Oracle® Key Vault
Administrator's Guide

Release 18.4
F29574-09
September 2020

Oracle Key Vault Administrator's Guide, Release 18.4

F29574-09

Copyright © 2013, 2020, Oracle and/or its affiliates.

Primary Author: Mark Doran

Contributing Authors: Hans Forbrich, Mark Fuller, James Womack

Contributors: Alexis Abell, Bharathi Baskaran, Lalitha Chowdary, Shubham Goyal, Srivatsan Kannan, Usha Krishnamurthy, Shirley Kumamoto, Swapna Jawarikapisha, Peter Knaggs, Michael Leong, Hui Li, William Maroulis, Khushal Melana, Rahil Mir, Dongwon Park, Sunil Pulla, Vipin Samar, Radhika Siravara, Ajay Srivastava, Peter Wahl

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

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

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

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government’s use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

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

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

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

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

Contents

Preface

Audience xxiii
Documentation Accessibility xxiii
Related Documents xxiii
Conventions xxiv

Changes in This Release for Oracle Key Vault

Changes for Oracle Key Vault Release 18.4 xxv
Changes for Oracle Key Vault Release 18.3 xxvii

1 Introduction to Oracle Key Vault

1.1 About Oracle Key Vault and Key Management 1-1
1.2 Benefits of Using Oracle Key Vault 1-2
1.3 Oracle Key Vault Use Cases 1-4
1.3.1 Centralized Storage of Oracle Wallet Files and Java Keystores 1-4
1.3.2 Centralized Management of TDE Master Encryption Keys Using Online Master Keys 1-5
1.3.3 Storage of Credential Files 1-7
1.3.4 Online Management of Endpoint Keys and Secret Data 1-7
1.4 Who Should Use Oracle Key Vault 1-8
1.5 Major Features of Oracle Key Vault 1-8
1.5.1 Centralized Storage and Management of Security Objects 1-9
1.5.2 Management of Key Lifecycle 1-10
1.5.3 Reporting and Alerts 1-10
1.5.4 Separation of Duties for Oracle Key Vault Users 1-11
1.5.5 Support for a Primary-Standby Environment 1-11
1.5.6 Persistent Master Encryption Key Cache 1-12
1.5.7 Backup and Restore Functionality for Security Objects 1-12
1.5.8 Automation of Endpoint Enrollment Using RESTful Services 1-13
1.5.9 Key Management Support Using RESTful Services 1-13
1.5.10 Support for OASIS Key Management Interoperability Protocol (KMIP) 1-13

1.5.11 Database Release and Platform Support 1-14
1.5.12 Integration with External Audit and Monitoring Services 1-14
1.5.13 Integration of MySQL with Oracle Key Vault 1-14
1.5.14 Automatic Storage Management Cluster File System (ACFS) Encryption 1-14
1.5.15 Support for Oracle Cloud Database as a Service Endpoints 1-15
1.5.16 Oracle Key Vault Hardware Security Module Integration 1-15
1.6 Oracle Key Vault Interfaces 1-15
1.6.1 Oracle Key Vault Management Console 1-15
1.6.2 Oracle Key Vault okvutil Endpoint Utility 1-16
1.6.3 Oracle Key Vault RESTful Services 1-16
1.7 Overview of an Oracle Key Vault Deployment 1-16

2 Oracle Key Vault Concepts

2.1 Overview of Oracle Key Vault Concepts 2-1
2.2 Oracle Key Vault Deployment Architecture 2-2
2.3 Access Control Configuration 2-3
2.3.1 About Access Control Configuration 2-3
2.3.2 Access Grants 2-4
2.3.3 Access Control Options 2-4
2.4 Administrative Roles within Oracle Key Vault 2-5
2.4.1 About Administrative Roles in Oracle Key Vault 2-5
2.4.2 Separation of Duties in Oracle Key Vault 2-6
2.4.3 System Administrator Role Duties 2-7
2.4.4 Key Administrator Role Duties 2-7
2.4.5 Audit Manager Role Duties 2-7
2.5 Emergency System Recovery Process 2-8
2.6 Root and Support User Accounts 2-8
2.7 Endpoint Administrators 2-9
2.8 FIPS Mode 2-9

3 Oracle Key Vault Multi-Master Cluster Concepts

3.1 Oracle Key Vault Multi-Master Cluster Overview 3-1
3.2 Benefits of Oracle Key Vault Multi-Master Clustering 3-2
3.3 Multi-Master Cluster Architecture 3-3
3.3.1 Oracle Key Vault Cluster Nodes 3-4
3.3.2 Cluster Node Limitations 3-4
3.3.3 Cluster Subgroup 3-5
3.3.4 Critical Data in Oracle Key Vault 3-5
3.3.5 Oracle Key Vault Read-Write Nodes 3-5

3.3.6 Oracle Key Vault Read-Only Nodes 3-6
3.3.7 Cluster Node Mode Types 3-6
3.3.8 Operations Permitted on Cluster Nodes in Different Modes 3-7
3.4 Building and Managing a Multi-Master Cluster 3-7
3.4.1 About Building and Managing a Multi-Master Cluster 3-7
3.4.2 Creation of the Initial Node in a Multi-Master Cluster 3-7
3.4.3 Expansion of a Multi-Master Cluster 3-9
3.4.3.1 About the Expansion of a Multi-Master Cluster 3-9
3.4.3.2 Management of Cluster Reconfiguration Changes Using a Controller Node 3-9
3.4.3.3 Addition of a Candidate Node to the Multi-Master Cluster 3-10
3.4.3.4 Addition of More Nodes to a Multi-Master Cluster 3-10
3.4.4 Migration to the Cluster from an Existing Deployment 3-12
3.4.4.1 Conversion of an Oracle Key Vault Standalone Server to a Multi-Master Cluster 3-13
3.4.4.2 Conversion from a Primary-Standby Server to a Multi-Master Cluster 3-13
3.5 Oracle Key Vault Multi-Master Cluster Deployment Scenarios 3-13
3.5.1 Cluster Size and Availability in Deployments 3-14
3.5.2 Two-Node Cluster Deployment 3-14
3.5.3 Mid-Size Cluster Across Two Data Centers Deployment 3-16
3.6 Multi-Master Cluster Features 3-19
3.6.1 Cluster Inconsistency Resolution in a Multi-Master Cluster 3-19
3.6.2 Name Conflict Resolution in a Multi-Master Cluster 3-19
3.6.3 Endpoint Node Connection Lists (Endpoint Node Scan Lists) 3-20

4 Oracle Key Vault Installation and Configuration

4.1 About Oracle Key Vault Installation and Configuration 4-1
4.2 Oracle Key Vault Installation Requirements 4-1
4.2.1 System Requirements 4-2
4.2.2 Network Port Requirements 4-3
4.2.3 Supported Endpoint Platforms 4-3
4.2.4 Endpoint Database Requirements 4-4
4.3 Installing and Configuring Oracle Key Vault 4-4
4.3.1 Downloading the Oracle Key Vault Appliance Software 4-5
4.3.2 Installing the Oracle Key Vault Appliance Software 4-6
4.3.3 Performing Post-Installation Tasks 4-10
4.4 Logging In to the Oracle Key Vault Management Console 4-15
4.5 Upgrading a Standalone or Primary-Standby Oracle Key Vault Server 4-16
4.5.1 About Upgrading the Oracle Key Vault Server Software 4-16
4.5.2 Step 1: Back Up the Server Before You Upgrade 4-17

4.5.3 Step 2: Perform Pre-Upgrade Tasks 4-17
4.5.4 Step 3: Upgrade the Oracle Key Vault Server or Server Pair 4-18
4.5.4.1 About Upgrading an Oracle Key Vault Server or Server Pair 4-18
4.5.4.2 Upgrading a Standalone Oracle Key Vault Server 4-19
4.5.4.3 Upgrading a Pair of Oracle Key Vault Servers in a Primary-Standby Deployment 4-21
4.5.5 Step 4: Upgrade the Endpoint Software 4-22
4.5.6 Step 5: If Necessary, Remove Old Kernels 4-23
4.5.7 Step 6: If Necessary, Add Disk Space to Extend Swap Space 4-25
4.5.8 Step 7: If Necessary, Remove SSH-Related DSA Keys 4-28
4.5.9 Step 8: Back Up the Upgraded Oracle Key Vault Server 4-28
4.6 Upgrading Oracle Key Vault in a Multi-Master Cluster Environment 4-29
4.6.1 About Upgrading Oracle Key Vault in a Multi-Master Cluster Environment 4-29
4.6.2 Step 1: Perform Pre-Upgrade Tasks 4-30
4.6.3 Step 2: If Upgrading from Release 18.1, Run the Pre-Upgrade Script on Each Node 4-31
4.6.4 Step 3: Upgrade Each Multi-Master Cluster Node 4-31
4.6.5 Step 4: Check the Node Version and the Cluster Version 4-32
4.6.6 Rolling Back the Pre-Upgrade Script 4-33
4.7 Overview of the Oracle Key Vault Management Console 4-34
4.8 Performing Actions and Searches 4-34
4.8.1 Actions Menus 4-34
4.8.2 Search Bars 4-35

5 Managing Oracle Key Vault Multi-Master Clusters

5.1 About Managing Oracle Key Vault Multi-Master Clusters 5-2
5.2 Creating the First (Initial) Node of a Cluster 5-2
5.3 Adding a Node to the Cluster 5-3
5.3.1 Creating a Read-Write Pair of Nodes in a Cluster 5-3
5.3.2 Creating a Read-Only Node in a Cluster 5-5
5.3.3 Creating an Additional Read-Write Pair in a Cluster 5-7
5.4 Terminating the Pairing of a Node 5-7
5.5 Disabling a Cluster Node 5-8
5.6 Enabling a Disabled Cluster Node 5-9
5.7 Deleting a Cluster Node 5-9
5.8 Force Deleting a Cluster Node 5-10
5.9 Managing Replication Between Nodes 5-10
5.9.1 Restarting Cluster Services 5-11
5.9.2 Disabling Node Replication 5-11
5.9.3 Enabling Node Replication 5-11

5.10 Cluster Management Information 5-11
5.11 Cluster Monitoring Information 5-13
5.12 Naming Conflicts and Resolution 5-14
5.12.1 About Naming Conflicts and Resolution 5-15
5.12.2 Naming Conflict Resolution Information 5-15
5.12.3 Changing the Suggested Conflict Resolution Name 5-16
5.12.4 Accepting the Suggested Conflict Resolution Name 5-16
5.13 Multi-Master Cluster Deployment Recommendations 5-16

6 Managing an Oracle Key Vault Primary-Standby Configuration

6.1 Overview of the Oracle Key Vault Primary-Standby Configuration 6-1
6.1.1 About the Oracle Key Vault Primary-Standby Configuration 6-2
6.1.2 Benefits of an Oracle Key Vault Primary-Standby Configuration 6-4
6.1.3 Difference Between Primary-Standby Configuration and Multi-Master Cluster 6-4
6.1.4 Primary Server Role in a Primary-Standby Configuration 6-5
6.1.5 Standby Server Role in a Primary-Standby Configuration 6-5
6.2 Configuring the Primary-Standby Environment 6-5
6.2.1 Step 1: Configure the Primary Server 6-5
6.2.2 Step 2: Configure the Standby Server 6-7
6.2.3 Step 3: Complete the Configuration on the Primary Server 6-9
6.3 Switching the Primary and Standby Servers 6-11
6.4 Restoring Primary-Standby After a Failover 6-12
6.5 Disabling (Unpairing) the Primary-Standby Configuration 6-13
6.6 Read-Only Restricted Mode in a Primary-Standby Configuration 6-14
6.6.1 About Read-Only Restricted Mode in a Primary-Standby Configuration 6-15
6.6.2 Primary-Standby with Read-Only Restricted Mode 6-16
6.6.3 Primary-Standby without Read-Only Restricted Mode 6-16
6.6.4 States of Read-Only Restricted Mode 6-17
6.6.4.1 About the States of Read-Only Restricted Mode 6-17
6.6.4.2 Read-Only Restricted State Functionality During a Primary Server Failure 6-19
6.6.4.3 Read-Only Restricted Mode Functionality During a Standby Server Failure 6-19
6.6.4.4 Read-Only Restricted State Functionality During a Network Failure 6-19
6.6.5 Enabling Read-Only Restricted Mode 6-20
6.6.6 Disabling Read-Only Restricted Mode 6-21
6.6.7 Recovering from Read-Only Restricted Mode 6-21
6.6.8 Read-Only Restricted Mode Notifications 6-22
6.7 Best Practices for Using Oracle Key Vault in a Primary-Standby Configuration 6-22

7 Managing Oracle Key Vault Users

7.1 Managing User Accounts 7-1
7.1.1 About Oracle Key Vault User Accounts 7-1
7.1.2 How a Multi-Master Cluster Affects User Accounts 7-2
7.1.2.1 Multi-Master Cluster Effect on System Administrator Users 7-3
7.1.2.2 Multi-Master Cluster Effect on Key Administrator Users 7-3
7.1.2.3 Multi-Master Cluster Effect on Audit Manager Users 7-3
7.1.2.4 Multi-Master Cluster Effect on Administration Users 7-4
7.1.2.5 Multi-Master Cluster Effect on System Users 7-4
7.1.3 Creating an Oracle Key Vault User Account 7-4
7.1.4 Viewing User Account Details 7-6
7.1.5 Deleting an Oracle Key Vault User Account 7-7
7.2 Managing Administrative Roles and Privileges 7-7
7.2.1 About Managing Administrative Roles 7-8
7.2.2 Granting or Changing an Administrative Role of a User 7-8
7.2.3 Granting a User Access to a Virtual Wallet 7-9
7.2.4 Revoking an Administrative Role from a User 7-10
7.3 Managing User Passwords 7-10
7.3.1 About Changing User Passwords 7-11
7.3.2 Changing Your Own Password 7-12
7.3.3 Changing Another User's Password 7-12
7.3.3.1 Changing a Password Manually 7-13
7.3.3.2 Changing a Password Through Email Notification 7-13
7.3.3.3 Changing Operating System User Account Passwords 7-14
7.3.4 Controlling the Use of Password Reset Methods 7-16
7.3.4.1 About Controlling the Use of Password Reset Methods 7-16
7.3.4.2 Configuring the Use of Password Reset Operations 7-17
7.4 Managing User Email 7-17
7.4.1 Changing the User Email Address 7-18
7.4.2 Disabling Email Notifications for a User 7-18
7.5 Managing User Groups 7-19
7.5.1 About Managing User Groups 7-19
7.5.2 How a Multi-Master Cluster Affects User Groups 7-19
7.5.3 Creating a User Group 7-20
7.5.4 Adding a User to a User Group 7-22
7.5.5 Granting a User Group Access to a Virtual Wallet 7-22
7.5.6 Renaming a User Group 7-23
7.5.7 Changing a User Group Description 7-23
7.5.8 Removing a User from a User Group 7-23

7.5.9 Deleting a User Group 7-24

8 Managing Oracle Key Vault Virtual Wallets and Security Objects

8.1 Managing Virtual Wallets 8-1
8.1.1 About Virtual Wallets 8-1
8.1.2 Creating a Virtual Wallet 8-2
8.1.3 Adding Security Objects to a Virtual Wallet 8-4
8.1.4 Removing Security Objects from a Virtual Wallet 8-5
8.1.5 Deleting a Virtual Wallet 8-5
8.2 Managing Access to Virtual Wallets from Keys and Wallets Tab 8-6
8.2.1 About Managing Access to Virtual Wallets from Keys and Wallets Tab 8-6
8.2.2 Granting Access to Users, User Groups, Endpoints, and Endpoint Groups 8-6
8.2.3 Modifying Access to Users, User Groups, Endpoints, and Endpoint Groups 8-7
8.3 Managing Access to Virtual Wallets from User's Menu 8-8
8.3.1 Granting a User Access to a Virtual Wallet 8-8
8.3.2 Revoking User Access from a Virtual Wallet 8-9
8.3.3 Granting a User Group Access to a Virtual Wallet 8-9
8.3.4 Revoking User Group Access from a Virtual Wallet 8-10
8.4 Managing the State of a Key or a Security Object 8-10
8.4.1 About Managing the State of a Key or a Security Object 8-11
8.4.2 How a Multi-Master Cluster Affects Keys and Security Ojects 8-11
8.4.3 Activating a Key or Security Object 8-12
8.4.4 Deactivating a Key or Security Object 8-12
8.4.5 Revoking a Key or Security Object 8-12
8.4.6 Destroying a Key or Security Object 8-13
8.5 Managing Details of Security Objects 8-13
8.5.1 About Managing the Details of Security Objects 8-13
8.5.2 Searching for Security Object Items 8-14
8.5.3 Viewing the Details of a Security Object 8-15
8.5.4 Adding or Modifying Details of a Security Object 8-18

9 Managing Oracle Key Vault Endpoints

9.1 Overview of Managing Endpoints 9-1
9.1.1 About Managing Endpoints 9-1
9.1.2 How a Multi-Master Cluster Affects Endpoints 9-2
9.2 Managing Endpoints 9-3
9.2.1 Types of Endpoint Enrollment 9-3
9.2.2 Endpoint Enrollment in a Multi-Master Cluster 9-4

9.2.3 Adding an Endpoint as an Oracle Key Vault System Administrator 9-5
9.2.4 Adding Endpoints Using Self-Enrollment 9-8
9.2.4.1 About Adding Endpoints Using Self-Enrollment 9-8
9.2.4.2 Adding an Endpoint Using Self-Enrollment 9-9
9.2.5 Deleting, Suspending, or Reenrolling Endpoints 9-9
9.2.5.1 About Deleting Endpoints 9-10
9.2.5.2 Deleting One or More Endpoints 9-10
9.2.5.3 Deleting One Endpoint (Alternative Method) 9-11
9.2.5.4 Suspending an Endpoint 9-11
9.2.5.5 Reenrolling an Endpoint 9-12
9.3 Default Wallets and Endpoints 9-13
9.3.1 Associating a Default Wallet with an Endpoint 9-13
9.3.2 Setting the Default Wallet for an Endpoint 9-13
9.4 Managing Endpoint Access to a Virtual Wallet 9-14
9.4.1 Granting an Endpoint Access to a Virtual Wallet 9-15
9.4.2 Revoking Endpoint Access to a Virtual Wallet 9-16
9.4.3 Viewing Wallet Items Accessed by Endpoints 9-16
9.5 Managing Endpoint Groups 9-17
9.5.1 How a Multi-Master Cluster Affects Endpoint Groups 9-17
9.5.2 Creating an Endpoint Group 9-18
9.5.3 Modifying Endpoint Group Details 9-20
9.5.4 Granting an Endpoint Group Access to a Virtual Wallet 9-21
9.5.5 Adding an Endpoint to an Endpoint Group 9-21
9.5.6 Removing an Endpoint from an Endpoint Group 9-22
9.5.7 Deleting Endpoint Groups 9-23
9.6 Managing Endpoint Details 9-23
9.6.1 About Endpoint Details 9-23
9.6.2 Modifying Endpoint Details 9-24
9.6.3 Global Endpoint Configuration Parameters 9-25
9.6.3.1 About Global Endpoint Configuration Parameters 9-25
9.6.3.2 Setting Global Endpoint Configuration Parameters 9-26
9.7 Upgrading Endpoints 9-27
9.7.1 Upgrading Endpoint Software from an Endpoint 9-27
9.7.1.1 Step 1: Prepare the Endpoint Environment 9-27
9.7.1.2 Step 2: Download the Oracle Key Vault Software onto the Endpoint 9-28
9.7.1.3 Step 3: Install the Oracle Key Vault Software onto the Endpoint 9-29
9.7.1.4 Step 4: Perform Post-Installation Tasks 9-30
9.7.2 Upgrading Endpoint Software on an Enrolled Endpoint 9-32

10 Enrolling Endpoints for Oracle Key Vault

10.1 About Endpoint Enrollment and Provisioning 10-1
10.2 Finalizing Enrollment and Provisioning 10-3
10.2.1 Step 1: Enroll the Endpoint and Download the Software 10-3
10.2.2 Step 2: Prepare the Endpoint Environment 10-5
10.2.3 Step 3: Install the Oracle Key Vault Software onto the Endpoint 10-5
10.2.4 Step 4: Perform Post-Installation Tasks 10-7
10.3 Environment Variables and Endpoint Provisioning Guidance 10-8
10.3.1 How the Location of JAVA_HOME Location Is Determined 10-8
10.3.2 Location of the okvclient.ora File and Environment Variables 10-9
10.3.3 Setting OKV_HOME for Non-Database Utilities to Communicate with Oracle Key Vault 10-9
10.3.4 Environment Variables in sqlnet.ora File 10-10
10.4 Endpoints That Do Not Use the Oracle Key Vault Client Software 10-10
10.5 Transparent Data Encryption Endpoint Management 10-10
10.6 Endpoint okvclient.ora Configuration File 10-11

11 Deploying Oracle Key Vault on an Oracle Cloud Infrastructure VM Compute Instance

11.1 About Deploying Oracle Key Vault on an Oracle Cloud Infrastructure Compute Instance 11-1
11.2 Benefits of Using Oracle Key Vault in Oracle Cloud Infrastructure 11-2
11.3 Provisioning an Oracle Key Vault Compute Instance 11-3
11.3.1 About Provisioning an Oracle Key Vault Compute Instance 11-3
11.3.2 Launching the Oracle Key Vault Compute Instance 11-4
11.3.2.1 About Launching the Oracle Key Vault Compute Instance 11-4
11.3.2.2 Step 1: Ensure That You Have Prerequisites in Place 11-4
11.3.2.3 Step 2: Find the Oracle Key Vault Image 11-5
11.3.2.4 Step 3: Launch the Oracle Key Vault VM Compute Instance 11-5
11.3.2.5 Step 4: Perform Post-Launch and Post-Installation Tasks 11-6
11.4 General Management of an Oracle Key Vault Compute Instance 11-6
11.4.1 Starting, Restarting, or Stopping an Oracle Key Vault Compute Instance 11-7
11.4.2 System Settings in an Oracle Key Vault Compute Instance 11-8
11.4.3 Backup and Restore Operations for Oracle Key Vault Compute Instances 11-8
11.4.4 Terminating an Oracle Key Vault Compute Instance 11-8
11.5 Migrating Oracle Key Vault Deployments Between On-Premises and OCI 11-9
11.5.1 About Performing Migrations with Oracle Key Vault Compute Instance Data 11-9

11.5.2 Migrating Oracle Key Vault Deployments into OCI Using Backup and Restore 11-10
11.5.3 Migrating Oracle Key Vault Deployments Out of OCI Using Backup and Restore 11-10

12 Oracle Database Instances in Oracle Cloud Infrastructure

12.1 About Managing Oracle Cloud Infrastructure Database Instance Endpoints 12-1
12.2 Preparing a Database Instance on OCI to be an Oracle Key Vault Endpoint 12-2
12.2.1 About Preparing a Database Instance on OCI to be an Oracle Key Vault Endpoint 12-2
12.2.2 Configuring a Database Cloud Service Instance 12-2
12.2.3 Creating a Low Privileged Operating System User on Database as a Service 12-3
12.3 Using an SSH Tunnel Between Oracle Key Vault and Database as a Service 12-4
12.3.1 Creating an SSH Tunnel Between Oracle Key Vault and a DBaaS Instance 12-5
12.3.2 Managing a Reverse SSH Tunnel in a Multi-Master Cluster 12-8
12.3.3 Managing a Reverse SSH Tunnel in a Primary-Standby Configuration 12-8
12.3.4 Viewing SSH Tunnel Configuration Details 12-9
12.3.5 Disabling an SSH Tunnel Connection 12-9
12.3.6 How the Connection Works if the SSH Tunnel Is Not Active 12-11
12.3.7 Deleting an SSH Tunnel Configuration 12-11
12.4 Registering and Enrolling a Database as a Service Instance as an Oracle Key Vault Endpoint 12-12
12.4.1 About Registering and Enrolling a Database as a Service Instance as an Oracle Key Vault Endpoint 12-12
12.4.2 Step 1: Register the Endpoint in the Oracle Key Vault Management Console 12-13
12.4.3 Step 2: Prepare the Endpoint Environment 12-15
12.4.4 Step 3: Install the Oracle Key Vault Software onto the Endpoint 12-15
12.4.5 Step 4: Perform Post-Installation Tasks 12-17
12.5 Suspending Database Cloud Service Access to Oracle Key Vault 12-18
12.5.1 About Suspending Database Cloud Service Access to Oracle Key Vault 12-18
12.5.2 Suspending Access for a Database Cloud Service to Oracle Key Vault 12-19
12.6 Resuming Database Cloud Service Access to Oracle Key Vault 12-20
12.7 Resuming a Database Endpoint Configured with a Password-Based Keystore 12-20

13 Oracle Key Vault Administration and Key Management with RESTful Services

13.1 About RESTful Services 13-1

13.2 Required Privileges for Using RESTful Services 13-2
13.3 Enabling RESTful Services 13-3
13.3.1 Step 1: Check the Endpoint System Requirements 13-3
13.3.2 Step 2: Enable Network Services 13-4
13.3.3 Step 3: Enable RESTful Services 13-4
13.3.4 Step 4: Download the RESTful Software Utility 13-4
13.4 Managing the RESTful Services Configuration File 13-5
13.4.1 About Managing the RESTful Services Configuration File 13-5
13.4.2 Configuration File Creation Guidelines 13-6
13.4.3 Creating the RESTful Services Configuration File 13-6
13.4.4 Examples of Configuration Files 13-8
13.4.5 Executing a Single RESTful Command 13-8
13.4.6 Executing Multiple RESTful Administrative Commands Using a Script 13-9
13.4.7 Creating a Script to Automatically Enroll Oracle Databases as Endpoints 13-10
13.5 Disabling RESTful Services 13-14
13.6 Oracle Key Vault Administrative REST Client Tool Commands 13-14
13.6.1 RESTful Services Command Syntax 13-15
13.6.2 RESTful Services Wallet Command Syntax 13-16
13.6.3 Commands to Add and Enroll Endpoints 13-17
13.6.3.1 create_endpoint Command 13-18
13.6.3.2 create_unique_endpoint Command 13-19
13.6.3.3 delete_endpoint Command 13-21
13.6.3.4 download Command 13-22
13.6.3.5 get_enrollment_token Command 13-23
13.6.3.6 provision Command 13-24
13.6.3.7 re_enroll Command 13-27
13.6.3.8 re_enroll_all Command 13-28
13.6.4 Commands to Modify Endpoint Details 13-28
13.6.4.1 modify_endpoint_email Command 13-29
13.6.4.2 modify_endpoint_desc Command 13-30
13.6.4.3 modify_endpoint_name Command 13-31
13.6.4.4 modify_endpoint_platform Command 13-32
13.6.4.5 modify_endpoint_type Command 13-33
13.6.5 Endpoint Group Commands 13-34
13.6.5.1 add_epg_member Command 13-35
13.6.5.2 create_endpoint_group Command 13-36
13.6.5.3 create_unique_endpoint_group Command 13-37
13.6.5.4 delete_endpoint_group Command 13-38
13.6.5.5 drop_epg_member Command 13-39
13.6.5.6 modify_endpoint_group_desc Command 13-40

13.6.5.7 modify_endpoint_group_name Command 13-41
13.6.6 Virtual Wallet Commands 13-42
13.6.6.1 add_wallet_access_ep Command 13-43
13.6.6.2 add_wallet_access_epg Command 13-45
13.6.6.3 check_object_status Command 13-46
13.6.6.4 create_unique_wallet Command 13-47
13.6.6.5 create_wallet Command 13-49
13.6.6.6 delete_wallet Command 13-50
13.6.6.7 drop_wallet_access_ep Command 13-51
13.6.6.8 drop_wallet_access_epg Command 13-52
13.6.6.9 get_default_wallet Command 13-53
13.6.6.10 get_object_name Command 13-54
13.6.6.11 get_wallets Command 13-55
13.6.6.12 modify_wallet_access_ep Command 13-56
13.6.6.13 modify_wallet_access_epg Command 13-58
13.6.6.14 modify_wallet_desc Command 13-59
13.6.6.15 modify_wallet_name Command 13-60
13.6.6.16 set_default_wallet Command 13-61
13.6.7 Error Reporting 13-62
13.6.7.1 About Error Reporting 13-62
13.6.7.2 Command Line Error Reporting 13-63
13.6.7.3 Error Reporting while Running Commands from a Script 13-63
13.6.8 Help Information 13-64
13.7 Oracle Key Vault Key Management REST Client Tool Commands 13-64
13.7.1 About Oracle Key Vault Key Management REST Client Tool Commands 13-65
13.7.2 Oracle Key Vault Key Management REST Client API Using OKVRESTSERVICE 13-65
13.7.3 List of Key Management REST Client Tool Commands 13-67
13.7.4 Key Creation and Registration Commands 13-68
13.7.4.1 create_key Command 13-68
13.7.4.2 reg_key Command 13-69
13.7.4.3 get_cert Command 13-70
13.7.4.4 get_key Command 13-70
13.7.4.5 get_opaque Command 13-71
13.7.4.6 get_secret Command 13-71
13.7.4.7 reg_cert Command 13-72
13.7.4.8 reg_opaque Command 13-73
13.7.4.9 reg_secret Command 13-73
13.7.5 Key Attribute Management Commands 13-74
13.7.5.1 add_attr Command 13-75
13.7.5.2 add_custom_attr Command 13-75

13.7.5.3 all_attr Command 13-76
13.7.5.4 del_attr Command 13-77
13.7.5.5 del_custom_attr Command 13-78
13.7.5.6 get_attr Command 13-78
13.7.5.7 list_attr Command 13-79
13.7.5.8 mod_attr Command 13-80
13.7.5.9 mod_custom_attr Command 13-81
13.7.6 Key Life Cycle Management Commands 13-81
13.7.6.1 activate Command 13-82
13.7.6.2 destroy Command 13-82
13.7.6.3 locate Command 13-83
13.7.6.4 revoke Command 13-84
13.7.6.5 query Command 13-84
13.7.7 Wallet Commands 13-85
13.7.7.1 add_member Command 13-85
13.7.7.2 del_member Command 13-86
13.7.7.3 list_wallet Command 13-86

14 Backup and Restore Operations

14.1 About Backing Up and Restoring Data in Oracle Key Vault 14-1
14.2 Oracle Key Vault Backup Destinations 14-2
14.2.1 About the Oracle Key Vault Backup Destination 14-2
14.2.2 Creating a Remote Backup Destination 14-4
14.2.3 Changing Settings on a Remote Backup Destination 14-6
14.2.4 Deleting a Remote Backup Destination 14-7
14.3 Backup Schedules and States 14-7
14.3.1 About Backup Schedule Types and States 14-8
14.3.2 Types of Oracle Key Vault Backups 14-8
14.3.3 Scheduled Backup States in Oracle Key Vault 14-9
14.4 Scheduling and Managing Oracle Key Vault Backups 14-9
14.4.1 Scheduling a Backup for Oracle Key Vault 14-10
14.4.2 Changing a Backup Schedule for Oracle Key Vault 14-11
14.4.3 Deleting a Backup Schedule from Oracle Key Vault 14-11
14.4.4 How Primary-Standby Affects Oracle Key Vault Backups 14-11
14.4.5 Protecting the Backup Using the Recovery Passphrase 14-12
14.5 Restoring Oracle Key Vault Data 14-12
14.5.1 About the Oracle Key Vault Restore Process 14-13
14.5.2 Procedure for Restoring Oracle Key Vault Data 14-13
14.5.3 Multi-Master Cluster and the Restore Operation 14-14
14.5.4 Primary-Standby and the Restore Operation 14-15

14.5.5 Third-Party Certificates and the Restore Operation 14-15
14.5.6 Changes Resulting from a System State Restore 14-15
14.6 Backup and Restore Best Practices 14-16

15 Oracle Key Vault General System Administration

15.1 Overview of Oracle Key Vault General System Administration 15-1
15.1.1 About Oracle Key Vault General System Administration 15-2
15.1.2 Viewing the Oracle Key Vault Dashboard 15-2
15.1.3 Using the Status Panes in the Dashboard 15-4
15.2 Configuring Oracle Key Vault in a Non-Multi-Master Cluster Environment 15-5
15.2.1 Configuring the Network Details 15-5
15.2.2 Configuring the Network Services 15-6
15.2.3 Configuring the System Time 15-7
15.2.4 Configuring DNS 15-9
15.2.5 Configuring FIPS Mode 15-9
15.2.6 Configuring Syslog 15-10
15.2.7 Configuring RESTful Services 15-11
15.2.8 Configuring Oracle Audit Vault Integration 15-11
15.2.9 Configuring the Oracle Key Vault Management Console Web Session Timeout 15-11
15.2.10 Restarting or Powering Off Oracle Key Vault 15-12
15.3 Configuring Oracle Key Vault in a Multi-Master Cluster Environment 15-12
15.3.1 Configuring System Settings for Individual Multi-Master Cluster Nodes 15-13
15.3.1.1 Configuring the Network Details for the Node 15-13
15.3.1.2 Configuring the Network Services for the Node 15-14
15.3.1.3 Configuring the System Time for the Node 15-14
15.3.1.4 Configuring DNS for the Node 15-15
15.3.1.5 Configuring the FIPS Mode for the Node 15-15
15.3.1.6 Configuring Syslog for the Node 15-16
15.3.1.7 Configuring Oracle Audit Vault Integration for the Node 15-16
15.3.1.8 Restarting or Powering Off Oracle Key Vault from a Node 15-16
15.3.2 Managing Oracle Key Vault Multi-Master Clusters 15-17
15.3.2.1 About Configuring Cluster System Settings 15-17
15.3.2.2 Configuring the System Time for the Cluster 15-18
15.3.2.3 Configuring DNS for the Cluster 15-18
15.3.2.4 Configuring Maximum Disable Node Duration for the Cluster 15-18
15.3.2.5 Configuring RESTful Services for the Cluster 15-19
15.3.2.6 Configuring Syslog for the Cluster 15-19
15.3.2.7 Configuring SNMP Settings for the Cluster 15-19
15.3.2.8 Configuring the Oracle Key Vault Management Console Web Session Timeout for the Cluster 15-20

15.4 Managing System Recovery 15-20
15.4.1 About Managing System Recovery 15-21
15.4.2 Recovering Credentials for Administrators 15-21
15.4.3 Changing the Recovery Passphrase in a Non-Clusters Environment 15-21
15.4.4 Changing the Recovery Passphrase in a Multi-Master Cluster 15-22
15.4.4.1 Step 1: Initiate the Recovery Passphrase Change Across the Nodes 15-22
15.4.4.2 Step 2: Change the Recovery Passphrase 15-23
15.4.5 Changing the Installation Passphrase 15-24
15.4.5.1 About Changing the Installation Passphrase 15-25
15.4.5.2 Changing an Installation Passphrase 15-25
15.5 Support for a Primary-Standby Environment 15-26
15.6 Commercial National Security Algorithm Suite Support 15-27
15.6.1 About Commercial National Security Algorithm Suite Support 15-27
15.6.2 Running the Commercial National Security Algorithm Scripts 15-28
15.6.3 Performing Backup and Restore Operations with CNSA 15-29
15.6.4 Upgrading a Standalone Oracle Key Vault Server with CNSA 15-30
15.6.5 Upgrading Primary-Standby Oracle Key Vault Servers to Use CNSA 15-31
15.7 Minimizing Downtime 15-32

16 Managing Certificates

16.1 Rotating Certificates 16-1
16.1.1 About Rotating Certificates 16-1
16.1.2 Advice for Managing Certificate Rotations 16-2
16.1.3 Factors That May Affect the Certificate Rotation Process 16-2
16.1.4 Rotating All Certificates 16-3
16.1.5 Checking the Certificate Rotation Status 16-5
16.2 Managing Console Certificates 16-6
16.2.1 About Managing Console Certificates 16-6
16.2.2 Step 1: Download the Certificate Request 16-6
16.2.3 Step 2: Have the Certificate Signed 16-7
16.2.4 Step 3: Upload the Signed Certificate to Oracle Key Vault 16-7
16.2.5 Console Certificates in Special Use Case Scenarios 16-8

17 Monitoring and Auditing Oracle Key Vault

17.1 Managing System Monitoring 17-1
17.1.1 Configuring Remote Monitoring to Use SNMP 17-1
17.1.1.1 About Using SNMP for Oracle Key Vault 17-2
17.1.1.2 Granting SNMP Access to Users 17-3
17.1.1.3 Changing the SNMP User Name and Password 17-4

17.1.1.4 Changing SNMP Settings on the Standby Server 17-4
17.1.1.5 Remotely Monitoring Oracle Key Vault Using SNMP 17-5
17.1.1.6 SNMP Management Information Base Variables for Oracle Key Vault 17-6
17.1.1.7 Example: Simplified Remote Monitoring of Oracle Key Vault Using SNMP 17-7
17.1.2 Configuring Email Notification 17-9
17.1.2.1 About Email Notification 17-9
17.1.2.2 Configuring Email Settings 17-10
17.1.2.3 Testing the Email Configuration 17-12
17.1.2.4 Disabling Email Notifications for a User 17-13
17.1.3 Configuring the Syslog Destination for Individual Multi-Master Cluster Nodes 17-13
17.1.3.1 Setting the Syslog Destination Setting for the Node 17-13
17.1.3.2 Clearing the Syslog Destination Setting for the Node 17-14
17.1.4 Capturing System Diagnostics 17-14
17.1.4.1 About Capturing System Diagnostics 17-14
17.1.4.2 Installing the Diagnostics Generation Utility 17-15
17.1.4.3 Generating a System Diagnostics File 17-16
17.1.4.4 Removing the Diagnostic Generation Utility Temporary Files 17-16
17.1.4.5 Removing the Diagnostic Generation Utility 17-16
17.1.5 Configuring Oracle Audit Vault Integration for the Node 17-17
17.2 Configuring Oracle Key Vault Alerts 17-17
17.2.1 About Configuring Alerts 17-17
17.2.2 Configuring Alerts 17-19
17.2.3 Viewing Open Alerts 17-20
17.3 Managing System Auditing 17-22
17.3.1 About Auditing in Oracle Key Vault 17-22
17.3.2 Configuring Syslog to Store Audit Records 17-23
17.3.3 Configuring Audit Settings for a Multi-Master Cluster 17-24
17.3.4 Viewing Audit Records 17-24
17.3.5 Exporting and Deleting Audit Records 17-25
17.3.6 Audit Consolidation with Audit Vault and Database Firewall 17-25
17.4 Using Oracle Key Vault Reports 17-26
17.4.1 About Oracle Key Vault Reports 17-26
17.4.2 Viewing Endpoint Reports 17-27
17.4.3 Viewing User Reports 17-28
17.4.4 Viewing Keys and Wallets Reports 17-28
17.4.5 Viewing System Reports 17-29

18 Managing Security Objects in Oracle Key Vault

18.1 Configuring an Oracle Key Vault-to-New TDE-Enabled Database Connection 18-1
18.1.1 About Configuring an Oracle Key Vault-to-New TDE-Enabled Database Connection 18-2
18.1.2 Limitations to Transparent Data Encryption Endpoint Integration 18-2
18.1.3 Step 1: Configure the Oracle Key Vault Server Environment 18-3
18.1.4 Step 2: Integrate Transparent Data Encryption with Oracle Key Vault 18-4
18.2 Migrating Existing TDE Wallets to Oracle Key Vault 18-5
18.2.1 About Migrating Existing TDE Wallets to Oracle Key Vault 18-6
18.2.2 Migrating an Existing TDE Wallet to Oracle Key Vault 18-6
18.2.3 Restoring Database Contents Previously Encrypted by TDE Using an Oracle Wallet 18-8
18.3 Using the Persistent Master Encryption Key Cache 18-9
18.3.1 About the Persistent Master Encrption Key Cache 18-10
18.3.2 About Oracle Key Vault Persistent Master Encyrption Key Cache Architecture 18-10
18.3.3 Caching Master Encryption Keys in the In-Memory and Persistent Master Encryption Key Cache 18-11
18.3.4 Storage Location of Persistent Master Encryption Key Cache 18-11
18.3.5 Persistent Master Encryption Key Cache Modes of Operation 18-12
18.3.5.1 Oracle Key Vault First Mode 18-12
18.3.5.2 Persistent Master Encryption Key Cache First Mode 18-12
18.3.6 Persistent Master Encryption Key Cache Refresh Window 18-13
18.3.7 Persistent Master Encryption Key Cache Parameters 18-13
18.3.7.1 PKCS11_CACHE_TIMEOUT Parameter 18-14
18.3.7.2 PKCS11_PERSISTENT_CACHE_TIMEOUT Parameter 18-14
18.3.7.3 PKCS11_PERSISTENT_CACHE_FIRST Parameter 18-15
18.3.7.4 PKCS11_CONFIG_PARAM_REFRESH_INTERVAL Parameter 18-15
18.3.7.5 PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW Parameter 18-16
18.3.7.6 EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN Parameter 18-16
18.3.8 Listing the Contents of the Persistent Master Encryption Key Cache 18-17
18.3.9 Oracle Database Deployments and Persistent Master Encryption Key Cache 18-18
18.4 Uploading and Downloading Oracle Wallets 18-19
18.4.1 About Uploading and Downloading Oracle Wallets 18-19
18.4.2 Uploading Oracle Wallets 18-20
18.4.3 Downloading Oracle Wallets 18-21
18.4.4 Guidelines for Uploading and Downloading Oracle Wallets 18-22
18.5 Uploading and Downloading JKS and JCEKS Keystores 18-23
18.5.1 About Uploading and Downloading JKS and JCEKS Keystores 18-23

18.5.2 Uploading JKS or JCEKS Keystores 18-23
18.5.3 Downloading JKS or JCEKS Keystores 18-24
18.5.4 Guidelines for Uploading and Downloading JKS and JCEKS Keystores 18-25
18.6 Uploading and Downloading Credential Files 18-25
18.6.1 About Uploading and Downloading Credential Files 18-25
18.6.2 Uploading a Credential File 18-26
18.6.3 Downloading a Credential File 18-27
18.6.4 Guidelines for Uploading and Downloading Credential Files 18-28
18.7 Using a User-Defined Key as the TDE Master Encryption Key 18-28
18.7.1 About Using a User-Defined Key as the TDE Master Encryption Key 18-28
18.7.2 Step 1: Upload the User-Defined Key 18-29
18.7.3 Step 2: Activate the User-Defined Key as a TDE Master Encryption Key 18-30

19 Using Oracle Key Vault with Other Features

19.1 Using a TDE-Configured Oracle Database in an Oracle RAC Environment 19-1
19.2 Using a TDE-Configured Oracle Database in an Oracle GoldenGate Environment 19-2
19.2.1 Oracle Wallets in an Oracle GoldenGate Environment 19-2
19.2.2 Online Master Keys in an Oracle GoldenGate Deployment 19-3
19.2.3 Migration of TDE Wallets in Oracle GoldenGate to Oracle Key Vault 19-3
19.3 Using a TDE-Configured Oracle Database in an Oracle Data Guard Environment 19-4
19.3.1 About Uploading Oracle Wallets in an Oracle Data Guard Environment 19-5
19.3.2 Uploading Oracle Wallets in an Oracle Data Guard Environment 19-5
19.3.3 Performing an Online Master Key Connection in an Oracle Data Guard Environment 19-6
19.3.4 Migrating Oracle Wallets in an Oracle Data Guard Environment 19-6
19.3.5 Reverse Migrating Oracle Wallets in an Oracle Data Guard Environment 19-7
19.3.6 Migrating an Oracle TDE Wallet to Oracle Key Vault for a Logical Standby Database 19-8
19.3.7 Checking the Oracle TDE Wallet Migration for a Logical Standby Database 19-8
19.4 Uploading Keystores from Automatic Storage Management to Oracle Key Vault 19-9
19.4.1 About Uploading Keystores from Automatic Storage Management to Oracle Key Vault 19-9
19.4.2 Uploading a Keystore from Automatic Storage Management to Oracle Key Vault 19-10
19.4.3 Copying a Keystore from Oracle Key Vault to Automatic Storage Management 19-11
19.5 MySQL Integration with Oracle Key Vault 19-12

19.6 Other Oracle Database Features That Oracle Key Vault Supports 19-12

A Oracle Key Vault Multi-Master Cluster Operations

B Oracle Key Vault okvutil Endpoint Utility Reference

B.1 About the okvutil Utility B-1
B.2 okvutil Command Syntax B-1
B.3 okvutil changepwd Command B-3
B.4 okvutil diagnostics Command B-3
B.5 okvutil download Command B-4
B.6 okvutil list Command B-6
B.7 okvutil upload Command B-8

C Troubleshooting Oracle Key Vault

C.1 Oracle Key Vault Pre-Installation Checklist C-2
C.2 Integrating Oracle Key Vault with Oracle Audit Vault and Database Firewall C-3
C.2.1 Step 1: Check the Environment C-3
C.2.2 Step 2: Register Oracle Key Vault as a Secured Target with AVDF C-3
C.2.3 Step 3: Register Oracle Key Vault as a Host with AVDF C-4
C.2.4 Step 4: Download the AVDF Agent and Upload it to Oracle Key Vault C-4
C.2.5 Step 5: Install the AVDF agent.jar File on the Oracle Key Vault Server C-5
C.2.6 Step 6: Add the Oracle Key Vault Audit Trail to AVDF C-5
C.2.7 Step 7: View Oracle Key Vault Audit Data Collected by AVDF C-6
C.3 RESTful Services Troubleshooting Help C-6
C.4 Error: Cannot Open Keystore Message C-6
C.5 KMIP Error: Invalid Field C-7
C.6 WARNING: Could Not Store Private Key Errors C-7
C.7 Errors After Upgrading Oracle Key Vault C-8
C.8 Error: Failed to Open Wallet C-8
C.9 Transaction Check Error: Diagnostics Generation Utility C-8
C.10 Fast-Start Failover (FSFO) Suspended (ORA-16818) C-9
C.11 SSH Tunnel Add Failure C-9
C.12 Error: Provision Command Fails if /usr/bin/java Does Not Exist C-10
C.13 TDE Endpoint Integration Issues C-10
C.14 Failover Situations in Primary-Standby Mode C-10
C.14.1 About Failover Situations in Primary-Standby Mode C-11
C.14.2 Failover Situations Without Read-Only Restricted Mode C-11
C.14.2.1 Primary Server: Planned Shutdown During an Upgrade C-11
C.14.2.2 Primary Server: Planned Shutdown During Maintenance C-12

C.14.2.3 Standby Server: Planned Shutdown C-12
C.14.2.4 Primary Server: Unplanned Shutdown C-13
C.14.2.5 Standby Server: Unplanned Shutdown C-13
C.14.3 Failover Situations with Read-Only Restricted Mode C-13
C.14.3.1 Primary Server: Planned Shutdown During an Upgrade C-14
C.14.3.2 Primary Server: Planned Shutdown During Maintenance C-14
C.14.3.3 Standby Server: Planned Shutdown C-15
C.14.3.4 Primary Server: Unplanned Shutdown C-15
C.14.3.5 Standby Server: Unplanned Shutdown C-16
C.15 Performing a Planned Shutdown C-16
C.15.1 Primary Server Planned Shutdown C-16
C.15.1.1 Performing a Primary Server Planned Shutdown During an Upgrade C-16
C.15.1.2 Performing a Primary Server Planned Shutdown During Maintenance C-17
C.15.2 Standby Server Planned Shutdown C-17
C.15.2.1 Performing a Standby Server Planned Shutdown During an Upgrade C-17
C.15.2.2 Performing a Standby Server Planned Shutdown During Maintenance C-18

D Security Technical Implementation Guides Compliance Standards

D.1 About Security Technical Implementation Guides D-1
D.2 Enabling and Disabling STIG Rules on Oracle Key Vault D-2
D.2.1 Enabling STIG Rules on Oracle Key Vault D-2
D.2.2 Disabling STIG Rules on Oracle Key Vault D-2
D.3 Current Implementation of STIG Rules on Oracle Key Vault D-2
D.4 Current Implementation of Database STIG Rules D-3
D.5 Current Implementation of Operating System STIG Rules D-6

Glossary

Index

Preface

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

- Audience
- Documentation Accessibility
- Related Documents
- Conventions

Audience

Oracle Key Vault Administrator's Guide is written for Oracle security administrators who are responsible for managing and centralizing encryption keys and other security objects.

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 Key Vault Root of Trust HSM Configuration Guide*
- *Oracle Database Security Guide*
- *Oracle Database Advanced Security Guide*
- *Oracle Database Administrator's Guide*
- *Oracle Data Guard Concepts and Administration*
- *Oracle Real Application Clusters Administration and Deployment Guide*
- *Oracle Fusion Middleware Understanding Oracle GoldenGate*
- *Key Management Interoperability Protocol Specification Version 1.1*

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

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

Conventions

The following text conventions are used in this document:

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

Changes in This Release for Oracle Key Vault

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

- Changes for Oracle Key Vault Release 18.4
  Oracle Key Vault release 18.4 introduces the ability to set a time-out value for the Oracle Key Vault management console.
- Changes for Oracle Key Vault Release 18.3
  Oracle Key Vault release 18.3 has two new features.

Changes for Oracle Key Vault Release 18.4

Oracle Key Vault release 18.4 introduces the ability to set a time-out value for the Oracle Key Vault management console.

- Management Console Idle Session Timeout
  Starting with this release, Oracle Key Vault will detect if the user session is idle, log the user out, and redirect the user to the login screen.
- Oracle Key Vault HSM Integration Supports Use of Token Labels
  Oracle Key Vault now provides the ability to choose a softcard slot based on a token label.
- Utimaco as a Supported HSM Vendor
  Utimaco is now a supported vendor for integration with Oracle Key Vault as the Root of Trust.

Management Console Idle Session Timeout

Starting with this release, Oracle Key Vault will detect if the user session is idle, log the user out, and redirect the user to the login screen.

Starting with Oracle Key Vault release 18.4, Oracle Key Vault will detect if the user's management console session is idle, log the user out, and redirect the user to the login screen. The user's session remains active as long as the user clicks a button, moves the mouse or presses a key, or is performing other management console related activities. If the user's session is idle for more than the management console timeout duration, then the user is logged out and redirected to the login screen.

The management console timeout is configurable and has a default setting of 10 minutes. Before the management console session ends, the user is notified and is given an option to extend the session. The notification is raised 2 minutes before session expiry if the timeout value is 10 minutes or longer. For smaller timeout values, the notification will be raised 10 seconds or 30 seconds prior to session

expiry depending on whether the timeout value is less than or greater than 5 minutes respectively. For example, if the timeout was set to 20 minutes, then the user will be notified after 18 minutes of inactivity and can extend the session. After the session is extended, if there is another period of 18 minutes with no activity detected, the user would be requested to extend the session once again. If the user does not extend the session this time, the user is logged out and redirected to login screen.

The management console idle session timeout applies to standalone, primary-standby, and multi-master cluster environments. In a multi-master cluster environment, setting the timeout value in one node applies the value to all nodes in the cluster. The timeout value takes effect after you click **Save** in the **System Settings** page, or click **Save to Cluster** in the **Cluster System Settings** page. Any update to the timeout value affects all new management console sessions immediately but affects the currently active sessions when the user's session is extended, the user refreshes the page, or the user navigates to another page.

Related Topics

- Configuring the Oracle Key Vault Management Console Web Session Timeout
  In a non-multi-master cluster environment, you can configure a timeout value in minutes for the Oracle Key Vault management console Web session.
- Configuring the Oracle Key Vault Management Console Web Session Timeout for the Cluster
  You can configure a timeout value in minutes for the Oracle Key Vault management console for all nodes in a multi-master cluster.

Oracle Key Vault HSM Integration Supports Use of Token Labels

Oracle Key Vault now provides the ability to choose a softcard slot based on a token label.

HSMs may support multiple tokens each with a token label. You can now specify a token label in order to create or use keys from tokens in specific slots for HSM integration. If you do not specify a token label, Oracle Key Vault will fall back to the previous behavior where it will choose the token in the first slot of the slot list. Token Labels in Oracle Key Vault enable support for softcards in nCipher HSMs and partitions in Thales (Safenet) HSMs.

Related Topics

- *Oracle Key Vault Root of Trust HSM Configuration Guide*

Utimaco as a Supported HSM Vendor

Utimaco is now a supported vendor for integration with Oracle Key Vault as the Root of Trust.

Starting with this release, you can configure Utimaco as a supported HSM vendor for integration with Oracle Key Vault, in addition to the current Safenet and nCipher vendors. For both the Hardware Security Modules and associated client-side libraries and tools, only version 4.31.1 is supported at this time.

Related Topics

- *Oracle Key Vault Root of Trust HSM Configuration Guide*

Changes for Oracle Key Vault Release 18.3

Oracle Key Vault release 18.3 has two new features.

- Oracle Key Vault Available in the Oracle Cloud Marketplace
  Starting with this release, you can deploy Oracle Key Vault to run on an Oracle Cloud Infrastructure (OCI) VM compute instance.
- Ability to Rename Endpoint Groups and Virtual Wallets Using RESTful Services
  Starting with this release, you can rename endpoint groups and virtual wallets using RESTful services.

Oracle Key Vault Available in the Oracle Cloud Marketplace

Starting with this release, you can deploy Oracle Key Vault to run on an Oracle Cloud Infrastructure (OCI) VM compute instance.

This functionality is available as click-to-deploy software in the Oracle Cloud Marketplace. Another benefit of this type of deployment is that provisioning in OCI is more streamlined and provides for a faster way to get an application running than in an on-premises installation, which requires an administrator to manage the hardware on which Oracle Key Vault is installed.

Related Topics

- Deploying Oracle Key Vault on an Oracle Cloud Infrastructure VM Compute Instance
  You can install Oracle Key Vault on an Oracle Cloud Infrastructure (OCI) VM compute instance from Oracle Cloud Marketplace.

Ability to Rename Endpoint Groups and Virtual Wallets Using RESTful Services

Starting with this release, you can rename endpoint groups and virtual wallets using RESTful services.

In previous releases, this ability was available in the Oracle Key Vault management console only, but is now available with the following new RESTful API commands:

- `modify_endpoint_group_name`
- `modify_wallet_name`

1

Introduction to Oracle Key Vault

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

- 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.
- Benefits of Using Oracle Key Vault
  Oracle Key Vault helps you to fight security threats, centralize key storage, and centralize key lifecycle management.
- Oracle Key Vault Use Cases
  The most typical use cases for Oracle Key Vault are centralized storage and management of security objects.
- Who Should Use Oracle Key Vault
  Oracle Key Vault is designed for users who are responsible for deploying, maintaining, and managing security within the enterprise.
- Major Features of Oracle Key Vault
  Oracle Key Vault enhances security in key management with a wide range of features that support different database deployments.
- Oracle Key Vault Interfaces
  Oracle Key Vault provides both a graphical user interface and command-line interfaces.
- Overview of an Oracle Key Vault Deployment
  There are three different Oracle Key Vault deployment options.

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.

Security objects that you can manage with Oracle Key Vault include as encryption keys, Oracle wallets, Java keystores (JKS), Java Cryptography Extension keystores (JCEKS), and credential files.

Oracle Key Vault centralizes encryption key storage across your organization quickly and efficiently. Built on Oracle Linux, Oracle Database, Oracle Database security features like Oracle Transparent Data Encryption, Oracle Database Vault, Oracle Virtual Private Database, and Oracle GoldenGate technology, Oracle Key Vault's centralized, highly available, and scalable security solution helps to overcome the biggest key-management challenges facing organizations today. With Oracle Key Vault you can retain, back up, 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). In addition, it complies

with the industry standard OASIS Key Management Interoperability Protocol (KMIP) for compatibility with KMIP-based clients.

You can use Oracle Key Vault to manage a variety of other endpoints, such as MySQL TDE encryption keys.

Oracle Key Vault also provides the multi-master cluster mode of operation, which increases availability and supports geographic distribution.

Related Topics

- Support for OASIS Key Management Interoperability Protocol (KMIP)
  You can use Oracle Key Vault with a range of OASIS KMIP Version 1.1 profiles.
- *Oracle Database Advanced Security Guide*

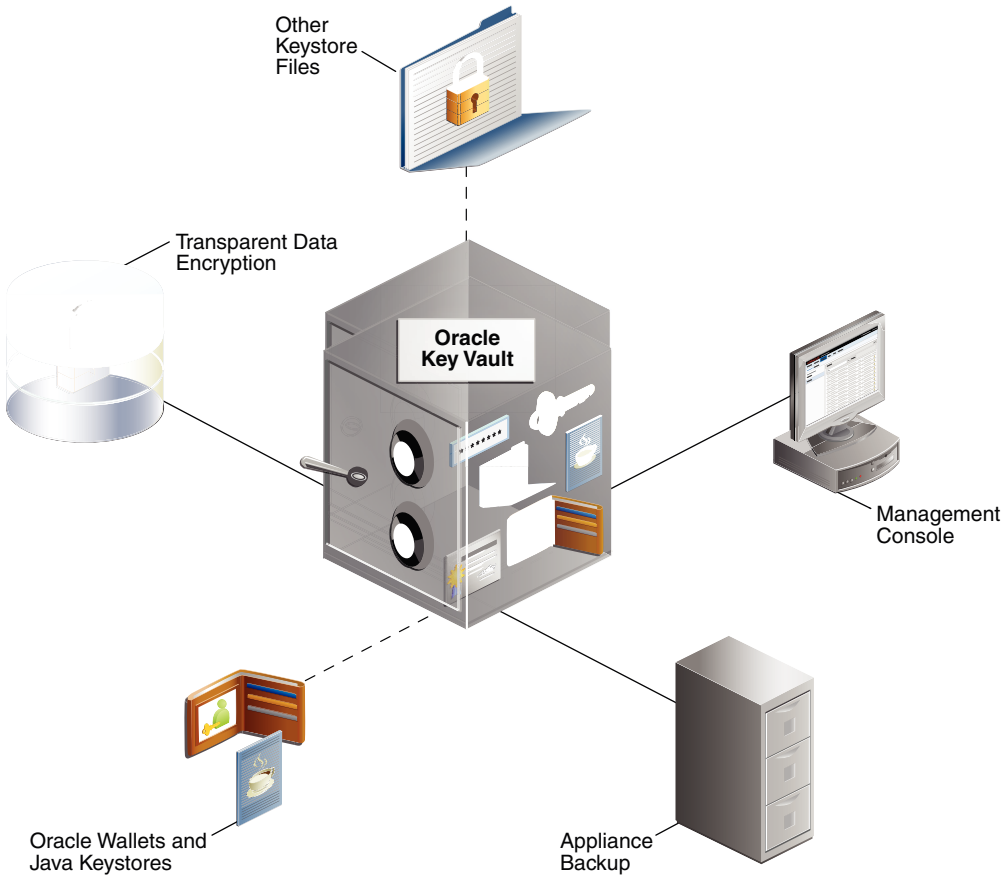
1.2 Benefits of Using Oracle Key Vault

Oracle Key Vault helps you to fight security threats, centralize key storage, and centralize key lifecycle management.

Deploying Oracle Key Vault in your organization will help you accomplish the following:

- Manage the lifecycle for endpoint security objects and keys, 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 between authorized endpoints across the organization.
- 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 Transparent Data Encryption (TDE), such as Oracle Real Application Clusters (Oracle RAC), Oracle Active Data Guard, pluggable databases, 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 keys and sensitive 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 keytab files, SSH key files, and server password files, that you upload to Oracle Key Vault from endpoints.
- **Oracle Key Vault Management Console** refers to the Oracle Key Vault graphical user interface, where you can log in to manage your security objects and administer the Oracle Key Vault system.
- **Oracle Key Vault Backup** refers to a backup device, where security objects in Oracle Key Vault can be backed up on-demand or on-schedule.

Oracle Key Vault multi-master cluster provides additional benefits, such as:

- Maximum key availability by providing multiple Oracle Key Vault nodes from which data may be retrieved
- Zero endpoint downtime during Oracle Key Vault multi-master cluster maintenance

1.3 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
  You can store security objects centrally in Oracle Key Vault, and manage them with automatic mechanisms for tracking, backup, and recovery.
- Centralized Management of TDE Master Encryption Keys Using Online Master Keys
  You can use an online master key to centralize the management of TDE master encryption keys over a direct network connection.
- 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.
- Online Management of Endpoint Keys and Secret Data
  You can use the RESTful key management interface to manage the storage and retrieval of keys.

1.3.1 Centralized Storage of Oracle Wallet Files and Java Keystores

You can store security objects centrally in Oracle Key Vault, and manage them with automatic mechanisms for tracking, backup, and recovery.

This will help you address 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 encryption keys), passwords (Secure External Password Store), and X.509 certificates (network encryption).

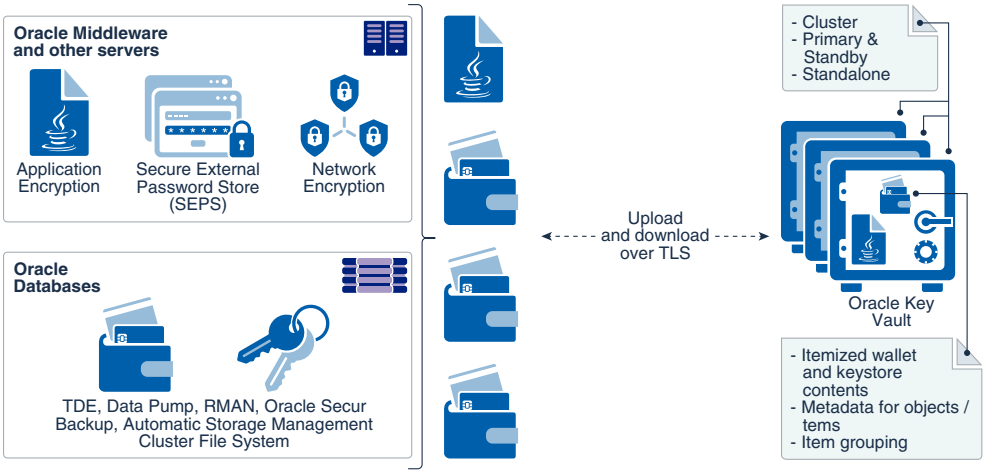
  Oracle Key Vault supports wallet files from all supported releases of the Oracle Database.
- Java keystores

  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 1-2 Centralized Storage of Oracle Wallet Files and Java Keystores



Related Topics

- Managing Secure External Password Store

1.3.2 Centralized Management of TDE Master Encryption Keys Using Online Master Keys

You can use an online master key to centralize the management of TDE master encryption keys over a direct network connection.

This feature applies only to Oracle databases that use Transparent Data Encryption (TDE). The term online master key replaces the previous term TDE direct connection.

Online master keys enable you to centrally manage Transparent Data Encryption (TDE) master encryption 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 encryption keys will be stored and managed in Oracle Key Vault. There are two scenarios that you can use:

- If the database does not yet have TDE wallets
- If the database has already been configured for TDE

The online master key feature works as follows: TDE generates the master encryption key and stores it in Oracle Key Vault. Oracle Key Vault administrators have full control of the TDE master encryption keys. They can revoke access of the keys from certain endpoints, share the keys with other endpoints, and perform other operations. The online master key is also a convenient alternative to copying local wallet files to multiple endpoints manually. Sharing TDE master encryption keys, rather than maintaining local wallet copies, is especially useful when TDE is running on database clusters such as Oracle Real Application Clusters (Oracle RAC) or Oracle Data Guard. The following comparison illustrates the difference:

- Local wallet copy

  In a Data Guard scenario, re-key operations on the primary database cause the managed recovery process on the standby databases to fail. You must copy the wallet to the standby database, and then an administrator must open the wallet (if the wallet is not an auto-login wallet). Afterward, you must restart the managed recovery process.

- Shared TDE key in a virtual wallet in Oracle Key Vault

  In a database cluster, after a key rotation operation, Oracle Key Vault immediately shares the new TDE master encryption key with other nodes in the cluster. There is no need to copy the wallet manually to the other nodes. In a Data Guard configuration, after key rotation, the new keys are immediately available to the standby databases, making the key management operations seamless.

Centralized management facilitates copying encrypted data between databases using Oracle Data Pump export, import, and the transportable tablespaces features of Oracle Database when master encryption 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 encryption keys are easily shared when you place them in a virtual wallet in Oracle Key Vault, and then grant each endpoint access to the virtual wallet.

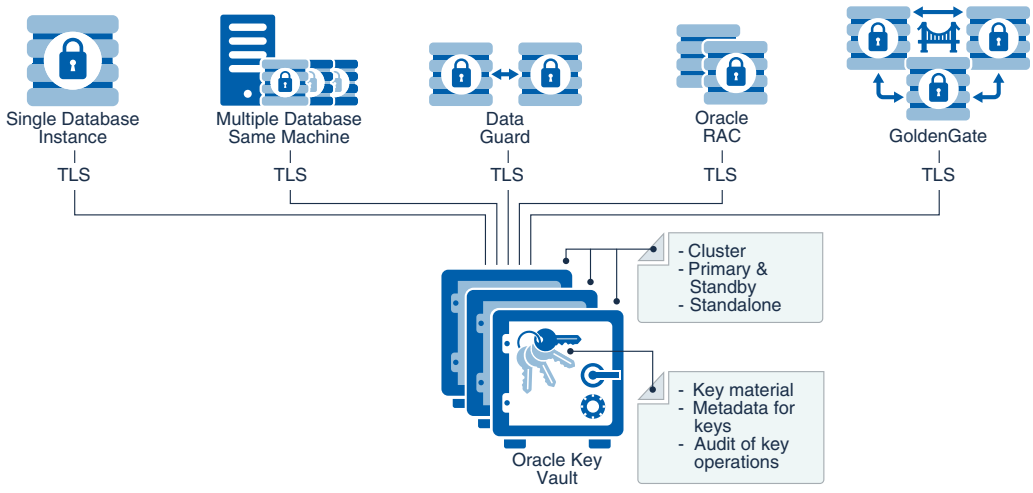
You must open the wallet before encryption and decryption. After you close the wallet, then encrypted data in tables and tablespaces is unavailable to you. You should rotate the TDE master encryption key regularly to remain in compliance with the applicable regulations.

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.

Online master keys are supported on Oracle Database 11g release 2 and later versions.

The following figure illustrates the centralized management of online master keys.

Figure 1-3 Centralized Management of Online Master Keys



Related Topics

- *Oracle Database Advanced Security Guide*

1.3.3 Storage of Credential Files

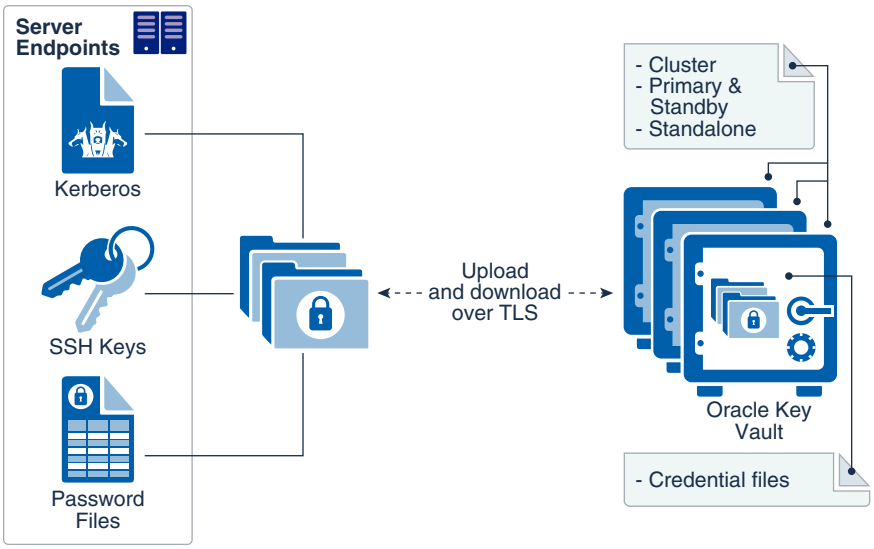
Oracle Key Vault can back up credential files other than Oracle wallets and Java keystores for long-term retention and recovery.

Oracle Key Vault does not interpret the actual content of a credential file. It simply 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 keytab files, and X.509 certificates.

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

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

Figure 1-4 Backing Up Credential Files



Related Topics

- Uploading and Downloading Credential Files
  The `okvutil upload` and `okvutil download` commands can upload and download credential files.

1.3.4 Online Management of Endpoint Keys and Secret Data

You can use the RESTful key management interface to manage the storage and retrieval of keys.

Applications, scripts, and third-party software can use the new interfaces to manage their keys and secrets in the Oracle Key Vault. They can retrieve the secrets or keys at run time and also generate and store new secrets or keys in Oracle Key Vault at run

time. All objects managed by the user or operations executed by the user using the RESTful utility have the same security and availability attributes and the same access control as those created by other Oracle Key Vault endpoint utilities such as `okvutil`.

1.4 Who Should Use Oracle Key Vault

Oracle Key Vault is designed 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.

1.5 Major Features of Oracle Key Vault

Oracle Key Vault enhances security in key management with a wide range of features that support different database deployments.

- Centralized Storage and Management of Security Objects
  You can store and manage security objects, such as TDE master encryption keys, wallets and keystores, and certificates, using Oracle Key Vault.
- Management of Key Lifecycle
  The management of the key lifecycle is critical for maintaining security and regulatory compliance, and consists of creation, backup, rotation, and expiration.
- Reporting and Alerts
  Oracle Key Vault provides reports and alerts to track system activity in depth.
- Separation of Duties for Oracle Key Vault Users
  Oracle Key Vault provides for separation of duties in the form of three console user roles: Key Administrator, System Administrator, and Audit Manager.
- Support for a Primary-Standby Environment
  To ensure that Oracle Key Vault can always access security objects, you can deploy Oracle Key Vault in a primary-standby (highly available) configuration.
- Persistent Master Encryption Key Cache
  The persistent master encryption key cache feature of the endpoint software enables databases to operate when the Oracle Key Vault server is unavailable.
- Backup and Restore Functionality for Security Objects
  Oracle Key Vault enables you to back up all security objects including keys, certificates, and passwords.
- Automation of Endpoint Enrollment Using RESTful Services
  The RESTful Services utility is an automation tool that enables you to quickly enroll and provision endpoints at scale.

- Key Management Support Using RESTful Services
  Oracle Key Vault extends RESTful services to enable key management at scale by providing a simplified interface to Key Management Interoperability Protocol (KMIP) operations.
- Support for OASIS Key Management Interoperability Protocol (KMIP)
  You can use Oracle Key Vault with a range of OASIS KMIP Version 1.1 profiles.
- Database Release and Platform Support
  Oracle Key Vault supports both full Oracle Database releases and bundle patches of Oracle Database.
- Integration with External Audit and Monitoring Services
  You can use Oracle Key Vault with Oracle Audit Vault and Database Firewall and Simple Network Management Protocol (SNMP).
- Integration of MySQL with Oracle Key Vault
  Oracle Key Vault can manage MySQL TDE encryption keys.
- Automatic Storage Management Cluster File System (ACFS) Encryption
  Oracle Key Vault supports key management for Automatic Storage Management (ASM) cluster file system (ACFS) encryption.
- Support for Oracle Cloud Database as a Service Endpoints
  An Oracle Key Vault on-premises server can manage Transparent Data Encryption (TDE) master encryption keys for Oracle Cloud Database as a Service instance.
- Oracle Key Vault Hardware Security Module Integration
  Oracle Key Vault can use a hardware security module (HSM) as a Roots of Trust (RoT) that protects encryption keys.

Related Topics

- Benefits of Oracle Key Vault Multi-Master Clustering
  The Oracle Key Vault multi-master cluster configuration addresses with regard to primary-standby environments.

1.5.1 Centralized Storage and Management of Security Objects

You can store and manage security objects, such as TDE master encryption keys, wallets and keystores, and certificates, using Oracle Key Vault.

- TDE master encryption keys

  For Oracle databases that use Transparent Data Encryption (TDE), Oracle Key Vault manages master encryption keys over a direct network connection using an online master encryption key 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 Real Application Clusters (Oracle RAC), Oracle Data Guard, or Oracle GoldenGate. You can easily migrate master encryption keys from Oracle wallets to Oracle Key Vault. Direct connections between TDE and Oracle Key Vault are supported for Oracle Database 11g release 2 and later.
- 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

being 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 wallets also facilitates the movement of encrypted data using Oracle Data Pump and the transportable tablespaces feature of Oracle Database, or when migrating (unplugging or plugging) a PDB. 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.

1.5.2 Management of Key Lifecycle

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

Oracle Key Vault provides mechanisms for facilitating periodic key rotations, backup, and recovery, which ensure that you can stay in regulatory compliance, unlike other systems that create keys and passwords. You can create policies to track the key lifecycle, and configure Oracle Key Vault to report key lifecycle changes as they happen. In this manner, 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.

1.5.3 Reporting and Alerts

Oracle Key Vault provides reports and alerts to track system activity in depth.

- Reports

  The Oracle Key Vault audit and management reports provide 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 that you want to receive. These include alerts for the expiration of keys, endpoint certificates, and user passwords, disk

utilization, system backup, and primary-standby events. You can choose to send alerts to syslog to allow for external monitoring.

1.5.4 Separation of Duties for Oracle Key Vault Users

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

Each user role possesses privileges for a type of task and may be assigned 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. You cannot retrieve security objects using the Oracle Key Vault management console.

Related Topics

- Administrative Roles within Oracle Key Vault
  Oracle Key Vault provides separation of duty compliant administrative roles that you can combine in various ways to meet enterprise needs.

1.5.5 Support for a Primary-Standby Environment

To ensure that Oracle Key Vault can always access security objects, you can deploy Oracle Key Vault in a primary-standby (highly available) configuration.

This configuration also supports disaster recovery scenarios.

You can deploy two Oracle Key Vault servers in a primary-standby configuration. The primary server services the requests that come from endpoints. If the primary server fails, then the standby server 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.

The primary-standby configuration was previously known as the high availability configuration. The primary-standby configuration and the multi-master cluster configuration are mutually exclusive.

Oracle Key Vault supports primary-standby read-only restricted mode. When the primary server is affected by server, hardware, or network failures, primary-standby read-only restricted mode ensures that an Oracle Key Vault server is available to service endpoints, thus ensuring operational continuity. However, key and sensitive operations, such as generation of keys are disabled, while operations such as generation of audit logs are unaffected.

When an unplanned shutdown makes the standby server unreachable, the primary server is still available to the endpoints in read-only mode.

Related Topics

- About the Oracle Key Vault Primary-Standby Configuration
  You configure a primary-standby environment by providing the primary and standby servers with each other's IP address and certificate, and then pairing them.

1.5.6 Persistent Master Encryption Key Cache

The persistent master encryption key cache feature of the endpoint software enables databases to operate when the Oracle Key Vault server is unavailable.

The TDE master encryption key is cached in the persistent master encryption key cache in addition to the in-memory cache, to make the master encryption 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 start-up operations.

The persistent master encryption key cache is not necessary in a multi-master cluster deployment. It is primarily used for standalone or primary-standby Oracle Key Vault deployments.

Related Topics

- Using the Persistent Master Encryption Key Cache
  The persistent master encryption key cache feature enables databases to be operational when the Oracle Key Vault server is unavailable.

1.5.7 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 server failure, because you can restore a new Oracle Key Vault server 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:

- Managing incremental and full backups
- Creating, deleting, and modifying remote backup locations
- Setting up, modifying, or disabling the current backup schedule
- Initiating an immediate one-time backup
- Scheduling a future one-time backup

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

Related Topics

- Backup and Restore Operations
  You may configure automatic backups for continuous, reliable, and protected access to security objects with minimum downtime.

1.5.8 Automation of Endpoint Enrollment Using RESTful Services

The RESTful Services utility is an automation tool that enables you to quickly enroll and provision endpoints 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 systems, who might need to enroll and provision many hundreds of endpoints simultaneously using the protective security measures of RESTful services.

Related Topics

- Oracle Key Vault Administration and Key Management with RESTful Services
  The Oracle Key Vault RESTful Services utility automates Oracle Key Vault administration tasks for a large distributed deployment.

1.5.9 Key Management Support Using RESTful Services

Oracle Key Vault extends RESTful services to enable key management at scale by providing a simplified interface to Key Management Interoperability Protocol (KMIP) operations.

The KMIP REST tool allows operations on managed objects such as keys, certificates, and other objects in a simple manner without complicated client side development. The KMIP REST tool also provides the ability to script or automate most key management functions.

Related Topics

- Oracle Key Vault Administration and Key Management with RESTful Services
  The Oracle Key Vault RESTful Services utility automates Oracle Key Vault administration tasks for a large distributed deployment.

1.5.10 Support for OASIS Key Management Interoperability Protocol (KMIP)

You can use Oracle Key Vault with a range of OASIS KMIP Version 1.1 profiles.

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.

Related Topics

- Key Management Interoperability Protocol Specification Version 1.1

1.5.11 Database Release and Platform Support

Oracle Key Vault supports both full Oracle Database releases and bundle patches of Oracle Database.

Oracle Key Vault supports Oracle Database releases 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 the bundle patches of Oracle Database release 11g release 2 and 12c release 1 (12.1.0.2) and later on Windows Server 2012.

Related Topics

- Oracle Key Vault Installation Requirements
  The Oracle Key Vault installation requirements cover system requirements such as CPU, memory, disk space, network interfaces, and supported endpoint platforms.

1.5.12 Integration with External Audit and Monitoring Services

You can use Oracle Key Vault with Oracle Audit Vault and Database Firewall and Simple Network Management Protocol (SNMP).

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 version 3 to monitor the health and availability of the system.

1.5.13 Integration of MySQL 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
  You can manage TDE encryption keys in MySQL with Oracle Key Vault.

1.5.14 Automatic Storage Management Cluster File System (ACFS) Encryption

Oracle Key Vault supports key management for Automatic Storage Management (ASM) cluster file system (ACFS) encryption.

1.5.15 Support for Oracle Cloud Database as a Service Endpoints

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

Related Topics

- Oracle Database Instances in Oracle Cloud Infrastructure
  Oracle Key Vault deployed on-premises can manage the TDE master encryption keys for Oracle Database instances running in Oracle Cloud Infrastructure (OCI).

1.5.16 Oracle Key Vault Hardware Security Module Integration

Oracle Key Vault can use a hardware security module (HSM) as a Roots of Trust (RoT) that protects encryption keys.

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, you can use HSMs in FIPS 140-2 Level 3 mode which can help meet certain compliance requirements.

Related Topics

- Why HSM?

1.6 Oracle Key Vault Interfaces

Oracle Key Vault provides both a graphical user interface and command-line interfaces.

- Oracle Key Vault Management Console
  The Oracle Key Vault management console is a browser-based graphical user interface that Key Vault administrators use to perform day-to-day tasks.
- Oracle Key Vault okvutil Endpoint Utility
  Endpoint administrators can use the `okvutil` command-line utility to upload and download security objects between Oracle Key Vault and endpoints.
- Oracle Key Vault RESTful Services
  You can use the Oracle Key Vault RESTful Services utility to automate processes for a large distributed enterprise deployment.

1.6.1 Oracle Key Vault Management Console

The Oracle Key Vault management console is a browser-based graphical user interface that Key Vault administrators use to perform day-to-day tasks.

It enables Oracle Key Vault administrators to manage keys and sensitive objects, wallets, endpoints, and users. The console can also configure settings for individual Oracle Key Vault servers, as well as multi-master clusters, primary-standby environments, backup, and recovery.

Related Topics

- Logging In to the Oracle Key Vault Management Console
  To use Oracle Key Vault, you can log in to the Oracle Key Vault management console.

1.6.2 Oracle Key Vault okvutil Endpoint Utility

Endpoint administrators can use the `okvutil` command-line utility 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.

Related Topics

- Oracle Key Vault okvutil Endpoint Utility Reference
  The `okvutil` utility enables you to perform tasks uploading and downloading security objects.

1.6.3 Oracle Key Vault RESTful Services

You can use the Oracle Key Vault RESTful Services utility to automate processes for a large distributed enterprise deployment.

This utility enables you to automate endpoint enrollment, virtual wallet management, and key management.

Related Topics

- Oracle Key Vault Administration and Key Management with RESTful Services
  The Oracle Key Vault RESTful Services utility automates Oracle Key Vault administration tasks for a large distributed deployment.

1.7 Overview of an Oracle Key Vault Deployment

There are three different Oracle Key Vault deployment options.

- A standalone deployment is simplest to deploy. However, it does not provide continuous availability of the key service in the event an Oracle Key Vault server becomes unavailable.
- A primary-standby configuration enables the deployment of a second Oracle Key Vault server as a passive standby to an active primary server. If a primary server becomes unavailable, then the standby server becomes the new primary Oracle Key Vault server to service read and write requests from the endpoints. This is the high availability mode that was introduced in Oracle Key Vault release 12.2.
- A multi-master cluster configuration allows for up to 16 nodes and is recommended for deployments requiring high availability.

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

1. Understand important concepts described in Oracle Key Vault Concepts and Oracle Key Vault Multi-Master Cluster Concepts.
2. Install and configure Oracle Key Vault as described in Oracle Key Vault Installation and Configuration.

3. Create a multi-master cluster as an alternative for a primary-standby configuration by adding up to 16 Oracle Key Vault servers for maximum redundancy and reliability. This is described in Managing Oracle Key Vault Multi-Master Clusters.

   You must have a separate license for each Oracle Key Vault server installation in a multi-master cluster environment.

4. Create a primary-standby configuration by adding a second Oracle Key Vault server. Enable primary-standby read-only restricted mode to ensure operational continuity of the endpoints. This is described in Managing an Oracle Key Vault Primary-Standby Configuration.

   You must have a separate license for each Oracle Key Vault server installation in a primary-standby environment.

5. Create users to manage the day-to-day tasks for Oracle Key Vault as described in Managing Oracle Key Vault Users .

6. Register endpoints so that they can use Oracle Key Vault to store and manage their security objects described in Managing Oracle Key Vault Endpoints.

7. Register endpoints in the cloud described in Oracle Database Instances in Oracle Cloud Infrastructure.

8. Enroll endpoints so that you can upload or download security objects between the endpoints and Oracle Key Vault described in Enrolling Endpoints for Oracle Key Vault .

9. Upload or add virtual wallets to Oracle Key Vault described in Managing Security Objects in Oracle Key Vault.

10. Use automating endpoint enrollment and provisioning for large-scale deployments in Oracle Key Vault Administration and Key Management with RESTful Services .

11. Read about using Oracle Key Vault with other features, such as Oracle GoldenGate, in Using Oracle Key Vault with Other Features.

12. Automate key management to perform online key management with other software using RESTful services, as described in Oracle Key Vault Administrative REST Client Tool Commands.

13. Learn how to perform periodic maintenance tasks such as administering and monitoring the system, as described in Oracle Key Vault General System Administration.

14. Learn how to monitor Oracle Key Vault by performing tasks such as creating alerts, as described in Monitoring and Auditing Oracle Key Vault.

2

Oracle Key Vault Concepts

To successfully deploy Oracle Key Vault, you must understand the deployment architecture, use cases, access control, administrative roles, and endpoints.

- Overview of Oracle Key Vault Concepts
  Endpoints are computer systems such as database and application servers, and other information systems where keys and credentials access data.
- Oracle Key Vault Deployment Architecture
  Oracle Key Vault is packaged as a software appliance preconfigured with an operating system, a database, and the Oracle Key Vault application.
- Access Control Configuration
  Oracle Key Vault enables you to control access to security objects at various access levels and time intervals.
- Administrative Roles within Oracle Key Vault
  Oracle Key Vault provides separation of duty compliant administrative roles that you can combine in various ways to meet enterprise needs.
- 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.
- Root and Support User Accounts
  Both the `root` and `support` user accounts are used with the command-line interface.
- Endpoint Administrators
  An endpoint administrator owns and manages endpoints, which are entities such as Oracle databases that use Oracle Key Vault.
- FIPS Mode
  FIPS mode enables Oracle Key Vault to adhere to FIPS 140-2 compliance.

2.1 Overview of Oracle Key Vault Concepts

Endpoints are computer systems such as database and application servers, and other information systems where keys and credentials access data.

These endpoint systems must store and manage their encryption keys and secrets 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.

You must register and enroll an endpoint so that it can 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.

You can group security objects such as master encryption keys and credential files 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. A privileged user can create a virtual wallet, add keys to the empty wallet, and then grant other users, endpoints, user groups, and endpoint groups various levels of access to the wallet. A user must have access to security objects before he or she can grant access on those same security objects to other users. The access level they grant can be equal to or less than 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 that 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 Oracle Key Vault, you can specify a default wallet for the endpoint. The default wallet ensures that endpoints are associated with a virtual wallet where the keys will be uploaded if no virtual wallet is specified at the time of wallet or key upload.

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 automatically audits all actions performed by users and endpoints.

2.2 Oracle Key Vault Deployment Architecture

Oracle Key Vault is packaged as a software appliance preconfigured with an operating system, a database, and the Oracle Key Vault application.

This way, you do not have to install and configure individual components. It is hardened for security according to operating system and database hardening best practices. The installation process does not include any unnecessary packages and software, and it enables only required ports and services.

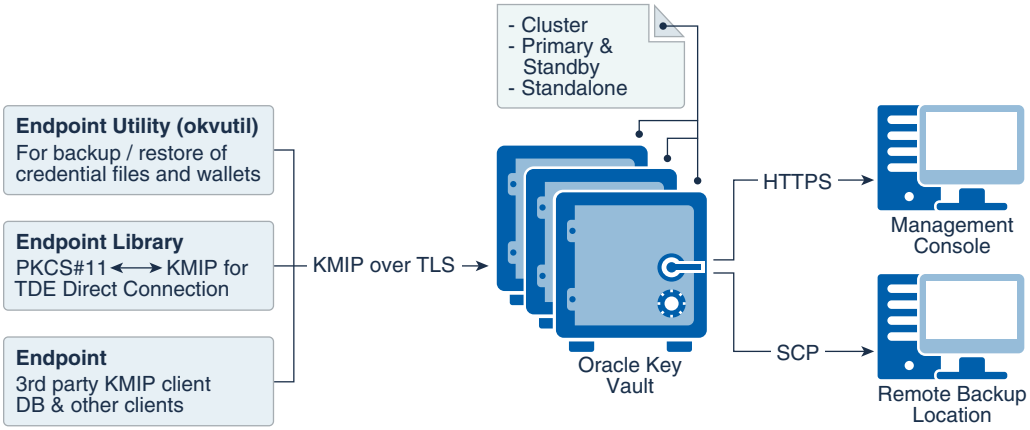
The endpoints communicate with Oracle Key Vault over a mutually authenticated Transport Layer Security (TLS) connection using the OASIS Key Management Interoperability Protocol (KMIP).

The Oracle Key Vault multi-master cluster configuration can contain up to 16 nodes, two of which must be read-write nodes with the remaining being a combination of either read-write pairs or read-only nodes. The multi-master configuration provides several benefits over a primary-standby configuration. The multi-master configuration and the primary-standby configuration are mutually exclusive.

The Oracle Key Vault primary-standby configuration defines one primary server and one standby server. The primary server is active and services requests from endpoints. If the primary fails to communicate with the standby for a time exceeding a configured time threshold, then the standby server takes over as primary. Communication related to data replication between the primary and standby servers is a mutually authenticated TLS connection. This was referred to as the primary-standby option (previously called high availability) in previous releases of Oracle Key Vault.

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

Figure 2-1 Oracle Key Vault Deployment Architecture



For multiple geographically distributed data centers with high load and extreme availability requirements, you should deploy Oracle Key Vault in a multi-master cluster configuration. The read-write pairs should span data centers. For single data centers where data does not leave the data center, consider using a classic primary-standby deployment of Oracle Key Vault with a periodic backup. A standalone deployment of Oracle Key Vault is useful for testing and development environments.

Related Topics

- Oracle Key Vault Multi-Master Cluster Overview
  The multi-master cluster nodes provide high availability, disaster recovery, load distribution, and geographic distribution to an Oracle Key Vault environment.
- Benefits of an Oracle Key Vault Primary-Standby Configuration
  The benefits of an Oracle Key Vault primary-standby configuration include high availability, necessary for business-critical operations.

2.3 Access Control Configuration

Oracle Key Vault enables you to control access to security objects at various access levels and time intervals.

- 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 Grants
  You can grant access to virtual wallets directly or indirectly.
- Access Control Options
  Access control options enable you to set the type of privileges that users have to read, write, and delete security objects.

2.3.1 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.

You can set access control on security objects individually, or collectively when you group them into a virtual wallet. Oracle Key Vault uses a virtual wallet to share a set of security objects with others. 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 users or endpoints individually, you can collectively grant access by using user groups or endpoint 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 .

2.3.2 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 through a 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.

From the Oracle Key Vault management console, you can grant access mappings on a virtual wallet in the following 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.

Related Topics

- Granting Access to Users, User Groups, Endpoints, and Endpoint Groups
  You can grant the **Read Only**, **Read and Modify**, and **Manage Wallet** access levels to users, user groups, endpoints, and endpoint groups.
- Granting an Endpoint Access to a Virtual Wallet
  An endpoint must have **Read and Modify** and **Manage Wallet** privileges on the wallet before security objects can be uploaded or downloaded.
- Granting a User Group Access to a Virtual Wallet
  You can modify the access level to a virtual wallet for a user group as functional needs change.

2.3.3 Access Control Options

Access control options enable you to set the type of privileges that users have to read, write, and delete security objects.

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:
  - Adding or removing 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.
  - Granting others access to the wallet
  - Modifying wallet settings, such as its description
  - Deleting the wallet

2.4 Administrative Roles within Oracle Key Vault

Oracle Key Vault provides separation of duty compliant administrative roles that you can combine in various ways to meet enterprise needs.

- About Administrative Roles in Oracle Key Vault
  Oracle Key Vault provides three administrative roles: System Administrator, Key Administrator, and Audit Manager.
- Separation of Duties in Oracle Key Vault
  When you grant the Oracle Key Vault roles to users, ensure that you adhere to separation of duty guidelines.
- System Administrator Role Duties
  The Oracle Key Vault System Administrator is responsible for general system-related tasks.
- Key Administrator Role Duties
  The Oracle Key Vault Key Administrator is responsible for managing security objects.
- Audit Manager Role Duties
  The Oracle Key Vault Audit Manager is responsible for audit-related tasks.

2.4.1 About Administrative Roles in Oracle Key Vault

Oracle Key Vault provides three administrative roles: System Administrator, Key Administrator, and Audit Manager.

- **System Administrator role** provides privileges for creating and managing users, creating and managing endpoints, configuring system settings and alerts, and generally administering Oracle Key Vault. This is the most powerful role.
- **Key Administrator role** provides privileges for managing the key life cycle and controlling access to all security objects in Oracle Key Vault.
- **Audit Manager role** provides privileges for managing the audit life cycle and audit policies.

These roles are designed to be flexible to support various organizational needs and structures. Administrative users can grant their roles to other users, but non-administrative users cannot. If one administrative user is performing two administrative functions, then that user will have two roles in Oracle 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, then he or she can grant another user both those roles or just one, depending on the needs of the organization.

One of the post-installation tasks is to create three administrative users for the three roles. The installation process also prompts you to create a recovery passphrase. 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 Oracle Key Vault.

You can enable users who have no specific administrative role to be responsible for a specific area, such as a virtual wallet. You can grant these users 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 manner, access to security objects is controlled, yet flexible to meet the evolving needs of the enterprise.

When you use the management console interface, your access to the various tabs, menus, and actions depends on your role and the objects that you have access to.

2.4.2 Separation of Duties in Oracle Key Vault

When you grant the Oracle Key Vault roles to users, ensure that you adhere to separation of duty guidelines.

Oracle Key Vault users can be assigned the three administrative roles by function, so there is a clear separation of duties between the System Administrator, Key Administrator, and the Audit Manager. You also can create users that have no administrative privileges.

In a strict separation of duties environment, 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.

You can achieve a separation of duties in two ways:

- Grant each person who has been granted one of the three roles the appropriate privileges for the functional area for which they are responsible. For example, grant the System Administrator privileges for system-related tasks, the Key Administrator user privileges to manage encryption keys (such as the `SYSKM` administrative privilege for Transparent Data Encryption), and the Audit Manager privileges such as the `AUDIT_ADMIN` role for Oracle Database unified auditing policies.
- 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 do not need to 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 unique user account and password. Because these roles are powerful, also ensure that these users are trustworthy and that they are knowledgeable about the areas that they manage.

2.4.3 System Administrator Role Duties

The Oracle Key Vault System Administrator is responsible for general system-related tasks.

- Creating and managing users
- Adding and managing endpoints
- Setting up the primary-standby servers
- Configuring alerts and key rotation reminders
- Scheduling backups
- Starting and stopping Oracle Key Vault
- Configuring SMTP server settings for email notification
- Enabling or disabling FIPS mode
- Configuring Oracle Key Vault to use a hardware security module
- Configuring SNMP for remote monitoring
- Enabling automated endpoint enrollment and key management through RESTful Services
- Enabling audit consolidation with Audit Vault Database Firewall
- Creating SSH tunnels for Oracle Cloud Database as a Service endpoints
- Setting up a cluster
- Managing and monitoring a cluster
- Managing the cluster configuration
- Granting the System Administrator role to other users

2.4.4 Key Administrator Role Duties

The Oracle Key Vault Key Administrator is responsible for managing security objects.

- Managing the lifecycle of security objects
- Having full access on all virtual wallets and security objects
- Controlling access to virtual wallets for users, endpoints, user groups, and endpoint groups
- Granting the Key Administrator role to other users

2.4.5 Audit Manager Role Duties

The Oracle Key Vault Audit Manager is responsible for audit-related tasks.

- Managing the audit trail as the only user who has privileges to export or delete Oracle Key Vault audit records
- Having read access on all security objects
- Managing audit settings
- Granting the Audit Manager role to other users

2.5 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 such as forgotten passwords.

The recovery passphrase is needed in the following situations:

- If there is no administrative user available to log into Oracle Key Vault, then you can use the recovery passphrase to repeat the post-installation tasks and create new administrative users for system, key, and audit management.
- If you want to restore Oracle Key Vault from a previous backup, then you must have the recovery passphrase that is associated with that backup.
- You will be prompted for the recovery passphrase during the node induction process in the cluster mode.
- You will need it if you want to reset the recovery passphrase periodically.

For these reasons, it is very important to store the recovery passphrase in a safe and accessible place and keep track of older recovery passphrases. The recovery passphrase is the same passphrase that you use when you add nodes to a multi-master cluster.

The only way to recover from a lost recovery passphrase is to reinstall Oracle Key Vault.

Related Topics

- Managing System Recovery
  System recovery includes tasks such as recovering lost administrative passwords.
- Restoring Oracle Key Vault Data
  Oracle Key Vault data from a remote backup destination can be restored onto a another Oracle Key Vault server.
- Performing Post-Installation Tasks
  After you install Oracle Key Vault, you must complete a set of post-installation tasks.

2.6 Root and Support User Accounts

Both the `root` and `support` user accounts are used with the command-line interface.

The `root` user account is the super user account for the operating system that hosts Oracle Key Vault. You do not need the `root` account for normal Oracle Key Vault administration. Instead, you must use the `root` account when you want to upgrade to a later bundle patch or perform some command-line operations such as adding disk space. The `support` user is the only account that can remotely log in to the operating system hosting the Oracle Key Vault when SSH is enabled.

Be aware that if you enter the `root` or `support` passwords incorrectly three times, then the account is locked for 15 minutes.

2.7 Endpoint Administrators

An endpoint administrator owns and manages endpoints, which are entities such as Oracle databases that use Oracle Key Vault.

This user is typically a system, security, or database administrator, but 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.

2.8 FIPS Mode

FIPS mode enables Oracle Key Vault to adhere to FIPS 140-2 compliance.

Federal Information Processing Standard (FIPS) publications are issued by the National Institute of Standards and Technology (NIST). The publication entitled *Security Requirements for Cryptographic Modules* (FIPS PUB 140-2) specifies the security requirements over several key areas that will be satisfied by a cryptographic module utilized within a security system protecting sensitive but unclassified information.

The Oracle Key Vault is FIPS 140–2 compliant. Selecting the option to install with FIPS 140–2 compliance performs all required changes during the installation. No additional modifications are necessary after the installation. You can also enable FIPS 140-2 compliance after the installation if it was not done during the initial installation of Oracle Key Vault. Enabling or disabling FIPS mode requires you to restart Oracle Key Vault.

Related Topics

-
- Configuring the FIPS Mode for the Node
  All multi-master cluster nodes must use the same FIPS mode setting or you will receive an alert.
- FIPS 140-2 Compliance in Oracle Linux
- *Oracle Database Security Guide*
- *Oracle Fusion Middleware Administering Oracle GoldenGate*

3

Oracle Key Vault Multi-Master Cluster Concepts

A multi-master cluster is a fully connected network of Oracle Key Vault servers called nodes.

- Oracle Key Vault Multi-Master Cluster Overview
  The multi-master cluster nodes provide high availability, disaster recovery, load distribution, and geographic distribution to an Oracle Key Vault environment.
- Benefits of Oracle Key Vault Multi-Master Clustering
  The Oracle Key Vault multi-master cluster configuration addresses with regard to primary-standby environments.
- Multi-Master Cluster Architecture
  An Oracle Key Vault node can be a read-write or a read-only node operating in different modes. Nodes can also form a subgroup.
- Building and Managing a Multi-Master Cluster
  You initialize a multi-master cluster using a single Oracle Key Vault server.
- Oracle Key Vault Multi-Master Cluster Deployment Scenarios
  All multi-master cluster nodes can serve endpoints actively and independently.
- Multi-Master Cluster Features
  Oracle Key Vault provides features that help with inconsistency resolution and name conflict resolution in clusters, and endpoint node scan lists.

3.1 Oracle Key Vault Multi-Master Cluster Overview

The multi-master cluster nodes provide high availability, disaster recovery, load distribution, and geographic distribution to an Oracle Key Vault environment.

An Oracle Key Vault multi-master cluster provides a mechanism to create read-write pairs of Oracle Key Vault nodes for maximum availability and reliability. You can add read-only Oracle Key Vault nodes to the cluster to provide even greater availability to endpoints that need Oracle wallets, encryption keys, Java keystores, certificates, credential files, and other objects.

An Oracle Key Vault multi-master cluster is an interconnected group of Oracle Key Vault nodes. Each node in the cluster is automatically configured to connect with all the other nodes, in a fully connected network. The nodes can be geographically distributed. Oracle Key Vault endpoints interact with any node in the cluster.

This configuration replicates data to all other nodes, reducing risk of data loss. To prevent data loss, you must configure pairs of nodes called read-write pairs to enable bi-directional synchronous replication. This configuration enables an update to one node to be replicated to the other node, and verifies this on the other node, before the update is considered successful. Critical data can only be added or updated within the read-write pairs. All added or updated data is asynchronously replicated to the rest of the cluster.

After you have completed the upgrade process, every node in the Oracle Key Vault cluster must be at Oracle Key Vault release 18.1 or later, and within one release update of all other nodes. Any new Oracle Key Vault server that is to join the cluster must be at the same release level as the cluster.

The clocks on all the nodes of the cluster must be synchronized. Consequently, all nodes of the cluster must have the Network Time Protocol (NTP) settings enabled.

Every node in the cluster can serve endpoints actively and independently while maintaining an identical dataset through continuous replication across the cluster. The smallest possible configuration is a two node cluster, and the largest configuration can have up to 16 nodes with several pairs spread across several data centers.

3.2 Benefits of Oracle Key Vault Multi-Master Clustering

The Oracle Key Vault multi-master cluster configuration addresses with regard to primary-standby environments.

To ensure high availability for geographically distributed endpoints, Oracle Key Vault nodes that are deployed in different data centers operate in active-active multi-master cluster configurations to create and share keys. With an active-active configuration, there are no passive machines in the cluster, which allows for better resource utilization. An added benefit of the multi-master cluster configuration is load distribution. When multiple Oracle Key Vault nodes in multi-master configuration are deployed in a data center, they can actively share the key requests of the endpoint databases in that data center.

In a typical large scale deployment, Oracle Key Vault must serve a large number of endpoints, possibly distributed in geographically distant data centers.

In comparison to a multi-master deployment, standalone Oracle Key Vault deployments provide the least availability, while primary-standby deployments offer limited availability:

- A primary-standby configuration only has a single primary Oracle Key Vault server that can actively serve clients.
- If the server running in the standby role is unavailable, then the server running in the primary role is in read-only mode and does not allow any write operations.
- The primary-standby mode can support either high availability in the same data center or disaster recovery across data centers.
- If the persistent master encryption key cache is not enabled, then database downtime is unavoidable during maintenance windows.

The Oracle Key Vault multi-master cluster configuration addresses these limitations. You can geographically disperse nodes to provide simultaneous high availability and disaster recovery capability.

An Oracle Key Vault multi-master cluster configuration offers significant benefits as follows:

- Data compatibility between multi-master cluster nodes similar to a primary-standby deployment

  Because all the nodes have an identical data set, the endpoints can retrieve information from any node. In a cluster, the unavailability of an Oracle Key Vault node does not affect the operations of an endpoint. If a given node is unavailable, then the endpoint interacts transparently with another node in the cluster.

- Fault tolerance

  Successfully enrolled clients transparently update their own list of available Oracle Key Vault nodes in the cluster. This enables clients to locate available nodes at any given time, without additional intervention. As such, unexpected failure in nodes or network disruptions do not lead to service interruption for endpoints as long as at least one operational Oracle Key Vault read-write pair remains accessible to the endpoint. If all read-write pairs are unavailable to an endpoint, but a read-only restricted node is available, then the endpoint can still invoke read-only operations.

- Zero data loss

  Data that has been added or updated at a read-write node is immediately replicated to its read-write peer and must be confirmed at the peer to be considered committed. It is then distributed across the cluster. Therefore, data updates are considered successful only if they are guaranteed to exist in multiple servers.

- No passive machines in the system

  A primary-standby configuration requires a passive standby server. The Oracle Key Vault multi-master cluster contains only active servers. This allows for better utilization of hardware.

- Scaling up and scaling down

  You can add extra Oracle Key Vault nodes to the cluster or remove existing nodes from the cluster without interrupting the overall Oracle Key Vault services to clients. This means the number of nodes in the cluster can be increased or decreased as required to meet the expected workload.

- Maintenance

  Whenever hardware or software maintenance is required, Oracle Key Vault nodes can leave the cluster and return back to the cluster after maintenance. The remaining nodes continue to serve the clients. Properly planned maintenance does not cause any service downtime, avoiding interruption of service to endpoints.

3.3 Multi-Master Cluster Architecture

An Oracle Key Vault node can be a read-write or a read-only node operating in different modes. Nodes can also form a subgroup.

- Oracle Key Vault Cluster Nodes
  An Oracle Key Vault node is an Oracle Key Vault server that operates as a member of a multi-master cluster.
- Cluster Node Limitations
  Limitations to cluster nodes depend on whether the node is asynchronous or synchronous.
- Cluster Subgroup
  A cluster subgroup is a group of one or more nodes of the cluster.
- Critical Data in Oracle Key Vault
  Oracle Key Vault stores critical data in Oracle Key Vault that is necessary for the endpoints to operate.

- Oracle Key Vault Read-Write Nodes
  A read-write node is a node in which critical data can be added or updated using the Oracle Key Vault or endpoint software.
- Oracle Key Vault Read-Only Nodes
  In a read-only node, users can add or update non-critical data but not add or update critical data.
- Cluster Node Mode Types
  Oracle Key Vault supports two types of mode for cluster nodes: read-only restricted mode or read-write mode.
- Operations Permitted on Cluster Nodes in Different Modes
  In an Oracle Key Vault multi-master cluster, operations are available or restricted based on the node and the operating mode of the node.

3.3.1 Oracle Key Vault Cluster Nodes

An Oracle Key Vault node is an Oracle Key Vault server that operates as a member of a multi-master cluster.

To configure an Oracle Key Vault server to operate as a member of the cluster, you must convert it to be a multi-master cluster node. The process is referred to as node induction. You initiate induction on the Cluster Management page of the Oracle Key Vault management console.

On induction, Oracle Key Vault modifies the **Cluster** tab to enable management, monitoring, and conflict resolution capabilities on the management console of the node. Cluster-specific features of the management console, such as cluster settings, audit replication, naming resolution, cluster alerts, and so on, are enabled as well.

The Primary-Standby Configuration page is not available on the Oracle Key Vault management console of a node. A node cannot have a passive standby server.

A node runs additional services to enable it to communicate with the other nodes of the cluster. Endpoints enrolled from a node are made aware of the cluster topology.

Each node in the cluster has a user-allocated node identifier. The node identifier must be unique in the cluster.

3.3.2 Cluster Node Limitations

Limitations to cluster nodes depend on whether the node is asynchronous or synchronous.

The nodes of a Oracle Key Vault multi-master cluster replicate data asynchronously between them. The only exception is replicating data to the read-write peer. There are various limitations arising as a result of the asynchronous replicate operations.

The IP addresses of a node in the cluster are static and cannot be changed after the node joins the cluster. If you want the node to have a different IP address, then delete the node from the cluster, and either add a new node with the correct IP address, or re-image the deleted node using the correct IP address before adding it back to the cluster.

You can perform only one cluster change operation (such as adding, disabling, or deleting a node) at a time.

Node IDs are unique across the cluster. You must ensure that the node ID is unique when you select the node during the induction process.

An Oracle Key Vault cluster does a best effort job at preventing users from performing nonsupported actions such as trying to remove a node ID that a controller node must have access to during the induction process. Similarly, a multi-master cluster will prevent the user from adding a second read-write peer if one already exists.

3.3.3 Cluster Subgroup

A cluster subgroup is a group of one or more nodes of the cluster.

A cluster can be conceptually divided into one or more cluster subgroups.

The node is assigned to a subgroup when you add the node to the multi-master cluster. The assignment cannot be changed for the life of the node. A node's cluster subgroup assignment is a property of the individual node, and members of a read-write pair may be in different cluster subgroups.

A cluster subgroup represents the notion of endpoint affinity. A node's cluster subgroup assignment is used to set the search order in the endpoint's node scan list. Nodes in the same cluster subgroup as the node where the endpoint was added are considered local to the endpoint. The nodes within an endpoint's local subgroup are scanned first, before communicating with nodes that are not in the local subgroup.

The cluster topology can change when you add or remove new nodes to and from the cluster. Nodes can also be added or removed from the local cluster subgroup. Each endpoint gets updates to this information along with the response message for any operation which the endpoint initiated. The updated endpoint's node scan list is sent back to the endpoint periodically even if there is no change to cluster topology. This is to make up for any lost messages.

3.3.4 Critical Data in Oracle Key Vault

Oracle Key Vault stores critical data in Oracle Key Vault that is necessary for the endpoints to operate.

The loss of this information can result in the loss of data on the endpoint. Endpoint encryption keys, certificates, and similar security objects that Oracle Key Vault manages are examples of critical data in Oracle Key Vault. Critical data must be preserved in the event of an Oracle Key Vault server failure to ensure endpoint recovery and continued operations.

Oracle Key Vault data that can be re-created or discarded after an Oracle Key Vault server failure is non-critical data. Cluster configuration settings, alert settings, email settings, and key sharing between virtual wallets are examples of non-critical data.

3.3.5 Oracle Key Vault Read-Write Nodes

A read-write node is a node in which critical data can be added or updated using the Oracle Key Vault or endpoint software.

The critical data that is added or updated can be data such as keys, wallet contents, and certificates.

Oracle Key Vault read-write nodes always exist in pairs. Each node in the read-write pair can accept updates to critical and non-critical data, and these updates are

immediately replicated to the other member of the pair, the read-write peer. A read-write peer is the specific member of one, and only one, read-write pair in the cluster. There is bi-directional synchronous replication between read-write peers. Replication to all nodes that are not a given node's read-write peer is asynchronous.

A node can be a member of, at most, one read-write pair. A node can have only one read-write peer. A node becomes a member of a read-write pair, and therefore a read-write node, during the induction process. A read-write node reverts to being a read-only node when its read-write peer is deleted, at which time it can form a new read-write pair.

A read-write node operates in read-write mode when it can successfully replicate to its read-write peer and when both peers are active. A read-write node is temporarily placed in read-only restricted mode when it is unable to replicate to its read-write peer or when its read-write peer is disabled.

Oracle Key Vault multi-master cluster requires at least one read-write pair to be fully operational. It can have a maximum of 8 read-write pairs.

3.3.6 Oracle Key Vault Read-Only Nodes

In a read-only node, users can add or update non-critical data but not add or update critical data.

Critical data is updated only through replication from other nodes.

A read-only node is not a member of a read-write pair and does not have an active read-write peer.

A read-only node can induct a new server into a multi-master cluster. The new node can be another read-only node. However, a read-only node becomes a read-write node if it inducts another node as its read-write peer.

The first node in the cluster is a read-only node. Read-only nodes are used to expand the cluster. A multi-master cluster, after it has been built, does not need to have any read-only nodes. A multi-master cluster with only read-only nodes is not ideal because no useful critical data can be added to such a multi-master cluster.

3.3.7 Cluster Node Mode Types

Oracle Key Vault supports two types of mode for cluster nodes: read-only restricted mode or read-write mode.

- **Read-only restricted mode**: In this mode, only non-critical data can be updated or added to the node. Critical data can be updated or added only through replication in this mode. There are two situations in which a node is in read-only restricted mode:
  - A node is read-only and does not yet have a read-write peer.
  - A node is part of a read-write pair but there has been a breakdown in communication with its read-write peer or if there is a node failure. When one of the two nodes is non-operational, then the remaining node is set to be in the read-only restricted mode. When a read-write node is again able to communicate with its read-write peer, then the node reverts back to read-write mode from read-only restricted mode.

- **Read-write mode**: This mode enables both critical and non-critical information to be written to a node. A read-write node should always operate in the read-write mode.

You can find the mode type of the cluster node on the Monitoring page of the **Cluster** tab of the node management console. The **Cluster** tab of any node management console displays the mode type of all nodes in the cluster.

3.3.8 Operations Permitted on Cluster Nodes in Different Modes

In an Oracle Key Vault multi-master cluster, operations are available or restricted based on the node and the operating mode of the node.

Related Topics

- Oracle Key Vault Multi-Master Cluster Operations
  There are restrictions and conditions for Oracle Key Vault multi-master cluster operations on cluster nodes.

3.4 Building and Managing a Multi-Master Cluster

You initialize a multi-master cluster using a single Oracle Key Vault server.

- About Building and Managing a Multi-Master Cluster
  After the initial cluster is created in the Oracle Key Vault server, you can add the different types of nodes that you need for the cluster.
- Creation of the Initial Node in a Multi-Master Cluster
  The initial node in a multi-master cluster must follow certain requirements before being made the initial node.
- Expansion of a Multi-Master Cluster
  After you initialize the cluster, you can expand it by adding up to 15 more nodes, as either read-write pairs or read-only nodes.
- Migration to the Cluster from an Existing Deployment
  You can migrate an existing Oracle Key Vault deployment to a multi-master cluster node.

3.4.1 About Building and Managing a Multi-Master Cluster

After the initial cluster is created in the Oracle Key Vault server, you can add the different types of nodes that you need for the cluster.

This Oracle Key Vault server seeds the cluster data and converts the server into the first cluster node, which is called the initial node. The cluster is expanded when you induct additional Oracle Key Vault servers, and add them as read-write nodes, or as simple read-only nodes.

A multi-master cluster can contain a minimum of 2 nodes and a maximum of 16 nodes.

3.4.2 Creation of the Initial Node in a Multi-Master Cluster

The initial node in a multi-master cluster must follow certain requirements before being made the initial node.

You create a multi-master cluster by converting a single Oracle Key Vault server to become the initial node. The Oracle Key Vault server can be a freshly installed Oracle Key Vault server, or it can already be in service with existing data. A standalone server, or a primary server of a primary-standby configuration can be converted to the initial node of a cluster.

Before using the primary server of the primary-standby configuration, you must unpair the primary-standby configuration. For a primary-standby configuration, you can use either of the following methods to upgrade to a cluster:

- Method 1:
  1. Back up the servers.
  2. Upgrade both the primary and standby servers to release 18.3.
  3. Unpair the paired primary and standby servers. (Before you unpair the servers, see *Oracle Key Vault Release Notes* for known issues regarding the unpair process.)
  4. Convert the primary server to be the first node of the cluster.
- Method 2:
  1. Back up the servers.
  2. Unpair the paired primary and standby servers.
  3. Upgrade the former primary server to release 18.3.
  4. Convert the primary server to be the first node of the cluster.

The initial node is special in that it provides the entirety of the data with which the cluster is initialized. This happens only once for the cluster when it is created. The data provided by the initial node will include but is not limited to the following components:

- Certificates, keys, wallets, and other security objects
- Users and groups
- Endpoint information
- Audits
- Reports

All other nodes added after the initial node must be created from freshly installed Oracle Key Vault servers.

The cluster name is chosen when the initial node is created. Once this name is chosen, you cannot change the cluster name.

The cluster subgroup of the initial node is also configured when the initial node is created and it cannot be changed. You must configure an Oracle Key Vault server that is converted to the initial node to use a valid Network Time Protocol (NTP) setting before you begin the conversion. The initial node always starts as a read-only node in read-only restricted mode.

To create the cluster, you select the **Add Node** button on the **Cluster** tab of the Oracle Key Vault Management console.

Related Topics

- Creating the First (Initial) Node of a Cluster
  To create a cluster, you must convert an existing standalone Oracle Key Vault server to become the first node in the cluster.
- *Oracle Key Vault Release Notes*

3.4.3 Expansion of a Multi-Master Cluster

After you initialize the cluster, you can expand it by adding up to 15 more nodes, as either read-write pairs or read-only nodes.

- About the Expansion of a Multi-Master Cluster
  Node induction is the process of configuring an Oracle Key Vault server to operate as a multi-master cluster node.
- Management of Cluster Reconfiguration Changes Using a Controller Node
  A controller node is the node that controls or manages a cluster reconfiguration change, such as adding, enabling, disabling, or removing nodes.
- Addition of a Candidate Node to the Multi-Master Cluster
  A freshly installed Oracle Key Vault server that is being added to a cluster is called a candidate node.
- Addition of More Nodes to a Multi-Master Cluster
  You add nodes one at a time, first as a single read-write node, and then later as read-write paired nodes.

3.4.3.1 About the Expansion of a Multi-Master Cluster

Node induction is the process of configuring an Oracle Key Vault server to operate as a multi-master cluster node.

A controller node inducts an Oracle Key Vault server converted to a candidate node into the cluster.

To expand a multi-master cluster, you use the induction process found on the **Cluster** tab of the Oracle Key Vault Management console. Nodes added to the Oracle Key Vault multi-master cluster are initialized with the current cluster data. You can add nodes either as read-write peers, or as read-only nodes.

Related Topics

- Creating a Read-Write Pair of Nodes in a Cluster
  After you create the initial node, you must add an additional read-write peer to the cluster.
- Creating a Read-Only Node in a Cluster
  To add a new read-only cluster node, you pair any existing cluster node with a newly configured server.

3.4.3.2 Management of Cluster Reconfiguration Changes Using a Controller Node

A controller node is the node that controls or manages a cluster reconfiguration change, such as adding, enabling, disabling, or removing nodes.

A node is only a controller node during the life of the change. During induction, the controller node provides the server certificate and the data that is used to initialize the candidate node. Another node can be the controller node for a subsequent cluster change. One controller node can only control one cluster configuration change at a time. Oracle Key Vault does not permit multiple cluster operations at the same time.

Oracle recommends that you perform one cluster operation at a time. Each concurrent operation will have its own controller node. One controller node can only control one cluster configuration transaction at a time.

The following table shows the role of the controller and controlled nodes during various cluster configuration.

Cluster Configuration Operation	Controller Node	Node Being Controlled
Induction as the first node	Any server	The controller node itself
Induction as a read-only node	Any node	Any server
Induction as a read-write node	Any node that does not have a read-write peer	Any server
Disable a node	Any node in the cluster	Any node in the cluster
Enable a node	Only the disabled node can only re-enable itself, and it can also not re-enable any other node	The disabled node
Delete a node	Any node in the cluster	Any other node in the cluster
Force Delete a node	Any node in the cluster	Any other node in the cluster
Manage inbound replication	Any node in the cluster	The node itself

3.4.3.3 Addition of a Candidate Node to the Multi-Master Cluster

A freshly installed Oracle Key Vault server that is being added to a cluster is called a candidate node.

In the process of adding the server to a cluster, Oracle Key Vault converts the server to a candidate node before it becomes a node of the cluster. The server is converted to a candidate node before it becomes a node of the cluster. To induct an Oracle Key Vault server to a cluster, you must provide necessary information such as the controller certificate, the server IP address, and the recovery passphrase so that the new candidate can successfully and securely communicate with the controller node.

You can convert an Oracle Key Vault server to be a candidate node by using the **Cluster** tab of the Oracle Key Vault Management console.

3.4.3.4 Addition of More Nodes to a Multi-Master Cluster

You add nodes one at a time, first as a single read-write node, and then later as read-write paired nodes.

When an Oracle Key Vault multi-master cluster is first created, and only contains one node, that initial node is a read-only node. After you have created the initial node, you can induct additional read-only nodes or a read-write peer. After these nodes have been added, you can further add read-only nodes or add a read-write peer.

Because the initial node is in read-only restricted mode and no critical data can be added to it, Oracle recommends that you induct a second node to form a read-write

pair with the first node. You should expand the cluster to have read-write pairs so that both critical and non-critical data can be added to the read-write nodes. Read-only nodes can help with load balancing or operation continuity during maintenance operations.

The general process for adding nodes to a cluster is to add one node at a time, and then pair these so that they become read-write pairs:

1. Add the initial node (for example, N1). N1 is a read-only node.
2. Add the second node, N2. N2 will be in read-only restricted node during the induction process.
   If you are adding N2 as the read-write peer of N1, then both N1 and N2 will become read-write nodes when N2 is added to the cluster. Otherwise, N1 and N2 will remain read-only nodes after you add N2 to the cluster. If you want to add N2 as the read-write peer of N1, then you must set **Add Candidate Node as Read-Write Peer** to **Yes** on N1 during the induction process. If you do not want N2 to be paired with N1, then set **Add Candidate Node as Read-Write Peer** to **No**. This example assumes that N2 will be made a read-write peer of N1.
3. Add a third node, N3, which must be a read-only node if N1 and N2 were made read-write peers, because the other two nodes in the cluster already are read-write peers.
   In fact, you can add multiple read-only nodes to this cluster, but Oracle recommends that you not do this, because when write operations take place, the few read-write nodes that are in the cluster will be overloaded. For optimum performance and load balancing, you must have more read-write pairs.
4. To create a second read-write pair for the cluster, when you add the next node (N4), set **Add Candidate Node as Read-Write Peer** to **Yes** to add node N4 to be paired with node N3.
   Node N4 must be added from N3 to make the second read-write pair, because N3 is the only node without a read-write peer at this point. After you complete this step, at this stage the cluster has two read-write pairs: the N1-N2 pair, and the N3-N4 pair.
5. To create the next pairing, add the next read-only node (for example, N5), followed by node N6.
   Be sure to set **Add Candidate Node as Read-Write Peer** to **Yes** when you add N6. Node N6 must be added to N5 because at this point, N5 is the only node without a read-write peer. By the time you complete this step, there will be three read-write pairs in the cluster: the N1-N2 pair, the N3-N4 pair, and the N5-N6 pair.

A freshly installed Oracle Key Vault server at the same version as the other nodes in the cluster, that is, at release 18.1 or later is converted to a candidate node. You should ensure that the candidate has Network Time Protocol (NTP) configured.

Any node in the cluster can be the controller node given no other cluster change operations are in progress. The candidate and the controller node exchange information that enables the controller node to ascertain the viability of induction. The controller node ships data over to the candidate node. Induction replicates the cluster data set to the candidate node. After a successful induction, you can configure the node to use the cluster-wide configuration settings. A cluster data set includes but is not limited to the following components:

- Certificates, keys, wallets, and other security objects
- Users and user groups
- Endpoint and endpoint group information

- Audit data
- Cluster name and cluster node details
- Cluster settings

The controller node assigns the node ID and the cluster subgroup for the candidate node. You cannot change these later on. If the controller node provides an existing cluster subgroup during induction, then the candidate node becomes part of that subgroup. If the controller node provides the name of the cluster subgroup that does not exist, then the cluster subgroup is created as part of the induction process and the candidate node is added to the cluster subgroup. You can have all endpoints associated with one subgroup, if you want. For example, all endpoints in data center A can be in one subgroup, and all endpoints in data center B can be in another subgroup. Endpoints that are in the same subgroup will prioritize connecting to the nodes in that subgroup before connecting to nodes in other subgroups.

A controller node that is a read-write node in read-write mode or read-only restricted mode can only add read-only nodes. A controller node that is a read-only node can add one candidate node as its read-write peer to form a read-write pair. A node becomes a member of a read-write pair, and therefore a read-write node, during the induction process. A controller node that is a read-only node can add additional nodes as read-only nodes, which can subsequently be used to form a new read-write pair.

A read-write node can become a read-only node if its read-write peer is deleted. This read-only node can be used to form another read-write pair.

Note the following:

- If the controller is a member of an existing read-write pair, then the node being added is inducted as a read-only node.
- If the controller is not a member of an existing read-write pair, then the node being added can be a read-only node or it can become the read-write peer to the controller node.

Related Topics

- Creating a Read-Only Node in a Cluster
  To add a new read-only cluster node, you pair any existing cluster node with a newly configured server.
- Creating an Additional Read-Write Pair in a Cluster
  Any node can be read-write paired with only one other node, and there can be multiple read-write pairs in a cluster.

3.4.4 Migration to the Cluster from an Existing Deployment

You can migrate an existing Oracle Key Vault deployment to a multi-master cluster node.

- Conversion of an Oracle Key Vault Standalone Server to a Multi-Master Cluster
  You can migrate a standalone Oracle Key Vault deployment that is at an older release to a multi-master deployment.
- Conversion from a Primary-Standby Server to a Multi-Master Cluster
  You can migrate an Oracle Key Vault primary-standby deployment that is at an earlier release than 18.1 to a multi-master deployment.

3.4.4.1 Conversion of an Oracle Key Vault Standalone Server to a Multi-Master Cluster

You can migrate a standalone Oracle Key Vault deployment that is at an older release to a multi-master deployment.

First, you must upgrade the server to the latest Oracle Key Vault release. After you complete the upgrade, you then can convert it to an initial node.

If your Oracle Key Vault server deployment is already at the current release, then you can directly convert it to an initial node.

The initial node will retain all the data of the existing Oracle Key Vault standalone deployment. After you create the initial node of the cluster, you can add more nodes to this cluster as necessary.

Related Topics

- *Oracle Key Vault Release Notes*
- Creating the First (Initial) Node of a Cluster
  To create a cluster, you must convert an existing standalone Oracle Key Vault server to become the first node in the cluster.

3.4.4.2 Conversion from a Primary-Standby Server to a Multi-Master Cluster

You can migrate an Oracle Key Vault primary-standby deployment that is at an earlier release than 18.1 to a multi-master deployment.

First, you must unpair the primary-standby server configuration. Then you should upgrade the unpaired former primary server to the lateset Oracle Key Vault release. After you have upgraded the former primary server, you can convert it to an initial node.

If you have the latest release of Oracle Key Vault in a primary-standby deployment, you must also unpair it before you can move it to a multi-master cluster. Then, you can directly convert the former primary server to an initial node.

Before you perform this kind of migration, you should back up the servers that will be used in the primary-standby deployment.

Related Topics

- Disabling (Unpairing) the Primary-Standby Configuration
  You can disable the primary-standby configuration by unpairing the primary and standby servers.
- Creating the First (Initial) Node of a Cluster
  To create a cluster, you must convert an existing standalone Oracle Key Vault server to become the first node in the cluster.

3.5 Oracle Key Vault Multi-Master Cluster Deployment Scenarios

All multi-master cluster nodes can serve endpoints actively and independently.

They can do this while striving to maintain an identical cluster data set through continuous replication across the cluster. Deployment scenarios of the multi-master cluster can range from a small two-node cluster to large 16-node deployments spanning across data centers.

- Cluster Size and Availability in Deployments
  In general, the availability of the critical data to the endpoints increases with the increasing size of the cluster.
- Two-Node Cluster Deployment
  A single read-write pair formed with two Oracle Key Vault nodes is the simplest multi-master cluster.
- Mid-Size Cluster Across Two Data Centers Deployment
  A two-data center configuration provides high availability, disaster recovery, and load distribution.

3.5.1 Cluster Size and Availability in Deployments

In general, the availability of the critical data to the endpoints increases with the increasing size of the cluster.

Usually, you must remove the read-write pairs from the cluster together to undergo maintenance such as patching or upgrade. While a two-node cluster provides better utilization, it does mean the endpoints will have a down time.

A two-node cluster is suitable for development and test environments where the endpoint downtime is non-critical. For small deployments where critical data is added or updated infrequently or can be controlled, a three-node cluster is acceptable. Large scale deployments under heavy load should deploy at least two read-write pairs to ensure endpoint continuity.

A three-node cluster with one read-write pair and one read-only node fares better than a two-node cluster because it provides endpoint continuity so long as no critical data is added or updated.

A four-node, two read-write pair cluster provides continuity for all endpoint operations while the nodes are in maintenance.

The cluster should ideally be comprised of read-write pairs. If network latency or network interruptions across data centers is of little concern, then you should deploy read-write pairs across data centers. In case of a disaster, the keys are preserved in the read-write peer. However, if the network latency or interruptions are of concern, then you should place the read-write pair in the same data center. A disaster resulting in the loss of read-write pair may result in the loss of keys if the disaster strikes within the replication time after the key is created.

3.5.2 Two-Node Cluster Deployment

A single read-write pair formed with two Oracle Key Vault nodes is the simplest multi-master cluster.

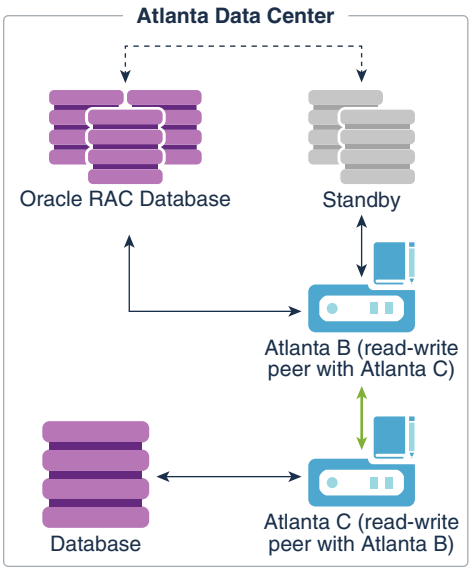
A two-node multi-master cluster looks similar to a standard primary-standby environment, in that there are only two nodes. The significant difference is that unlike the primary-standby configuration where the standby is passive, both nodes are active and can respond to endpoint requests at the same time.

A cluster made up of two read-only nodes is not useful because no critical data can be added to it. Using a two-node multi-master cluster provides the following advantages over a primary-standby environment:

- Both nodes can be actively queried and updated by endpoints unlike the primary-standby configuration where only the primary server can be queried.
- A multi-master cluster can be expanded to three or more nodes without downtime.
- If the nodes are in separate data centers, then endpoints can interact with local nodes, rather than reach across the network to the primary server node.

The following figure describes the deployment used for a two-node, single data center.

Figure 3-1 Oracle Key Vault Multi-Master Cluster Deployment in Single Data Center



In this scenario, the Atlanta Data Center hosts three databases, as follows:

- A single instance database
- A multi-instance Oracle Real Applications Clusters (Oracle RAC) database
- A multi-instance standby database for the Oracle RAC database

There are two Oracle Key Vault servers, labeled Atlanta B and Atlanta C, presenting a read-write pair of Oracle Key Vault nodes. These nodes are connected by a bidirectional line indicating that these are read-write peers. Read-write peer nodes are synchronous, which means that the transactions occur immediately.

The Oracle RAC database and the associated standby database are enrolled with the Oracle Key Vault, which would be at the head of the endpoint node scan lists for this database. Not shown is that each Oracle RAC instance would have a separate connection to the Oracle Key Vault server.

The database instance is enrolled with the Atlanta C node. To illustrate this connection, the database is connected by an arrow to Oracle Key Vault Atlanta C, which would be at the head of the endpoint node scan lists for this database.

In the event that either Oracle Key Vault server is offline, perhaps for maintenance, all endpoints automatically connect to the other (available) Oracle Key Vault server (Atlanta B).

3.5.3 Mid-Size Cluster Across Two Data Centers Deployment

A two-data center configuration provides high availability, disaster recovery, and load distribution.

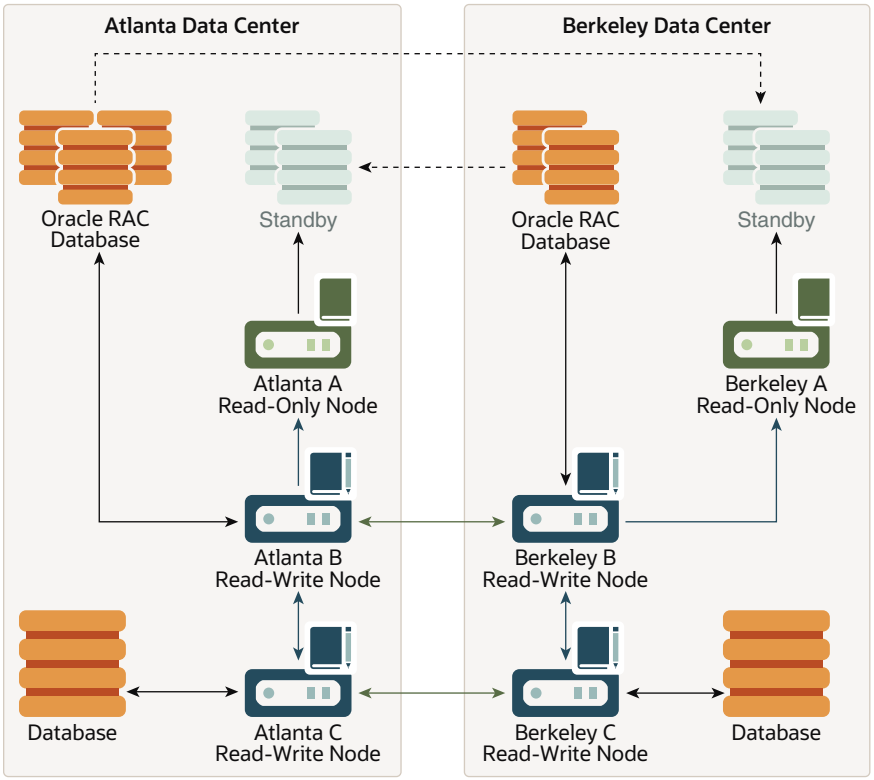
At least two read-write pairs are required. A read-write pair is only created when you pair a new node with a read-only node. As a best practice, you could configure the peers in different data centers if you are concerned about disaster recovery, or you could put the read-write peers in the same data center if you are concerned about network latency or network interruptions. Cluster nodes in the same data center should be part of the same cluster subgroup. You should enroll all endpoints in a data center with the nodes in that data center. This ensures that the nodes within a given Berkeley Data Center at the head of the endpoint node scan list for endpoints in the same data center.

For a large deployment, Oracle recommends that you have a minimum of four Oracle Key Vault servers in a data center for high availability. This enables additional servers to be available for key updates if one of the servers fails. When you register the database endpoints, balance these endpoints across the Oracle Key Vault servers. For example, if the data center has 1000 database endpoints to register, and you have Oracle Key Vault four servers to accommodate them, then enroll 250 endpoints with each of the four servers.

Each endpoint first contacts the Oracle Key Vault nodes in the local data center. If an outage causes all Oracle Key Vault nodes to be unavailable in one data center, then as long as connectivity to another data center is available, the endpoint node scan list will redirect the endpoints to available Oracle Key Vault nodes in another data center.

A possible deployment scenario with two data centers, each containing two read-write nodes, paired with read-write nodes in the other data center is shown in Figure 3-2. A data center can also host one or more read-only nodes as needed for load balancing, reliability, or expansion purposes. In the scenario described in the following figure, each data center hosts a single read-only node.

Figure 3-2 Oracle Key Vault Multi-Master Cluster Deployment across Two Data Centers



In this scenario, both the Atlanta Data Center and the Berkeley Data Center each hosts three databases, as follows:

- A single instance database
- A multi-instance Oracle RAC database
- A multi-instance standby database for the Oracle RAC database

The dotted lines connecting the Oracle RAC databases to the standby databases represent database transactions. Note that this data is unidirectional, going from the database to the standby only. The Atlanta Data Center Oracle RAC Database sends data to the Berkeley Standby database, and the Berkeley Oracle RAC Database sends data to the Atlanta Standby database.

Atlanta Data Center and Berkeley Data Center each have three Oracle Key Vault nodes, with two being read-write and one being read-only. These nodes are configured as follows:

- Atlanta A Read-Only Node and Berkeley A Read-Only Node are read-only restricted nodes, in which data is unidirectional, going from the read-write node to the read-only node
- Atlanta B Read-Write Node is a read-write peer with Berkeley B Read-Write Node. These two nodes are within one cluster. Their connection is bidirectional and enables these two nodes to be in sync at all times. You can extend these nodes to include up to 16 nodes combined, read-write pairs and read-only nodes. All these nodes can communicate with each other.

- Atlanta C Read-Write Node is a read-write peer with Berkeley C Read-Write Node. The relationship between these two nodes operates in the same way as the relationship between the Atlanta B and Berkeley B nodes.

All of the read-write nodes connect to all other nodes. To maintain legibility, only some of these connections are shown, specifically:

- The read-write pair connection between Atlanta B Read-Write Node and Berkeley B Read-Write Node across the two data centers, in which the data flow is bidirectional
- The read-write pair connection between Atlanta C Read-Write Node and Berkeley C Read-Write Node across the two data centers, in which the data flow is bidirectional
- The regular connection between Atlanta B Read-Write Node and Atlanta C Read-Write Node in the Atlanta Data Center, in which the data flow is bidirectional
- The regular connection between Berkeley B Read-Write Node and Berkeley C Read-Write Node in the Berkeley Data Center, in which the data flow is bidirectional
- The regular connection between Atlanta B Read-Write Node to Atlanta A Read-Only Node in the Atlanta Data Center, in which the data flow is unidirectional from the read-write node to the read-only node
- The regular connection between Berkeley B Read-Write Node and Berkeley A Read-Only Node in the Berkeley Data Center, in which the data flow is unidirectional from the read-write node to the read-only node

The endpoint node scan list is unique to each endpoint and controls the order of Oracle Key Vault nodes to which the endpoint connections from databases are established. Each endpoint can connect to all Oracle Key Vault nodes. Preference is given to the nodes in the same cluster subgroup as the node where the endpoint was added before moving to nodes in other cluster subgroups. In Figure 3-2, the following connections are shown, which imply the first entry in each client endpoint node scan list:

- In Atlanta Data Center:
  - The Oracle RAC Database connects to Atlanta B Read-Write Node, with the data going in a bidirectional flow.
  - The Atlanta A Read-Only Node connect to the Standby, with the data going from the read-only node to the standby in a unidirectional flow.
  - The Database connects to the Atlanta C Read-Write Node, with the data going in a bidirectional flow.
- In Berkeley Data Center:
  - The Oracle RAC Database connects to Berkeley B Read-Write Node, with the data going in a bidirectional flow.
  - The Berkeley A Read-Only Node connects to the Standby, with the data going from the read-only node to the standby in a unidirectional flow.
  - The Database connects to Berkeley C Read-Write Node, with the data going in a bidirectional flow.

In the event that Atlanta C Read-Write node (a read-write peer with Berkeley C Read-Write Node) cannot be reached or does not have the necessary key, the database to

which it connects in Atlanta Data Center will connect to other Oracle Key Vault nodes to fetch the key.

Related Topics

- Oracle Key Vault Read-Write Nodes
  A read-write node is a node in which critical data can be added or updated using the Oracle Key Vault or endpoint software.
- Oracle Key Vault Read-Only Nodes
  In a read-only node, users can add or update non-critical data but not add or update critical data.

3.6 Multi-Master Cluster Features

Oracle Key Vault provides features that help with inconsistency resolution and name conflict resolution in clusters, and endpoint node scan lists.

- Cluster Inconsistency Resolution in a Multi-Master Cluster
  Network outages can introduce inconsistency in data in a cluster, but when the outage is over, data is consistent again.
- Name Conflict Resolution in a Multi-Master Cluster
  Naming conflicts can arise when an object has the same name as another object in a different node.
- Endpoint Node Connection Lists (Endpoint Node Scan Lists)
  An endpoint node scan list is a list of nodes to which the endpoint can connect.

3.6.1 Cluster Inconsistency Resolution in a Multi-Master Cluster

Network outages can introduce inconsistency in data in a cluster, but when the outage is over, data is consistent again.

A node can be disconnected from other nodes in the cluster voluntarily or involuntarily. When a node becomes available to the cluster after being voluntarily disconnected, any data changes in the cluster are replicated to the node. Network disruptions, power outages, and other disconnects can happen any time for any Oracle Key Vault node, causing an involuntary disconnection from other nodes in the cluster. Such failures interrupt the data replication processes within a multi-master cluster. Temporary failures do not always introduce inconsistency to a cluster. As soon as the problem is addressed, the data replication process will resume from the moment it was halted. This ensures that even after some disconnections, disconnected Oracle Key Vault nodes will be able to synchronize themselves with the other nodes in the cluster eventually.

Any change made in a read-write node is guaranteed to be replicated to the other paired read-write node. Therefore, even if the read-write node suffers a failure, the data is available on one other node in the cluster.

3.6.2 Name Conflict Resolution in a Multi-Master Cluster

Naming conflicts can arise when an object has the same name as another object in a different node.

Users must specify names when creating virtual wallets, users, user groups, endpoints, and endpoint groups. A name conflict arises when two or more users create the same object with the same name on different nodes before the object has been replicated. If the object has been replicated on other nodes, then the system prevents the creation of objects with duplicate names. But replication in the Oracle Key Vault cluster is not instantaneous, so there is a possibility that during the replication window (which can be in the order of seconds), another object with the same name may have been created in this cluster. If this happens, it becomes a name conflict. Name conflicts have obvious drawbacks. For example, the system cannot distinguish between the references to two objects with duplicate names. Uniqueness in names is thus enforced to avoid inconsistencies in the cluster. All other object names must be unique within their object type, such as wallets, endpoint groups, user groups, and any other object type. For example, no two wallets may have the same name within the cluster. User names and endpoint names must not conflict.

While rare, a naming conflict can still arise. When this occurs, Oracle Key Vault detects this name conflict and raises an alert. Oracle Key Vault then will append `_0KV`*xx* (where *xx* is a node number) to the name of the conflicting object that was created later. You can choose to accept this suggested object name or rename the object.

To accept or change a conflicting object name, click the **Cluster** tab, then **Conflict Resolution** from the left navigation bar to see and resolve all conflicts.

3.6.3 Endpoint Node Connection Lists (Endpoint Node Scan Lists)

An endpoint node scan list is a list of nodes to which the endpoint can connect.

An endpoint connects to an Oracle Key Vault server or node to manage or access wallets, keys, certificates, and credentials.

In a standalone situation the endpoint node scan list has one entry. In a primary-standby configuration, the endpoint can connect to one of two nodes.

In an Oracle Key Vault multi-master cluster, the endpoint node scan list is the list of all the nodes in the cluster. There is a read-only node list and read-write node list. Node subgroup assignments and node modes influence the order of nodes in the endpoint node scan list. The list is made available to the endpoint at the time of endpoint enrollment. The list is maintained automatically to reflect the available nodes in the cluster. This node tracks changes to the cluster and makes them available to the endpoints. The following events will trigger a change to the endpoint node scan list:

- A change of cluster size, for example due to node addition or node removal
- A change to the mode of the node, for example when a node in read-only restricted mode changes to read-write mode
- An hour has passed since the last endpoint update

The endpoint gets the updated scan list along with the next non-empty response to the next request. Once the scan list is sent by the node, it marks the scan list as sent to the endpoint. It is possible that a scan list sent to the endpoint and marked sent in the node, may not be applied at the endpoint. As such the cluster periodically sends the scan list to the endpoint even if there are no changes to the cluster nodes or the modes of any of the cluster nodes.

4

Oracle Key Vault Installation and Configuration

Installing Oracle Key Vault entails ensuring that the environment meets the necessary requirements before you begin the installation and configuration.

- About Oracle Key Vault Installation and Configuration
  Oracle Key Vault is a software appliance that is delivered as an ISO image.
- Oracle Key Vault Installation Requirements
  The Oracle Key Vault installation requirements cover system requirements such as CPU, memory, disk space, network interfaces, and supported endpoint platforms.
- Installing and Configuring Oracle Key Vault
  You must download the Oracle Key Vault application software, and then you can perform the installation.
- Logging In to the Oracle Key Vault Management Console
  To use Oracle Key Vault, you can log in to the Oracle Key Vault management console.
- Upgrading a Standalone or Primary-Standby Oracle Key Vault Server
  This upgrade includes the Oracle Key Vault server software and utilities that control the associated endpoint software.
- Upgrading Oracle Key Vault in a Multi-Master Cluster Environment
  Similar to a standalone or primary-standby upgrade, this type of upgrade includes the Oracle Key Vault server software and endpoint software-related utilities.
- Overview of the Oracle Key Vault Management Console
  The Oracle Key Vault management console provides a graphical user interface for System Administrators, Key Administrators, and Audit Managers.
- Performing Actions and Searches
  The Oracle Key Vault management console enables you to perform standard actions and search operations, as well as get help information.

4.1 About Oracle Key Vault Installation and Configuration

Oracle Key Vault is a software appliance that is delivered as an ISO image.

The software appliance consists of a pre-configured operating system, an Oracle database, and the Oracle Key Vault application. You must install Oracle Key Vault onto its own dedicated server.

4.2 Oracle Key Vault Installation Requirements

The Oracle Key Vault installation requirements cover system requirements such as CPU, memory, disk space, network interfaces, and supported endpoint platforms.

- System Requirements
  System requirements include CPU, memory, disk, network interface, hardware compatibility, and RESTful services client.
- Network Port Requirements
  Network port requirements includes requirements for SSH/SCP, SNMP, HTTPS, listeners, KMIP, and TCP ports.
- Supported Endpoint Platforms
  Oracle Key Vault supports both UNIX and Windows endpoint platforms.
- Endpoint Database Requirements
  For endpoints, Oracle Key Vault supports Oracle Database release 10 and later.

4.2.1 System Requirements

System requirements include CPU, memory, disk, network interface, hardware compatibility, and RESTful services client.

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 16 cores. Recommended: 24-48 cores with cryptographic acceleration support (Intel AESNI).
- **Memory**: Minimum 16 GB of RAM. Recommended: 32–64 GB.
- **Disk**: Minimum 2 TB. Recommended: 4 TB.
- **Network interface**: One network interface.
- **Hardware Compatibility**: Refer to the hardware compatibility list (HCL) for Oracle Linux Release 6 Update 10 at the link in the Related Topics section.

> **Note:**
>
> You can find the supported hardware from the hardware certification list for Oracle Linux and Oracle VM. Filter the results by selecting **All Operating Systems** and choosing **Oracle Linux 6.10**. However, be aware that Oracle Key Vault does not support the QLogic QL4* family of network cards.

  Oracle Key Vault 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. Oracle Key Vault can be installed on Oracle X7–2 servers as a standalone server, a primary-standby configuration, or a multi-master cluster configuration.
- **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 you have installed a cURL version that supports Transport Layer Security (TLS) 1.2 or later on the endpoint 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.

Related Topics

- Hardware Certification List for Oracle Linux and Oracle VM

4.2.2 Network Port Requirements

Network port requirements includes requirements for SSH/SCP, SNMP, HTTPS, listeners, KMIP, and TCP ports.

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

The following table lists the required network ports for Oracle Key Vault:

Table 4-1 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
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 Administrative commands to communicate with Oracle Key Vault
5695	HTTPS port	Used by RESTful Key Management commands to communicate with Oracle Key Vault
1521 and 1522	Database TCPS listener ports	Listener ports used in a primary-standby configuration by Oracle Data Guard to communicate between the primary and standby server
7443	Database TCPS listener port	Listener port used in a primary-standby configuration to run OS commands like synchronizing wallets and configuration files through HTTPS. This port is also used when you add a new node to a cluster.
5696	KMIP port	Used by Oracle Key Vault endpoints and third party KMIP clients to communicate with the Oracle Key Vault KMIP Server
7093	TCP port	Used by Oracle GoldenGate for transmitting data in a Multi-Master Cluster configuration.

4.2.3 Supported Endpoint Platforms

Oracle Key Vault supports both UNIX and Windows endpoint platforms.

Oracle supports 64-bit Linux endpoints, and only 64-bit endpoints are supported for Oracle databases that use the online master key. The operating systems on which the endpoint runs must be compatible with Transport Layer Security (TLS) 1.2, either directly or with appropriate patches.

The supported endpoint platforms in this release are as follows:

- Oracle Linux (6 and 7)
- Oracle Solaris (10 and 11)
- Oracle Solaris Sparc (10 and 11)
- RHEL 6 and 7
- IBM AIX (6.1, and 7.1) and AIX 5.3 in a limited capacity
- HP-UX (IA) (11.31)
- Windows Server 2012

4.2.4 Endpoint Database Requirements

For endpoints, Oracle Key Vault supports Oracle Database release 10 and later.

Administrators who manage endpoints that are Oracle Database 10g release 2 and later can use the `okvutil upload` command to upload Oracle wallets to Oracle Key Vault. Administrators who manage endpoints that are Oracle Database 11g release 2 and later can use the online master key to manage TDE master encryption keys.

Administrators who manage endpoints that are Oracle Database may need to set the `COMPATIBLE` initialization parameter.

For an endpoint that is Oracle Database release 11.2 or 12.1, set the `COMPATIBLE` initialization parameter to `11.2.0.0` or later. A `COMPATIBLE` setting of 11.2 or later enables Transparent Data Encryption to work with Oracle Key Vault. For example:

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

This applies to an Oracle Database endpoint that use the online master key to manage TDE master encryption keys. 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 such as `10.2`. After you set the `COMPATIBLE` parameter, you must restart the database.

Related Topics

- *Oracle Database Administrator's Guide*

4.3 Installing and Configuring Oracle Key Vault

You must download the Oracle Key Vault application software, and then you can perform the installation.

- Downloading the Oracle Key Vault Appliance Software
  You can download executable files for both a fresh Oracle Key Vault installation or an upgrade.

- Installing the Oracle Key Vault Appliance Software
  The Oracle Key Vault installation process installs all the required software components onto a dedicated server.
- Performing Post-Installation Tasks
  After you install Oracle Key Vault, you must complete a set of post-installation tasks.

4.3.1 Downloading the Oracle Key Vault Appliance Software

You can download executable files for both a fresh Oracle Key Vault installation or an upgrade.

For a fresh installation, you can download the Oracle Key Vault appliance software from Software Delivery Cloud. You cannot use this package to upgrade Oracle Key Vault. For an upgrade, you can download the Oracle Key Vault upgrade software from the My Oracle Support website.

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

   https://edelivery.oracle.com

2. Click **Sign In**, and if prompted, enter your **User ID** and **Password**.
3. In the **All Categories** menu, select **Release**. In the next field, enter **Oracle Key Vault** and then click **Search**.
4. From the list that is displayed, select **Oracle Key Vault 18.4.0.0.0** or click the **+Add to Cart** button next to the **Oracle Key Vault 18.4.0.0.0**.

   The download is added to your cart. (To check the cart contents, click **View Cart** in the upper right of the screen.)
5. Click **Checkout**.
6. On the next page, verify the details of the installation package, and then click **Continue**.
7. In the **Oracle Standard Terms and Restrictions** page, select **I have reviewed and accept the terms of the Commercial License, Special Programs License, and/or Trial License**, and click **Continue**.

   The download page appears, which lists the following Oracle Key Vault ISO files:

   - V*part_number*.iso (Oracle Key Vault 18.4.0.0.0 - Disc 1)
   - V*part_number*.iso (Oracle Key Vault 18.4.0.0.0 - Disc 2)
8. To the right of the **Print** button, click **View Digest Details**.

   The listing for the two ISO files expands to display the SHA-1 and SHA-256 checksum reference numbers for each ISO file.
9. Copy the SHA-256 checksum reference numbers and store them for later reference.
10. Click **Download** and select a location to save the ISO files.

    You can save each file individually by clicking its name and then specifying a location for the download.
11. 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.

12. After the ISO files are downloaded to the specified location, verify the SHA-256 checksums of the downloaded files:
    a. Generate a SHA256 checksum for the first `V*part_number*.iso`:

       ```
       $ sha256sum V*part_number*.iso
       ```

       Ensure that the checksum matches the value that you copied from the **File Download** dialog box in 9.
    b. Generate a SHA-256 checksum for the second `V*part_number*.iso`:

       ```
       $ sha256sum V*part_number*.iso
       ```

       Ensure that the checksum matches the value that you copied from the **File Download** dialog box in 9.
13. Burn each of the two `V*part_number*.iso` files to a DVD-ROM disc and then label the discs:
    - `OKV Disc 1`
    - `OKV Disc 2`

You can now install Oracle Key Vault on the server.

4.3.2 Installing the Oracle Key Vault Appliance Software

The Oracle Key Vault installation process installs all the required software components onto a dedicated server.

The installation process may take from 30 minutes or longer to complete, depending on the server resources where you are installing Oracle Key Vault.

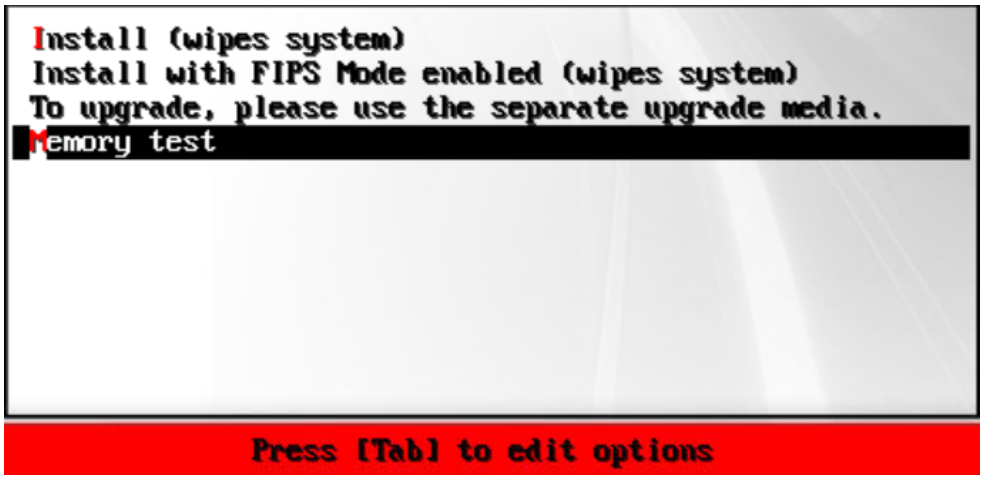
> **Caution:**
>
> The Oracle Key Vault installation wipes the server and installs a customized Oracle Linux 6 Update 10. The installation erases existing software and data on the server.

- 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.

To install the Oracle Key Vault appliance:

1. Insert `OKV Disc 1` into the DVD drive and then restart the computer.

   The installation starts, and the initial screen appears.



2. Using the up and down arrow keys, select the desired installation option or the option to perform a memory test, and then press **Enter**.

   Choosing **Install with FIPS Mode enabled (wipes system)** automatically enables FIPS mode on the system.

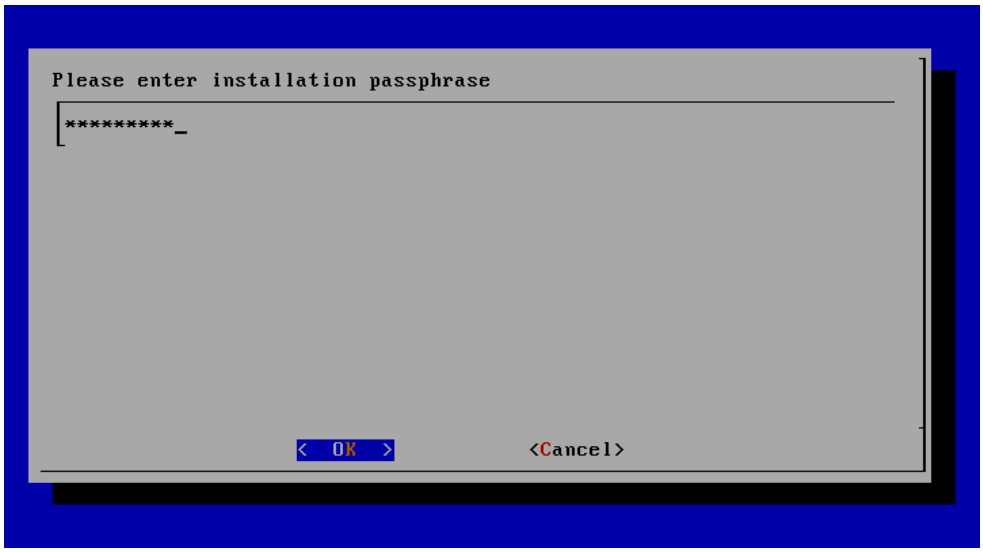
   The installation begins and after several minutes, the message **Please insert disc 2** is displayed.

3. Insert `OKV Disc 2` into the DVD drive, and then press **Enter**.

   The installation proceeds and after some time the message **Please insert disc 1** is displayed.

4. Insert `OKV Disc 1` into the DVD drive, and press **Enter**.

   The installation proceeds and after some time the message **Please enter installation passphrase** is displayed.

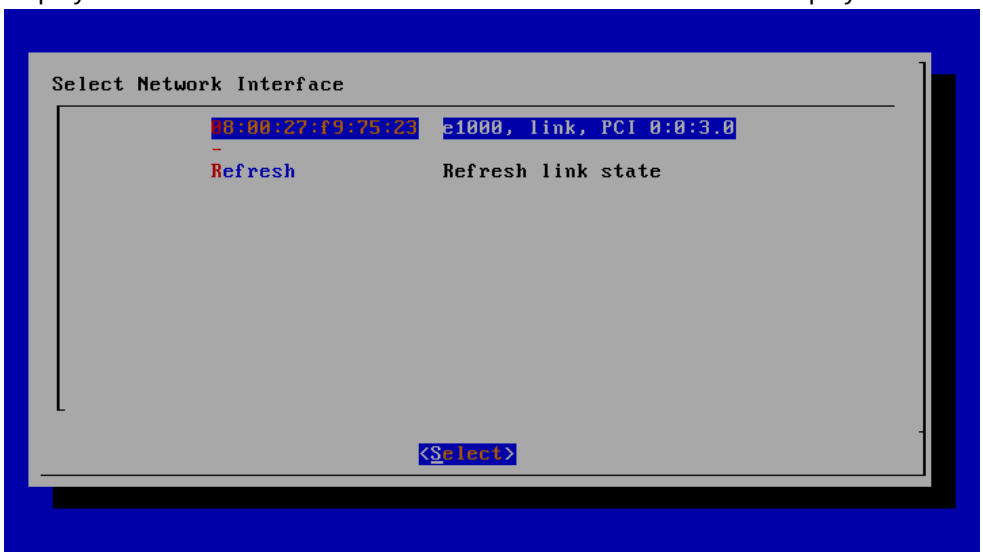


The installation passphrase must have 8 or more characters and contain at least one of each of the following: an uppercase letter, a lowercase letter, a number, and a special character from the set: period (`.`), comma (`,`), underscore (`_`), plus sign (`+`), colon (`:`), exclamation mark (`!`). In addition, the passphrase may include a space character ( ) provided it is not used as the first or last character of the passphrase.

It is important to store the installation passphrase securely. You will need it later to authenticate yourself at the Oracle Key Vault management console, complete the post-installation tasks, and add nodes in a multi-master cluster.

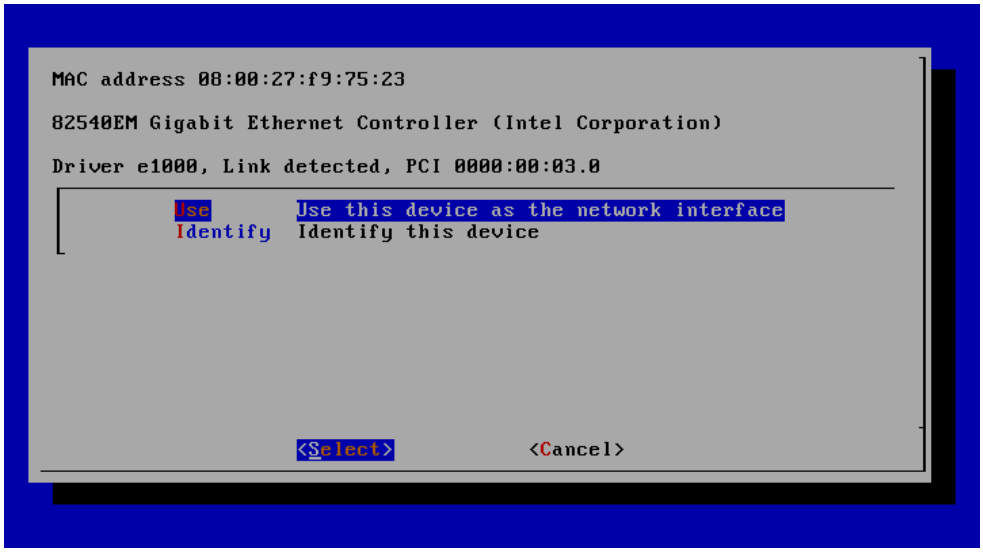
5. Enter the installation passphrase, and then press **Enter**.
6. Confirm the installation passphrase, and then press **Enter**.

   The message **Installation passphrase was successfully configured** is displayed. Press **Enter**. The **Select Network Interface** screen is displayed.



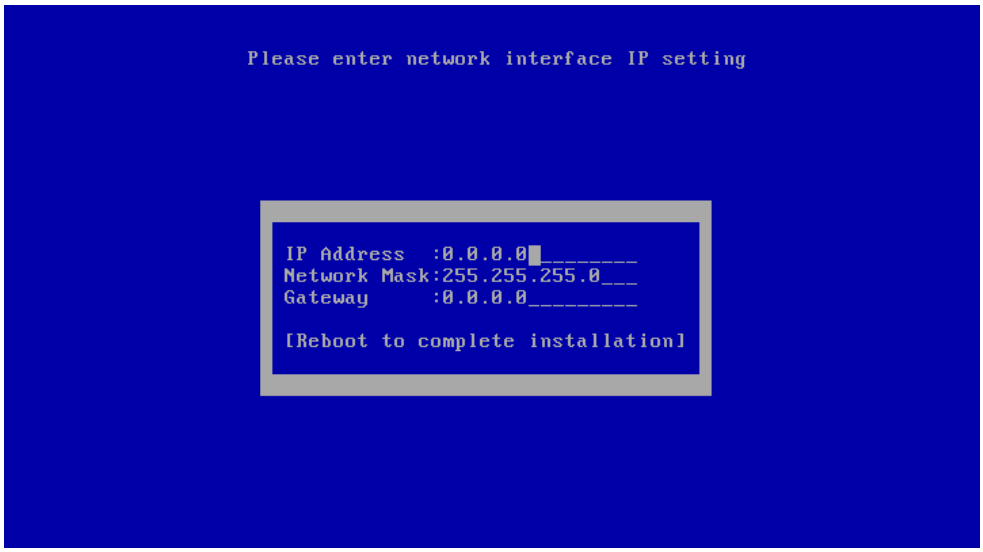
7. Select the interface and press **Enter**. If more than one network interface is available, select the interface that you want to serve as the management interface, and to communicate with endpoints.

   The **Network Selection Interface** screen is displayed.



8. Press **Enter**.

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



9. 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 or longer.

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

10. 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, then 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 the post-installation tasks, but be aware that after you set these passwords, you can only change them by using Secure Shell (SSH) on the computer on which these passwords were created.

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.

> **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.

4.3.3 Performing Post-Installation Tasks

After you install Oracle Key Vault, you must complete a set of post-installation tasks.

These tasks include configuring the administrative user accounts and passwords for recovery, and operating system accounts and passwords for `root` and `support`.

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

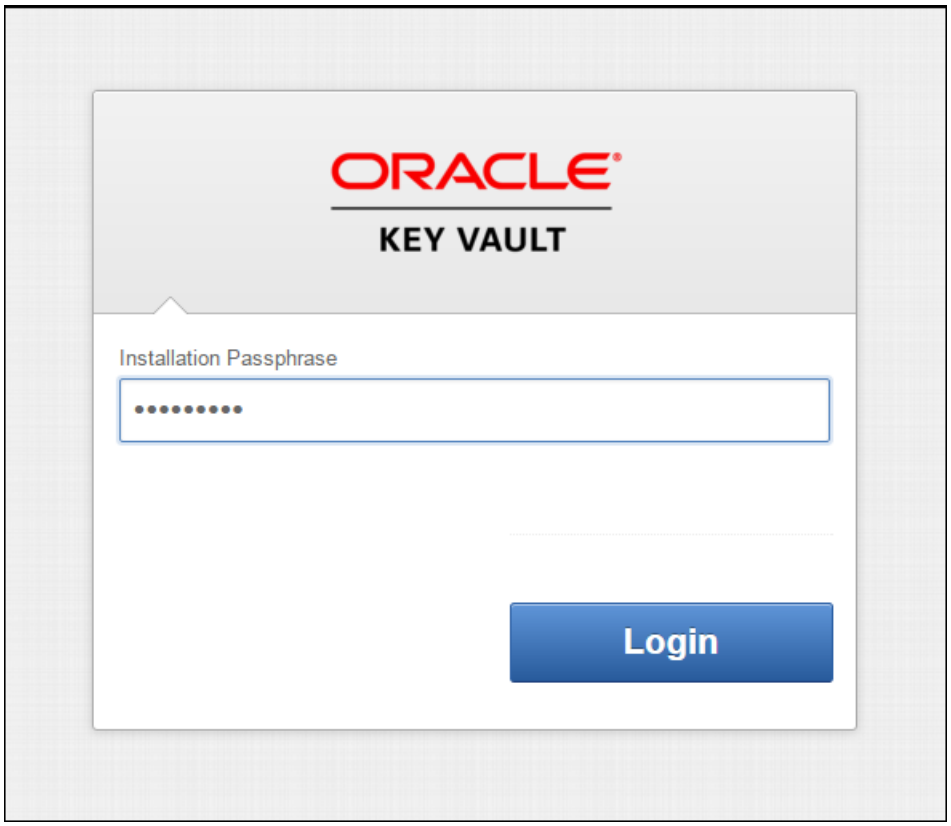
   For example, to connect in to an Oracle Key Vault server whose IP address is 192.0.2.254, enter the following in the address bar:

   ```
   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.

   This message is only temporary. When you configure third-party certificates, this message will no longer appear. 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 Managing Console Certificates.

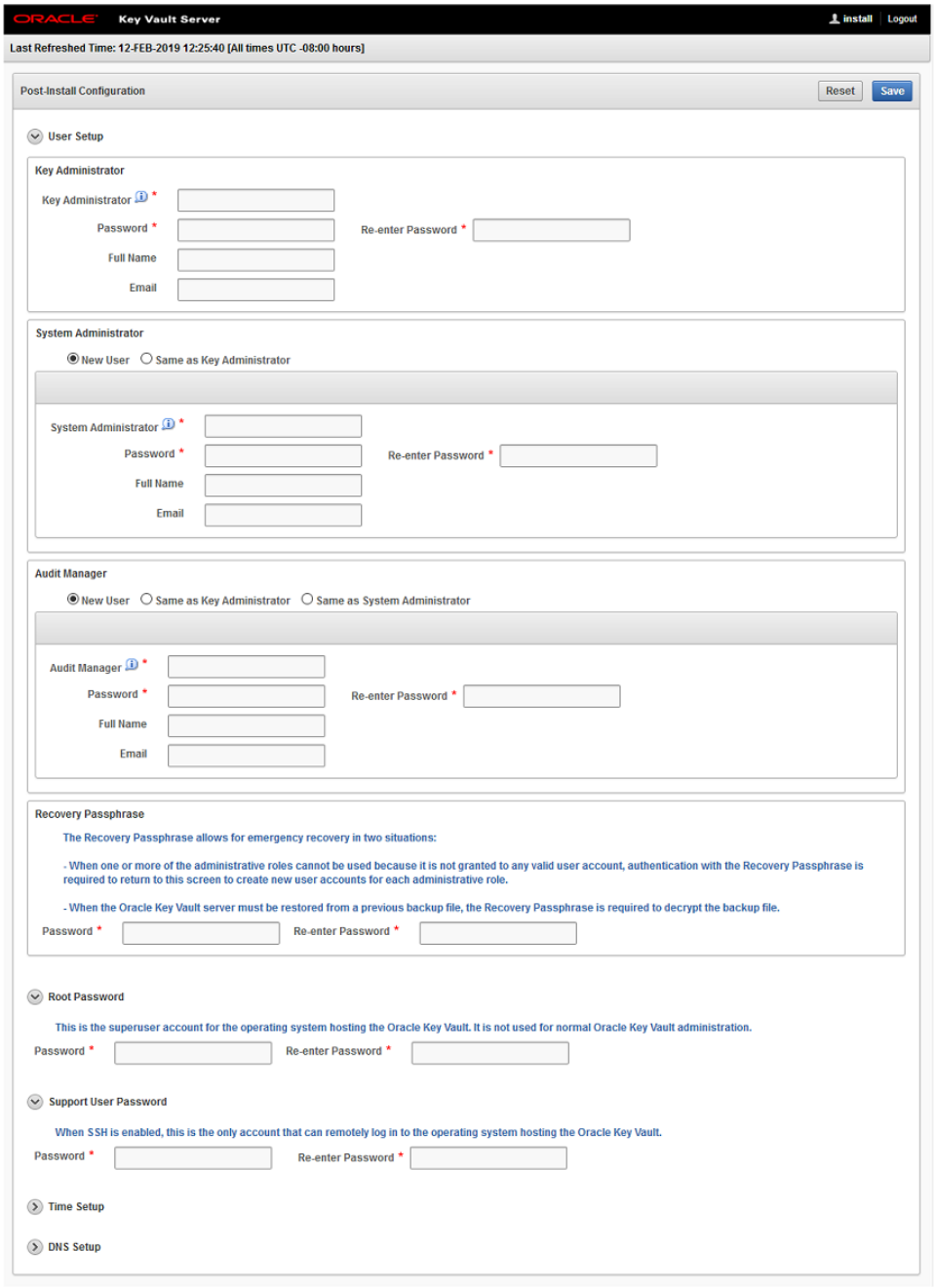
3. The **Installation Passphrase** screen is displayed.



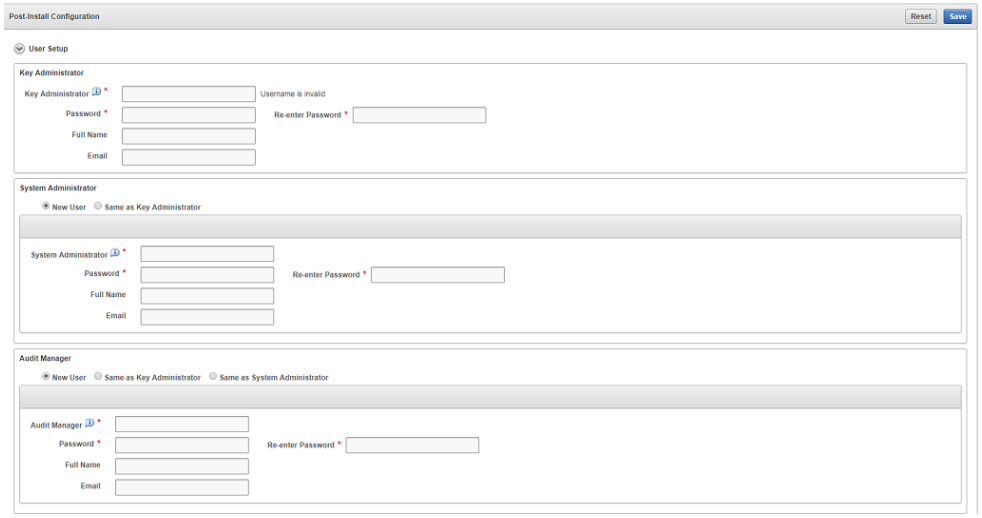
The **Installation Passphrase** screen is displayed when you connect to the Oracle Key Vault server for the first time, in order to complete the post-installation tasks. After you complete the post-installation tasks, the Oracle Key Vault login screen is displayed when you access the Oracle Key Vault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

4. Enter the installation passphrase.

   The **Post-Install Configuration** screen is displayed.



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



In the **User Setup** section:

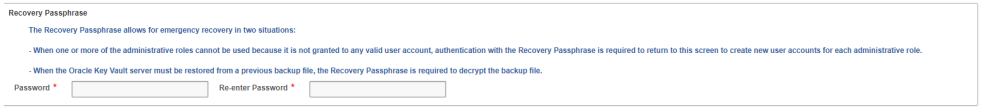
- Enter the user name and password, the full name (optional), and email (optional) for each administrative user account.

  Note that the passwords are one-time use passwords which must be changed when the user logs in the first time.

- Ideally, create a different user account for each of these administrative roles for a strict separation of duties, or combine roles as necessary.

- Ensure that passwords have 8 or more characters and contain 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 (`:`), exclamation mark (`!`). In addition, the passphrase may include a space character ( ) provided it is not used as the first or last character of the passphrase.

6. In the **Recovery Passphrase** section, create the recovery password.

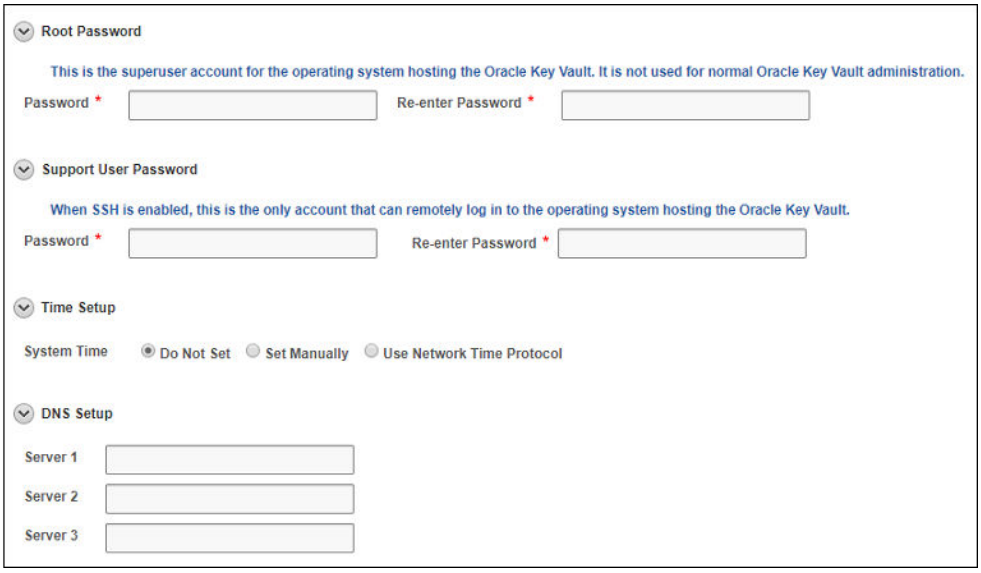


The recovery passphrase has the same minimum requirements as user passwords. For greater security, Oracle recommends that you make the recovery passphrase longer and more complex. Because this is a critical password, you must properly secure and safeguard the recovery password. The recovery password is required in the following scenarios:

- In an emergency, when there are no administrative users available to access Oracle Key Vault
- To restore Oracle Key Vault data from a backup
- To reset the recovery password
- Induct a new node into a multi-master cluster
- To configure a hardware security module (HSM)

> **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. Be aware that if you enter either of these passwords incorrectly three times in a row, then the account is locked for 15 minutes.**

7. 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** screen in the previous procedure, Installing the Oracle Key Vault Appliance Software.



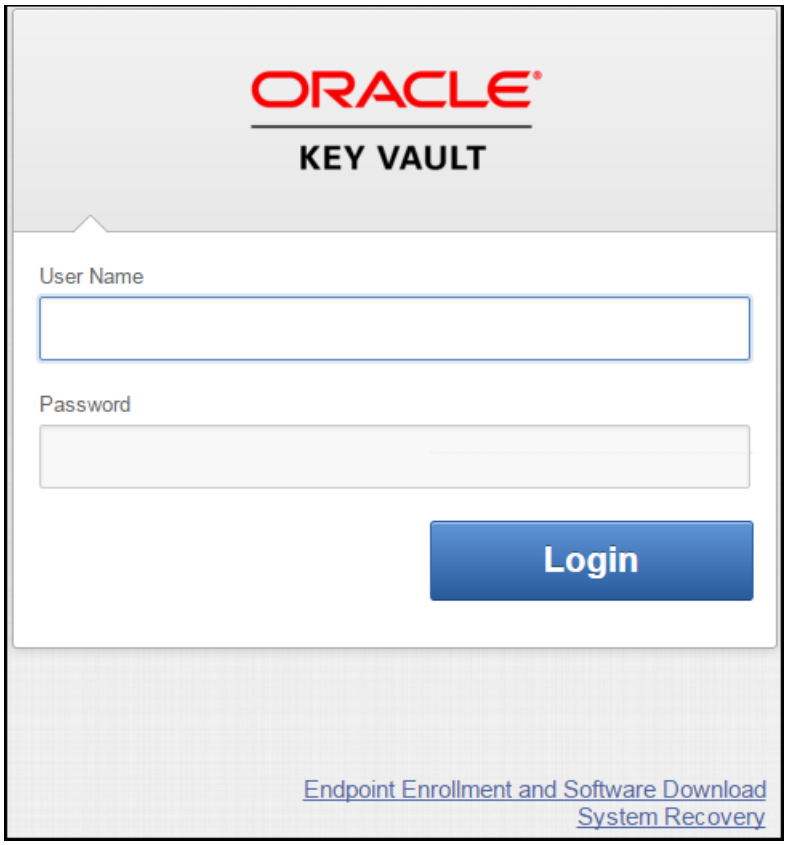
The root password is the super user account for the operating system hosting Oracle Key Vault. You will need the support password to log into Oracle Key Vault remotely using the SSH protocol.

> **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. The System Administrator can configure these later on.

8. Click **Save** in the upper right corner of the **Post-Install Configuration** screen.

   The Oracle Key Vault management console login screen is displayed.



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

Related Topics

- 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.
- 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.

4.4 Logging In to the Oracle Key Vault Management Console

To use Oracle Key Vault, you 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
   ```
3. After the login screen appears, enter your user name and password.
4. Click **Login**.

4.5 Upgrading a Standalone or Primary-Standby Oracle Key Vault Server

This upgrade includes the Oracle Key Vault server software and utilities that control the associated endpoint software.

- About Upgrading the Oracle Key Vault Server Software
  When you upgrade the Oracle Key Vault server software appliance, also upgrade the endpoint software to get access to the latest enhancements.
- Step 1: Back Up the Server Before You Upgrade
  Before you upgrade the Oracle Key Vault server, perform a one-time backup to a remote destination so that you can recover data in case the upgrade fails.
- Step 2: Perform Pre-Upgrade Tasks
  To ensure a smooth upgrade to Oracle Key Vault, you should prepare the server you are upgrading.
- Step 3: Upgrade the Oracle Key Vault Server or Server Pair
  You can upgrade a standalone Oracle Key Vault server or a pair of Oracle Key Vault servers in a primary-standby deployment.
- Step 4: Upgrade the Endpoint Software
  As part of the upgrade, you must reenroll endpoints created in earlier releases of Oracle Key Vault, or update the endpoint software.
- Step 5: If Necessary, Remove Old Kernels
  Oracle recommends that you clean up the older kernels that were left behind after the upgrade.
- Step 6: If Necessary, Add Disk Space to Extend Swap Space
  If you upgraded from an earlier release, you should extend swap space to accommodate the new Oracle Key Vault software.
- Step 7: If Necessary, Remove SSH-Related DSA Keys
  You should remove SSH-related DSA keys left behind after the upgrade, because they can cause problems with some code analysis tools.
- Step 8: Back Up the Upgraded Oracle Key Vault Server
  You must perform server backup and user password tasks after completing a successful upgrade.

4.5.1 About Upgrading the Oracle Key Vault Server Software

When you upgrade the Oracle Key Vault server software appliance, also upgrade the endpoint software to get access to the latest enhancements.

However, the endpoint software downloaded from the previous Oracle Key Vault release will continue to function with the upgraded Oracle Key Vault server.

You must upgrade in the order shown: first perform a full backup of Oracle Key Vault, upgrade the Oracle Key Vault server or server pair in the case of a primary-standby deployment, the endpoint software, and last, perform another full backup of the upgraded server. Note that upgrading requires a restart of the Oracle Key Vault server.

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.

Before you begin the upgrade, refer to *Oracle Key Vault Release Notes* for additional information about performing upgrades.

Related Topics

- *Oracle Key Vault Release Notes*
- Using the Persistent Master Encryption Key Cache
  The persistent master encryption key cache feature enables databases to be operational when the Oracle Key Vault server is unavailable.

4.5.2 Step 1: Back Up the Server Before You Upgrade

Before you upgrade the Oracle Key Vault server, perform a one-time backup to a remote destination so that you can recover data in case the upgrade fails.

> **Caution:**
>
> Do not bypass this step. Back up the server before you perform the upgrade so that your data is safe and recoverable.

Related Topics

- *Oracle Database Backup and Recovery User's Guide*

4.5.3 Step 2: Perform Pre-Upgrade Tasks

To ensure a smooth upgrade to Oracle Key Vault, you should prepare the server you are upgrading.

- Use SSH to log in to the server where Oracle Key Vault is installed.
- Ensure that the server meets the minimum disk space requirement for an upgrade. If the `/usr/local/dbfw/tmp` directory does not have sufficient free space, then delete any diagnostics `.zip` files that maybe stored in that directory.
- To increase available disk space, remove the temporary jar files located in `/usr/local/okv/ssl`. *Be careful in doing so.* If you accidentally delete any files other than the jar files in `/usr/local/okv/ssl`, then the Oracle Key Vault server becomes non-functional.
- 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 (NO with persistent cache)

Oracle Key Vault Usage	Downtime required
Primary Server Upgrade in a primary-standby deployment	YES (NO with persistent cache)

- Plan for downtimes:
  - If Oracle Key Vault uses an online master key, 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 primary-standby deployment, plan for a downtime of a few hours. The persistent cache allows you to upgrade Oracle Key Vault servers without database downtime. The default duration of the persistent cache from the moment the Oracle Key Vault server becomes unavailable is 1,440 minutes (one day).
- Set the `$OKV_HOME` to the location where endpoint is installed so that the upgrade process for the endpoint software can complete successfully.
- 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 the Syslog Destination for Individual Multi-Master Cluster Nodes
  On each node, you can forward syslog entries to a remote service such as Splunk or SIEM.

4.5.4 Step 3: Upgrade the Oracle Key Vault Server or Server Pair

You can upgrade a standalone Oracle Key Vault server or a pair of Oracle Key Vault servers in a primary-standby deployment.

- About Upgrading an Oracle Key Vault Server or Server Pair
  You can deploy Oracle Key Vault as a standalone server in test and development environments or in a primary-standby configuration in production environments.
- Upgrading a Standalone Oracle Key Vault Server
  A single Oracle Key Vault server in a standalone deployment is the most typical deployment in test and development environments.
- Upgrading a Pair of Oracle Key Vault Servers in a Primary-Standby Deployment
  You should allocate several hours to upgrade the primary server after upgrading the standby.

4.5.4.1 About Upgrading an Oracle Key Vault Server or Server Pair

You can deploy Oracle Key Vault as a standalone server in test and development environments or in a primary-standby configuration in production environments.

In a standalone deployment you must upgrade a single Oracle Key Vault server, but in a primary-standby deployment you must upgrade both primary and standby Oracle Key Vault servers. Note that persistent caching enables endpoints to continue to be operational during the upgrade process.

> **Note:**
>
> If you are upgrading from a system with 4 GB memory, first add an additional 12 GB memory to the system before upgrading.

Related Topics

- About the Persistent Master Encrption Key Cache
  The persistent master encryption key cache ensures the availability of TDE master encryption keys.

4.5.4.2 Upgrading a Standalone Oracle Key Vault Server

A single Oracle Key Vault server in a standalone deployment is the most typical deployment in test and development environments.

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

3. Ensure that SSH access is enabled.

   Log in to the Oracle Key Vault management console as a user who has the System Administrator role. Select the **System Settings** tab, then under Network Services, select **SSH Access**. Select **IP address(es)** and then enter only the IP addresses that you need. Click **Save**.

4. Ensure you have enough space in the destination directory for the upgrade ISO files.

5. Log in to the Oracle Key Vault server through SSH as user `support`, then switch user `su` to `root`.

   If the SSH connection times out while you are executing any step of the upgrade, then the operation will not complete successfully. Oracle recommends that you ensure tht you use the appropriate values for the `ServerAliveInterval` and `ServerAliveCountMax` options for your SSH sessions to avoid upgrade failures.

6. Copy the upgrade ISO file to the destination directory using **Secure Copy Protocol** or other secure transmission method.

   ```
   root# scp remote_host:remote_path/okv-upgrade-disc-18.3.0.0.0.iso /var/lib/
   oracle/destination_directory_for_iso_file
   ```

   In this specification:

   - `remote_host` is the IP address of the computer containing the ISO upgrade file
   - `remote_path` is the directory of the ISO upgrade file

7. Make the upgrade accessible by using the `mount` command:

   ```
   root# /bin/mount -o loop,ro /var/lib/oracle/okv-upgrade-disc-18.3.0.0.0.iso /
   images
   ```

8. Clear the cache using the `clean all` command:

```
root# yum -c /images/upgrade.repo clean all
```

9. Apply the upgrade with `upgrade.rb` 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.

   If you are performing an HSM upgrade using nCipher, at this point you must execute the following commands:

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

10. Restart the Oracle Key Vault server by running `reboot` command:

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

    On the first restart of the computer after the upgrade, the system will apply the necessary changes. This can take a few hours. Do not shut down the system during this time.

    The upgrade is completed when the screen with heading: `Oracle Key Vault Server 18.3.0.0.0` appears. The revision should reflect the upgraded release. Following 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.

11. Confirm that Oracle Key Vault has been upgraded to the correct version.

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

    b. Select the **System** tab, and then select **Status**.

    c. Verify that the version displayed is 18.3.0.0.0.

       The release number is also at the bottom of each page, to the right of the copyright information.

12. If your site uses the Commercial National Security Algorithm (CNSA) suite, then re-install these algorithms onto the standalone server.

13. Disable SSH access.

    Log in to the Oracle Key Vault management console as a user who has the System Administrator role. Select the **System Settings** tab, then under Network Services, select **Disabled**. Click **Save**.

Related Topics

- Upgrading a Standalone Oracle Key Vault Server with CNSA
  You can upgrade a standalone Oracle Key Vault while using Commercial National Security Algorithm (CNSA) compliance by upgrading and then executing the `okv_cnsa` script.

- *Oracle Key Vault Root of Trust HSM Configuration Guide*

4.5.4.3 Upgrading a Pair of Oracle Key Vault Servers in a Primary-Standby Deployment

You should allocate several hours to upgrade the primary server after upgrading the standby.

You must perform the upgrade standby and primary servers in one session with as little time between the standby and primary upgrade. 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 longer than several hours.

1. Prepare for the upgrade.
   - 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 shut down during the upgrade, if no persistent cache has been configured. With persistent cache enabled, endpoints will continue to be operational during the upgrade process.
   - Ensure that both the primary and standby systems have 8 GB memory.
2. Ensure that you have backed up the server you are upgrading so your data is safe and recoverable.

   You can use Oracle Backup and Recovery (Oracle RMAN) to perform this backup. Ensure that in the time between the backup and shutting down the Oracle Key Vault servers for upgrade, that no databases perform a set or rekey operation (for example, using the `ADMINISTER KEY MANAGEMENT` statement), since these new keys will not included in the backup.

   Do not proceed without completing this step.
3. First, upgrade the standby server while the primary server is running.

   Follow Step 2 through to Step 10 of the standalone mode upgrade process.
4. Ensure that the upgraded standby Oracle Key Vault server is restarted and running.
5. Upgrade the primary Oracle Key Vault server following Steps 1-10 of the standalone mode upgrade.

   After both the standby and primary Oracle Key Vault servers are upgraded, the two servers will automatically synchronize.
6. Log in to the Oracle Key Vault management console as a user with the System Administrator role.
7. Select the **System** tab, and then **Status**.
8. Verify that the **Version** field displays the new software version 18.3.0.0.0.
9. If your site uses the Commercial National Security Algorithm (CNSA) suite, then re-install these algrorithms onto the primary and standby servers.

Related Topics

- Upgrading a Standalone Oracle Key Vault Server
  A single Oracle Key Vault server in a standalone deployment is the most typical deployment in test and development environments.
- Upgrading Primary-Standby Oracle Key Vault Servers to Use CNSA
  You can upgrade Oracle Key Vault primary-standby servers while using Commercial National Security Algorithm (CNSA) compliance by upgrading and then executing the `okv_cnsa` script.
- *Oracle Database Backup and Recovery User's Guide*

4.5.5 Step 4: Upgrade the Endpoint Software

As part of the upgrade, you must reenroll endpoints created in earlier releases of Oracle Key Vault, or update the endpoint software.

If you are upgrading from an earlier release to the latest release of Oracle Key Vault, then you must reenroll the endpoint instead of upgrading the endpoint software. Reenrolling the endpoint automatically updates the endpoint software.

1. Ensure that you have upgraded the Oracle Key Vault servers. If you are upgrading the endpoint software for an Oracle database configured for direct-connect, then shut down 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. In the **Download Endpoint Software Only** section, select the appropriate platform from the drop-down list.
   d. Click the **Download** button.
3. Identify the path to your existing endpoint installation that you are about to upgrade (for example, `/home/oracle/okvutil`).
4. Install the endpoint software by executing the following command:

   ```
   java -jar okvclient.jar -d existing_endpoint_directory_path
   ```

   For example:

   ```
   java -jar okvclient.jar -d /home/oracle/okvutil
   ```

   If you are installing the `okvclient.jar` file on a Windows endpoint system that has Oracle Database release 11.2.0.4 **only**, then include the `-db112` option. (This option is not necessary for any other combination of endpoint platform or Oracle Database version.) For example:

   ```
   java -jar okvclient.jar -d /home/oracle/okvutil -v -db112
   ```

5. Install the updated PKCS#11 library file.
   This step is needed only for online TDE master encryption key management by Oracle Key Vault.
   - **On UNIX/Linux platforms:** Run `root.sh` from the `bin` directory of endpoint installation directory to copy the latest `liborapkcs.so` file for Oracle Database endpoints.

```
$ sudo $OKV_HOME/bin/root.sh
```

Or

```
$ su - root
# bin/root.sh
```

- **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. You will be prompted for the version of the database in use.

```
bin\root.bat
```

6. Restart the endpoint if it was shut down.

Related Topics

- Reenrolling an Endpoint
  When you reenroll an endpoint, the enrollment process automatically upgrades the endpoint software.

4.5.6 Step 5: If Necessary, Remove Old Kernels

Oracle recommends that you clean up the older kernels that were left behind after the upgrade.

While the older kernel is not in use, it may be marked as an issue by some code analysis tools.

1. Log in to the Oracle Key Vault server as the `support` user.
2. Switch to the `root` user.

```
su - root
```

3. Mount `/boot` if it was not mounted on the system.
   a. Check if the `/boot` is mounted. The following command should display `/boot` information if it was mounted.

```
df -h /boot;
```

   b. Mount it if `/boot` is not mounted.

```
/bin/mount /boot;
```

4. Check the installed kernels and the running kernel.
   a. Search for any kernels that are installed.

```
rpm -q kernel-uek | sort;
```

   The following example output shows that two kernels are installed:

```
kernel-uek-4.1.12-103.9.4.el6uek.x86_64
kernel-uek-4.1.12-112.16.7.el6uek.x86_64
```

   b. Check the latest kernel.

```
uname -r;
```

   The following output shows an example of a kernel version that was installed at the time:

```
4.1.12-112.16.7.el6uek.x86_64
```

This example assumes that `4.1.12-112.16.7.el6uek.x86_64` is the latest version, but newer versions may be available by now. Based on this output, you will need to remove the `kernel-uek-4.1.12-103.9.4.el6uek.x86_64` kernel. You should remove all kernels that are older than the latest kernel.

5. Remove the older kernel and its associated RPMs.

   For example, to remove the `kernel-uek-4.1.12-103.9.4.el6uek.x86_64` kernel:

   ```
   yum --disablerepo=* remove `rpm -qa | grep 4.1.12-103.9.4.el6uek`;
   ```

   Output similar to the following appears:

   ```
   Loaded plugins: security
   Setting up Remove Process
   Resolving Dependencies
   --> Running transaction check
   ---> Package kernel-uek.x86_64 0:4.1.12-103.9.4.el6uek will be erased
   ---> Package kernel-uek-devel.x86_64 0:4.1.12-103.9.4.el6uek will be erased
   ---> Package kernel-uek-firmware.noarch 0:4.1.12-103.9.4.el6uek will be
   erased
   --> Finished Dependency Resolution

   Dependencies Resolved

   ================================================================================
   ================================================================================
   ================================================
    Package
   Arch
   Version
   Repository
    Size
   ================================================================================
   ================================================================================
   ================================================
   Removing:
    kernel-uek
   x86_64
   4.1.12-103.9.4.el6uek                          @anaconda-
   OracleLinuxServer-201410181705.x86_64/6.6                          241 M
    kernel-uek-devel
   x86_64
   4.1.12-103.9.4.el6uek                          @anaconda-
   OracleLinuxServer-201410181705.x86_64/6.6                           38 M
    kernel-uek-firmware
   noarch
   4.1.12-103.9.4.el6uek                          @anaconda-
   OracleLinuxServer-201410181705.x86_64/6.6                          2.9 M

   Transaction Summary
   ================================================================================
   ================================================================================
   ================================================
   Remove        3 Package(s)

   Installed size: 282 M
   Is this ok [y/N]:
   ```

6. Enter y to accept the deletion output.

7. Repeat these steps starting with Step 4 for all kernels that are older than the latest kernel.

4.5.7 Step 6: If Necessary, Add Disk Space to Extend Swap Space

If you upgraded from an earlier release, you should extend swap space to accommodate the new Oracle Key Vault software.

By default, Oracle Key Vault releases earlier than release 18.1 were installed with approximately 4 GB of disk space. After you complete the upgrade to release 18.1, Oracle recommends that you increase the swap space allocation for the server on which you upgraded Oracle Key Vault. A new Oracle Key Vault installation is automatically configured with sufficient swap space. However, if you upgraded from a previous release, then you must manually add disk space to extend the swap space, particularly if the intention is to convert the upgraded server into the first node of a multi-master cluster.

1. Log in to the server in which you upgraded Oracle Key Vault and connect as `root`.
2. Check the current amount of swap space.

   ```
   [root@my_okv_server support]# swapon -s
   ```

   Output similar to the following appears. This example shows that the system has 4 GB of swap space.

   ```
   Filename Type Size Used Priority
   /dev/dm-0 partition 4194300 3368 -1
   ```

3. Check the amount of space on the system by executing the `vgdisplay` and `vgs` commands.

   a. Run the `vgdisplay` command.

      ```
      [root@my_okv_server support]# vgdisplay
      ```

      Output similar to the following appears. Note the values that are displayed after `Alloc PE` and `Free PE` (in **bold**).

      ```
      --- Volume group ---
      VG Name vg_root
      System ID
      Format lvm2
      Metadata Areas 1
      Metadata Sequence No 17
      VG Access read/write
      VG Status resizable
      MAX LV 0
      Cur LV 12
      Open LV 12
      Max PV 0
      Cur PV 1
      Act PV 1
      VG Size 2048.78 GiB
      PE Size 32.00 MiB
      Alloc PE / Size 7289 / 2027.78 GiB
      Free PE / Size 672 / 21.00 GiB
      VG UUID HGesFT-0JiY-C47e-kuVn-yzZ0-Htlw-KnUni0
      ```

   b. Run the vgs command.

      ```
      [root@my_okv_server support]# vgs
      ```

Output similar to the following appears.

```
VG #PV #LV #SN Attr VSize VFree
vg_root 1 12 0 wz--n- 2048.78g 21.00g
```

4. Follow these guidelines to determine if you need more swap space:
   - If the hard disk is equal to or greater than 1 TB in size, then you should have approximately 64 GB of swap space.
   - If the hard disk is less than 1 TB in size, then you should have approximately 20 to 25 percent of hard disk space set aside for swap space.

   If you need more swap space, then complete the rest of the steps in this procedure.
5. Shut down the Oracle Key Vault system server.
   a. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
   b. Select the **System** tab, and then select **System Settings**.
   c. Click the **Power Off** button.

   After you shut down the Oracle Key Vault server, you can add a new disk if needed, and then power the server back on.
6. Run the `fdisk -l` to find if there are any available partitions on the new disk.

   At this stage, there should be no available partitions.
7. Run the `fdisk` *`disk_device_to_be_added`* command to create the new partition.

   For example, to create a disk device named `/dev/sdb`:

   ```
   fdisk /dev/sdb
   ```

   In the prompts that appear, enter the following commands in sequence:

   `n` for new partition

   `p` for primary

   `1` for partition number

   Accept the default values for cylinder (press **Enter** twice)

   `w` to write and exit
8. Use the `pvcreate` *`disk_device_partition`* command to add the newly added disk to the physical volume.

   For example, for a disk device named `/dev/sdb1`, which is the name of the partition on that disk to be created (based on the name used for the disk device that was added).

   ```
   [root@my_okv_server support]# pvcreate /dev/sdb1
   ```

   Output similar to the following appears:

   ```
   Physical volume "/dev/sdb1" successfully created
   ```
9. Extend the logical volume with this disk space that you just added:

   ```
   [root@my_okv_server support]# vgextend vg_root /dev/sdb1
   ```

   Output similar to the following appears.

```
Volume group "vg_root" successfully extended
```

10. Check that the disk space has been successfully extended by running the `vgdisplay` and `vgs` commands again.

```
[root@my_okv_server support]# vgdisplay
 --- Volume group ---
 VG Name vg_root
 System ID
 Format lvm2
 Metadata Areas 2
 Metadata Sequence No 18
 VG Access read/write
 VG Status resizable
 MAX LV 0
 Cur LV 12
 Open LV 11
 Max PV 0
 Cur PV 2
 Act PV 2
 VG Size 328.75 GiB
 PE Size 32.00 MiB
 Total PE 10520
 Alloc PE / Size 7289 / 227.78 GiB
 Free PE / Size 3231 / 100.97 GiB
 VG UUID GeaZEb-Fivt-fFCv-i60c-x598-04Ot-J3GmEF

[root@my_okv_server support]# vgs
 VG #PV #LV #SN Attr VSize VFree
 vg_root 2 12 0 wz--n- 328.75g 100.97g
```

    This output indicates that the space allocation has increased after you added the new disk.

11. Disable swapping.

```
[root@my_okv_server support]# swapoff -v /dev/vg_root/lv_swap
```

12. To extend the swap space, run the `lvresize` command.

```
[root@my_okv_server support]# lvresize -L +60G /dev/vg_root/lv_swap
```

    Output similar to the following appears:

```
Size of logical volume vg_root/lv_swap changed from 4.00 GiB (128 extents)
to 64.00 GiB (2048 extents
Logical volume lv_swap successfully resized.
```

13. Format the newly added swap space.

```
[root@my_okv_server support]# mkswap /dev/vg_root/lv_swap
```

    Output similar to the following appears:

```
mkswap: /dev/vg_root/lv_swap: warning: don't erase bootbits sectors
on whole disk. Use -f to force.
Setting up swapspace version 1, size = 67108860 KiB
no label, UUID=fea7fc72-0fea-43a3-8e5d-e29955d46891
```

14. Enable swapping again.

```
[root@my_okv_server support]# swapon -v /dev/vg_root/lv_swap
```

15. Verify the amount of swap space that is available.

```
[root@my_okv_server support]# swapon -s
```

Output similar to the following appears.

```
Filename Type Size Used Priority

/dev/dm-0 partition 67108860 0 -1
```

16. Restart the Oracle Key Vault server.
    a. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
    b. Select the **System** tab, and then select **System Settings**.
    c. Click the **Reboot** button.

4.5.8 Step 7: If Necessary, Remove SSH-Related DSA Keys

You should remove SSH-related DSA keys left behind after the upgrade, because they can cause problems with some code analysis tools.

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

   Log in to the Oracle Key Vault management console as a user who has the System Administrator role. Select the **System Settings** tab, then under Network Services, select **SSH Access**. Select **IP address(es)** and then enter only the IP addresses that you need. Click **Save**.
3. Login to the Oracle Key Vault support account using SSH.

   ```
   ssh support@OracleKeyVault_serverIPaddress
   ```
4. Switch to the `root` user.

   ```
   su - root
   ```
5. Change directory to `/etc/ssh`.

   ```
   cd /etc/ssh
   ```
6. Rename the following keys.

   ```
   mv ssh_host_dsa_key.pub ssh_host_dsa_key.pub.retire
   mv ssh_host_dsa_key ssh_host_dsa_key.retire
   ```
7. Disable SSH access.

   Log in to the Oracle Key Vault management console as a user who has the System Administrator role. Select the **System Settings** tab, then under Network Services, select **Disabled**. Click **Save**.

4.5.9 Step 8: Back Up the Upgraded Oracle Key Vault Server

You must perform server backup and user password 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 than in earlier releases. 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.

Related Topics

- *Oracle Database Backup and Recovery User's Guide*

4.6 Upgrading Oracle Key Vault in a Multi-Master Cluster Environment

Similar to a standalone or primary-standby upgrade, this type of upgrade includes the Oracle Key Vault server software and endpoint software-related utilities.

- About Upgrading Oracle Key Vault in a Multi-Master Cluster Environment
  To perform this upgrade, you must upgrade each multi-master cluster node.
- Step 1: Perform Pre-Upgrade Tasks
  Similar to a standalone or primary-standby environment, you must prepare the Oracle Key Vault server for the pre-upgrade multi-master cluster process.
- Step 2: If Upgrading from Release 18.1, Run the Pre-Upgrade Script on Each Node
  If you are upgrading from Oracle Key Vault release 18.1, then run the pre-upgrade on each multi-master cluster node before performing the full upgrade.
- Step 3: Upgrade Each Multi-Master Cluster Node
  Do not use other Oracle Key Vault features until you have completed upgrading *all* multi-master cluster nodes.
- Step 4: Check the Node Version and the Cluster Version
  After you complete the upgrade of at least one node, you can log into any of the upgraded nodes to check the node and cluster versions.
- Rolling Back the Pre-Upgrade Script
  After you run the pre-upgrade script, you can roll it back if none of the nodes in the cluster have been successfully upgraded.

4.6.1 About Upgrading Oracle Key Vault in a Multi-Master Cluster Environment

To perform this upgrade, you must upgrade each multi-master cluster node.

The upgrade process involves two main steps: running a pre-upgrade script to prepare all the nodes for upgrade, and then performing the upgrade on each multi-master cluster node. If you are upgrading from Oracle Key Vault release 18.1, then you must run the pre-upgrade script. If you are upgrading from release 18.2 and later, then you must bypass running the pre-upgrade script. After you have begun a cluster upgrade, ensure that you upgrade all the nodes in the cluster one after the other, without too much intervening time between upgrades of two nodes. If you run the pre-upgrade script but then realize that you still must use the previous version of Oracle Key Vault, you can run a rollback script to undo the changes done by pre-upgrade script, so long

as no nodes have yet successfully been upgraded. You will need to run pre-upgrade again if you decide to proceed with the upgrade later.

Upgrading an Oracle Key Vault multi-master cluster includes upgrading each cluster node to the new later version. You must upgrade all nodes to the same Oracle Key Vault version. You should first upgrade the read-only nodes of the cluster, and then upgrade the read-write pairs. As each cluster node is upgraded, its node version is updated to the new version of the Oracle Key Vault. After you complete the upgrade of all cluster nodes, the cluster version is updated to the new version of the Oracle Key Vault. (You can check node version or the cluster version by selecting the **Cluster** tab, then in the left navigation bar, selecting **Management**.) Oracle Key Vault multi-master cluster upgrade is considered complete when node version and cluster version at each cluster node is updated to the latest version of Oracle Key Vault.

Before you perform the upgrade, note the following:

- Perform the entire upgrade process on *all* multi-master cluster nodes, without interruption. (That is, after you have started the cluster upgrade process, ensure that you try and upgrade all nodes, one after the other.) Do not perform other Oracle Key Vault activities until you have completed upgrading all the nodes in your environment.
- Be aware that you cannot use certain new features (for example, certificate rotation) until you have completed upgrading all of the multi-master cluster nodes. An error is returned when such features are used from the node that has been upgraded. Oracle recommends that you plan the upgrade of all cluster nodes close to each other to ensure availability of the new features sooner.

Related Topics

- Step 4: Check the Node Version and the Cluster Version
  After you complete the upgrade of at least one node, you can log into any of the upgraded nodes to check the node and cluster versions.

4.6.2 Step 1: Perform Pre-Upgrade Tasks

Similar to a standalone or primary-standby environment, you must prepare the Oracle Key Vault server for the pre-upgrade multi-master cluster process.

1. Back up the server so that you can recover data in case the upgrade fails.
2. Perform the pre-upgrade tasks that are described for standalone or primary-standby environments, which include tasks such as ensuring that the server meets the minimum disk space requirements, ensuring that no full or incremental backup jobs are runing, and planning for downtimes.

Related Topics

- *Oracle Database Backup and Recovery User's Guide*
- Step 2: Perform Pre-Upgrade Tasks
  To ensure a smooth upgrade to Oracle Key Vault, you should prepare the server you are upgrading.

4.6.3 Step 2: If Upgrading from Release 18.1, Run the Pre-Upgrade Script on Each Node

If you are upgrading from Oracle Key Vault release 18.1, then run the pre-upgrade on each multi-master cluster node before performing the full upgrade.

If you are upgrading from Oracle Key Vault release 18.2 or later, then you **must** bypass this step. The pre-upgrade script sets the stage for the upgrade by making some preparatory changes to the nodes that will be upgraded, such as updating the Oracle GoldenGate parameter files and blocking user operations. The `cluster_preupgrade_181.zip` file is available after you mount the upgrade ISO, at `/images/preupgrade/cluster_preupgrade_181.zip`.

1. Log in to the Oracle Key Vault server.
2. If necessary, enable SSH access.

   Log in to the Oracle Key Vault management console as a user who has the System Administrator role. Select the **System Settings** tab, then under Network Services, select **SSH Access**. Select **IP address(es)** and then enter only the IP addresses that you need. Click **Save**.
3. SSH into the multi-master cluster node in order to run the pre-upgrade script.

   ```
   ssh support@Oracle_Key_Vault_IP_address
   ```
4. Switch to the `root` user.

   ```
   su - root
   ```
5. Unzip the pre-upgrade files into the `/tmp` directory.

   ```
   /usr/bin/unzip /images/preupgrade/cluster_preupgrade_181.zip -d /tmp
   ```
6. Execute the pre-upgrade script.

   ```
   /tmp/cluster_preupgrade_181.sh
   ```
7. After you have successfully completed this procedure, repeat these pre-upgrade steps on all multi-master cluster nodes.

   After you complete these pre-upgrade steps, you are ready to perform the actual upgrade on each multi-master cluster node.

Related Topics

- Rolling Back the Pre-Upgrade Script
  After you run the pre-upgrade script, you can roll it back if none of the nodes in the cluster have been successfully upgraded.

4.6.4 Step 3: Upgrade Each Multi-Master Cluster Node

Do not use other Oracle Key Vault features until you have completed upgrading *all* multi-master cluster nodes.

You must perform these steps on each node of the cluster, one after the other.

1. SSH into the first multi-master cluster node that you want to upgrade.

   ```
   ssh support@Oracle_Key_Vault_IP_address
   ```
2. Disable the multi-master cluster node.

In the node's Management page (under the **Cluster** tab), the node's status will change from `DISABLING` to `DISABLED`.

3. Perform the upgrade as you would upgrade a standalone Oracle Key Vault server (but not a primary-standby pair).

   When you run the `/usr/bin/ruby /images/upgrade.rb --confirm` step during the upgrade, you will be asked to confirm that you completed the pre-upgrade steps.

4. After the node has been successfully upgraded, re-enable it.

   After you re-enable the disabled multi-master cluster node, its status changes from `DISABLED` to `ENABLING`, then to `ACTIVE`. The status of the node will remain at `ENABLING` and will not change to `ACTIVE` unless bidirectional replication between it and all other nodes is occurring successfully.

5. As necessary, disable SSH access on this node.

   Log in to the Oracle Key Vault management console as a user who has the System Administrator role. Select the **System Settings** tab, then under Network Services, select **Disabled**. Click **Save**.

6. After you have successfully completed this procedure, repeat these upgrade steps on all multi-master cluster nodes.

Related Topics

- Disabling a Cluster Node
  You can temporarily disable a cluster node, which is required for upgrades and maintenance.
- Step 3: Upgrade the Oracle Key Vault Server or Server Pair
  You can upgrade a standalone Oracle Key Vault server or a pair of Oracle Key Vault servers in a primary-standby deployment.
- Upgrading a Standalone or Primary-Standby Oracle Key Vault Server
  This upgrade includes the Oracle Key Vault server software and utilities that control the associated endpoint software.
- Enabling a Disabled Cluster Node
  You can enable any cluster node that was previously disabled. You must perform this operation from the disabled node.

4.6.5 Step 4: Check the Node Version and the Cluster Version

After you complete the upgrade of at least one node, you can log into any of the upgraded nodes to check the node and cluster versions.

Oracle Key Vault tracks the version information of each cluster node as well as the version of the cluster as a whole. The node version represents the version of the Oracle Key Vault software on a given node. When a node is upgraded, its node version is updated to the new version of the Oracle Key Vault software. The cluster version is derived from the version information of the cluster nodes and is set to the minimum version of any cluster node. During cluster upgrade, node version is updated as each cluster node is upgraded to the later version. When all of the cluster nodes have been upgraded, the cluster version is then updated to the new version. (The Cluster Version and Node Version fields are available in Oracle Key Vault release 18.2 and later.)

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **Cluster** tab.
3. In the left navigation bar, select **Management**.
4. Check the following areas:
   - To find the node version, check the Cluster Details area.
   - To find the cluster version, check the Cluster Information area.

4.6.6 Rolling Back the Pre-Upgrade Script

After you run the pre-upgrade script, you can roll it back if none of the nodes in the cluster have been successfully upgraded.

Remember that this pre-upgrade script is only necessary for an upgrade from Oracle Key Vault release 18.1. Upgrades from 18.2 and later do not need this pre-upgrade script run.

Do not roll back the pre-upgrade script if any nodes have been successfully upgraded. You may want to roll back the pre-upgrade script if, for example, you realize that you must still continue using the previous version of Oracle Key Vault. Another reason to roll back the pre-upgrade script is in the event that the upgrade on any node that you attempted to upgrade fails. Then you must roll back the pre-upgrade script from each cluster node. You will need to start the upgrade process from the beginning if you decide to upgrade Oracle Key Vault multi-master cluster later. If you choose to run the rollback script on one node, then you must run through the rollback steps on all other nodes as well, before you a) either continue working, or b) attempt to upgrade the cluster again.

1. Log in to the Oracle Key Vault server.
2. If SSH access is disabled, then enable it.

   Log in to the Oracle Key Vault management console as a user who has the System Administrator role. Select the **System Settings** tab, then under Network Services, select **SSH Access**. Select **IP address(es)** and then enter only the IP addresses that you need. Click **Save**.
3. SSH into the first multi-master cluster node where you want to perform the rollback operation.

   ```
   ssh support@Oracle_Key_Vault_IP_address
   ```
4. Switch to the `root` user.

   ```
   su - root
   ```
5. Execute the following command:

   ```
   /tmp/cluster_preupgrade_181.sh ROLLBACK
   ```
6. Disable SSH access.

   Log in to the Oracle Key Vault management console as a user who has the System Administrator role. Select the **System Settings** tab, then under Network Services, select **Disabled**. Click **Save**.

4.7 Overview of the Oracle Key Vault Management Console

The Oracle Key Vault management console provides a graphical user interface for System Administrators, Key Administrators, and Audit Managers.

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

- Setting up and managing the cluster
- 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 primary-standby
- Performing backups

4.8 Performing Actions and Searches

The Oracle Key Vault management console enables you to perform standard actions and search operations, as well as get help information.

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. The **Help** selection of the **Actions** list provides detailed help for using these features.

- Actions Menus
  The actions available from an **Actions** drop-down menu can vary but typically include a set of standard menu items.
- Search Bars
  Along with **Actions** menus, many tabs in the Oracle Key Vault management console contain search bars.

4.8.1 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.

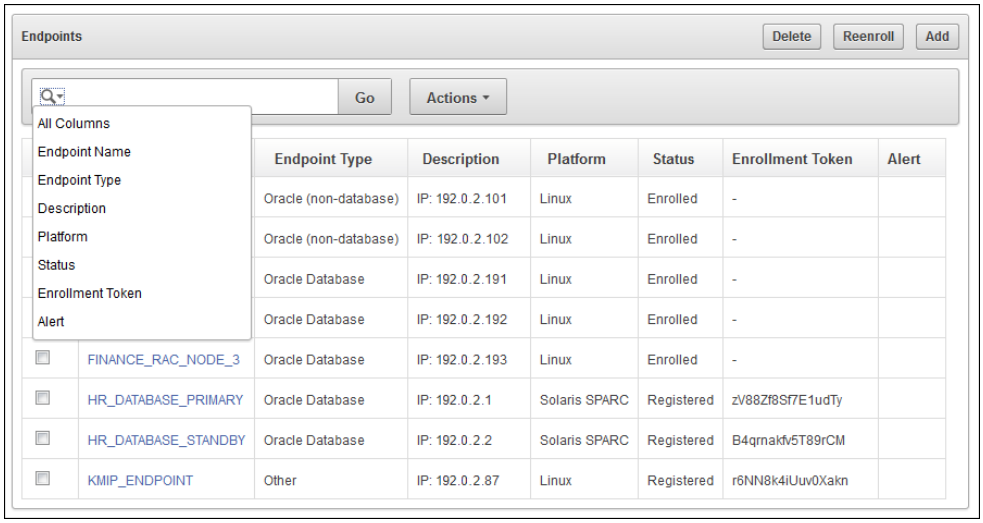
4.8.2 Search Bars

Along with **Actions** menus, many tabs in the Oracle Key Vault management console 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.

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 4-1 Endpoints Page



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.

5

Managing Oracle Key Vault Multi-Master Clusters

You can create, configure, manage, and administer an Oracle Key Vault multi-master cluster by using the Oracle Key Vault management console.

- About Managing Oracle Key Vault Multi-Master Clusters
  You can add or remove nodes from the cluster, disable or enable cluster nodes, and manage activities such as node conflicts and replication.
- Creating the First (Initial) Node of a Cluster
  To create a cluster, you must convert an existing standalone Oracle Key Vault server to become the first node in the cluster.
- Adding a Node to the Cluster
  You can create a read-write pair of nodes or a read-only node.
- Terminating the Pairing of a Node
  On the controller node, you can terminate the pairing process for a new node.
- Disabling a Cluster Node
  You can temporarily disable a cluster node, which is required for upgrades and maintenance.
- Enabling a Disabled Cluster Node
  You can enable any cluster node that was previously disabled. You must perform this operation from the disabled node.
- Deleting a Cluster Node
  You can permanently delete a node from the cluster.
- Force Deleting a Cluster Node
  You can permanently force delete a node from a cluster that is dead, unresponsive, or has exceeded the maximum disabled node time limit.
- Managing Replication Between Nodes
  You can enable and disable node replication from the Oracle Key Vault management console.
- Cluster Management Information
  The Cluster Management page provides a concise overview of the cluster and the status of each node.
- Cluster Monitoring Information
  The Cluster Monitoring page provides the replication health of the cluster and the current node.
- Naming Conflicts and Resolution
  Oracle Key Vault can resolve naming conflicts that can arise as users create objects such as endpoints, endpoint groups, and user groups.
- Multi-Master Cluster Deployment Recommendations
  Oracle provides deployment recommendations for deployments that have two or more nodes.

5.1 About Managing Oracle Key Vault Multi-Master Clusters

You can add or remove nodes from the cluster, disable or enable cluster nodes, and manage activities such as node conflicts and replication.

5.2 Creating the First (Initial) Node of a Cluster

To create a cluster, you must convert an existing standalone Oracle Key Vault server to become the first node in the cluster.

This first node is called the initial node. The standalone Oracle Key Vault server can also be a server that has been upgraded to Oracle Key Vault release 18.1 or later from a previous release or can also be the server that is unpaired from a primary-standby configuration. Check *Oracle Key Vault Release Notes* for known issues about unpair operations and upgrades.

You can use this node to add one or more nodes to the cluster. The node operates in read-only restricted mode until it is part of a read-write pair.

1. Perform a server backup.
2. Log into the Oracle Key Vault management console as a user who has the System Administrator role.
3. If the Oracle Key Vault server was upgraded from a release earlier than Oracle Key Vault release 12.2 (bundle patch 8) , then generate and activate (rotate) a new certificate for the node.
4. Select the **Cluster** tab.

   The Configure as Candidate Node page appears, with the IP address of the current server listed in the **Current Server IP** field.
5. On the Configure as Candidate Node page, enter the following information:
   - **First Node of Cluster**: Select the **Yes** button.
   - **Node Name**: Enter a unique name for this node. You cannot change this name after it has been accepted in the name resolution process.
   - **Cluster Name**: Enter a name for this cluster of nodes. You cannot change this name after it has been accepted in the name resolution process.
   - **Cluster Subgroup**: Enter a name for this sub-group of nodes, such as a data center name or a logical group name. You cannot change this name after it has been accepted in the name resolution process.
6. Click **Convert to Candidate Node**.

After the conversion is complete, the Cluster Management page is displayed and the node is now operating in read-only restricted mode.

Related Topics

- Rotating All Certificates
  You can use the Oracle Key Vault management console to rotate certificates.
- Managing Certificates
  In addition to Oracle Key Vault-generated certificates, you can manage third-party certificates.

- *Oracle Database Backup and Recovery User's Guide*
- *Oracle Key Vault Release Notes*

5.3 Adding a Node to the Cluster

You can create a read-write pair of nodes or a read-only node.

- Creating a Read-Write Pair of Nodes in a Cluster
  After you create the initial node, you must add an additional read-write peer to the cluster.
- Creating a Read-Only Node in a Cluster
  To add a new read-only cluster node, you pair any existing cluster node with a newly configured server.
- Creating an Additional Read-Write Pair in a Cluster
  Any node can be read-write paired with only one other node, and there can be multiple read-write pairs in a cluster.

5.3.1 Creating a Read-Write Pair of Nodes in a Cluster

After you create the initial node, you must add an additional read-write peer to the cluster.

You can configure any two nodes as a read-write pair. However, any single node can be read-write paired with only one other node. A node can have one or more read-only nodes connected to it, as long as the total number of nodes in the cluster does not exceed 16.

To create the read-write pair using two nodes in a cluster, you pair a node (referred to as the controller node) with a newly configured server (referred to as the candidate node). Note that this will take some time: an hour or more, depending on the speed of your server, network, and volume of data in the cluster.

1. Perform a backup of the controller node before continuing.
2. Ensure that the following network requirements are in place:
   - There is good network connectivity between the servers that host the controller node and the candidate node.
   - The ports that are required for Oracle Key Vault are open in the network firewall. These ports are described in Network Port Requirements.
3. Log into the controller node Oracle Key Vault management console as a user who has the System Administrator role.

   You can use any existing node, including the first node, that does not have a read-write peer to be the controller for this operation. If necessary, add a read-only node first.
4. Select the **Cluster** tab.
5. Click **Add**.
6. In **Recovery Passphrase of the Cluster**, enter the recovery passphrase.

   This value will be used later when you pair with the candidate node.
7. Select **Yes** for **Add Node as Read-Write Peer**.

8. Enter the following details under **Add Candidate Node Details**.

   While you enter these details, do not save any of this information or click the **Add Node** button until you reach Step 9.

   - **Candidate Node ID**: Select a unique ID for the candidate node. Remember that after you create this ID, you cannot change it.
   - **Candidate Node Name**: Enter a unique name of the candidate node. After you create this name, you cannot change it.
   - **Cluster subgroup Name for Candidate Node**: Enter the sub-group name for the candidate node. You can provide an existing subgroup name. If you provide a subgroup name that does not exist, it will be created. Remember that you cannot change the subgroup of this node after the node joins the cluster.
   - **IP Address of Candidate Node**: Enter the IP address of the candidate node.

9. In a new browser window, log into the Oracle Key Vault management console of the candidate node as a user who has the System Administrator role.
10. Select the **Cluster** tab.

    The Configure as Cluster Candidate page is displayed.

11. For **First Node of Cluster**, select **No**.
12. For **Recovery Passphrase of the Cluster**, enter the recovery passphrase of the cluster that you created earlier for the controller node.
13. For **IP Address of the Controller Node**, enter the IP address of the controller node.
14. In the browser window for the controller node, scroll to the bottom of the screen. Select and copy the entire node certificate.
15. In the browser window for the candidate node, paste the copied certificate from the controller node into the **Certificate of the Controller Node** field.

    Check the recovery passphrase, the IP address, and the pasted in certificate very carefully to ensure that you copied it correctly. If there is an error, after you click **Convert to Candidate Node**, you will need to reinstall Oracle Key Vault.

16. Click **Convert to Candidate Node**.

    After the conversion is complete, the screen will refresh and show the certificate of the candidate node. The **Adding Candidate Node to Cluster** page is displayed. This can take several minutes.

17. Select and copy the entire candidate node certificate.
18. In the browser window of the controller node, paste the copied certificate from the candidate node into the **Certificate of Candidate Node** box.
19. Click **Add Node**.
20. Click **OK** to confirm in the confirmation dialog box.

    This process will take an hour or more, depending on the speed of your server, network, and volume of data in the cluster. During this time, the network management interface of the Oracle Key Vault will be restarted and you might momentarily get a `Server Error 500` on the controller node. On the candidate node, errors may also appear, such as `Bad Gateway`. The candidate node will

restart as part of the induction process. This is normal. During the pairing process, the status of the candidate node will display as `PAIRING` on all cluster nodes.

To view the status of any server, view the output on the management console.

After the candidate node restarts, you can log in to either cluster node to view the cluster status by selecting the Cluster tab. Both nodes will now show as `ACTIVE`. The candidate node may briefly display that it is in read-only restricted mode after it automatically restarts. This node is now a synchronously paired cluster node and no longer a candidate node. After a node is part of a cluster, the console will display the node name, subgroup name, and cluster name in the top right area of the console header.

Related Topics

- *Oracle Database Backup and Recovery User's Guide*

5.3.2 Creating a Read-Only Node in a Cluster

To add a new read-only cluster node, you pair any existing cluster node with a newly configured server.

The existing cluster node is referred to as the controller node, and the newly configured server is referred to as the candidate node. This process will take an hour or more, depending on the speed of your server, network, and volume of data in the cluster.

1. Perform a server backup before continuing.
2. Ensure that the following network requirements are in place:
   - There is good network connectivity between the servers that host the controller node and the candidate node.
   - The ports that are required for Oracle Key Vault are open in the network firewall. These ports are described in Network Port Requirements.
3. Log into the controller node Oracle Key Vault management console as a user who has the System Administrator role.

   You can use any existing node as a controller for this operation.
4. Select the **Cluster** tab.
5. Ensure that **Management** is selected.
6. In the Cluster Details section click **Add**.
7. In the Add Cluster Details section, enter the cluster recovery passphrase in the **Recovery Passphrase of the Cluster** field.

   This value will be used later when pairing with the candidate node.
8. Under **Add Candidate Node Details**, enter the following information:
   - **Add Node as Read-Write Peer**: Select **No** .
   - **Node ID**: Select a unique ID for the candidate node. Remember that after you create this ID, you cannot change it.
   - **Node Name**: Enter a unique name of the candidate node. After you create this name, you cannot change it.

- **Cluster Subgroup**: Enter the subgroup name for the candidate node. You can provide an existing subgroup name. If you provide a subgroup name that does not exist, then it will be created. Remember that you cannot change the subgroup of this node after the node joins the cluster.
- **Cluster Name**: This name is populated automatically.
- **IP Address**: Enter the IP address of the candidate node.

Do not exit this page.

9. In a new browser window, log into the Oracle Key Vault management console of the candidate node as a user who has the System Administrator role.
10. Select the **Cluster** tab.

    The Configure as Cluster Candidate page appears.
11. For **First Node of Cluster**, select the **No** option.
12. For **Recovery Passphrase of the Cluster**, enter the same recovery passphrase of the cluster value that was entered for the controller node. This is the same value that was entered on the controller node.
13. For **IP Address of the Controller Node**, enter the IP address of controller node.
14. In the browser window for the controller node, scroll to the bottom of the screen. Select and copy the entire node certificate.
15. In the browser window for the candidate node, paste the copied certificate from the controller node into the **Certificate of the Controller Node** box.

    Check the recovery passphrase, the IP address, and the pasted in certificate very carefully to ensure that you copied it correctly. If there is an error, after you click **Convert to Candidate Node**, you will need to reinstall Oracle Key Vault.
16. Click **Convert to Candidate**.

    After the conversion is complete, the screen will refresh and show the certificate of the candidate node. The Adding Candidate Node to Cluster page is displayed. This can take several minutes.
17. Select and copy the entire candidate node certificate.
18. In the browser window of the controller node, paste the copied certificate from the candidate node into the **Certificate of Candidate Node** box.
19. Click **Add Node**.
20. Click **OK** to confirm in the confirmation dialog box.

    This process will take an hour or more, depending on the speed of your server, network, and volume of data in the cluster. During this time, the Oracle Key Vault console of the controller node will become unresponsive and can display an error such as `Server Error 500`. On the candidate node, errors may also appear, such as `Bad Gateway`. The candidate node will restart as part of the synchronization process. This is normal. During the pairing process, the status of the candidate node will display as `PAIRING` on all other cluster nodes not involved in this pairing process.

    To view the status of any server, view the output on the server console.

    After the candidate node restarts, you can log into either cluster node to view the cluster status by selecting the **Cluster** tab. Both nodes will now show as `ACTIVE`. The candidate node may briefly display that it is in read-only restricted mode after

it automatically restarts. This node is now a read-only paired cluster node and no longer a candidate node. After a node is part of a cluster, the console will display the node name, sub-group name, and cluster name in the top right area of the console header.

Related Topics

- *Oracle Database Backup and Recovery User's Guide*

5.3.3 Creating an Additional Read-Write Pair in a Cluster

Any node can be read-write paired with only one other node, and there can be multiple read-write pairs in a cluster.

1. Select a read-only cluster node as the controller node to pair with a new candidate node.
2. Follow the steps to create a read-write pair of nodes in a cluster.

Related Topics

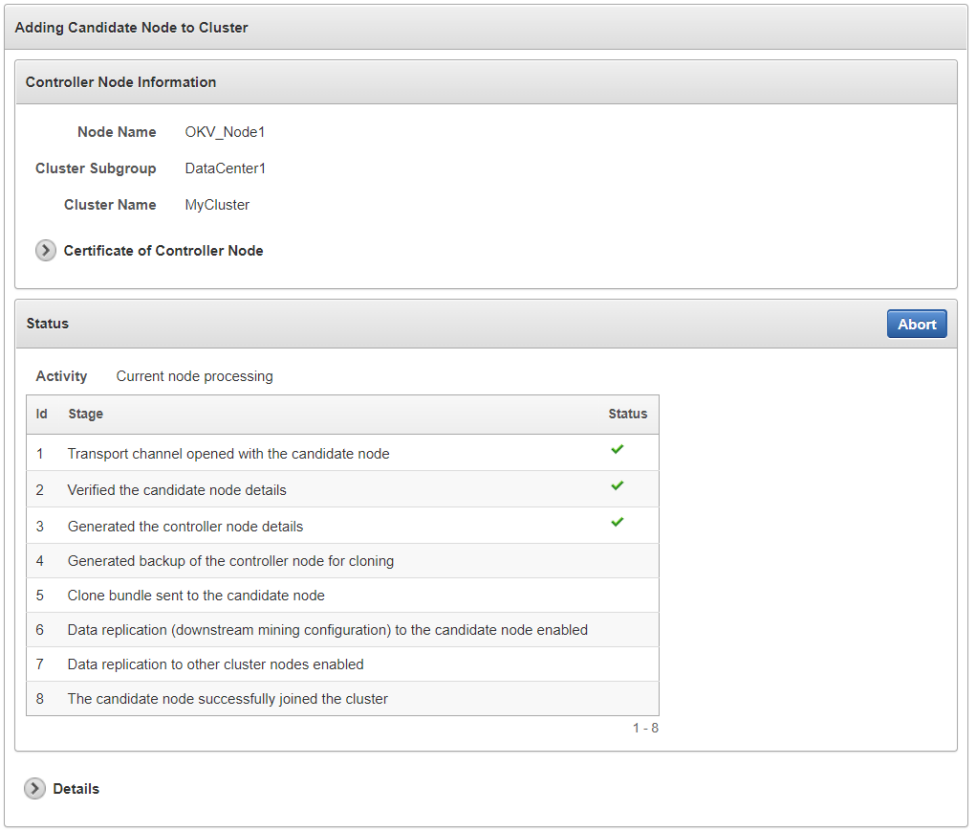
- Creating a Read-Write Pair of Nodes in a Cluster
  After you create the initial node, you must add an additional read-write peer to the cluster.

5.4 Terminating the Pairing of a Node

On the controller node, you can terminate the pairing process for a new node.

Be aware that when the controller node performs this operation, then it will experience network changes that will temporarily prevent it from serving endpoints.

1. On the controller node, in the **Status** section of the Adding Candidate Node to Cluster page, click the **Abort** button.



2. A dialog with the message **Are you sure you want to ABORT the addition of the new node?** will appear. Select **OK**.

   After the pairing process terminates, you will be returned to the **Add Node to Cluster** page on the controller node.

If you terminate the pairing of a candidate node, then the candidate node is no longer usable in its current state. If you want to use the server as a node, then you must re-image the server with the Oracle Key Vault appliance software. However, if it is early enough in the termination process (before any bundle parts have reached the candidate node from the controller node), then you can terminate the pairing on the candidate node. This returns the candidate node to its standalone state. In this case, you do not have to re-image the candidate node.

5.5 Disabling a Cluster Node

You can temporarily disable a cluster node, which is required for upgrades and maintenance.

However, be aware that a node can only be disabled for a set period of time. When it exceeds that time, it cannot be enabled again. The default maximum disable node duration time is 24 hours, but you can set it for as high as 99 hours.

1. Log into any cluster Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **Cluster** tab.
3. In the **Select Node** column, select the checkbox of the node to disable.
4. Click **Disable**.

On the node that is being disabled, the node status will display as `DISABLING` during the disabling process. The other nodes will display the status for this node as `DISABLED`. When the disabling process is complete, the node that you disabled displays the `DISABLED` status.

Related Topics

- Configuring Maximum Disable Node Duration for the Cluster
  You can set the Configuring Maximum Node Duration time for the cluster in hours.

5.6 Enabling a Disabled Cluster Node

You can enable any cluster node that was previously disabled. You must perform this operation from the disabled node.

1. As a user who has the System Administrator role, log into the Oracle Key Vault management console of any active node in the cluster.
2. Select the **Cluster** tab.

   By default, the Management page should appear.
3. In the Cluster Details section, note the dates in which the nodes have been disabled.

   Oracle recommends that you enable the nodes in the reverse order in which they were disabled. Otherwise, the enabling action may not be able to complete.
4. In the Cluster Details section, under Node Name, click the name of the node that was disabled most recently.

   Clicking the node name enables you to log in to this node. You can only enable disabled nodes from the disabled node itself.
5. On the Management page of the disabled node, select **Enable**.

   You do not need to check the checkbox of the disabled node in the Cluster Details table.
6. Repeat this step for each disabled node, from the most recent to the node that was disabled first.

5.7 Deleting a Cluster Node

You can permanently delete a node from the cluster.

Deleted nodes cannot be added back to any cluster, not just to the current cluster from which they were deleted. However, you can reinstall the Oracle Key Vault appliance software on this server and add the deleted node to a cluster. All data will be synchronized with the cluster before the node is deleted. A node cannot delete itself. Be aware that if a deleted node has a read-write peer, then this read-write peer node will experience network changes that will temporarily prevent it from serving endpoints.

1. As a user who has the System Administrator role, on a different node, log into the Oracle Key Vault management console.

   A node cannot delete itself.
2. Select the **Cluster** tab.
3. In the **Select Node** column, select the checkbox of the node to delete.

4. Click **Delete**.

   The node status will display as `DELETING`. After it is deleted, it will show as `DELETED`, and later be removed from the cluster management page.
   This action is immediate. The node status will display as `DELETING`. Do not shut down the deleted server until it no longer shows in the cluster status. However, Oracle recommends that you wait an hour after deleting a cluster node before reusing the node ID of the node that was deleted.

5.8 Force Deleting a Cluster Node

You can permanently force delete a node from a cluster that is dead, unresponsive, or has exceeded the maximum disabled node time limit.

Forcefully deleting a node that is still a part of a cluster may cause inconsistency in the cluster. Be aware that if the read-write peer of the node that was forcefully deleted is also removed from the cluster before confirming that all critical data from the forcefully deleted node has reached other nodes, then data loss can result. When you forcefully delete a node, ensure that the node to be deleted has first been shut down. A node cannot be deleted from its own management console. When you must forcefully delete a node, ensure that the node to be deleted has first been shut down. Deleted nodes cannot be added back to the cluster. However, you can reinstall the Oracle Key Vault appliance software on a server and then add the deleted node to a cluster. Be aware that if a deleted node has a read-write peer, then this read-write peer node will experience network changes that will temporarily prevent it from serving endpoints.

1. On a different node, log into the Oracle Key Vault management console as a user who has the System Administrator role.

   A node cannot delete itself. Oracle recommends that if the node to be deleted has a read-write peer, to force delete the node from its read-write peer.

2. Select the **Cluster** tab.
3. In the **Select Node** column, select the checkbox of the node to force delete.
4. Click **Force Delete**.

   The node status will display as `DELETING`. After it is deleted, it will show as `DELETED`, and later be removed from the cluster management page. Oracle recommends that you wait an hour after force deleting a cluster node before reusing the node ID of the node that was deleted.

5.9 Managing Replication Between Nodes

You can enable and disable node replication from the Oracle Key Vault management console.

- Restarting Cluster Services
  While managing replication between nodes, you can restart a node's cluster services when the cluster service status for the node is down.
- Disabling Node Replication
  You can disable the replication link between the current node and any other node in a cluster.

- Enabling Node Replication
  You can enable the replication link between the current node and any other node in a cluster.

5.9.1 Restarting Cluster Services

While managing replication between nodes, you can restart a node's cluster services when the cluster service status for the node is down.

1. Log into Oracle Key Vault management console of any cluster node as a user who has the System Administrator role.
2. Select the **Cluster** tab, and then **Monitoring** from the left navigation bar.
3. In the Node State pane, click the **Restart Cluster Services** button.

5.9.2 Disabling Node Replication

You can disable the replication link between the current node and any other node in a cluster.

1. Log into Oracle Key Vault management console of any cluster node as a user who has the System Administrator role.
2. Select the **Cluster** tab, and then **Monitoring** from the left navigation bar.
3. Select the nodes for which you want to disable replication.
4. Click **Disable**.
5. Click **OK** to confirm in the dialog box.

5.9.3 Enabling Node Replication

You can enable the replication link between the current node and any other node in a cluster.

1. As a user who has the System Administrator role, log in to the Oracle Key Vault management console of the node for which replication should be managed.
2. Select the **Cluster** tab, and then **Monitoring** from the left navigation bar.
3. Select the nodes for which you want to enable replication.
4. Click **Enable**.
5. Click **OK** to confirm in the dialog box.

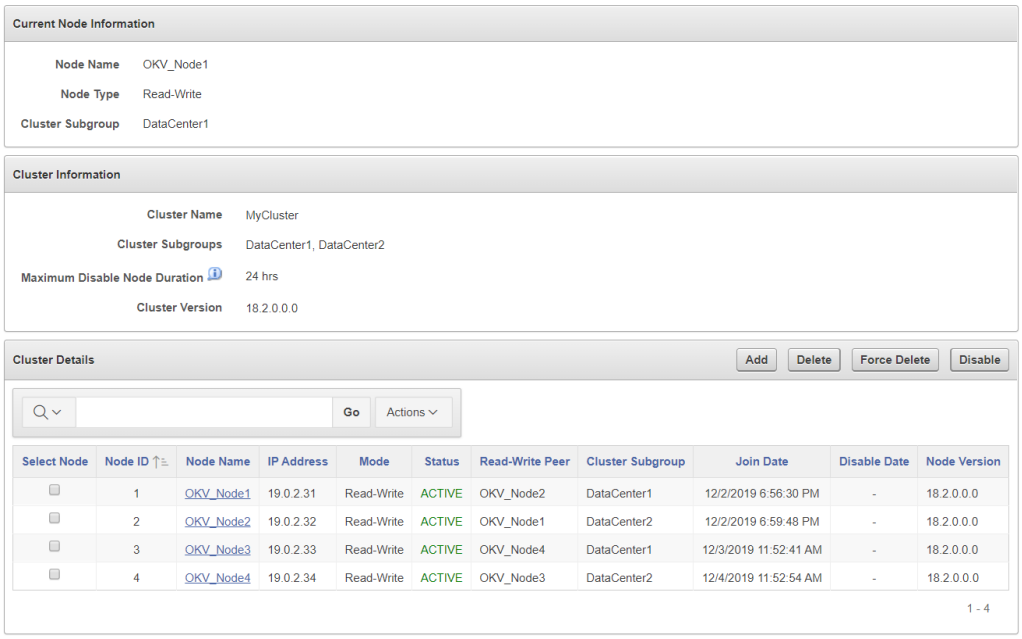
5.10 Cluster Management Information

The Cluster Management page provides a concise overview of the cluster and the status of each node.

You can also manage the cluster from the cluster details section. When a node is performing a cluster operation it becomes the controller node.

Be aware that because the replication across the cluster takes time, there may be a delay before the Cluster Management page refreshes with the new cluster status. The replication lag in the monitoring page will help estimate the delay.

To view the Cluster Management page, click the **Cluster** tab, and then **Management** from the left navigation bar.



Current Node Information

- **Node Name**: The name of this node.
- **Node Type:** The type of node, such as whether it is read-only or read-write.
- **Cluster Subgroup**: The subgroup to which this node belongs.

Cluster Information

- **Cluster Name**: The name of the cluster.
- **Cluster Subgroups**: All subgroups within the cluster.
- **Maximum Disable Node Duration**: The maximum time, in hours, that a node can be disabled before it is evicted from the cluster.
- **Cluster Version**: The version of Oracle Key Vault in which the cluster is operating.

Cluster Details

- **Select Node**: Used to select a node for a specific operation, such as delete, force delete, or disable.
- **Node ID**: The ID of the node.
- **Node Name**: The name of the node. Clicking the node name takes you to the Cluster Management page of that node.
- **IP Address**: The IP address of the node.
- **Mode**: The type of node, such as read-write, read-only, or read-only restricted.
- **Status**: The status of the node, such as active, pairing, disabling, disabled, enabling, deleting, or deleted.
- **Read-Write Peer**: The read-write peer of the node. If blank, it has no read-write peer.

- **Cluster Subgroup**: The subgroup to which the node belongs.
- **Join Date**: The date and time that the node was added to the cluster.
- **Disable Date**: The date and time that the node was disabled.
- **Node Version**: The current version of the Oracle Key Vault node.

5.11 Cluster Monitoring Information

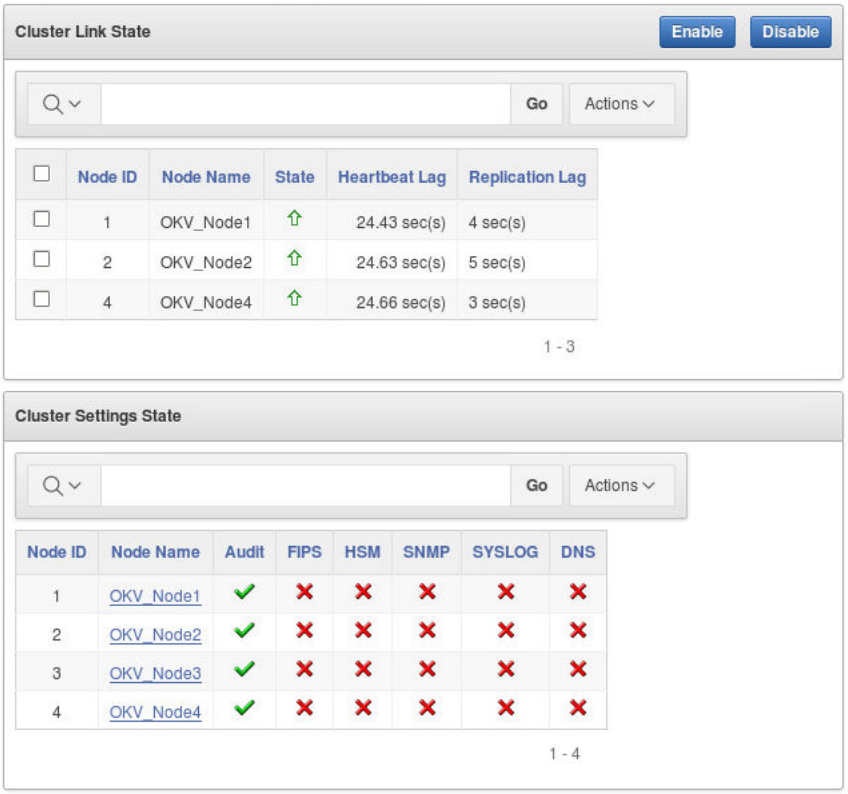
The Cluster Monitoring page provides the replication health of the cluster and the current node.

This page also provides a concise overview of the settings enabled in the cluster. You cannot update the settings on this page. Because the replication across the cluster takes time, there may be a delay before the Cluster Monitoring page refreshes with the new cluster state. Replication lag will help estimate the delay.

To view the cluster monitoring page, click the **Cluster** tab, and then **Monitoring** from the left navigation bar.

You can hover the mouse over the checkmarks or X's in the Cluster Settings State pane. It will display one of the following explanations of the state:

- Enabled in Cluster
- Enabled in Node
- Disabled in Cluster
- Disabled in Node



Cluster Link State

- **Select Node**: Used to select nodes for a specific operation, such as enabling or disabling replication. You can click the checkbox on the label row to select all nodes.
- **Node ID**: The ID of the node.
- **Node Name**: The name of the node.
- **State**: The current state of the node. The server is either up or down.
- **Heartbeat Lag**: The average time of the heartbeat.
- **Replication Lag**: The average time to replicate an object.
- **Enable**: Enables the replication between the current node and the node selected.
- **Disable**: Disables the replication between the current node and the node selected.

Cluster Settings State

- **Node ID**: The ID of the node.
- **Node Name**: The name of the node.
- **Audit**: Indicates if auditing is enabled or disabled.
- **FIPS**: Indicates if FIPS mode is enabled or disabled.
- **HSM**: Indicates if HSM integration is enabled or disabled.
- **SNMP**: Indicates if SNMP is enabled or disabled.
- **SYSLOG**: Indicates if syslog is enabled or disabled.
- **DNS**: Indicates if DNS is enabled or disabled.

5.12 Naming Conflicts and Resolution

Oracle Key Vault can resolve naming conflicts that can arise as users create objects such as endpoints, endpoint groups, and user groups.

- About Naming Conflicts and Resolution
  If you create an object that has the same name as another object on another node, Oracle Key Vault resolves this conflict.
- Naming Conflict Resolution Information
  The Cluster Conflict Resolution page provides a list of objects with names that conflict with objects created on different nodes.
- Changing the Suggested Conflict Resolution Name
  You can change the suggested name for an object that conflicts with another object of the same type.
- Accepting the Suggested Conflict Resolution Name
  You can accept the suggested name for an object name that conflicts with another object of the same type.

5.12.1 About Naming Conflicts and Resolution

If you create an object that has the same name as another object on another node, Oracle Key Vault resolves this conflict.

You can create a new object with a name that conflicts with an object of the same type created on another node. If a conflict happens, then Oracle Key Vault will make the name of the conflicting object unique by adding `_OKV`*xx*, where *xx* is a node number, to the end of the user-supplied name. You can choose to accept this new name or change the object name.

System administrators can resolve the following naming conflicts:

- User names
- Endpoint names

Key administrators can resolve the following naming conflicts:

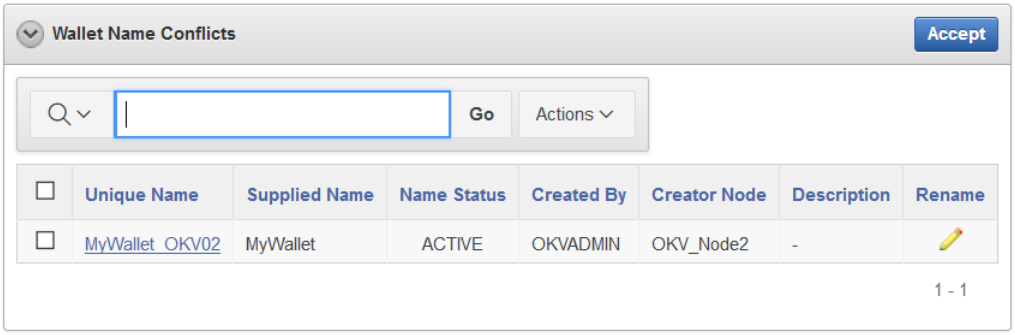
- Endpoint groups
- Security objects
- User groups
- Wallets

If an object is stuck in the `PENDING` state and will not transition to `ACTIVE`, then check for any broken replication links in the cluster. You can find cluster links in the Oracle Key Vault management console by selecting the **Cluster** tab and then selecting **Monitoring**.

5.12.2 Naming Conflict Resolution Information

The Cluster Conflict Resolution page provides a list of objects with names that conflict with objects created on different nodes.

On this page, you can accept the suggested unique name or edit the object name. To view the Cluster Conflict Resolution page, click the **Cluster** tab, and then **Conflict Resolution** from the left navigation bar. Alternatively, you can click on the **Click here for details** button on a Naming Conflict alert from the Alerts table on the Home page.



Wallet Name Conflicts

- **Unique Name**: The unique name assigned to the object by the system.
- **Supplied Name**: The original name of the object that conflicts with another object of this type.
- **Name Status**: The status of the object. The status can be `PENDING` or `ACTIVE`.
- **Created By**: The user that created the conflicting object name.
- **Creator Node**: The node that the conflicting object was created on.
- **Description**: The description of the object as entered by the user.
- **Rename**: The button that links to the object page where it can be renamed.
- **Accept**: Allows you to accept the suggested name for the selected objects.

5.12.3 Changing the Suggested Conflict Resolution Name

You can change the suggested name for an object that conflicts with another object of the same type.

1. As a user who has the appropriate administrator role, log in to the Oracle Key Vault management console.
2. Select the **Cluster** tab, and then **Conflict Resolution** from the left navigation bar.
3. Locate the object that requires a name change.
4. Click the edit icon to the right of the object.
5. On the object overview page, enter the new name for the object.
6. Click **Save**.

5.12.4 Accepting the Suggested Conflict Resolution Name

You can accept the suggested name for an object name that conflicts with another object of the same type.

1. As a user who has the appropriate administrator role, log in to the Oracle Key Vault management console.
2. Select the **Cluster** tab, and then **Conflict Resolution** from the left navigation bar.
3. Select the objects for which you want to accept the suggested name.
4. Click **Accept**.

5.13 Multi-Master Cluster Deployment Recommendations

Oracle provides deployment recommendations for deployments that have two or more nodes.

Two-Node Deployment Recommendations

Use a two-node deployments for the following situations:

- Non-critical environments, such as test and development

- Simple deployment of read-write pairs with both nodes active, replacing classic primary-standby
- Single data center environments

Considerations for a two-node deployment:

- Availability is provided by multiple nodes.
- Maintenance will require down time.
- Good network connectivity between data centers is mandatory.

Three-Node Deployment Recommendations

Use a three-node deployment for the following situations:

- Single data center environments with minimal downtime requirement
- Single read-write pair with additional read-only node to handle load
- One read-only node is available for zero downtime during maintenance

Considerations for a three-node deployment:

- Take regular backups to remove destinations for disaster recovery.

Four or More Node Deployment Recommendations

Use a deployment of four or more nodes for the the following situations:

- Large data centers distributed across geographical locations
- Deployment of read-write pairs with pair members spanning geography

Considerations for a large deployment:

- Availability is provided by multiple nodes.
- Additional read-only nodes can be used to handle load.
- Good network connectivity between data centers is mandatory.

6

Managing an Oracle Key Vault Primary-Standby Configuration

You can deploy Oracle Key Vault in a primary-standby server configuration.

- Overview of the Oracle Key Vault Primary-Standby Configuration
  The Oracle Key Vault primary-standby configuration provides benefits based on the type of deployment your site needs.
- Configuring the Primary-Standby Environment
  To configure a primary-standby environment, you must have the System Administrator role and have access to the two servers (one primary and one standby).
- Switching the Primary and Standby Servers
  You can switch the roles of the primary and standby server for situations such as maintenance periods.
- Restoring Primary-Standby After a Failover
  A failover takes place if the primary server fails.
- Disabling (Unpairing) the Primary-Standby Configuration
  You can disable the primary-standby configuration by unpairing the primary and standby servers.
- Read-Only Restricted Mode in a Primary-Standby Configuration
  The read-only restricted mode is the default mode in a primary-standby configuration.
- Best Practices for Using Oracle Key Vault in a Primary-Standby Configuration
  Oracle provides guidelines for ensuring operational continuity and minimal downtime of Oracle Key Vault.

6.1 Overview of the Oracle Key Vault Primary-Standby Configuration

The Oracle Key Vault primary-standby configuration provides benefits based on the type of deployment your site needs.

- About the Oracle Key Vault Primary-Standby Configuration
  You configure a primary-standby environment by providing the primary and standby servers with each other's IP address and certificate, and then pairing them.
- Benefits of an Oracle Key Vault Primary-Standby Configuration
  The benefits of an Oracle Key Vault primary-standby configuration include high availability, necessary for business-critical operations.

- Difference Between Primary-Standby Configuration and Multi-Master Cluster
  In both primary-standby and multi-master cluster configurations, one server will always operate in read-write mode.
- Primary Server Role in a Primary-Standby Configuration
  A primary-standby deployment consists of two Oracle Key Vault servers operating in a primary-standby configuration.
- Standby Server Role in a Primary-Standby Configuration
  In a primary-standby environment, one server runs in the standby server role.

6.1.1 About the Oracle Key Vault Primary-Standby Configuration

You configure a primary-standby environment by providing the primary and standby servers with each other's IP address and certificate, and 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 the primary server.

> **Note:**
>
> Oracle strongly recommends that you keep the primary and standby systems as identical as possible, because their roles can be reversed in maintenance periods and failure situations. These include the following:
>
> - Oracle Key Vault software versions
> - Disk size
> - RAM size
> - System clocks on both systems must be synchronized

If your deployment requires a primary-standby configuration, then Oracle recommends that you configure it *before* adding endpoints to Oracle Key Vault. This enables the endpoints to know about both the primary and standby servers. An endpoint that is added before the standby server configuration will not know about the standby server, unless you re-enroll the endpoint. If you configure the primary-standby environment after adding endpoints, then you must re-enroll the endpoints to ensure the endpoints recognize both servers that were previously enrolled with the primary and standby servers in standalone mode.

> **WARNING:**
>
> Configure primary-standby deployments before adding endpoints to ensure that the endpoints know about both nodes.

If you want to add SNMP support in a primary-standby environment, then ideally, configure SNMP on both the primary and the 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 also add SNMP support to the standby after pairing the servers by accessing the standby using SSH.

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

If you want to enable FIPS mode in a primary-standby environment, then you must ensure that both the primary and standby servers use the same FIPS mode: either both are enabled, or both are disabled for FIPS mode. 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.

With persistent cache enabled, both the primary and the standby will cache the master encryption keys from Oracle Key Vault independently. Ensure that TDE operations have executed on the primary and standby servers after these servers have started to verify the persistent cache. The persistent cache feature also enables endpoints to be operational during primary-standby operations, such as configuration, switchovers, and failovers.

If enabled, read-only restricted mode ensures endpoint operational continuity (such as enabling the endpoints to fetch keys) if either the standby or primary server is not available. For example, if the standby shuts down, then the primary will go into read-only restricted mode and enable the endpoints to fetch keys and continue operations.

A primary-standby configuration is characterized by continuous synchronization between the primary server and the 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 the primary-standby environment by selecting the **Allow Read-Only Restricted Mode** option to **Yes** or **No** on the Configure Primary-Standby 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 allows keys to be retrieved, but does 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. You can use the persistent master encryption key cache feature to avoid endpoint downtime. With this feature, data integrity is ensured by allowing endpoints to communicate with one primary server at any given time. This avoids split-brain situations, and the risk of data loss associated with such situations.

Related Topics

- Changing SNMP Settings on the Standby Server
  You change the SNMP settings from the command line on the standby server.

6.1.2 Benefits of an Oracle Key Vault Primary-Standby Configuration

The benefits of an Oracle Key Vault primary-standby configuration include high availability, necessary for business-critical operations.

Users performing business-critical operations must have data to be accessible and recoverable with minimum downtime. These requirements are met in a primary-standby configuration.

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. This is a significant reason to deploy Oracle Key Vault in a primary-standby configuration. In a classic primary-standby configuration, the emphasis is on key preservation. In a multi-master cluster, emphasis is on both key preservation and availability of the keys.

You can create a cluster of Oracle Key Vault server nodes for greater availability and redundancy. A primary-standby configuration is limited to two servers, whereas a multi-master cluster can have up to 16 geographically distributed nodes. The primary-standby configuration and the multi-master configuration are mutually exclusive.

Related Topics

- Oracle Key Vault Multi-Master Cluster Overview
  The multi-master cluster nodes provide high availability, disaster recovery, load distribution, and geographic distribution to an Oracle Key Vault environment.

6.1.3 Difference Between Primary-Standby Configuration and Multi-Master Cluster

In both primary-standby and multi-master cluster configurations, one server will always operate in read-write mode.

In a primary-standby configuration, when both servers are available, one of the servers operates in read-write mode in the primary server role, and the other operates in the standby server role. The endpoints only connect to the server running in the primary server role. The roles can be switched manually to support maintenance operations, or automatically due to server or connectivity failure. If either the primary or standby server becomes unavailable, then the remaining server operates in a read-only restricted mode, limiting normal updates while allowing audits and other internal updates.

In a multi-master cluster, the endpoints can connect to any Oracle Key Vault server. Some servers are configured as bi-directional read-write pairs in which information updated in either node must be successfully replicated to the other node immediately. If one of the nodes in a read-write pair becomes unavailable, the surviving node operates in read-only restricted mode until the other node is restored and synchronization resumes. A fully functional multi-master cluster must have at least one read-write pair.

When a successful update occurs in a read-write pair, the update is propagated to all other nodes in the cluster.

A primary-standby configuration and a multi-master cluster configuration are mutually exclusive and incompatible configurations. The specific configuration of an Oracle Key Vault deployment has no ramification on the endpoint side configuration.

6.1.4 Primary Server Role in a Primary-Standby Configuration

A primary-standby deployment consists of two Oracle Key Vault servers operating in a primary-standby configuration.

By default, endpoints only connect to the primary server until is becomes unavailable. At any time, only one server operates in the primary server role and that server actively accepts client connections. The other server operates in the standby server role, which receives updates from the primary server. On failure of the server running in the primary server role, the standby assumes the primary role. There may be restrictions in operations if the primary-standby pair is not fully available and operational.

6.1.5 Standby Server Role in a Primary-Standby Configuration

In a primary-standby environment, one server runs in the standby server role.

This standby server does not accept client connections while in that role. The server receives updates only from the paired server running in primary server role. If the primary server is no longer available, including being available to the administrator, then the server running in the standby role switches to assume the primary server role. There may be restrictions in operations if the primary-standby pair is not fully available and operational.

6.2 Configuring the Primary-Standby Environment

To configure a primary-standby environment, you must have the System Administrator role and have access to the two servers (one primary and one standby).

After you complete the configuration, with persistent cache enabled, endpoints will continue to operate while the primary-standby configuration is enabled.

- Step 1: Configure the Primary Server
  To configure the primary server, you must enable it to connect to the standby server.
- Step 2: Configure the Standby Server
  To configure the standby server, you must enable it to connect to the primary server.
- Step 3: Complete the Configuration on the Primary Server
  After you configure the primary and standby servers, you can enable the primary-standby on the designated primary server.

6.2.1 Step 1: Configure the Primary Server

To configure the primary server, you must enable it to connect to the standby server.

If you plan to configure an HSM (such as SafeNet, nCipher, or Utimaco) with Oracle Key Vault, then you must first enable this HSM in Oracle Key Vault before configuring the primary server.

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. Check if the server has FIPS mode enabled, and if necessary, enable or disable it.

   You must ensure that both the primary and standby servers use the same FIPS mode setting: either both are enabled, or both are disabled, for FIPS mode. Changing the FIPS mode setting requires a restart of Oracle Key Vault.

   a. Select the **System** tab, and then select **System Settings** in the left pane.

   b. In the FIPS Mode section, either select or clear the **Enable** check box, depending on whether both servers will use FIPS mode.

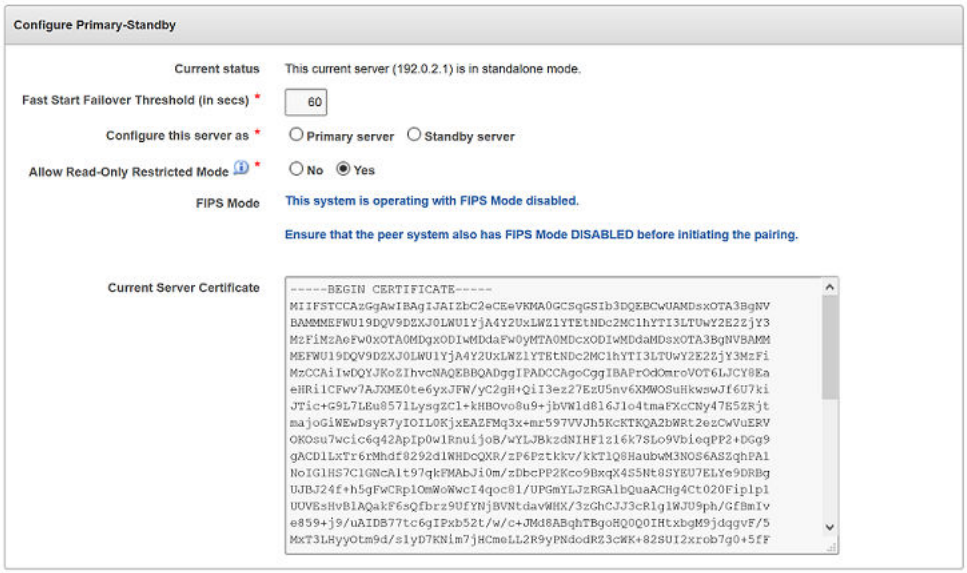
   c. Click **Save**.

      In a moment, Oracle Key Vault will restart.

4. If you changed the FIPS mode, then log back into the designated primary server as a user who has the System Administrator role.

5. Click the **System** tab, then click **Primary-Standby** in the left pane.

   The Configure Primary-Standby page appears.



The following are the fields on the Configure Primary-Standby 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

- **FIPS Mode:** Displays the current FIPS mode status of the server.
- **Current Server Certificate**: Displays the server certificate.

6. Copy the following information, and then store it in a text file named `primary.txt`.

   You will need 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 on a new line in `primary.txt` after the IP address.

   Save `primary.txt`.

Next, you are ready to configure the standby server.

Related Topics

- Step 2: Configure the Standby Server
  To configure the standby server, you must enable it to connect to the primary server.
- *Oracle Key Vault Root of Trust HSM Configuration Guide*Configuring HSM

6.2.2 Step 2: Configure the Standby Server

To configure the standby server, you must enable it to connect to the primary server.

If you plan to configure an HSM (such as SafeNet, nCipher, or Utimaco) with Oracle Key Vault, then you must first enable this HSM in Oracle Key Vault before configuring the standby server.

1. Open a web browser and enter the IP address of the designated standby server.

   The **Oracle Key Vault Management Console login screen** is displayed.

2. Log in as the System Administrator.
3. Check if the server has FIPS mode enabled, and if necessary, enable or disable it.

   You must ensure that both the primary and standby servers use the same FIPS mode setting: either both are enabled, or both are disabled, for FIPS mode. Changing the FIPS mode setting requires a restart of Oracle Key Vault.

   a. Select the **System** tab, and then select **System Settings** in the left pane.
   b. In the FIPS Mode section, either select or clear the **Enable** check box, depending on whether both servers will use FIPS mode.
   c. Click **Save**.

      In a moment, Oracle Key Vault will restart.

4. If you changed the FIPS mode, then log back into the designated standby server as a user who has the System Administrator role.
5. Click the **System** tab, then click **Primary-Standby** in the left pane.

   The Configure Primary-Standby page is displayed.

6. Copy the following information, and store it in a text file named `standby.txt`.

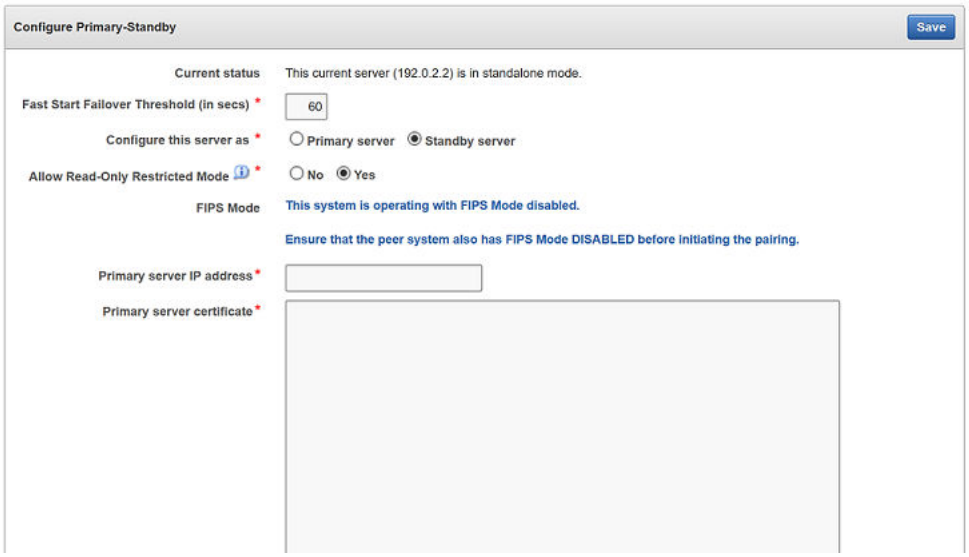
You will need this information when you configure the primary server.

- 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 on a new line in `standby.txt` after the IP address.

Save `standby.txt`.

7. In the **Configure this server as** field, select **Standby server**.

   The **Primary server IP address** and **Primary server certificate** fields are displayed.

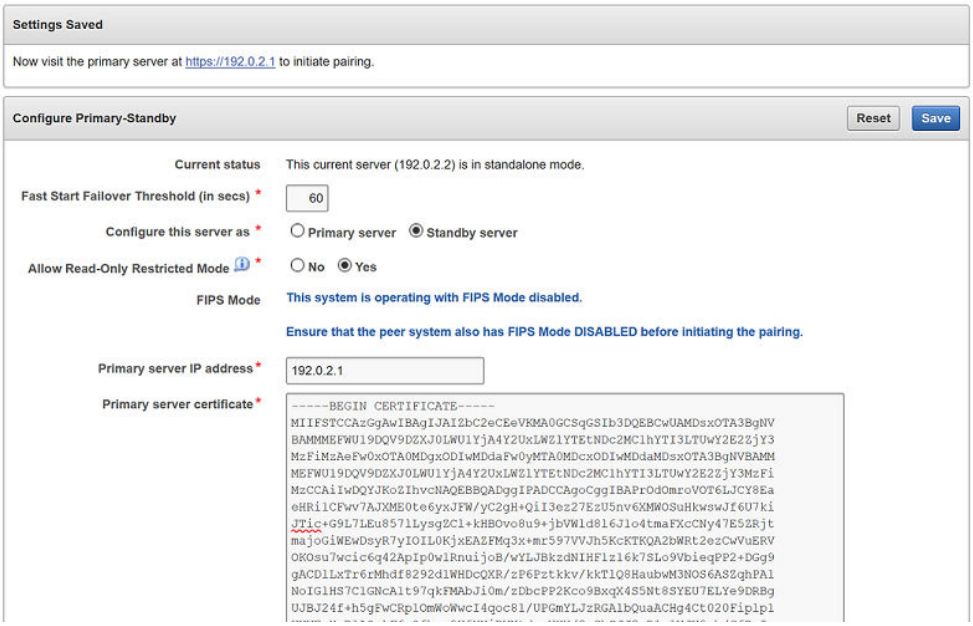


   Ensure that **Yes** is selected in the **Allow Read-Only Restricted Mode** field.

   Do not disable read only restricted mode unless necessary. If the primary-standby configuration is configured with read only restricted mode disabled, then you must enable it by reinstalling and configuring Oracle Key Vault again.

8. Copy the following information from `primary.txt`, and paste it in the **Configure Primary-Standby** 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.
9. Click **Save**.

   The **Settings Saved** page is displayed.



The **Reset** button enables you to delete the primary-standby configuration, if necessary.

10. Do not exit the management console.

At this stage, the primary-standby configuration is complete on the designated standby server. The next step is to enable primary-standby on the designated primary server.

Related Topics

- Step 3: Complete the Configuration on the Primary Server
  After you configure the primary and standby servers, you can enable the primary-standby on the designated primary server.
- *Oracle Key Vault Root of Trust HSM Configuration Guide*Configuring HSM

6.2.3 Step 3: Complete the Configuration on the Primary Server

After you configure the primary and standby servers, you can enable the primary-standby on the designated primary server.

1. Ensure that you are logged in to the standby server as a user with the System Administrator Role and that the Oracle Key Vault management console Configure Primary-Standby page is displayed.
2. 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.
3. Log in as the System Administrator.
4. Click the **System** tab, then click **Primary-Standby** in the left pane.

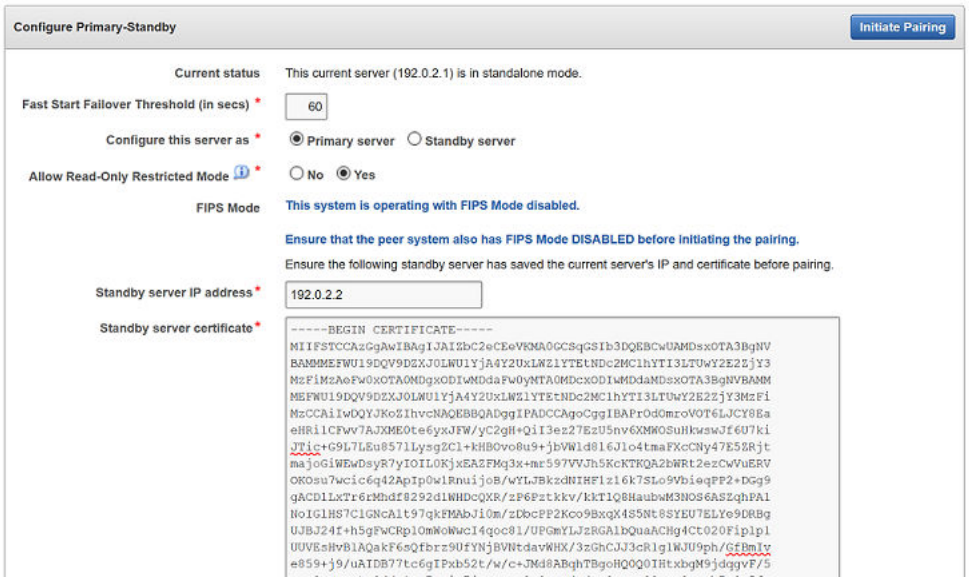
   The Configure Primary-Standby page appears.
5. In the **Configure this server as** field, select **Primary server**.

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.

Do not disable read only restricted mode unless necessary. If the primary-standby configuration is configured with read only restricted mode disabled, then you must enable it by reinstalling and configuring Oracle Key Vault again.

6. Copy the following information from `standby.txt`, and paste it in the Configure Primary-Standby 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.



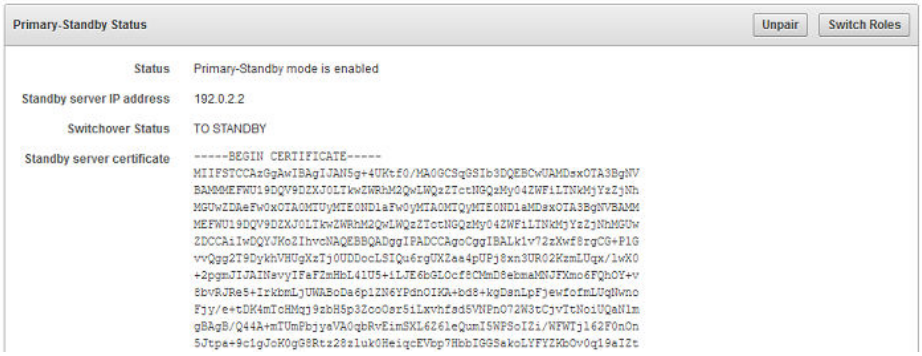
7. Click **Initiate Pairing**.
8. In the confirmation message that is displayed, click **OK**. The **Operation in Progress** page is displayed.

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

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

   If the pairing of primary and standby servers is successful, then the current session is terminated. The **Oracle Key Vault Management Console login screen** of the primary server is displayed. The primary-standby configuration is now complete.
10. Check that the configuration was successful.
    a. Log in as the System Administrator.
    b. Click the **System** tab, then click **Primary-Standby** in the left pane.

       The Primary-Standby Status page appears.



c. Ensure that the **Status** label is set to **Primary-Standby mode is enabled**.

d. Ensure that the **Switchover Status** is correct. In this example, the status is correctly set to **TO STANDBY**.

At this stage, the primary-standby configuration should be ready to use. Note the following:

- When the primary-standby configuration is complete, you cannot log in to the standby server using a web browser because all configuration is propagated from the primary.
- To manage the primary-standby deployment, log in to the primary server using a web browser.

> **Caution:**
>
> Ensure that you leave read-only restricted mode enabled while configuring primary-standby. Enabling it later requires a reinstall of the Oracle Key Vault server software on the standby server.
>
> After configuring the primary-standby environment, 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 primary-standby configuration.

6.3 Switching the Primary and Standby Servers

You can switch the roles of the primary and standby server for situations such as maintenance periods.

During such maintenance periods, you might want to shut down a server 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.

1. Log in to the Oracle Key Vault management console of the primary node as a user with the System Administrator role.
2. Before switching the primary and standby servers, ensure that there are no primary-standby 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 primary-standby related alerts on the **Alerts** page are addressed before switching the primary and standby servers.

3. Click the **System** tab, then **Primary-Standby** from the left side bar.

   The **Primary-Standby Status** page appears.

4. 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.

   Click **OK** in 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.

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

   This logs you out of the current session and then opens a login page to the switched primary server. Otherwise, try accessing the new primary server's URL directly.

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

6. Log in to the primary server to see the IP address of the switched standby node.

7. Click the **System** tab, then **Primary-Standby** from the left side bar.

   The Primary-Standby Status page appears. The **Standby server IP address** field displays the IP address.

6.4 Restoring Primary-Standby After a Failover

A failover takes place if the primary server fails.

If the primary server is not available, then the standby server takes over the primary role. If the standby server does not hear from the primary server for a time exceeding the **Fast Start Failover Threshold** value, then it will assume that the primary is shut down and start the failover process. You can configure the value in the **Fast Start Failover Threshold** field from the Oracle Key Vault management console from the default of 60 seconds. If the failed server (the old primary) becomes available again, then in most cases it will automatically become the new standby server. If the primary server fails permanently, then the standby server will take over as the primary. In this case, you must restore the primary-standby configuration.

1. Reinstall the Oracle Key Vault image on the failed server.

   Ensure that you use the original IP address for the failed server.

2. Log on to the newly installed server and follow the steps to configure the primary-standby environment.

   You can designate the new server as the standby server (because the cluster has a functional primary) and then pair it with the functioning primary.

3. If you want to restore the original configuration and set the new server as the primary, then click the **Switch Roles** option after you successfully pair the two nodes and enable primary-standby.

   The **Switch Roles** button enables 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.

Related Topics

- Failover Situations in Primary-Standby Mode
  Failover situations can occur with or without read-only restricted mode or during a planned shutdown operation for both primary and standby servers.

6.5 Disabling (Unpairing) the Primary-Standby Configuration

You can disable the primary-standby configuration by unpairing the primary and standby servers.

After the two servers are unpaired, the primary and standby servers will operate in standalone mode. To prevent endpoints from connecting to the old standby (now standalone) Oracle Key Vault server, you must take the old standby off the network. See *Oracle Key Vault Release Notes* for guidance about setting the permissions of the `/var/lib/oracle/diag/rdbms/dbfwdb/dbfwdb/metadata_pv` directory beforehand. Check the *Release Notes* for additional issues related to unpair operations.

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 **Primary-Standby** from the left side bar.

   The Primary-Standby 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. After 10 minutes, click the **Refresh** button to be logged out of the current session.
5. Log back in to the management console of the primary server.
6. Select **System**, then **Primary-Standby** from the left side bar.

   The Configure Primary-Standby page appears. The **Current status** field shows the server in standalone mode.

> **Caution:**
>
> If you want to use the old standby (now standalone) Oracle Key Vault server as a standby in a new primary-standby deployment, or as part of a multi-master cluster, then you must re-install the Oracle Key Vault software

Related Topics

- *Oracle Key Vault Release Notes*

6.6 Read-Only Restricted Mode in a Primary-Standby Configuration

The read-only restricted mode is the default mode in a primary-standby configuration.

- About Read-Only Restricted Mode in a Primary-Standby Configuration
  Primary-standby read-only restricted mode ensures endpoint operational continuity.
- Primary-Standby with Read-Only Restricted Mode
  Read-only restricted mode is the default primary-standby mode in Oracle Key Vault.
- Primary-Standby without Read-Only Restricted Mode
  When a primary-standby environment is configured without read-only restricted mode, the impact on endpoint operations differs.
- 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.

- Enabling Read-Only Restricted Mode
  Read-only restricted mode is enabled by default when primary-standby is configured.
- Disabling Read-Only Restricted Mode
  Read-only restricted mode is enabled by default when primary-standby is configured.
- 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.
- Read-Only Restricted Mode Notifications
  When the primary or standby server enters read-only restricted mode, an alert is generated.

6.6.1 About Read-Only Restricted Mode in a Primary-Standby Configuration

Primary-standby read-only restricted mode ensures endpoint operational continuity.

This endpoint operational continuity is essential when the primary or standby Oracle Key Vault servers are affected by server, hardware, or network failures.

When an unplanned shutdown makes the primary or standby server offline, the endpoints can still connect to the surviving peer server to perform critical operations. Primary-standby 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 a primary-standby Oracle Key Vault configuration, the single point of failure is eliminated when you replicate the primary server's data to the standby server. Read-only restricted mode enables the generation of non-critical data such as audit records. 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 primary-standby 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:

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

The following sections describe the behavior of:

6.6.2 Primary-Standby with Read-Only Restricted Mode

Read-only restricted mode is the default primary-standby mode in Oracle Key Vault.

> **Note:**
>
> You can disable read-only restricted mode during the primary-standby configuration. Oracle recommends that you configure primary-standby with read-only restricted mode enabled, which is the default mode. While configuring primary-standby, ensure that **Yes** is selected in the **Allow Read-Only Restricted Mode** field on the Configure Primary-Standby 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 primary-standby failover situations with read-only restricted mode, see Failover Situations with Read-Only Restricted Mode.

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 About the States of Read-Only Restricted Mode.

For more information about read-only restricted mode, see States of Read-Only Restricted Mode.

> **Note:**
>
> Read-only restricted mode has no impact on a standalone server.

6.6.3 Primary-Standby without Read-Only Restricted Mode

When a primary-standby environment is configured without read-only restricted mode, the impact on endpoint operations differs.

This impact depends 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, because 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 primary-standby 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.

6.6.4 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.

- About the States of Read-Only Restricted Mode
  Read-only restricted mode puts the Oracle Key Vault instance into the read-only restricted mode state.
- Read-Only Restricted State Functionality During a Primary Server Failure
  You can set a failover threshold value to determine when a standby server takes over for a failed primary server.
- Read-Only Restricted Mode Functionality During a Standby Server Failure
  If a standby fails, the primary server waits for the duration in the **Fast Start Failover Threshold** field on the **Configure Primary-Standby** page.
- Read-Only Restricted State Functionality During a Network Failure
  When a network failure affects communication between primary and standby servers, communication between certain endpoints and the primary server may also be affected.

6.6.4.1 About the States of Read-Only Restricted Mode

Read-only restricted mode puts the Oracle Key Vault instance into the read-only restricted mode state.

However, read-only restricted mode does not put the embedded Oracle Key Vault database into the read-only restricted mode state. In read-only restricted mode, the following behavior occurs when a primary or a standby server is unavailable:

- When the primary server is down, data cannot be replicated and so the standby server will failover and disable all operations that generate new data. However, the standby can fetch existing data.
- When the standby server is down, data cannot be replicated and so the primary server disables all operations that generate new data. However, the primary can fetch existing data.

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 6-1 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, then 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.

6.6.4.2 Read-Only Restricted State Functionality During a Primary Server Failure

You can set a failover threshold value to determine when a standby server takes over for a failed primary server.

In the event of a primary server failure, the standby server waits for the duration specified in the **Fast Start Failover Threshold (in secs)** field on the Configure Primary-Standby page. If the primary server is not reachable after the specified duration has elapsed, the standby server enters read-only restricted mode. In read-only restricted mode, only operations that fetch data are allowed. Endpoint operations that add new data or modify existing data on the Oracle Key Vault server are blocked.

Related Topics

- Configuring the Primary-Standby Environment
  To configure a primary-standby environment, you must have the System Administrator role and have access to the two servers (one primary and one standby).

6.6.4.3 Read-Only Restricted Mode Functionality During a Standby Server Failure

If a standby fails, the primary server waits for the duration in the **Fast Start Failover Threshold** field on the **Configure Primary-Standby** page.

If the standby server is not reachable after the specified duration has elapsed, the primary server enters read-only restricted mode. In read-only restricted mode, only operations that fetch data are allowed. Endpoint operations that add new data or modify existing data on the Oracle Key Vault server are blocked.

The primary server continues to provide limited service to the endpoints.

Related Topics

- Configuring the Primary-Standby Environment
  To configure a primary-standby environment, you must have the System Administrator role and have access to the two servers (one primary and one standby).

6.6.4.4 Read-Only Restricted State Functionality During a Network Failure

When a network failure affects communication between primary and standby servers, communication between certain endpoints and the primary server may also be affected.

The primary server waits for the duration specified in the **Fast Start Failover Threshold** field on the Configure Primary-Standby page. If the standby server is not reachable after the specified duration has elapsed, the primary server enters read-only restricted mode.

The standby server will also wait for the same duration. If the primary server is not reachable after the specified duration has elapsed, the standby server enters read-only restricted mode. The standby server takes over as the new primary server,

and provides service to endpoints that cannot communicate with the affected primary server.

Related Topics

- Configuring the Primary-Standby Environment
  To configure a primary-standby environment, you must have the System Administrator role and have access to the two servers (one primary and one standby).

6.6.5 Enabling Read-Only Restricted Mode

Read-only restricted mode is enabled by default when primary-standby is configured.

Oracle recommends that you configure the primary-standby servers with read-only restricted mode enabled.

1. Unpair the primary server from the standby server, and then reinstall Oracle Key Vault on the standby server.
2. Perform post-installation tasks on the standby server.
3. Log in to the standby server as the System Administrator.
4. Select the **System** tab.
5. Select **Primary-Standby** and then configure primary-standby on the standby server.

   On the Configure Primary-Standby page, ensure that **Yes** is selected in the **Allow Read-Only Restricted Mode** field.
6. Log in to the primary server as the System Administrator.
7. Select **Primary-Standby** and then on the Configure Primary-Standby page, ensure that **Yes** is selected in the **Allow Read-Only Restricted Mode** field.
8. Click **Initiate Pairing**.

Read-only restricted mode takes effect if connectivity is lost between the primary and standby servers. Read-only restricted mode has no effect on a standalone server.

Related Topics

- Installing the Oracle Key Vault Appliance Software
  The Oracle Key Vault installation process installs all the required software components onto a dedicated server.
- Performing Post-Installation Tasks
  After you install Oracle Key Vault, you must complete a set of post-installation tasks.
- Configuring the Primary-Standby Environment
  To configure a primary-standby environment, you must have the System Administrator role and have access to the two servers (one primary and one standby).

6.6.6 Disabling Read-Only Restricted Mode

Read-only restricted mode is enabled by default when primary-standby is configured.

Oracle recommends that you configure primary-standby with read-only restricted mode enabled. Follow these steps if an existing primary-standby deployment with read-only restricted mode that is enabled must be converted to a deployment that has read-only restricted mode disabled.

1. Unpair the primary server from the standby server, and reinstall Oracle Key Vault on the standby server.
2. Perform post-installation tasks on the standby server.
3. Log in to the standby server as the System Administrator.
4. Select the **System** tab.
5. Select **Primary-Standby** and then configure primary-standby on the standby server.

   On the Configure Primary-Standby page, ensure that **No** is selected in the **Allow Read-Only Restricted Mode** field.
6. Log in to the primary server as the System Administrator.
7. Select **Primary-Standby** and on the Configure Primary-Standby page, ensure that **No** is selected in the **Allow Read-Only Restricted Mode** field.
8. Click **Initiate Pairing**.

After read-only restricted mode is disabled, it does not take effect if connectivity is lost between the primary and standby servers. Read-only restricted mode has no effect on a standalone server.

Related Topics

- Installing the Oracle Key Vault Appliance Software
  The Oracle Key Vault installation process installs all the required software components onto a dedicated server.
- Performing Post-Installation Tasks
  After you install Oracle Key Vault, you must complete a set of post-installation tasks.
- Configuring the Primary-Standby Environment
  To configure a primary-standby environment, you must have the System Administrator role and have access to the two servers (one primary and one standby).

6.6.7 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 Oracle Key Vault database, recovery may not be possible. You would then need to reinstate the Oracle Key Vault instance because of the partial failure. If the failed server is unable to again pair with the peer server within 20 minutes, then you must reinstantiate the server.

Even though the endpoint processes communicating with the Oracle Key Vault servers retain the IP address of the last known reachable server, they must determine the IP address of the new Oracle 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 wait for a response before trying to reach the server configured as the standby server in the configuration scripts. To minimize downtime, Oracle recommends 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. Reinstate the standby server if it does not automatically 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, Oracle recommends that you perform a switchover after recovery. You should also not enroll any new endpoints before the switchover.

Related Topics

- Restoring Primary-Standby After a Failover
  A failover takes place if the primary server fails.

6.6.8 Read-Only Restricted Mode Notifications

When the primary or standby server enters read-only restricted mode, an alert is generated.

You can view these alerts on the **Alerts** page. If email notifications are configured, then an email notification is sent.

Related Topics

- Viewing Open Alerts
  Users can view alerts depending on their privileges.
- Configuring Email Notification
  You can use email notifications to directly notify administrators of Key Vault status changes without logging into the Oracle Key Vault management console.

6.7 Best Practices for Using Oracle Key Vault in a Primary-Standby Configuration

Oracle provides guidelines for ensuring operational continuity and minimal downtime of Oracle Key Vault.

- Configure your Transparent Data Encryption (TDE)-enabled databases to have an auto-login connection into Oracle Key Vault. *Oracle Database Advanced Security Guide* describes how to configure auto-login keystores.
- Apply the database patch for Bug 22734547 to tune the Oracle Key Vault heartbeat.
- Ensure that read-only restricted mode is enabled in primary-standby Oracle Key Vault deployments.
- Set the duration in the **Fast Start Failover Threshold** field on the Configure Primary-Standby 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.
- Before attempting any unpair operations, check *Oracle Key Vault Release Notes* for known issues.

Related Topics

- *Oracle Database Advanced Security Guide*
- *Oracle Key Vault Release Notes*

7

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.

- Managing User Accounts
  You can create Oracle Key Vault user accounts, grant these users Key Vault administrative roles, and add the users to user groups.
- Managing Administrative Roles and Privileges
  Oracle Key Vault has predefined roles that you can grant to (or change) or revoke from users.
- Managing User Passwords
  You or the user can change the user's password. You also can have passwords reset automatically.
- Managing User Email
  Oracle Key Vault users should have their current email on file so that they can receive alerts such as system changes.
- Managing User Groups
  You can organize users who have a common purpose into a named user group.

7.1 Managing User Accounts

You can create Oracle Key Vault user accounts, grant these users Key Vault administrative roles, and add the users to user groups.

- About Oracle Key Vault User Accounts
  Oracle Key Vault users fulfill multiple functions.
- How a Multi-Master Cluster Affects User Accounts
  An Oracle Key Vault multi-master cluster environment affects users in various ways.
- Creating an Oracle Key Vault User Account
  A user with the System Administrator role can create user accounts from the Oracle Key Vault management console.
- Viewing User Account Details
  All administrative users can view the list of Oracle Key Vault user accounts and their details.
- Deleting an Oracle Key Vault User Account
  Deleting an Oracle Key Vault user removes the user from any user groups the user was part of in Oracle Key Vault.

7.1.1 About Oracle Key Vault User Accounts

Oracle Key Vault users fulfill multiple functions.

An important user function is to register and enroll Oracle Key Vault endpoints, enabling the user to manage his or her security objects by using Oracle Key Vault.

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, or Audit Manager
- Ordinary users who have none of the administrative roles, but who have access to security objects

Separation of duties in Oracle Key Vault means that users with an administrative role have access to functions pertaining to their role, but not other roles. For example, only a user with the System Administrator role has access to the full **System** tab, not users with the Key Administrator or Audit Manager roles. Similarly, a 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 that are specific to their function. For example, you can grant a user access to a specific virtual wallet. This user can log into the Oracle Key Vault management console and add, manage, and delete his or her own security objects, but he or she cannot see system menus, details of other users and endpoints, their wallets, or audit reports.

Although the separation of user duties is recommended, you can have a single user perform all the administrative functions by granting that user all the administrative roles.

Oracle Key Vault does not permit the user name to be the same as the name of another user or an endpoint. If you are creating users in a multi-master cluster environment, there is a chance that user with the same name will be created in another node at the same time. In that case, Oracle Key Vault checks for naming conflicts and will automatically rename the user account that was created after the first user account of that name. You must drop the second user and then recreate it with a different name.

7.1.2 How a Multi-Master Cluster Affects User Accounts

An Oracle Key Vault multi-master cluster environment affects users in various ways.

These can include expanding the activities that they can perform and ensuring that their names do not conflict with other objects in the cluster environment.

- Multi-Master Cluster Effect on System Administrator Users
  The user who is granted the System Administrator role is responsible for managing the cluster configuration.
- Multi-Master Cluster Effect on Key Administrator Users
  The user who is granted the Key Adminstrator role manages endpoint groups, user groups, wallets, and objects.
- Multi-Master Cluster Effect on Audit Manager Users
  The user who is granted the Audit Manager role is responsible for configuring audit settings.
- Multi-Master Cluster Effect on Administration Users
  Administrative users can have any combination of the administration roles, including the System Administrator, Key Administrator, and Audit Manager roles.

- Multi-Master Cluster Effect on System Users
  System users are responsible for the operating system of each Oracle Key Vault appliance, server, and node.

7.1.2.1 Multi-Master Cluster Effect on System Administrator Users

The user who is granted the System Administrator role is responsible for managing the cluster configuration.

The System Administrator role in a multi-master cluster includes the following responsibilities:

- All system administrator responsibilities for a single Oracle Key Vault server
- Cluster initialization, converting the first Oracle Key Vault server to the initial node
- Adding and removing nodes from the cluster
- Disabling and enabling nodes in the cluster
- Managing cluster-wide system settings
- Monitoring cluster operations and cluster health indicators
- Enabling and disabling replication between nodes
- Monitoring and resolving data and naming conflicts
- Monitoring and reacting to cluster alerts
- Managing cluster settings

7.1.2.2 Multi-Master Cluster Effect on Key Administrator Users

The user who is granted the Key Adminstrator role manages endpoint groups, user groups, wallets, and objects.

In a multi-master cluster, when these items are uploaded in separate nodes and in separate data centers, name conflicts can occur. The key administrator provides input to the system administrator to resolve these conflicts for wallets, KMIP objects, endpoint groups, and user groups.

Related Topics

- Naming Conflicts and Resolution
  Oracle Key Vault can resolve naming conflicts that can arise as users create objects such as endpoints, endpoint groups, and user groups.

7.1.2.3 Multi-Master Cluster Effect on Audit Manager Users

The user who is granted the Audit Manager role is responsible for configuring audit settings.

In a multi-master cluster environment, this user can configure audit settings for the entire cluster and for individual nodes. The audit manager user can use different setting for different nodes, if necessary. However, this user can also unify audit settings across the entire cluster.

The audit manager can replicate audit trails between nodes, if necessary. However, this can result in significant traffic between nodes, so the audit manager can turn on or off the audit trail replication. By default, the audit trails replication is turned off.

7.1.2.4 Multi-Master Cluster Effect on Administration Users

Administrative users can have any combination of the administration roles, including the System Administrator, Key Administrator, and Audit Manager roles.

Administrative user information created in the Oracle Key Vault server that is used as the initial node seeds the cluster.

New servers added to a cluster will get administrative user information from the cluster. Administrator information that is created on the server for the purpose of inducting the server into the cluster will be removed.

Administrative users that are created in a node after the node joins an Oracle Key Vault cluster will have a cluster-wide presence. New administrative users that are added to the Oracle Key Vault cluster on different Oracle Key Vault nodes may have name conflicts. When the user account is created, Oracle Key Vault automatically resolves the administrative user name conflicts. User and endpoint conflicts will displayed in the Conflicts Resolution page and administrators can choose to rename endpoint conflicts. If there is a user name conflict, then you must either accept the automatically generated user name, or delete and recreate the user. User accounts will not be available for use and will be placed in a `PENDING` state until the name resolution is completed.

Related Topics

- Naming Conflicts and Resolution
  Oracle Key Vault can resolve naming conflicts that can arise as users create objects such as endpoints, endpoint groups, and user groups.

7.1.2.5 Multi-Master Cluster Effect on System Users

System users are responsible for the operating system of each Oracle Key Vault appliance, server, and node.

Oracle Key Vault servers are first installed and later configured to become nodes of an Oracle Key Vault cluster. As part of the server configuration, the operating system users (`support` and `root`) are created. Those users will remain unchanged after the server joins a cluster.

The same `support` and `root` passwords should be used for all the Oracle Key Vault nodes. Unlike Oracle Key Vault administrative accounts that are replicated, the `support` and `root` accounts are operating system users, and their passwords are not automatically synchronized across the cluster. Therefore, each node can potentially have a different `support` or `root` user password, making it difficult to manage multiple nodes of the cluster.

7.1.3 Creating an Oracle Key Vault User Account

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

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.

4. Enter a user name in **User Name**.

   Enter a maximum of 30 characters for the user name. If you are in a multi-master cluster environment, then use a maximum of 24 characters for the user name. Ensure that the user name is not the same as an Oracle Key Vault endpoint name.

5. If you are using a multi-master cluster, then choose whether to select the **Make Unique** checkbox.

   Make Unique helps to control naming conflicts with user names across the multi-master cluster environment. When a server is converted to a cluster node, then the character limit for user names drops from 30 to 24 to allow for automatic renaming in case of a conflict. Users that were created before an Oracle Key Vault conversion to a cluster node are not affected by naming conflicts.

   - If you select **Make Unique**, then the user account will be active immediately and this user can perform operations.
   - If you do not select **Make Unique**, then the user account will be created in the `PENDING` state. Oracle Key Vault will then begin a name resolution operation and may rename the user account to a name that is unique across the cluster. If there is a naming collision, then the collision will be reported on the Conflicts page on any node in the cluster. The user account will then be renamed to a unique name. You will need to go to a read-write node of the cluster and either accept the renamed user account or change the user account name. If you change the user account name, then this will restart the name resolution operation and the user account will return to a `PENDING` state. A user account in the `PENDING` state cannot be used to perform most operations.

6. Optionally, add the user's full name in **Full Name**.

7. 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 will receive 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 or she will be asked to change the password.

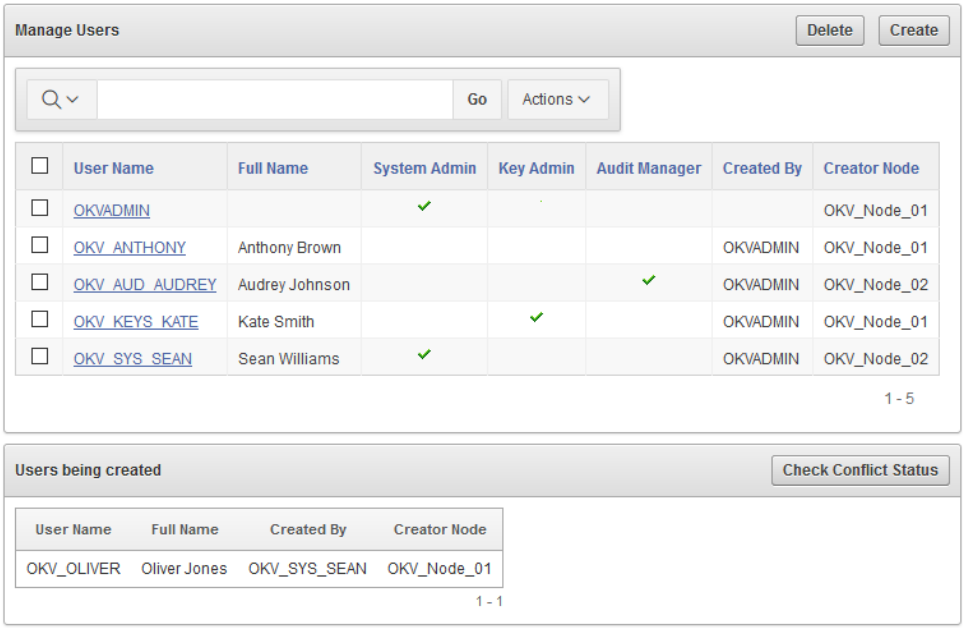
     The SMTP server configuration must be configured to use this option.

   - **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 character. The special

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

8. Click **Save**.

   The **Manage Users** page appears and lists the new user. If the user is in the PENDING state, then it remains in the Users being created section until it transitions to the ACTIVE state, similar to the following example.



Related Topics

- Naming Conflicts and Resolution
  Oracle Key Vault can resolve naming conflicts that can arise as users create objects such as endpoints, endpoint groups, and user groups.
- Managing Administrative Roles and Privileges
  Oracle Key Vault has predefined roles that you can grant to (or change) or revoke from users.
- Configuring Email Notification
  You can use email notifications to directly notify administrators of Key Vault status changes without logging into the Oracle Key Vault management console.

7.1.4 Viewing User Account Details

All administrative users can view the list of Oracle Key Vault user accounts 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 Oracle Key Vault user. This is the page where all user management tasks are performed.

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 the column user name, full name, or roles.

3. Click on a user name to display the **User Details** page.

Related Topics

- Administrative Roles within Oracle Key Vault
  Oracle Key Vault provides separation of duty compliant administrative roles that you can combine in various ways to meet enterprise needs.
- Performing Actions and Searches
  The Oracle Key Vault management console enables you to perform standard actions and search operations, as well as get help information.

7.1.5 Deleting an Oracle Key Vault User Account

Deleting an Oracle Key Vault user removes the user from any user groups the user was part of in Oracle Key Vault.

The operation does not delete any security objects managed by the user. Administrators can only delete users that are not in the `PENDING` state.

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

   The **Manage Users** page appears displaying the list of users.
3. Check the boxes by the users you want to delete.
4. Click **Delete**.
5. In the confirmation dialog box, click **OK**.
6. Click **Save**.

7.2 Managing Administrative Roles and Privileges

Oracle Key Vault has predefined roles that you can grant to (or change) or revoke from users.

- About Managing Administrative Roles
  You can grant or change an administrative role for a user account that you have added.
- Granting or Changing an Administrative Role of a User
  You can use the Manage Users page to grant or change a user administrative role.
- Granting 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.
- Revoking an Administrative Role from a User
  You can use the Manage User page to revoke a role from a user.

7.2.1 About Managing Administrative Roles

You can grant or change an administrative role for a user account that 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. You cannot add, change, or delete these roles.

If you are using a multi-master cluster environment, then you can not grant, change, and revoke administrative roles for users in the `PENDING` state.

7.2.2 Granting or Changing an Administrative Role of a User

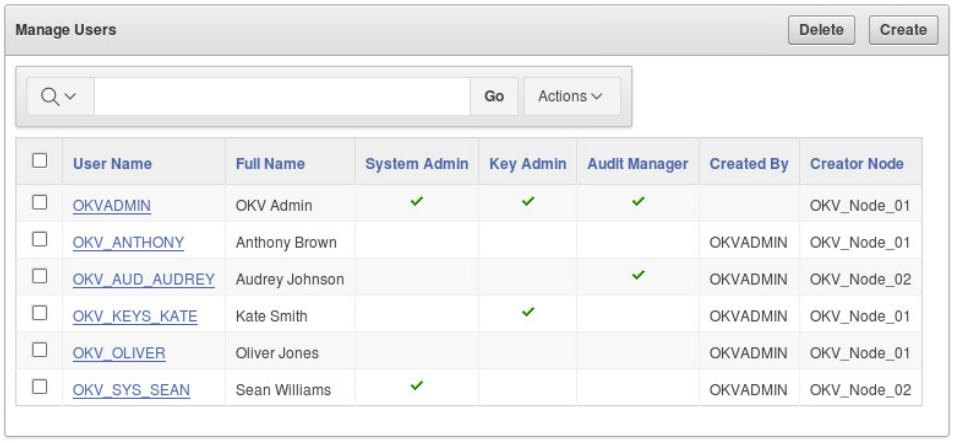
You can use the Manage Users page to grant or change a user administrative role.

1. Log in to the Oracle Key Vault management console as a user who has the same role that is to be granted.

   For example, if the user needs the System Administrator role, then log in as a user who has been granted the System Administrator role.

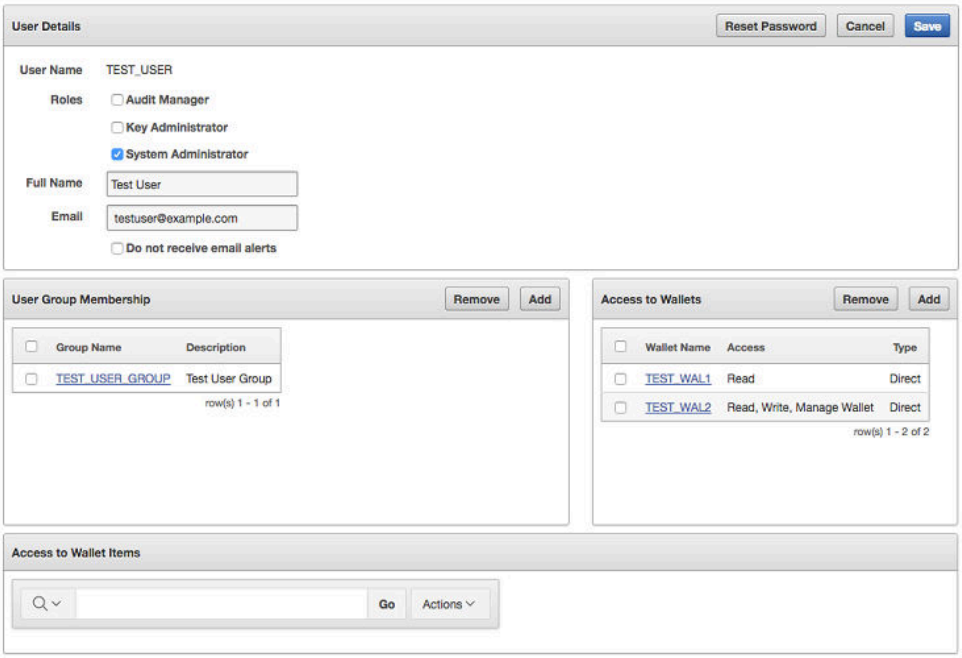
2. Click the **Users** tab.

   The Manage Users page appears displaying the list of users.



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 Oracle Key Vault user. It displays the following user information: user name, email, administrative roles, membership in user groups, and access to security objects.



4. To grant a role, check the **Roles** box for the role you want to grant.

   To change a role, uncheck the box for the previous role and check the box by the new role. If you do not see the role listed that you want to grant, then you are logged in as a user who does not have that role and therefore do not have the privilege to grant it.

5. Click **Save**.

7.2.3 Granting 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 can be granted access to security objects in Oracle Key Vault at a level that is appropriate to their function in the organization.

You cannot grant access to a virtual wallet if the wallet is in the `PENDING` state.

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 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** page appears.

5. Select the wallet under **Select Wallet**.
6. Set the access level to the selected wallet under **Select Access Level**: **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**.

Related Topics

- Access Control Configuration
  Oracle Key Vault enables you to control access to security objects at various access levels and time intervals.
- Managing Access to Virtual Wallets from Keys and Wallets Tab
  You can grant virtual wallet access to and revoke virtual wallet access from endpoint by using the **Keys and Wallets** tab.

7.2.4 Revoking an Administrative Role from a User

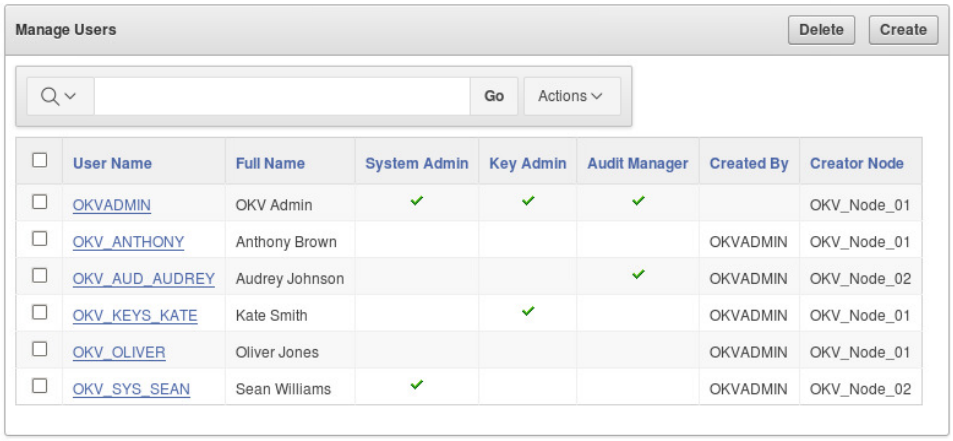
You can use the Manage User page to revoke a role from a user.

1. Log in to the Oracle Key Vault management console as a user who has the same role that is to be revoked.

   You can only grant and revoke roles for which you are an administrator.

2. Click the **Users** tab.

   The Manage Users page appears displaying the list of users.



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**.

7.3 Managing User Passwords

You or the user can change the user's password. You also can have passwords reset automatically.

- About Changing User Passwords
  Any valid Oracle Key Vault user can change his or her own password.

- Changing Your Own Password
  Any user can change his or her own Oracle Key Vault account password.
- Changing Another User's Password
  You can change another user's password if you have the identical administrative role (at minimum) as the user whose password you want to reset.
- Controlling the Use of Password Reset Methods
  You can restrict the ability of users to reset another user's password manually so that only password reset operations through email notifications are allowed.

7.3.1 About Changing User Passwords

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

You can 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 the Audit Manager 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. The following 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 the Audit Manager role.
- OKV_OLIVER: OKV_SYS_KEYS_JOE can change the password for user OKV_OLIVER, who has no roles at all.

Any user can change his or her own password.

Assuming you have privileges to do so, you can change the password of another user by using either of the following methods:

- Specify a new password for the other user and then notify this user of the new password by using any out-of-band method.
- Send the user a randomly generated one-time password to their email account.

Related Topics

- Controlling the Use of Password Reset Methods
  You can restrict the ability of users to reset another user's password manually so that only password reset operations through email notifications are allowed.

7.3.2 Changing Your Own Password

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

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

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

7.3.3 Changing Another User's Password

You can change another user's password if you have the identical administrative role (at minimum) as the user whose password you want to reset.

- Changing a Password Manually
  You can change the password manually for a user and then use any out-of-band method to notify the user of the new password.
- Changing a Password Through Email Notification
  You can change a user's password by sending them a randomly generated one-time password to their email account.
- Changing Operating System User Account Passwords
  Before you perform the post-installation configuration task after the Oracle Key Vault installation, you can change the passwords for the `root` and `support` accounts in the server terminal console.

7.3.3.1 Changing a Password Manually

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

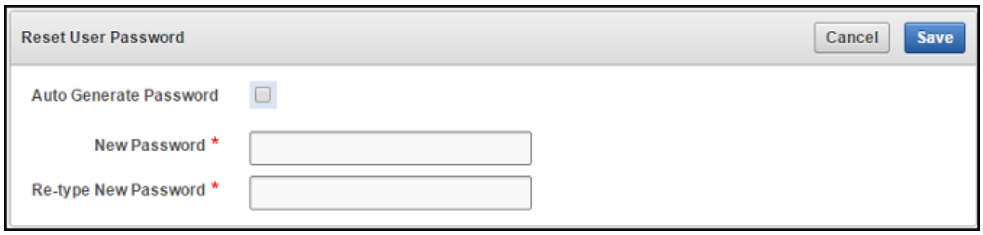
This method of changing password is available only when the `Reset passwords using email only` option in the User Password Recovery tab of the System Recovery page is not checked.

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 displays 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.



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

Related Topics

- About Controlling the Use of Password Reset Methods
  You can configure Oracle Key Vault to only allow users to change another user's password by sending them a randomly generated one-time password through email.

7.3.3.2 Changing a Password Through Email Notification

You can change a user's password by sending them a randomly generated one-time password to their email account.

This one-time password can be sent directly from Oracle Key Vault to the user. You must configure SMTP in email settings in order to use this feature. Oracle recommends that you restrict password recovery functionality to use this method by selecting the `Reset passwords using email only` option in the User Password Recovery tab of the System Recovery page.

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

   The Manage Users page appears displaying the list of users.
3. Click the user name of the user whose password you want to change.

The User Details page appears.

4. Click **Reset Password**.

   The Reset User Password page appears.

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

   An email address field appears.

6. Enter the email address of the user.

7. Click **Send One-Time Password**.

If you check **Auto Generate Password** without configuring SMTP, a link to **Email Settings** appears. Click the link to configure email settings and repeat the steps in this topic.

Related Topics

- Controlling the Use of Password Reset Methods
  You can restrict the ability of users to reset another user's password manually so that only password reset operations through email notifications are allowed.
- Configuring Email Settings
  You can configure the Simple Mail Transfer Protocol (SMTP) server properties to receive email notifications from Oracle Key Vault.

7.3.3.3 Changing Operating System User Account Passwords

Before you perform the post-installation configuration task after the Oracle Key Vault installation, you can change the passwords for the `root` and `support` accounts in the server terminal console.

After that, you can use SSH to change the `root` and `support` passwords. (When you install Oracle Key Vault, you create these accounts as part of the process.) 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.

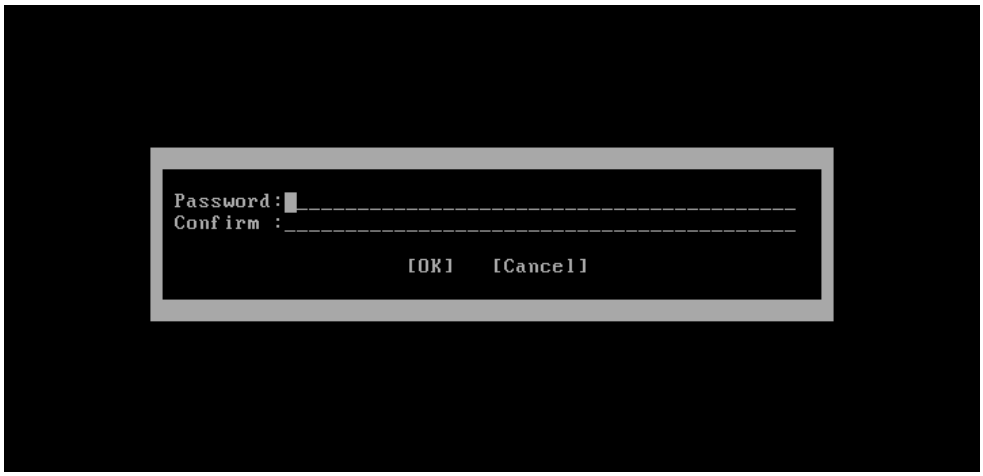
1. Connect to the server console.

2. Select **Set User Passwords** to set the `root` and `support` user passwords. Press **Enter**.

   The Set User Passwords screen appears.

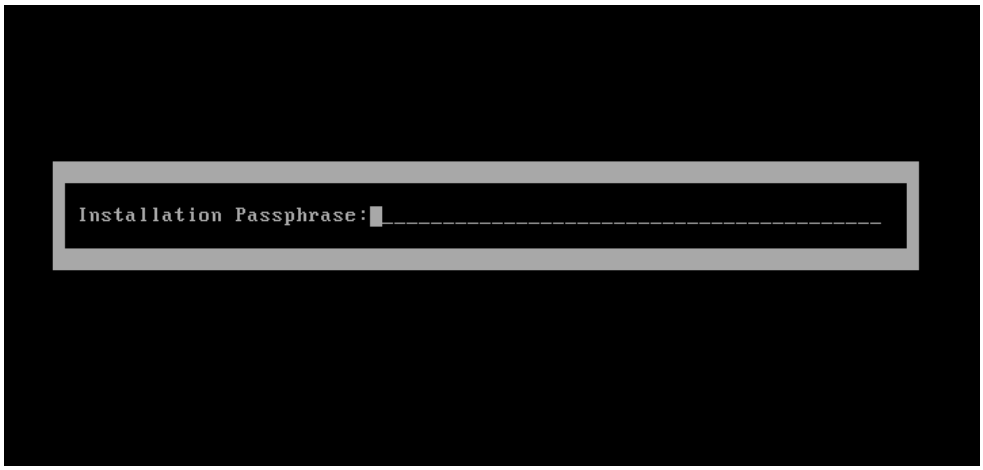
3. Select **Set root password** or **Set support password** and press **Enter**.

   The Set Password screen appears.



4. Type the new password in the **Password** and **Confirm** fields, and then select **OK** and press **Enter**.

   The Installation Passphrase screen appears.



5. Enter the installation passphrase and then press **Enter**.

7.3.4 Controlling the Use of Password Reset Methods

You can restrict the ability of users to reset another user's password manually so that only password reset operations through email notifications are allowed.

- About Controlling the Use of Password Reset Methods
  You can configure Oracle Key Vault to only allow users to change another user's password by sending them a randomly generated one-time password through email.
- Configuring the Use of Password Reset Operations
  A user who has access to the system recovery passphrase can configure the use of password reset operations

7.3.4.1 About Controlling the Use of Password Reset Methods

You can configure Oracle Key Vault to only allow users to change another user's password by sending them a randomly generated one-time password through email.

The user performing a password change for another user must be either an Oracle Key Vault administrator or have the same or higher privileges as the user whose password needs to be reset.

By default, there are two ways to change another user's password:

- Manually, in which you create a new password for the user. In this scenario, both you and the user will know the password (until this user manually changes his or her own password)
- Automatically, in which you trigger an automatically-generated password for the user, who is then emailed the new password on a one-time basis. In this scenario, only the user knows his or her new password.

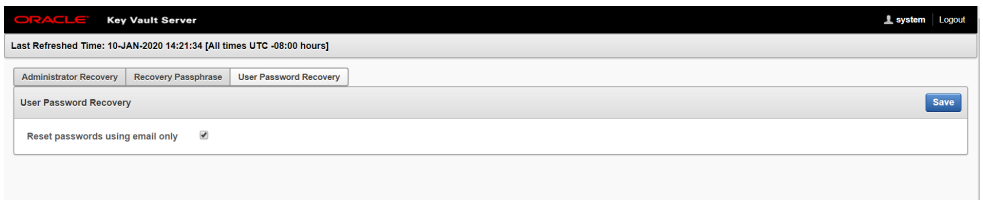
You can enable only automatic password generation through email notification and disable manual password reset operations. The email notification uses the email ID that is associated with the user's account. The benefit of this feature is that the newly generated password is known only to the user whose password needed to be reset, not to the user who initiated the user's password change. Users can still change their own passwords when this feature is enabled.

When this feature is disabled, then both methods of user creation are allowed: manual password reset operations, and automatic password reset operations.

7.3.4.2 Configuring the Use of Password Reset Operations

A user who has access to the system recovery passphrase can configure the use of password reset operations

1. Navigate to the Oracle Key Vault management console, but do not log in.
2. At the bottom of the login screen, click the **System Recovery** button.
3. When prompted, enter the system recovery passphrase.
4. Select the **User Password Recovery** tab.
5. In the User Password Recovery page, select the **Reset Passwords Using Email Only** option to enable or disable this option.



Related Topics

- Changing a Password Through Email Notification
  You can change a user's password by sending them a randomly generated one-time password to their email account.

7.4 Managing User Email

Oracle Key Vault users should have their current email on file so that they can receive alerts such as system changes.

- Changing the User Email Address
  After creating a user account, you can add or change the user's email address.
- Disabling Email Notifications for a User
  You can disable email notifications for a user on the User Details page.

7.4.1 Changing the User Email Address

After creating a user account, you can add or change the user's email address.

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

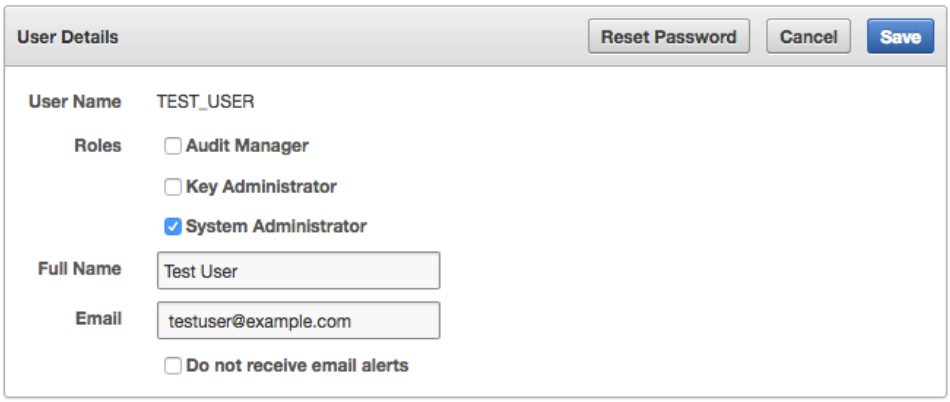
7.4.2 Disabling Email Notifications for a User

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

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
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. Select the **Do not receive email alerts** option.
5. Click **Save**.

7.5 Managing User Groups

You can organize users who have a common purpose into a named user group.

- About Managing User Groups
  Users who have the Key Administrator role can create, modify, and delete user groups.
- How a Multi-Master Cluster Affects User Groups
  User groups are used at the Oracle Key Vault server and cluster level to group user roles and permissions.
- Creating a User Group
  You can create a user group when a set of users must manage a set of common security objects.
- Adding a User to a User Group
  You can add an existing user to a user group if that user must manage the same security objects as the group.
- Granting a User Group Access to a Virtual Wallet
  You can modify the access level to a virtual wallet for a user group as functional needs change.
- Renaming a User Group
  Depending on its status, you can change the name of a user group.
- Changing a User Group Description
  A group description is useful for identifying the purpose of the group.
- Removing a User from a User Group
  Depending on the circumstances, you can remove a user from a user group.
- Deleting a User Group
  You can delete a user group when the users in the group do not need to access the same security objects.

7.5.1 About Managing User Groups

Users who have the Key Administrator role can create, modify, and delete user groups.

This enables them to manage their access to virtual wallets. After a user group is created, you can modify its details.

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, then you can assign these users to a group and grant the group access instead of granting access to each user or based on each security object. When certain users do not need access to the security objects any longer, you can remove them from the group. You can add new users to the group. You can modify the group's access level to security objects at any time.

7.5.2 How a Multi-Master Cluster Affects User Groups

User groups are used at the Oracle Key Vault server and cluster level to group user roles and permissions.

When new servers are inducted into the cluster, Oracle Key Vault replaces any user group information that is in the cluster. You can create new user groups in the cluster from a read-write pair.

User groups created in a node after the node is added to an Oracle Key Vault cluster will have a cluster-wide presence. User groups created on two different nodes could have name conflicts. Oracle Key Vault automatically resolves the user group name conflicts. These conflicts will be displayed in the Conflicts Resolution page and administrators can choose to rename them.

Note the following:

- You cannot change membership by adding or removing users when the user group is in a `PENDING` state. Similarly, users in a pending state cannot be added to, or removed from a user group in the `ACTIVE` state.
- You cannot change access mapping for users and user groups if a wallet is in the `PENDING` state. Similarly, users and user groups in a pending state cannot be added to, or removed from a wallet access mapping even when the wallet is in the `ACTIVE` state.

Related Topics

- Naming Conflicts and Resolution
  Oracle Key Vault can resolve naming conflicts that can arise as users create objects such as endpoints, endpoint groups, and user groups.

7.5.3 Creating a User Group

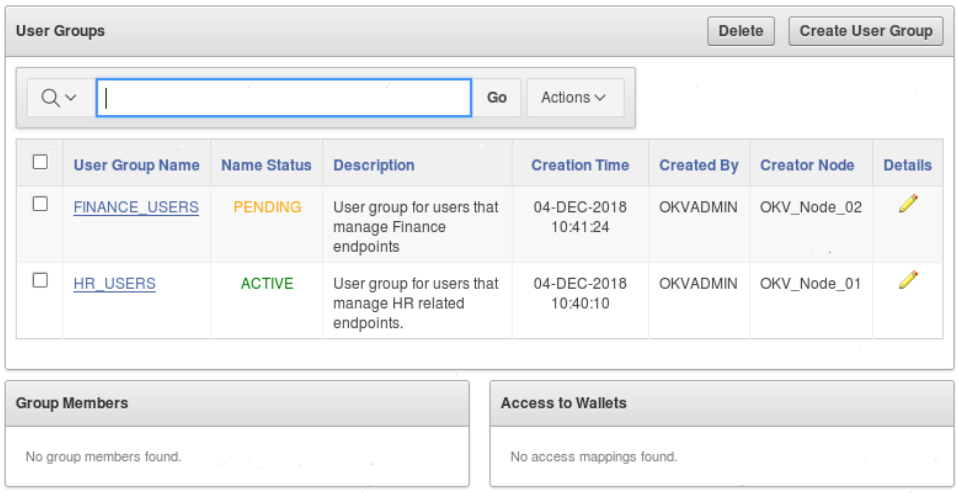
You can create a user group when a set of users must manage a set of common security objects.

You can add users to the group when you create the group or later after creating the group.

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

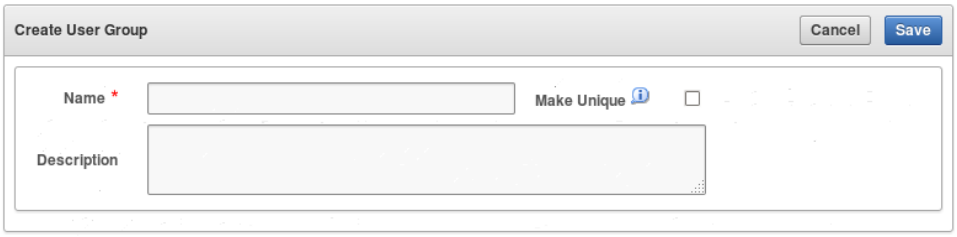
   The **Manage Users** page appears displaying the list of users.
3. Select **Manage Access** from the left sidebar.

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



4. Click **Create User Group**.

   The Create User Group page appears.



5. In the **Name** field, enter the name of the new group and in the **Description** field, a brief description.
6. If you are using a multi-master cluster, then choose whether to select the **Make Unique** checkbox.

   Make Unique helps to control naming conflicts with names across the multi-master cluster environment. User groups that were created before an Oracle Key Vault conversion to a cluster node are not affected by naming conflicts.

   - If you select **Make Unique**, then the group name will be active immediately and this user group can be used in user operations. Clicking **Make Unique** also displays a list of users that you can add to the group.
   - If you do not select **Make Unique**, then the user group will be created in the `PENDING` state. Oracle Key Vault will then begin a name resolution operation and may rename the user group to a name that is unique across the cluster. If there is a naming collision, then the collision will be reported on the Conflicts page on any node in the cluster. The user group will then be renamed to a unique name. You will need to go to a read-write node of the cluster and either accept the renamed user group or change the user group name. If you change the user group name, then this will restart the name resolution operation and the user group will return to a `PENDING` state. A user group in the `PENDING` state cannot be used to perform most operations.
7. In **Description**, optionally, enter a description for the user group.
8. Click **Save**.

Related Topics

- Naming Conflicts and Resolution
  Oracle Key Vault can resolve naming conflicts that can arise as users create objects such as endpoints, endpoint groups, and user groups.

7.5.4 Adding a User to a User Group

You can add an existing user to a user group if that user must manage the same security objects as the group.

If both the user and user group are in the `ACTIVE` state, then you can add users to a group when you create the group or later after creating the groups.

1. Log in to the Oracle Key Vault management console 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 who are not in the user group.
5. Check the boxes for the users you want to add.
6. Click **Save**.

7.5.5 Granting a User Group Access to a Virtual Wallet

You can modify the access level to a virtual wallet for a user group as functional needs change.

However, you can only modify the access level if the user group and wallet are in the `ACTIVE` state.

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**.

   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**.

Related Topics

- Access Control Configuration
  Oracle Key Vault enables you to control access to security objects at various access levels and time intervals.

7.5.6 Renaming a User Group

Depending on its status, you can change the name of a user group.

In a multi-master cluster, if the user group is in the `PENDING` state, then only the creator user can rename the user group.

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**.

   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 name in the **Name** field.

   If this node is part of a multi-master cluster and you do not select **Make Unique**, the user group will enter the `PENDING` state after being renamed.
5. Click **Save**.

7.5.7 Changing a User Group Description

A group description is useful for identifying the purpose of the group.

You can change this description at any time to match the purpose of the group. In a multi-master cluster, if the user group is in the `PENDING` state, then only the creator can modify the user group description.

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**.

   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**.

7.5.8 Removing a User from a User Group

Depending on the circumstances, you can remove a user from a user group.

In a multi-master cluster, if both the user and the user group are in the `ACTIVE` state, then you can remove users from a user group. You may want to remove these users

when their function in the organization changes, and they no longer need to manage the same security objects as the group.

1. Log in to the Oracle Key Vault management console 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 boxes for the users you want to remove.
5. Click **Remove**.
6. Click **OK** to confirm.

7.5.9 Deleting a User Group

You can delete a user group when the users in the group do not need to access the same security objects.

Removing a user group automatically deletes the group's access to wallets and security objects. In a multi-master cluster, if a user group is in the `PENDING` state, then only the creator can delete it.

1. Log in to the Oracle Key Vault management console 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 boxes for the user groups that you want to delete.
4. Click **Delete.**
5. Click **OK** to confirm.

8

Managing Oracle Key Vault Virtual Wallets and Security Objects

You can create a virtual wallet to store security objects, and then share this wallet with trusted peers at different access levels.

- Managing Virtual Wallets
  A virtual wallet is a container for security objects that you can create and then grant access to users.
- Managing Access to Virtual Wallets from Keys and Wallets Tab
  You can grant virtual wallet access to and revoke virtual wallet access from endpoint by using the **Keys and Wallets** tab.
- Managing Access to Virtual Wallets from User's Menu
  To manage access control on virtual wallets for users, endpoints, and their respective groups, you can use the **Users** menu or **Endpoints** menu.
- Managing the State of a Key or a Security Object
  You can set the date to activate or deactivate keys or security objects, and change the state of some virtual wallet security objects.
- Managing Details of Security Objects
  You can manage details about security objects, such as find details about these objects and modifying these details.

8.1 Managing Virtual Wallets

A virtual wallet is a container for security objects that you can create and then grant access to users.

- About Virtual Wallets
  A virtual wallet is a container for security objects.
- Creating a Virtual Wallet
  You can create a virtual wallet and add security objects to it at the same time.
- Adding Security Objects to a Virtual Wallet
  You can add new security objects to a virtual wallet at any time as needed.
- Removing Security Objects from a Virtual Wallet
  You can remove security objects from virtual wallets at any time as needed.
- Deleting 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.

8.1.1 About Virtual Wallets

A virtual wallet is a container for security objects.

These security objects can be public and private encryption keys, including Transparent Data Encryption (TDE) keystores, Oracle wallets, Java keystores, certificates, secret data, and credential files. You can use 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 groups, and endpoint groups access to the virtual wallet at various levels of access. You can modify a virtual wallet and its wallet contents at any time. You can also modify virtual wallet user lists and their respective access level.

Other than the Key Administrator, access to the virtual wallet must be granted explicitly to 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.

8.1.2 Creating a Virtual Wallet

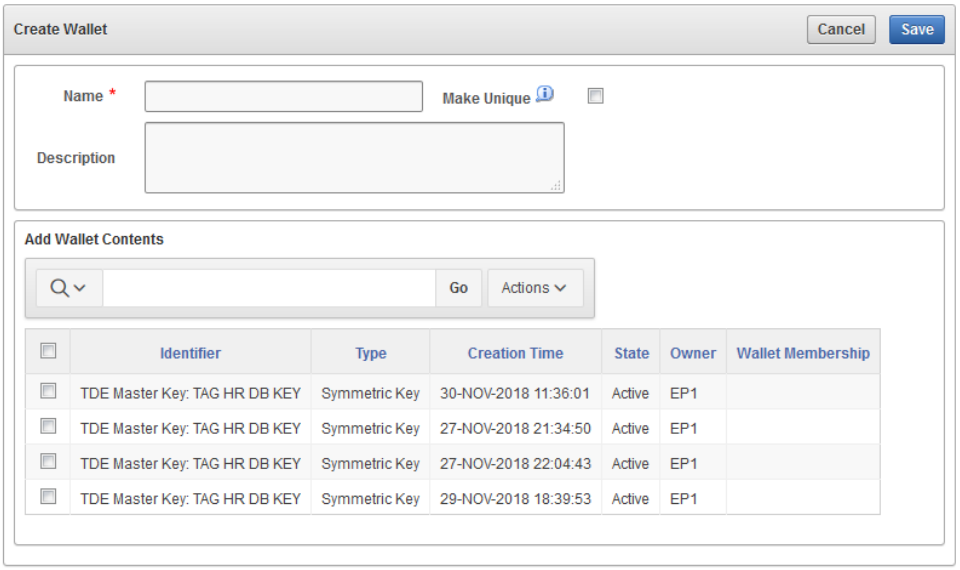
You can create a virtual wallet and add security objects to it at the same time.

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.

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.



4. Enter a name for the wallet in the **Name** field and an identifying description in **Description**.

Virtual wallet names are case-sensitive. For example, `wallet1` and `Wallet1` are two different wallets. Oracle recommends that you add a user-friendly description to the wallet to identify it easily.

5. If you are using a multi-master cluster, then choose whether to select the **Make Unique** checkbox.

   Make Unique helps to control naming conflicts with virtual wallet names across the multi-master cluster environment. Virtual wallets that were created before an Oracle Key Vault conversion to a cluster node are not affected by naming conflicts.

   - If you select **Make Unique**, then the virtual wallet will be active immediately and this wallet can be used in operations.
   - If you do not select **Make Unique**, then the wallet will be created in the `PENDING` state. Oracle Key Vault will then begin a name resolution operation and may rename the wallet to a name that is unique across the cluster. If there is a naming collision, then the collision will be reported on the Conflicts page on any node in the cluster. The wallet will then be renamed to a unique name. You will need to go to a read-write node of the cluster and either accept the renamed wallet name or change the wallet name. If you change the wallet name, then this will restart the name resolution operation and the wallet will return to a `PENDING` state. A wallet in the `PENDING` state cannot be used to perform most operations.

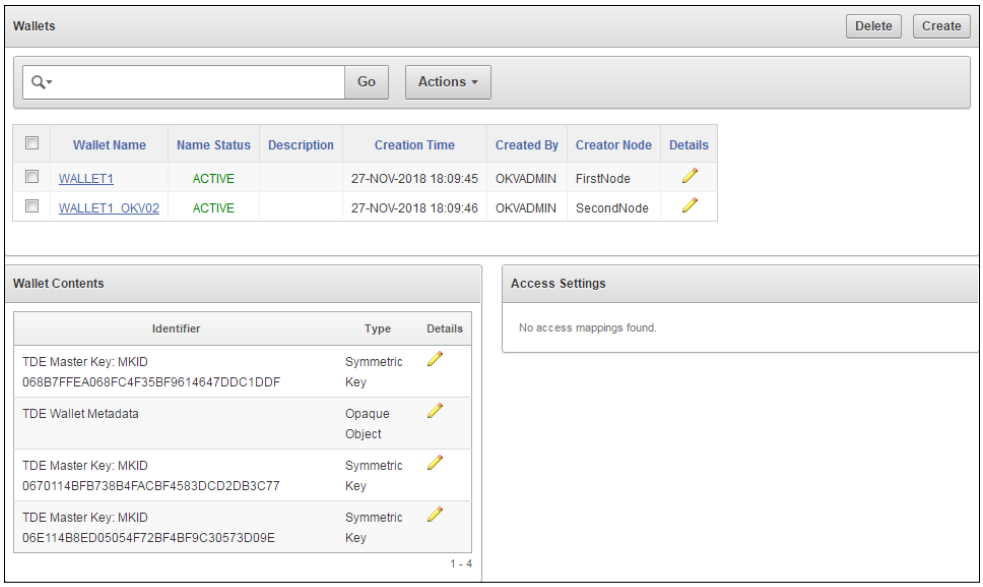
6. In the **Add Wallet Contents** pane, check the boxes by the names of the listed security objects that you want to add to the wallet.

   The **Add Wallet Contents** pane lists the security objects you have **Read and Modify** access to. If the list is empty, then you have no access to the security objects already in Oracle Key Vault. In this case, you would add security objects to the wallet after you upload them to Oracle Key Vault.

7. Click **Save** to create the new wallet with any associated security objects.

   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.



Related Topics

- Name Conflict Resolution in a Multi-Master Cluster
  Naming conflicts can arise when an object has the same name as another object in a different node.

8.1.3 Adding Security Objects to a Virtual Wallet

You can add new security objects to a virtual wallet at any time as needed.

In a multi-master cluster, you cannot add security objects to a virtual wallet when it is in the `PENDING` state.

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 boxes by the security objects that you want to add to the wallet.
6. Click **Save**.

   A confirmation message appears, then the **Wallet Overview** page appears. **Wallet Contents** lists the new security objects added.

8.1.4 Removing Security Objects from a Virtual Wallet

You can remove security objects from virtual wallets at any time as needed.

In a multi-master cluster, you can remove security objects from a virtual wallet when it is in the `PENDING` state.

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 boxes by the security objects you want to remove from the wallet.
5. Click **Remove Items.**

   The **Wallet Contents** pane in the **Wallet Overview** page displays the revised list.

8.1.5 Deleting 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 Oracle Key Vault. Endpoints that have downloaded this virtual wallet will continue to retain their local copy. In a multi-master cluster, you delete a virtual wallet when it is in the `PENDING` state.

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 boxes next to the name of the wallet that you want to delete from the **Wallets** table.

   You can delete more than one virtual wallet at the same time.
4. Click **Delete**.
5. Click **OK** to confirm.
6. Select the **Keys & Wallets** tab to see the updated list of wallets in the **Wallets** page.

8.2 Managing Access to Virtual Wallets from Keys and Wallets Tab

You can grant virtual wallet access to and revoke virtual wallet access from endpoint by using the **Keys and Wallets** tab.

- About Managing Access to Virtual Wallets from Keys and Wallets Tab
  Access control is deciding which users and endpoints share virtual wallets and security objects, and what operations they can perform on those virtual wallets.
- Granting Access to Users, User Groups, Endpoints, and Endpoint Groups
  You can grant the **Read Only**, **Read and Modify**, and **Manage Wallet** access levels to users, user groups, endpoints, and endpoint groups.
- Modifying Access to Users, User Groups, Endpoints, and Endpoint Groups
  You can modify access settings on a virtual wallet for users, user groups, endpoints, and endpoint groups from the **Keys & Wallets** tab.

8.2.1 About Managing Access to Virtual Wallets from Keys and Wallets Tab

Access control is deciding which users and endpoints share virtual wallets and security objects, and what operations they can perform on those virtual wallets.

You must have access to a virtual wallet or be a key administrator to manage access control for users, endpoints, and their respective groups.

To manage access to virtual wallets, you can use the **Keys & Wallets** tab, where you select the wallet, you grant an endpoint, endpoint group, user, or user group access to the wallet.

Related Topics

- Managing Access to Virtual Wallets from User's Menu
  To manage access control on virtual wallets for users, endpoints, and their respective groups, you can use the **Users** menu or **Endpoints** menu.

8.2.2 Granting Access to Users, User Groups, Endpoints, and Endpoint Groups

You can grant the **Read Only**, **Read and Modify**, and **Manage Wallet** access levels to users, user groups, endpoints, and endpoint groups.

After they have access to the wallet, they will have access to all the security objects in the wallet. In a multi-master cluster, you cannot grant access to endpoints, endpoint groups, users, or user groups while the virtual wallet is in the `PENDING` state.

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.

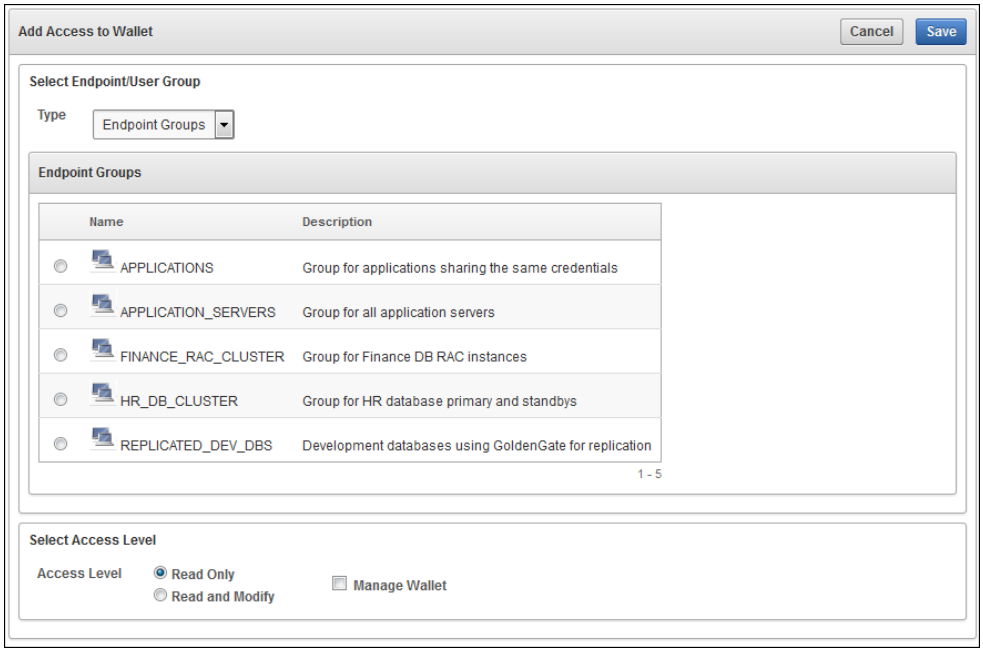
   The **Wallets** page appears.

3. Click the pencil icon in the **Details** column corresponding to the wallet to which you want to grant access.

   The **Wallet Overview** page appears.

4. In the **Wallet Access Settings** pane, click **Add**.

   The Add Access to Wallet page appears.



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 Oracle Key Vault endpoint groups is displayed under the heading **Endpoint Groups**. If you select **Users**, the list of Oracle 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**.

   The **Wallet Access Settings** pane displays the new entity.

8.2.3 Modifying Access to Users, User Groups, Endpoints, and Endpoint Groups

You can modify access settings on a virtual wallet for users, user groups, endpoints, and endpoint groups from the **Keys & Wallets** tab.

In a multi-master cluster, you cannot modify access to endpoints, endpoint groups, users, or user groups while the virtual wallet is in the `PENDING` state.

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. **Wallet Access Settings** lists all the entities that have access to this wallet under **Subject Name**, and can include users, endpoints, user groups, and endpoint groups.
5. Select the access settings that 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.

8.3 Managing Access to Virtual Wallets from User's Menu

To manage access control on virtual wallets for users, endpoints, and their respective groups, you can use the **Users** menu or **Endpoints** menu.

- Granting a User Access to a Virtual Wallet
  You can grant access to a virtual wallet by using the **Users** tab.
- Revoking User Access from a Virtual Wallet
  You can revoke access to a virtual wallet for a user by using the **Users** tab.
- Granting a User Group Access to a Virtual Wallet
  You can grant user group access to a virtual wallet by using the **Users** tab.
- Revoking User Group Access from a Virtual Wallet
  You can remove user group access to a virtual wallet by using the **Users** tab.

Related Topics

- Managing 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.
- Managing Access to Virtual Wallets from Keys and Wallets Tab
  You can grant virtual wallet access to and revoke virtual wallet access from endpoint by using the **Keys and Wallets** tab.

8.3.1 Granting a User Access to a Virtual Wallet

You can grant access to a virtual wallet by using the **Users** tab.

In a multi-master cluster, you cannot grant a user access to a virtual wallet while the virtual wallet is in the `PENDING` state.

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**. You can check **Access to Wallets** in **User Details** for the user to see the wallet added.

Related Topics

- Access Control Options

8.3.2 Revoking User Access from a Virtual Wallet

You can revoke access to a virtual wallet for a user by using the **Users** tab.

In a multi-master cluster, you cannot revoke user access from a virtual wallet while the virtual wallet is in the `PENDING` state.

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.

   The Manage Users page appears.
3. Click the user's name under **User Name**.

   The User Details page appears.
4. In **Access to Wallets**, check the box by the virtual wallet that 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**. You can check **Access to Wallets** in **User Details** for the user to see the wallet deleted.

8.3.3 Granting a User Group Access to a Virtual Wallet

You can grant user group access to a virtual wallet by using the **Users** tab.

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. In a multi-master cluster, you

cannot grant a user group access to a virtual wallet while the virtual wallet is in the `PENDING` state.

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**. You can check **Access to Wallets** in **User Groups** for the user to see the wallet added.

8.3.4 Revoking User Group Access from a Virtual Wallet

You can remove user group access to a virtual wallet by using the **Users** tab.

In a multi-master cluster environment, you cannot revoke user group access from a virtual wallet while the virtual wallet is in the `PENDING` state.

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.** You can check **Access to Wallets** in **User Groups** to see the wallet removed from the list.

8.4 Managing the State of a Key or a Security Object

You can set the date to activate or deactivate keys or security objects, and change the state of some virtual wallet security objects.

- About Managing the State of a Key or a Security Object
  You can control the dates when a key or a security object is active, that is, when it can be used.

- How a Multi-Master Cluster Affects Keys and Security Ojects
  Keys that you create on one node of a multi-master cluster will take some time to appear on other nodes in the cluster.
- Activating a Key or Security Object
  Keys can be in the **Active** or **Pre-Active** state.
- Deactivating a Key or Security Object
  A key deactivates or expires when it passes the date that has been set for deactivation.
- Revoking a Key or Security Object
  When you revoke a key, you can set its state to **Deactivated** or **Compromised**.
- Destroying a Key or Security Object
  When a key is no longer used or compromised in some way, then you can destroy it.

8.4.1 About Managing the State of a Key or a Security Object

You can control the dates when a key or a security object is active, that is, when it can be used.

You also revoke and destroy keys and security objects. Be aware that a multi-master cluster affects the activation or de-activation times of keys and security objects on different nodes, and that naming conflicts can arise.

Related Topics

- How a Multi-Master Cluster Affects Keys and Security Ojects
  Keys that you create on one node of a multi-master cluster will take some time to appear on other nodes in the cluster.

8.4.2 How a Multi-Master Cluster Affects Keys and Security Ojects

Keys that you create on one node of a multi-master cluster will take some time to appear on other nodes in the cluster.

The time is defined by the replication lag between nodes. The replication lag value is displayed on the Cluster Link State pane of the Monitoring page, which can be accessed by choosing the **Cluster** tab.

If you add a Transparent Data Encryption (TDE) master encryption key to two different keystores on two different nodes, then it will be shown in both keystores.

Adjusting the activation date, deactivation date, process start date, and protection stop date has restrictions. For these dates, if changes are made to the security object very close to the current time, then state changes can happen because of replication lag.

As with the creation of any object in a multi-master cluster, a security object can have a name conflict with an object created on a different node. If there is a conflict, then Oracle Key Vault will suggest a unique name or allow you to rename it.

Related Topics

- Naming Conflicts and Resolution
  Oracle Key Vault can resolve naming conflicts that can arise as users create objects such as endpoints, endpoint groups, and user groups.

8.4.3 Activating a Key or Security Object

Keys can be in the **Active** or **Pre-Active** state.

Most keys are in the **Active** state when they are created. However, for a key that will be used for securing data later than its creation date, you can set the **Process Start Date**. Currently, only keys uploaded with 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.

1. Log in to the Oracle Key Vault management console as a user who has 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 for which 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**.

8.4.4 Deactivating 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 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**.

8.4.5 Revoking a Key or Security Object

When you revoke a key, you can set its state to **Deactivated** or **Compromised**.

At this point, the key should no longer be used to encrypt new data. However, you can download and use the deactivated keys to decrypt old data.

1. Log in to the Oracle Key Vault management console as a user who has read and modify access on this key.
2. Select the **Keys & Wallets** tab.
3. Select **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.
5. In the Item Details page, click **Revoke**.

6. In the Revoke Item page, from the **Revocation Reason** drop-down list, select a reason for the revocation.
7. Optionally, add more details in **Revocation Message**
8. Click **Save**.

8.4.6 Destroying a Key or Security Object

When a key is no longer used or compromised in some way, then you can destroy it.

Metadata for destroyed keys and security objects are kept in Oracle Key Vault even after they have been destroyed.

1. Log in to the Oracle Key Vault management console as a user who has 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 for which you want to set.
4. On the Item Details page for the item, click **Destroy**.
5. Click **Save**.

8.5 Managing Details of Security Objects

You can manage details about security objects, such as find details about these objects and modifying these details.

- About Managing the Details of Security Objects
  You can search for security objects within a virtual wallet, and add, modify, or remove these security objects.
- Searching for Security Object Items
  You can search for individual security objects if you have privileges to view these objects.
- Viewing the Details of a Security Object
  An administrative user with the Key Administrator role can view, add, and modify the details of a security object.
- Adding or Modifying Details of a Security Object
  Only users who have the appropriate privileges can add or modify the details of a security object.

8.5.1 About Managing the Details of Security Objects

You can search for security objects within a virtual wallet, and add, modify, or remove these security objects.

Security objects are managed by Oracle 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 individual security objects are not even viewable to a user with the System Administrator role.

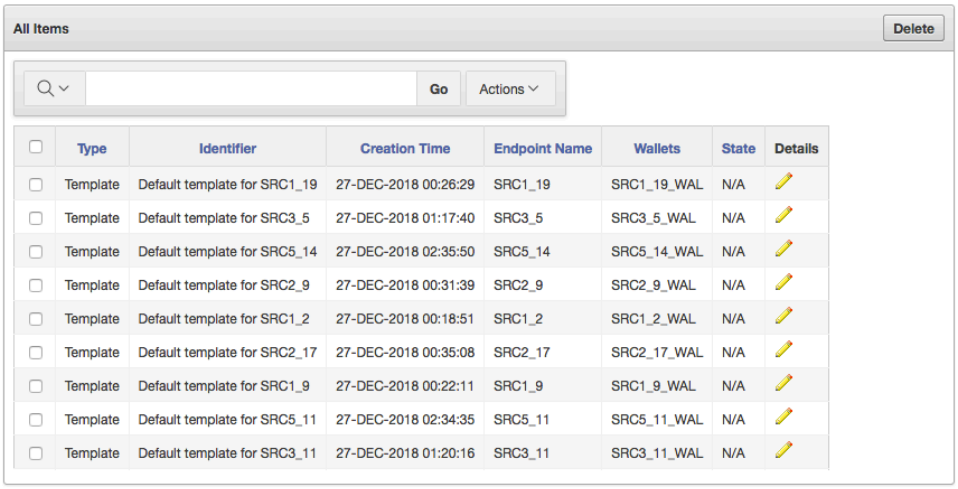
8.5.2 Searching for Security Object Items

You can search for individual security objects if you have privileges to view these objects.

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 page appears displaying all the security objects in a table.**All Items**



   The table has the following columns for each security object:

   - **Type:** Indicates the object type of security object. Valid values are **Symmetric Key**, **Private Key**, **Template**, **Opaque Object**, **Certificate**, and **Secret Data**.
   - **Identifier:** Lists the identifier for the security object and includes a prefix that helps identify a subtype for the item.
   - **Creation Time:** Date and time that the security object was added to Oracle Key Vault.
   - **Endpoint Name:** The endpoint that owns the security object.
   - **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.

Related Topics

- Performing Actions and Searches
  The Oracle Key Vault management console enables you to perform standard actions and search operations, as well as get help information.

8.5.3 Viewing the Details of a Security Object

An administrative user with the Key Administrator role can view, add, and modify the details of a security object.

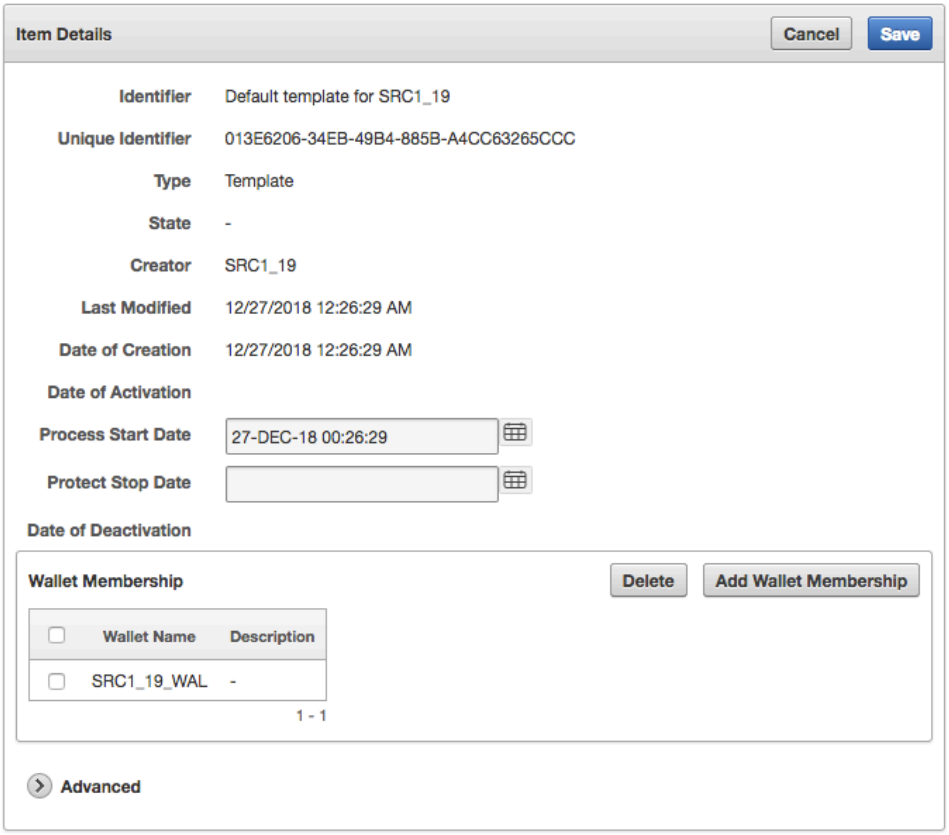
The administrative user can perform these actions on the security object from its corresponding Item Details page. Item details are attributes of a specific security object and depend on the type of security object.

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.



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**.

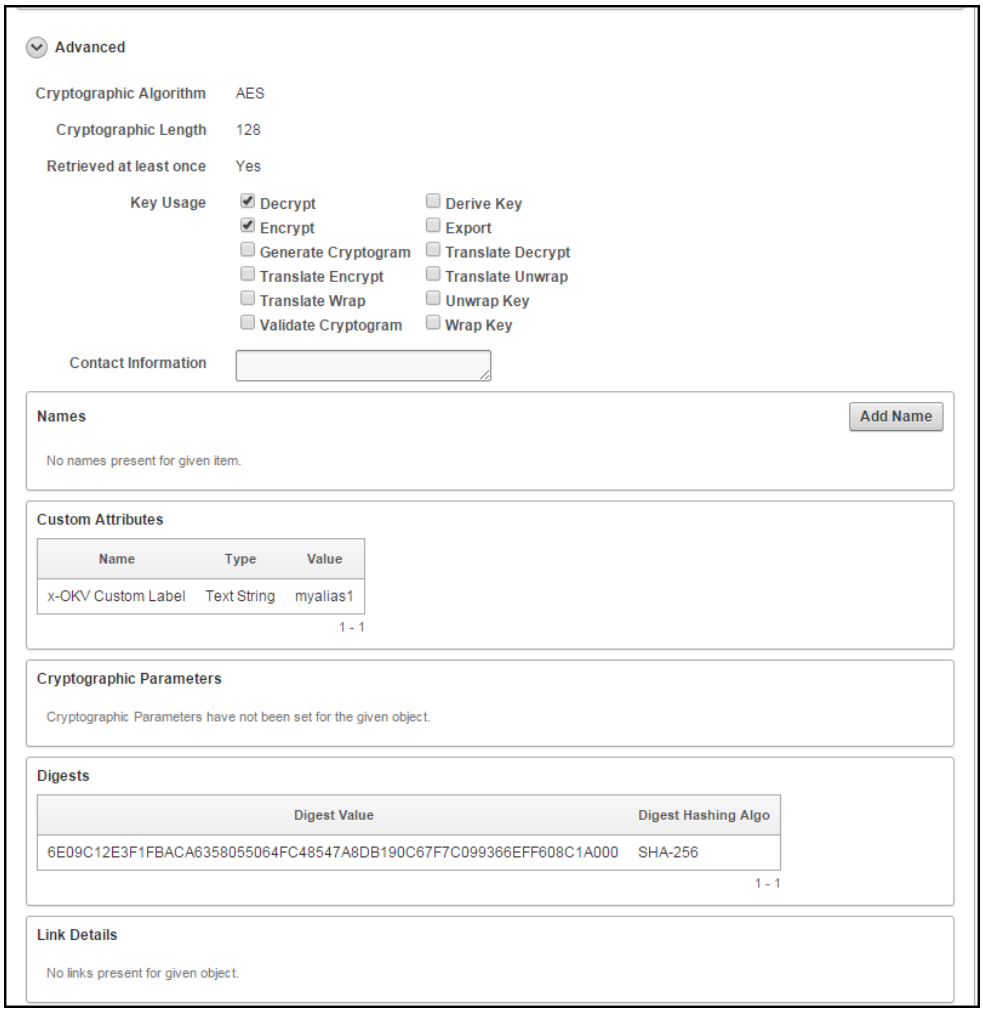
You can revoke or destroy a security object, and add or remove it to and from a wallet from the Item Details page.

The **Wallet Membership** pane in the Item Details page enables you to add the security object to a wallet or delete the security object 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 encryption key, then the **Identifier** shows the prefix TDE master encryption 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 the security objects. 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 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 **Date of Activation** setting 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 **Date of Deactivation** setting.
- **Date of Deactivation:** The date of deactivation.

5. Click **Advanced** to view the cryptographic attributes of the security object.



Attribute information and queries may vary depending on the item type. Examples of attributes are as follows:

- **Cryptographic Algorithms:** The encryption algorithm used by the item
- **Key Usage:** Operations that the key can be used for. Clients may or may not use these attributes. For example, Transparent Data Encryption does not consult the key usage attributes.
- **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
- **Cryptographic Length:** The length in bits of the key
- **Retrieved at Least Once:** Indicates if the object has been served to the client
- **Contact Information:** Used for contact purposes only
- **Digests:** Digest values of the security object
- **Link Details:** Links to related objects

Related Topics

- *Key Management Interoperability Protocol Specification Version 1.1*

8.5.4 Adding or Modifying Details of a Security Object

Only users who have the appropriate privileges can add or modify the 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 if you own the security object or if you have access to a wallet that contains the 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 necessary changes.
7. Click **Save** in the top right corner of the pane.

9

Managing Oracle Key Vault Endpoints

Oracle Key Vault endpoints are computer systems like database or application servers, where keys and credentials are used to access data.

- Overview of Managing Endpoints
  You can manage endpoints in both standalone environments and multi-master clusters in mostly the same way, except that multi-master clusters have more restrictions.
- Managing Endpoints
  You can enroll, reenroll, suspend, and delete endpoints.
- Default Wallets and Endpoints
  You can use a default wallet, which is a type of virtual wallet, with an endpoint.
- Managing 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.
- Managing Endpoint Groups
  An endpoint group is a named group of endpoints that share a common set of wallets.
- Managing Endpoint Details
  Endpoint details refers to endpoint name, type, description, platform, and email, and adding the endpoint to a group, or upgrading the endpoint software.
- Upgrading Endpoints
  You can perform endpoint upgrades from either the Oracle Key Vault management console login page or from the endpoint.

9.1 Overview of Managing Endpoints

You can manage endpoints in both standalone environments and multi-master clusters in mostly the same way, except that multi-master clusters have more restrictions.

- About Managing Endpoints
  You must register and enroll an endpoint to communicate with Oracle Key Vault.
- How a Multi-Master Cluster Affects Endpoints
  You should be aware of how a multi-master cluster affects endpoints, both in the way an endpoint connects to it and with restrictions.

9.1.1 About Managing Endpoints

You must register and enroll an endpoint to communicate with Oracle Key Vault.

Afterward, keys in the endpoint can be uploaded to Oracle Key Vault and be shared with other endpoints and then downloaded from these endpoints so that users can access their data. Only a user with the System Administrator role can add an endpoint

to Oracle Key Vault. After 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 Oracle Key Vault.

All users can create virtual wallets but only a user with Key Administrator role 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 Real Application Clusters (Oracle RAC) database 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.

If you have a large deployment, then install at least four Oracle Key Vault servers, and when you enroll the endpoints, balance them across these four servers to ensure high availability. For example, if a data center has 1000 database endpoints to register, and you have Oracle Key Vault four servers to accommodate them, then enroll 250 endpoints with each of the four servers.

When you name an endpoint, remember that an Oracle Key Vault user name cannot be the same as an Oracle Key Vault endpoint name.

The two administrative roles as they pertain to endpoints are as follows:

- A user with the System Administrator role:
  - Manages the endpoint metadata such as the name, type, platform, description, and email notifications
  - Manages the endpoint lifecycle, which consists of enrolling, suspending, reenrolling, and deleting 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

9.1.2 How a Multi-Master Cluster Affects Endpoints

You should be aware of how a multi-master cluster affects endpoints, both in the way an endpoint connects to it and with restrictions.

In a multi-master configuration, when an endpoint attempts to make a connection to Oracle Key Vault, it performs the following actions:

- First, it obtains the list of server IPs from its configuration file (`okvclient.ora`).
- Next, it picks one at random from those in the cluster subgroup to which the endpoint's creator node belongs.

Be aware of the following restrictions with how endpoints work in multi-master clusters:

- An endpoint can only be enrolled from the same node where it was most recently created or re-enrolled.
- An endpoint gets its initial and subsequent endpoint node scan list update based on the cluster subgroup to which the creator node belongs. Oracle Key Vault

creates the endpoint node scan list when you first add nodes from the same cluster subgroup as the creator node. Oracle Key Vault adds the other nodes later.

- You cannot assign a default wallet to an endpoint if one or both of them (wallet and endpoint) is in the `PENDING` state and if the assignment is attempted from a non-creator node. After both the endpoint and wallet are in the `ACTIVE` state, this restriction ends.

9.2 Managing Endpoints

You can enroll, reenroll, suspend, and delete endpoints.

- Types of Endpoint Enrollment
  The first step in enrolling an endpoint is to add the endpoint to Oracle Key Vault.
- Endpoint Enrollment in a Multi-Master Cluster
  Endpoints of a cluster are the client systems of the multi-master cluster.
- Adding an Endpoint as an Oracle Key Vault System Administrator
  A user who has been granted the System Administrator role can add an endpoint by using the **Endpoints** tab.
- Adding Endpoints Using Self-Enrollment
  The self-enrollment process immediately sends the endpoint to the **Enrolled** status without the intermediate **Registered** status.
- Deleting, Suspending, or Reenrolling Endpoints
  When endpoints no longer use Oracle Key Vault to store security objects, you can delete them, and then re-enroll when they are needed.

9.2.1 Types of Endpoint Enrollment

The first step in enrolling an endpoint is to add the endpoint to Oracle Key Vault.

There are two methods for adding, also known as registering, an endpoint:

- Initiated by an administrator
  An Oracle Key Vault user who has the System Administrator role initiates the enrollment from the Oracle Key Vault side by adding the endpoint to Oracle Key Vault. When the endpoint is added, a one-time enrollment token is generated. This token can be communicated to the endpoint administrator in two ways:
  - Directly from Oracle Key Vault by email. To use email notification you must configure SMTP in email settings.
  - 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. In a multi-master cluster, the same node that is used to add the endpoint must be used to enroll the endpoint.

  After the enrollment token is used to enroll an endpoint, it cannot be used again for another enrollment. If you must reenroll an endpoint, then 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`. In a cluster, a self-enrolled endpoint is created with a generic endpoint name in this format: `ENDPT_xx_001`, where *xx* is a 2-digit node identifier or node number. 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 then disable it when the self-enrollment period ends.

Related Topics

- Configuring Email Notification
  You can use email notifications to directly notify administrators of Key Vault status changes without logging into the Oracle Key Vault management console.

9.2.2 Endpoint Enrollment in a Multi-Master Cluster

Endpoints of a cluster are the client systems of the multi-master cluster.

Endpoint enrollment is divided into two steps. First you add the endpoint and then you enroll it.

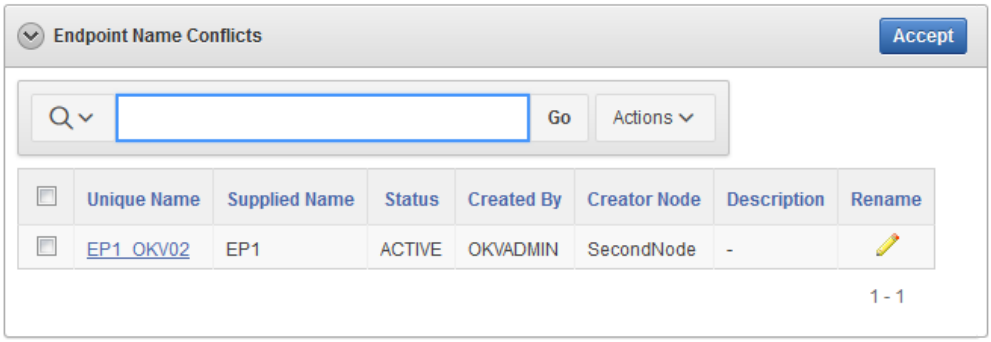
The Oracle Key Vault server that becomes the initial node can have endpoints already enrolled, especially if it was upgraded from a previous release. These existing endpoints initialize, or seed, the cluster. During induction, the endpoints enrolled in the cluster are replicated to a newly added node. During induction, Oracle Key Vault removes endpoints that were previously enrolled in all candidate nodes added to the cluster.

Endpoints can only be enrolled on a read-write node.

After you enroll the endpoint, the new endpoint will have a cluster-wide presence. You can add endpoints of the Oracle Key Vault multi-master cluster to any read-write node.

> **Note:**
>
> An endpoint must be enrolled on the same node where it was most recently added or re-enrolled.

New endpoints added concurrently to the multi-master cluster on different nodes could have name conflicts. Oracle Key Vault automatically resolves the endpoint name conflicts, and then displays the conflicts in a Conflicts Resolution page, similar to the following figure. From here, a system administrator can choose to rename them.



Related Topics

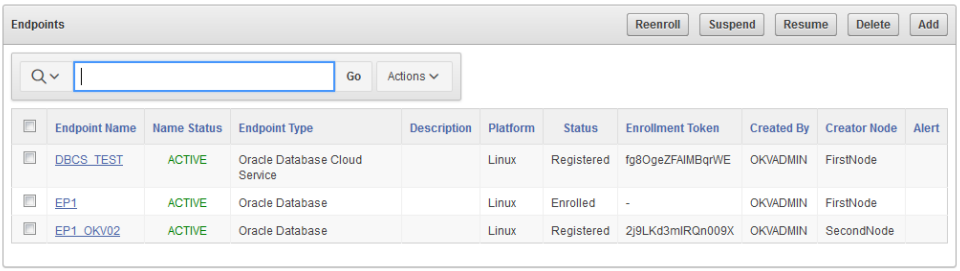
- Naming Conflicts and Resolution
  Oracle Key Vault can resolve naming conflicts that can arise as users create objects such as endpoints, endpoint groups, and user groups.

9.2.3 Adding an Endpoint as an Oracle Key Vault System Administrator

A user who has been granted the System Administrator role can add an endpoint by using the **Endpoints** tab.

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 Oracle Key Vault endpoints.

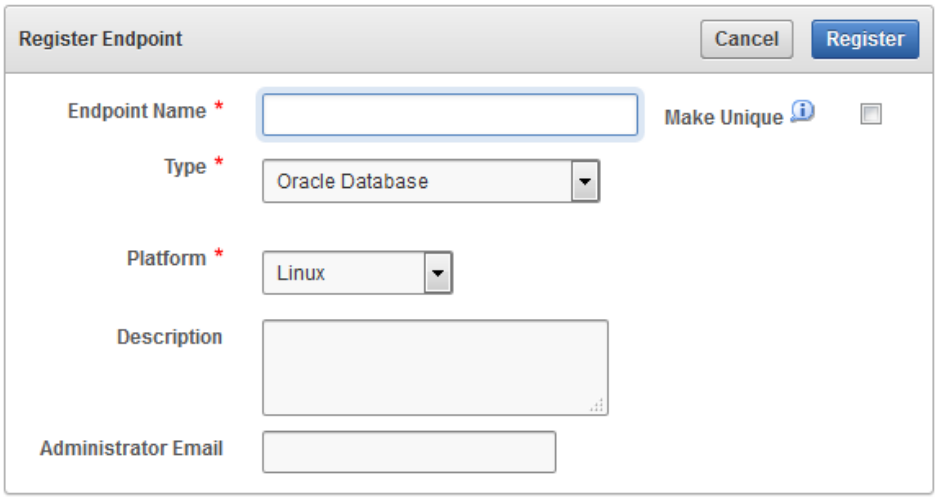


   The Endpoints page displays the list of registered and enrolled endpoints with the following endpoint details: name, type, description, platform, status, enrollment token, and alert. The endpoint status can be either **Registered** or **Enrolled**:

   - **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.
   - **Created By**: The user who created the endpoint.
   - **Creator Node**: The node on which the endpoint was created.
   - **Name Status**: The state of the endpoint. The state will be either `ACTIVE` or `PENDING`.

3. Click **Add** on the **Endpoints** page.

   The Register Endpoint page appears. The **Make Unique** checkbox only appears in multi-master clusters mode.



4. In the **Endpoint Name** field, enter a name for the endpoint.

   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 identified by this name throughout. The maximum length is 30 characters.

5. If you are using a multi-master cluster, then choose whether to select the **Make Unique** checkbox.

   Make Unique helps to control naming conflicts with names across the multi-master cluster environment. Endpoints that were created before an Oracle Key Vault conversion to a cluster node are not affected by naming conflicts.

   - If you select **Make Unique**, then the endpoint will be active immediately and users can use this endpoint.
   - If you do not select **Make Unique**, then the endpoint will be created in the `PENDING` state. Oracle Key Vault will then begin a name resolution operation and may rename the endpoint to a name that is unique across the cluster. If there is a naming collision, then the collision will be reported on the Conflicts page on any node in the cluster. The endpoint will then be renamed to a unique name. You will need to go to a read-write node of the cluster and either accept the renamed endpoint or change the endpoint name. If you change the endpoint name, then this will restart the name resolution operation and the endpoint will return to a `PENDING` state. An endpoint in the `PENDING` state cannot be used to perform most operations.

6. From the **Type** drop-down list, select the type of endpoint.

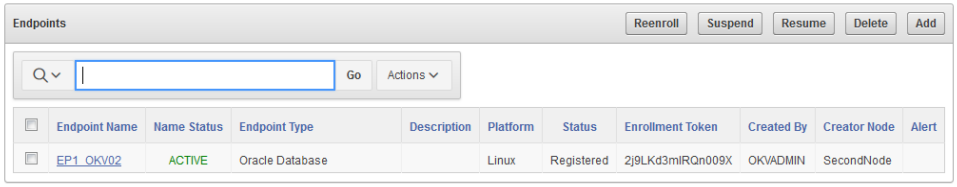
   Supported types are **Oracle Database**, **Oracle Database Cloud Service**, **Oracle (non-database)**, **Oracle ACFS**, **MySQL Database**, and **Other**. An example of **Other** is a third-party KMIP endpoint. If you are using Oracle Advanced Security Transparent Data Encryption (TDE) and want to use Oracle Key Vault to manage a TDE master encryption key or wallet, then you must set **Type** to **Oracle Database**.

7. Complete the following endpoint information:

- **Platform**: Supported platform choices are **Linux**, **Solaris SPARC**, **Solaris x64**, **AIX**, **AIX 5.3**, **HPUX**, and **Windows**.
- **Description**: Optionally, enter a useful identifying description such as the host name, IP address, function, or location of the endpoint.
- **Administrator Email**: Optionally, enter the email address of the endpoint administrator to have the enrollment token and other endpoint-related alerts sent directly from Oracle Key Vault. You must have SMPT configured to use the email notificaiton feature.

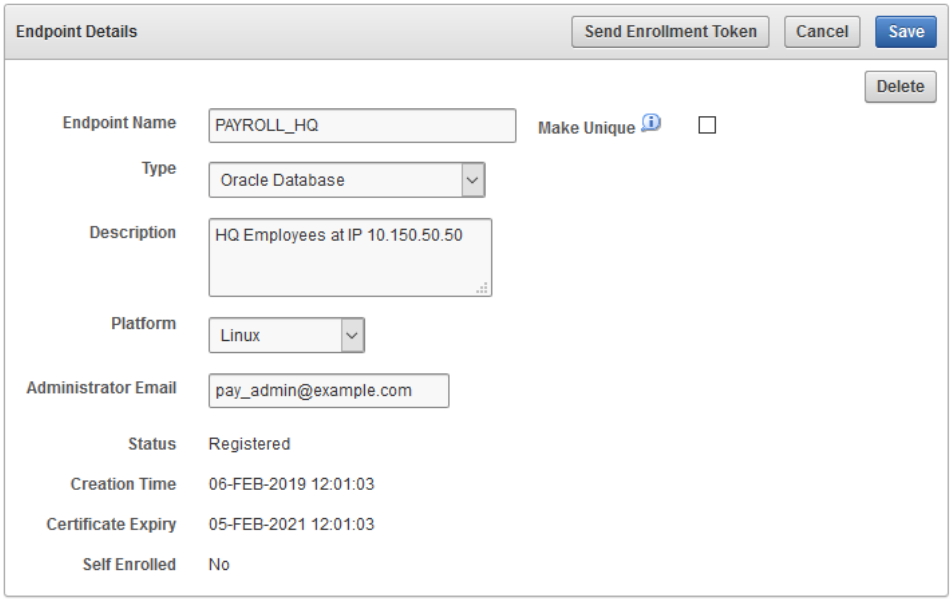
8. Click **Register**.

   The Endpoints page appears listing the new endpoint with a status of **Registered**. The Enrollment Token column displays the one-time enrollment token.



9. Click the **Endpoint Name** to see details for the endpoint.

   The Endpoint Details page appears.



   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:

   - If you did configure SMTP and entered the email address, you can have Oracle Key Vault send the enrollment token directly to the endpoint administrator, shown in the next step, where you click the **Send Enrollment Token** button.

- If you did not configure SMTP or enter the email address, then you must use an out-of-band method to send the enrollment token to the endpoint administrator.

The endpoint must be enrolled and the endpoint jar file must be downloaded from the node on which the endpoint was most recently created or reenrolled.

10. Click **Send Enrollment Token**.

    At this stage, the endpoint's administrator can 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**.

Related Topics

- Configuring Email Notification
  You can use email notifications to directly notify administrators of Key Vault status changes without logging into the Oracle Key Vault management console.
- Step 1: Enroll the Endpoint and Download the Software
  You must have the endpoint's enrollment token before you can download the endpoint software `okvclient.jar`.

9.2.4 Adding Endpoints Using Self-Enrollment

The self-enrollment process immediately sends the endpoint to the **Enrolled** status without the intermediate **Registered** status.

- About Adding Endpoints Using Self-Enrollment
  Oracle Key Vault associates a self-enrolled attribute with all endpoints that are enrolled through endpoint self-enrollment.
- Adding an Endpoint Using Self-Enrollment
  You can configure the self-enrollment process for endpoints from the Oracle Key Vault management console.

9.2.4.1 About Adding Endpoints Using 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 format `ENDPT_001`. In a multi-master cluster, system generated endpoint names are in the format `ENDPT_`*`node_id_sequential_number`*, where *`node_id`* is a value such as `01` or `02`. For example, `ENDPT_01_001` can be the generated name of an endpoint.

Endpoint self-enrollment is disabled by default and must be enabled by a user who has the System Administrator role.

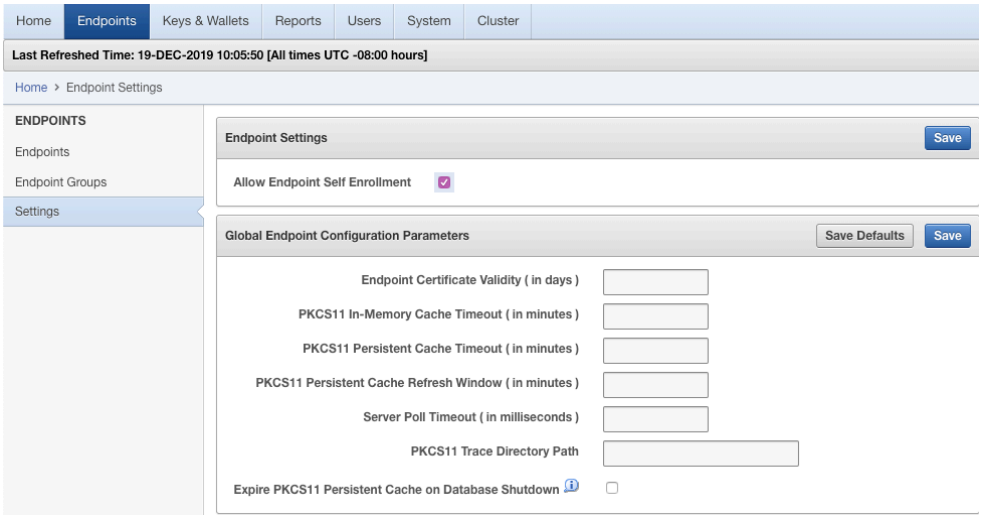
A best practice is to enable endpoint self-enrollment for limited periods when you expect endpoints to enroll. After the expected endpoints have been enrolled, you should disable endpoint self-enrollment.

9.2.4.2 Adding an Endpoint Using Self-Enrollment

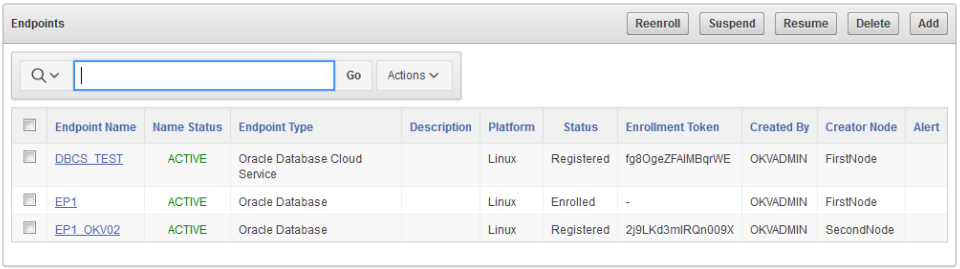
You can configure the self-enrollment process for endpoints from the Oracle Key Vault management console.

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.



3. Check the box to the right of **Allow Endpoint Self-Enrollment**.
4. Click **Save**.



Related Topics

- Step 1: Enroll the Endpoint and Download the Software
  You must have the endpoint's enrollment token before you can download the endpoint software `okvclient.jar`.

9.2.5 Deleting, Suspending, or Reenrolling Endpoints

When endpoints no longer use Oracle Key Vault to store security objects, you can delete them, and then re-enroll when they are needed.

- About Deleting Endpoints
  Deleting an endpoint removes it permanently from Oracle Key Vault.
- Deleting One or More Endpoints
  The Endpoints page enables you to delete a group of endpoints from Oracle Key Vault at one time.
- Deleting One Endpoint (Alternative Method)
  The **Endpoint Details** page provides a consolidated view for the selected endpoint including a mechanism to delete the endpoint from Oracle Key Vault.
- Suspending an Endpoint
  You can suspend an endpoint temporarily for security reasons, and then reinstate the endpoint once the threat has passed.
- Reenrolling an Endpoint
  When you reenroll an endpoint, the enrollment process automatically upgrades the endpoint software.

9.2.5.1 About Deleting Endpoints

Deleting an endpoint removes it permanently from Oracle Key Vault.

However, security objects that were previously created or uploaded by that endpoint will remain in Oracle Key Vault. Similarly, security objects that are associated with that endpoint also remain. To permanently delete or reassign these security objects, you must be a user with the Key Administrator role or authorized 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.

You cannot delete an endpoint that is in the `PENDING` state unless you are the user who created it. You must delete it on the node on which it was created.

9.2.5.2 Deleting One or More Endpoints

The Endpoints page enables you to delete a group of endpoints from Oracle Key Vault at one time.

You can also delete a single endpoint from this page.

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 boxes to the left of the endpoints you want to delete.
4. Click **Delete**.
5. Click **OK** in the confirmation dialog box that appears.

Related Topics

- Performing Actions and Searches
  The Oracle Key Vault management console enables you to perform standard actions and search operations, as well as get help information.

9.2.5.3 Deleting One Endpoint (Alternative Method)

The **Endpoint Details** page provides a consolidated view for the selected endpoint including a mechanism to delete the endpoint from Oracle Key Vault.

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. Click the endpoint name you want to delete.

   The Endpoint Details page appears.
4. Click **Delete**.
5. Click **OK** to confirm.

Related Topics

- Performing Actions and Searches
  The Oracle Key Vault management console enables you to perform standard actions and search operations, as well as get help information.

9.2.5.4 Suspending an Endpoint

You can suspend an endpoint temporarily for security reasons, and then reinstate the endpoint once the threat has passed.

When you suspend an endpoint, its status will change from **Enrolled** to **Suspended**. You cannot suspend an endpoint that is in the `PENDING` state unless you are the user who created it.

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. Click on the endpoint name you want to suspend. The **Endpoint Details** page appears.
4. Click **Suspend**.
5. In the confirmation window, click **OK.**

   When you suspend an endpoint, its **Status** on the **Endpoints** page will be **Suspended**.
6. 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**.

The following rules apply to suspending an endpoint in a multi-master cluster:

- For regular endpoints, the endpoint will continue to operate until all suspend operation requests have reached all nodes in the cluster.
- You can suspend the endpoint on any node.

- For cloud-based endpoints, the endpoint will continue to operate until the suspend operation has reached all nodes from where the reverse SSH tunnel is established.
- You can potentially suspend the endpoint on any node from the cloud-based endpoint from where the reverse SSH tunnel is established.

Related Topics

- Performing Actions and Searches
  The Oracle Key Vault management console enables you to perform standard actions and search operations, as well as get help information.

9.2.5.5 Reenrolling an Endpoint

When you reenroll an endpoint, the enrollment process automatically upgrades the endpoint software.

You must also reenroll an endpoint to accommodate changes such as pairing a primary Oracle Key Vault server with a new secondary server in a primary-standby configuration. The action of reenrolling an endpoint will immediately disallow any connections from the endpoint's old deployment. If you are reenrolling an endpoint, Oracle recommends that you immediately download `okvclient.jar` and deploy it in a directory that is separate from the existing deployment. When you deploy the software, use the `-o` option to overwrite the symbolic link pointing to the old `okvclient.ora`. You cannot reenroll an endpoint that is in the `PENDING` state unless you are the user who created the 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 access the Endpoints page.

   The Endpoints page lists all of the endpoints in Key Vault.
3. Check the boxes to the left of the endpoints that you want to reenroll.
4. Click **Reenroll**.

   After you deploy the `okvclient.jar` file, the `The endpoint software for Oracle Key Vault installed successfully` message should appear. If instead the `The endpoint software for Oracle Key Vault upgraded successfully` message appears, then the reenrollment was performed in the old deployment directory, and as a result, the endpoint software was upgraded but not successfully reenrolled.

   You can overwrite the symbolic link reference that points to `okvclient.ora` in the new directory by using the `okvclient.jar` option `-o`.

   A new enrollment token will be generated for each reenrolled endpoint and appear in the corresponding Enrollment Token column. You can use this one-time token to reenroll the endpoint. You must download the endpoint jar file from the same node on which the endpoint was reenrolled.

Related Topics

- Step 1: Enroll the Endpoint and Download the Software
  You must have the endpoint's enrollment token before you can download the endpoint software `okvclient.jar`.

9.3 Default Wallets and Endpoints

You can use a default wallet, which is a type of virtual wallet, with an endpoint.

- Associating a Default Wallet with an Endpoint
  A default wallet is a type of virtual wallet to which security objects are uploaded when a wallet is not explicitly specified.
- Setting the Default Wallet for an Endpoint
  Setting a default wallet for an endpoint automatically uploads the endpoint's security objects to the wallet if another wallet is not explicitly specified.

9.3.1 Associating a Default Wallet with an Endpoint

A default wallet is a type of virtual wallet to which security objects are uploaded when a wallet is not explicitly specified.

Default wallets are useful for sharing with other endpoints such as nodes in an Oracle Real Application Clusters (Oracle RAC), or primary and standby nodes in Oracle Data Guard by having all endpoints use the same default wallet.

If you want to use the default wallet, then you must set after you register the endpoint before you enroll it. If you decide to use a default wallet after enrollment, then you must remove the default wallet and subsequently reenroll the endpoint.

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. When the status is **registered**, then 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.

In a multi-master cluster, you can only assign the default wallet on the same node where the endpoint and wallet were created when either are still in the `PENDING` state. After both are in the `ACTIVE` state, then there are no restrictions. After the default wallet is assigned and the endpoint is enrolled, the default wallet can be accessed from any node, as long as both are in the `ACTIVE` state and the information has been replicated to that node.

9.3.2 Setting the Default Wallet for an Endpoint

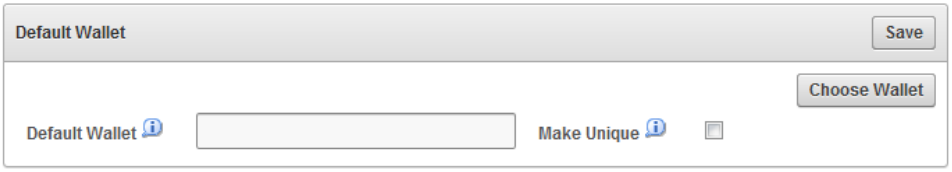
Setting a default wallet for an endpoint automatically uploads the endpoint's security objects to the wallet if another wallet is not explicitly specified.

Oracle requires that you set the default wallet right after registering the endpoint, and before downloading the endpoint software.

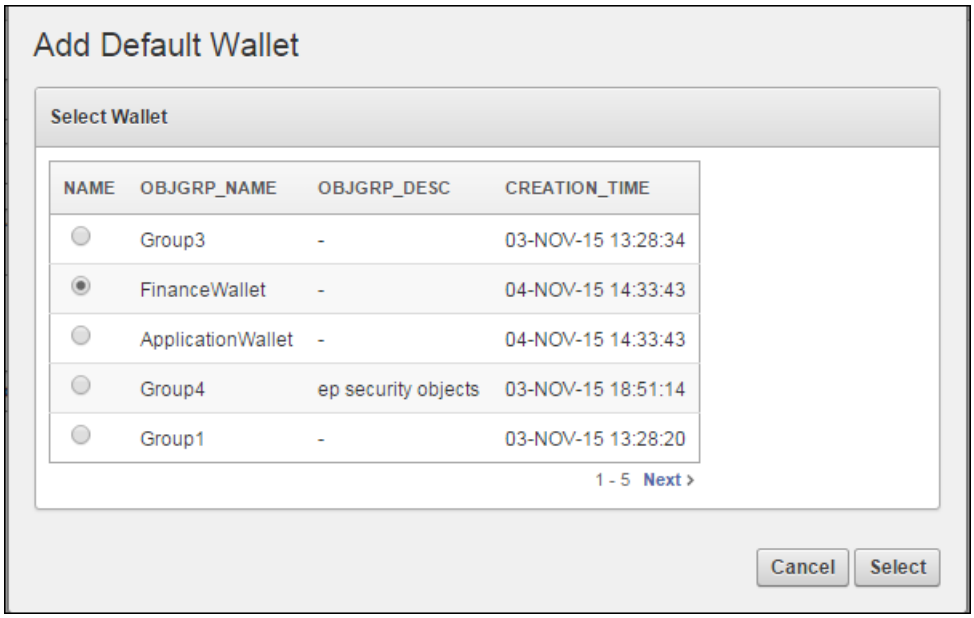
1. Log in to the Oracle Key Vault management console as an administrator who has the Key Administrator role.
2. Select the **Endpoints** tab, and then click on the endpoint name.

   The Endpoint Details page appears.

3. In the Default Wallet pane, select **Choose Wallet**.

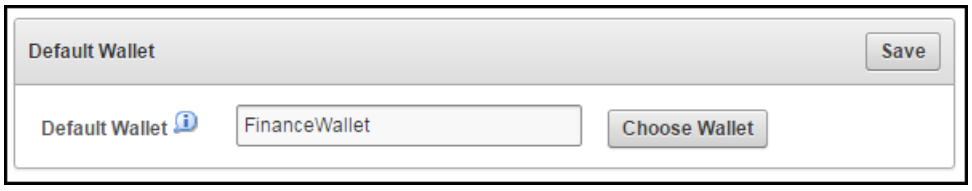


The Add Default Wallet page appears displaying a list of available wallets.



4. Select a wallet from the list to be the default wallet by clicking the option to the left of the wallet, and then click **Select**.

The selected wallet name appears in the **Default Wallet** pane.



5. Click **Save**.

9.4 Managing 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.

- Granting an Endpoint Access to a Virtual Wallet
  An endpoint must have **Read and Modify** and **Manage Wallet** privileges on the wallet before security objects can be uploaded or downloaded.
- Revoking Endpoint Access to a Virtual Wallet
  You can revoke access to a virtual wallet for an endpoint by using the **Endpoints** tab.

- Viewing Wallet Items Accessed by Endpoints
  The term wallet items refers to the security objects to which the endpoint has access.

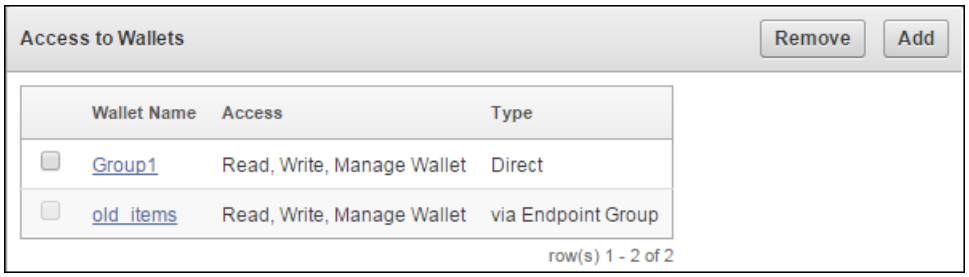
9.4.1 Granting an Endpoint Access to a Virtual Wallet

An endpoint must have **Read and Modify** and **Manage Wallet** privileges on the wallet before security objects can be uploaded or downloaded.

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.

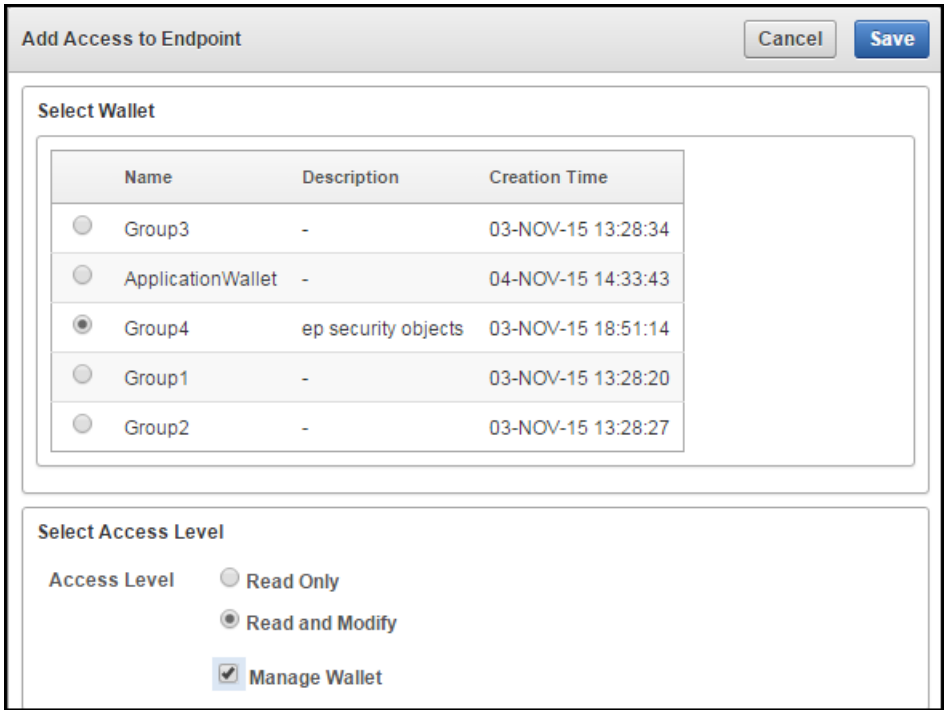
1. Log in to the Oracle Key Vault management console as an administrator who has the Key Administrator role.
2. Select the **Endpoints** tab to get to the **Endpoints** page.
3. On the **Endpoints** page, select the endpoint that must have access to the virtual wallet.

   The Endpoint Details page appears with the **Access to Wallets** pane.



4. In the **Access to Wallets** pane, which 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.



5. Select a wallet from the available list of wallets shown on the **Add Access to Endpoint** page.
6. Select the **Access Level** in the **Select Access Level** pane.
7. Click **Save**.

Related Topics

- Access Control Options
  Access control options enable you to set the type of privileges that users have to read, write, and delete security objects.

9.4.2 Revoking Endpoint Access to a Virtual Wallet

You can revoke access to a virtual wallet for an endpoint by using the **Endpoints** tab.

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 display the **Endpoints** page.
3. On the **Endpoints** page, select the endpoint name, which will display the **Endpoint Details** page.

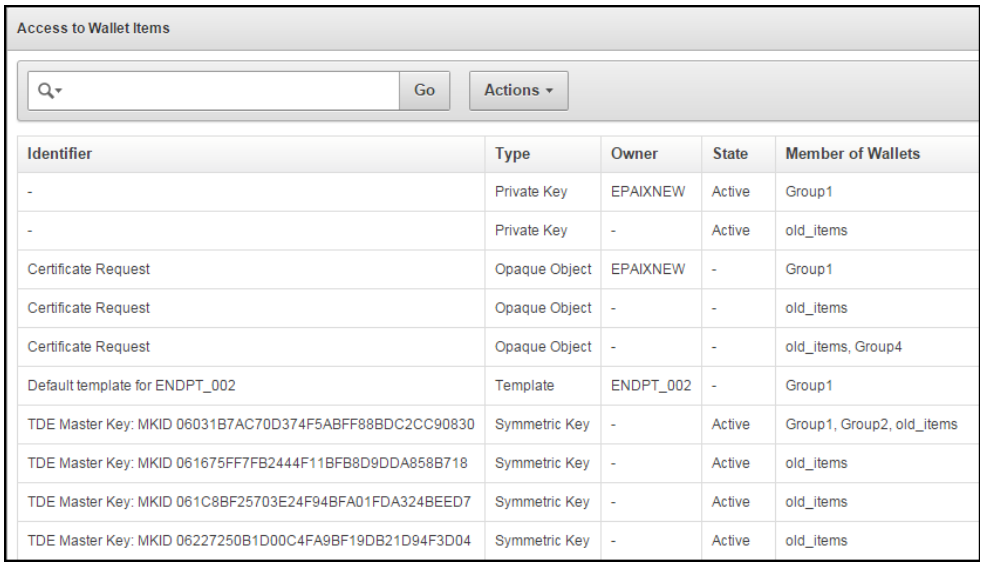
   Locate the **Access to Wallets** pane on this page. The Access to Wallets pane shows a list of wallets that the endpoint has access to.
4. Select the wallet that you want to revoke access to.
5. Click **Remove**.
6. In the confirmation dialog box, click **OK**.

9.4.3 Viewing Wallet Items Accessed by Endpoints

The term wallet items refers to the security objects to which the endpoint has access.

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 access **Endpoint Details**.

   The **Access to Wallet Items** pane in **Endpoint Details** lists the wallet items that the endpoint has access to.



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

9.5 Managing Endpoint Groups

An endpoint group is a named group of endpoints that share a common set of wallets.

- How a Multi-Master Cluster Affects Endpoint Groups
  You can create endpoint groups on any node and they will have a cluster-wide presence.
- Creating an Endpoint Group
  Endpoints that must share a common set of security objects stored in wallets can be grouped into an endpoint group.
- Modifying Endpoint Group Details
  You can add endpoints and access mappings to an endpoint group after creating the endpoint group.
- Granting an Endpoint Group Access to a Virtual Wallet
  You can grant an endpoint group access to a virtual wallet.
- Adding an Endpoint to an Endpoint Group
  You can add an endpoint to a named endpoint group.
- Removing an Endpoint from an Endpoint Group
  When you remove an endpoint from an endpoint group, this removes access to wallets that are associated with that endpoint group.
- Deleting Endpoint Groups
  You can delete endpoint groups if their member endpoints no longer require access to the same virtual wallets.

9.5.1 How a Multi-Master Cluster Affects Endpoint Groups

You can create endpoint groups on any node and they will have a cluster-wide presence.

You can add, update, or delete endpoint groups in any node, but in read-write mode only.

The Oracle Key Vault server that becomes the initial node can have endpoints groups already created. These endpoint groups are used to initialize, or seed, the cluster. During induction, the endpoint groups in the cluster are replicated to a newly added node. Endpoint groups previously created in all other nodes added to the cluster will be removed during induction.

New endpoint groups added concurrently to the multi-master cluster on different nodes may have name conflicts. Oracle Key Vault automatically resolves any endpoint group name conflicts. These conflicts are displayed in a Conflicts Resolution page and key administrators can choose to rename them.

Related Topics

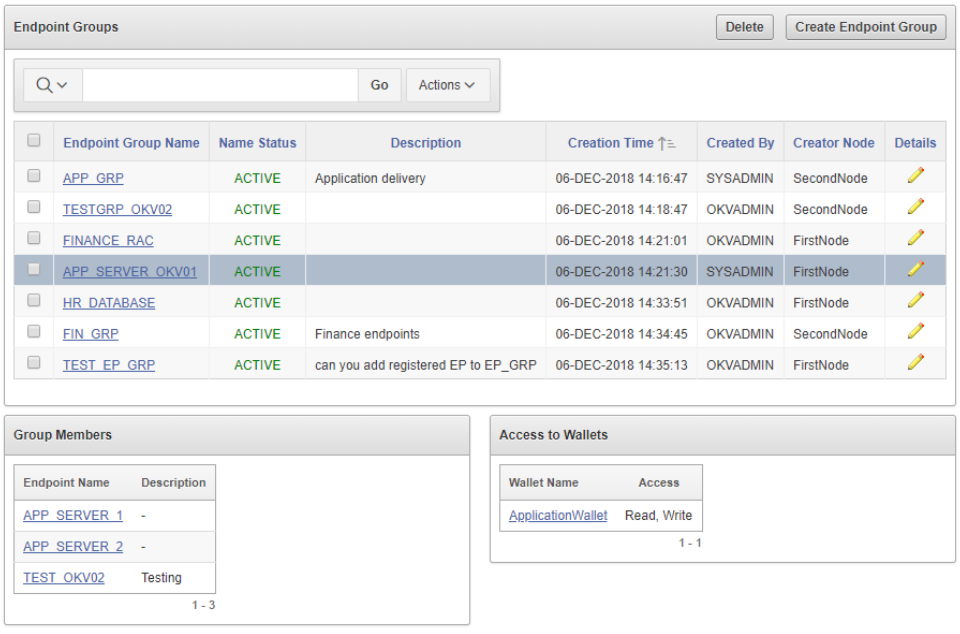
- Naming Conflicts and Resolution
  Oracle Key Vault can resolve naming conflicts that can arise as users create objects such as endpoints, endpoint groups, and user groups.

9.5.2 Creating an Endpoint Group

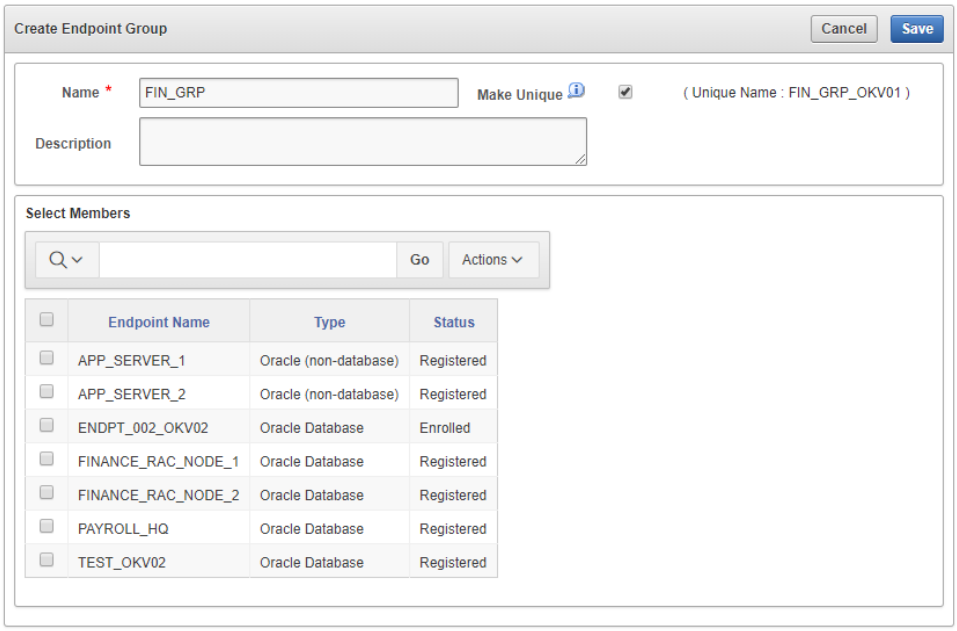
Endpoints that must share a common set of security objects stored in wallets can be grouped into an endpoint group.

For example, endpoints using Oracle Real Application Clusters (Oracle RAC), Oracle GoldenGate, or Oracle Active Data Guard may need to share keys for access to shared data.

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.



3. Click **Create Endpoint Group**.
   The Create Endpoint Group page appears.



4. Enter the name of the new group and a brief description.
5. If you are using a multi-master cluster, then choose whether to select the **Make Unique** checkbox.
   Make Unique helps to control naming conflicts with names across the multi-master cluster environment. Endpoint groups that were created before an Oracle Key Vault conversion to a cluster node are not affected by naming conflicts.
   - If you select **Make Unique**, then the endpoint group will be active immediately and users can use this endpoint group. Clicking **Make Unique** also displays a list of endpoints that you can add to the endpoint group.
   - If you do not select **Make Unique**, then the endpoint group will be created in the `PENDING` state. Oracle Key Vault will then begin a name resolution operation and may rename the endpoint group to a name that is unique across the cluster. If there is a naming collision, then the collision will be reported on the Conflicts page on any node in the cluster. The endpoint group will then be renamed to a unique name. You will need to go to a read-write node of the cluster and either accept the renamed endpoint group or change the endpoint name. If you change the endpoint group name, then this will restart the name resolution operation and the endpoint group will return to a `PENDING` state. An endpoint group in the `PENDING` state cannot be used to perform most operations.
6. In the **Select Members** pane, which lists all the endpoints, check the boxes to the left of each endpoint to add the endpoint to the group.
7. Click **Save** to complete creating the endpoint group.
   The new endpoint group now appears in the Endpoint Groups page.

Related Topics

- Modifying Endpoint Group Details
  You can add endpoints and access mappings to an endpoint group after creating the endpoint group.

- Performing Actions and Searches
  The Oracle Key Vault management console enables you to perform standard actions and search operations, as well as get help information.

9.5.3 Modifying Endpoint Group Details

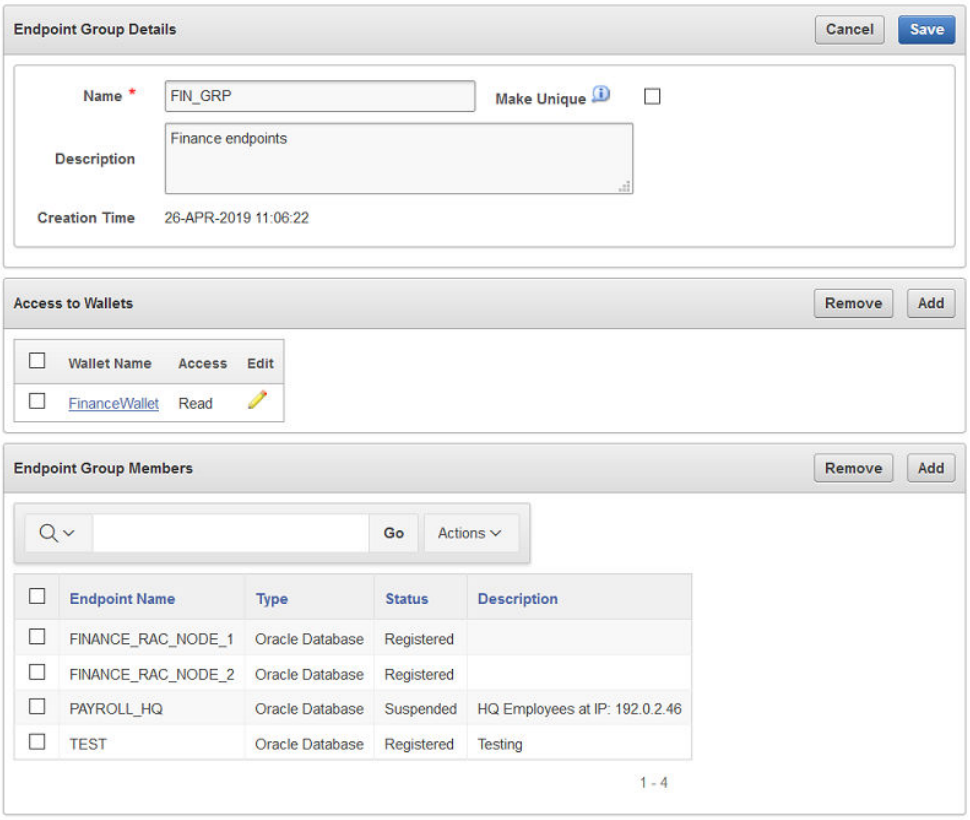
You can add endpoints and access mappings to an endpoint group after creating the endpoint group.

An endpoint can belong to more than one endpoint group. You cannot add one endpoint group to another endpoint group.

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.



4. Modify the description as needed.

   Add or remove access to wallets or endpoint group members by clicking **Add** or **Remove**.
5. Click **Save**.

9.5.4 Granting an Endpoint Group Access to a Virtual Wallet

You can grant an endpoint group access to a virtual wallet.

In a multi-master cluster, you cannot grant access an endpoint group that is in the `PENDING` state 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 **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**.

9.5.5 Adding an Endpoint to an Endpoint Group

You can add an endpoint to a named endpoint group.

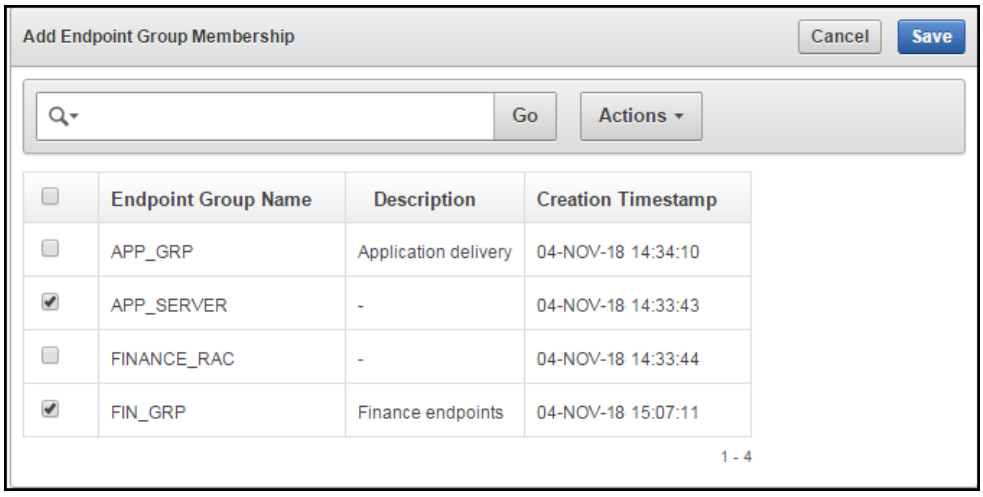
In a multi-master cluster, you cannot add an endpoint that is in the `PENDING` state to an endpoint group. Also, you cannot add an endpoint to an endpoint group that is in the `PENDING` state.

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**.

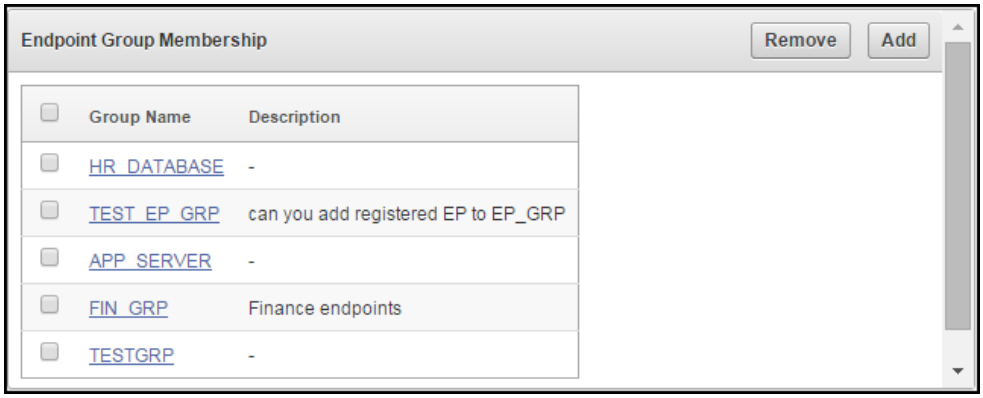
   The Add Endpoint Group Membership page appears.



A list of endpoint groups is displayed under **Endpoint Group Name**.

5. Check the boxes to the left of the endpoint groups you want to add the endpoint to.
6. Click **Save**.

   The Endpoint Group Membership pane displays the checked endpoint group.



Related Topics

- Creating an Endpoint Group
  Endpoints that must share a common set of security objects stored in wallets can be grouped into an endpoint group.

9.5.6 Removing an Endpoint from an Endpoint Group

When you remove an endpoint from an endpoint group, this removes access to wallets that are associated with that endpoint group.

The removal process completes the removal unless the endpoint has been separately granted access to the wallets, directly or through another endpoint group. In a multi-master cluster, you can remove multiple endpoints at the same time. In a multi-master cluster, you cannot remove an endpoint from an endpoint group that is in the `PENDING` state.

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 boxes to the left of the endpoint names to be removed.
5. Click **Remove**.
6. In the confirmation dialog box, click **OK**.

9.5.7 Deleting 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. You can only delete an endpoint group that is in the `PENDING` state if it has no members or access to wallets.

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 boxes to the left of the endpoint group name.
4. Click **Delete**.
5. In the confirmation dialog box, click **OK**.

9.6 Managing Endpoint Details

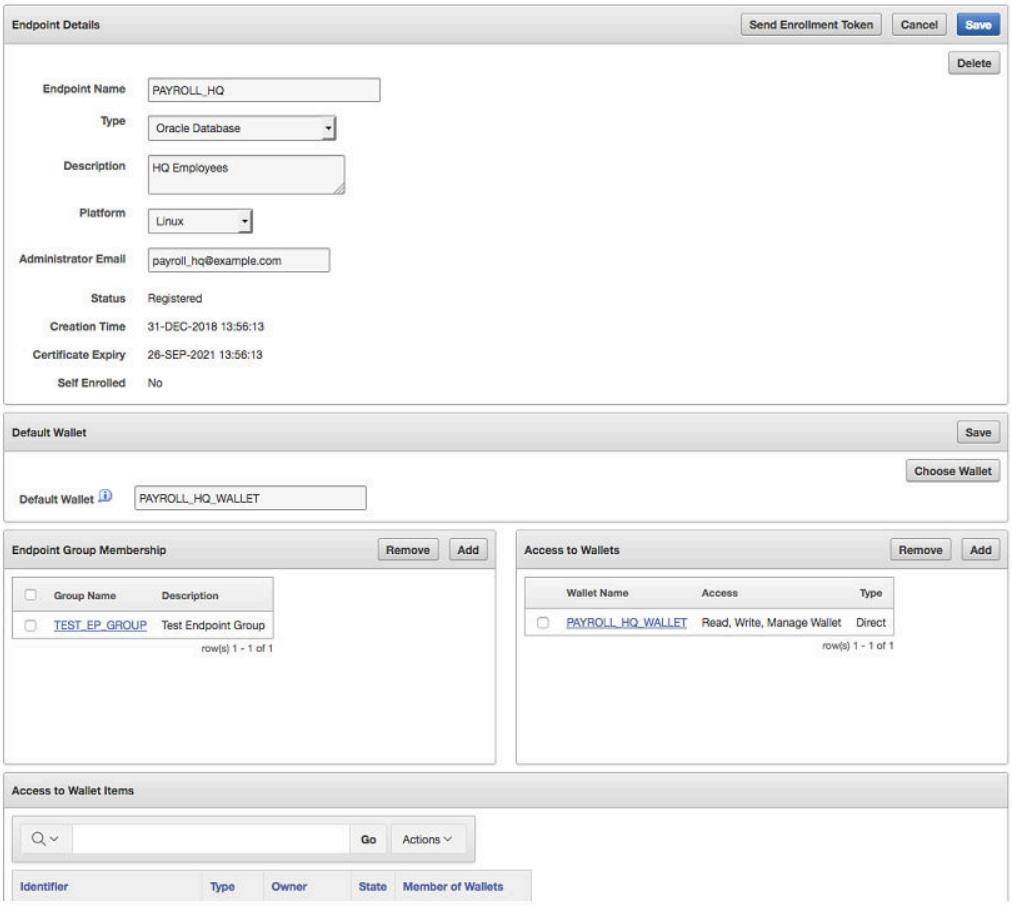
Endpoint details refers to endpoint name, type, description, platform, and email, and adding the endpoint to a group, or upgrading the endpoint software.

- About Endpoint Details
  The Endpoint Details page provides a consolidated view of the endpoint.
- Modifying Endpoint Details
  You can modify the endpoint name, endpoint type, description, platform, and email.
- Global Endpoint Configuration Parameters
  Oracle Key Vault provides endpoint-specific configuration parameters that you can set in the Oracle Key Vault management console.

9.6.1 About Endpoint Details

The Endpoint Details page provides a consolidated view of the endpoint.

To access this page, you can select the **Endpoints** tab and then click the name of an endpoint. From here you can modify endpoint details and complete endpoint management tasks.



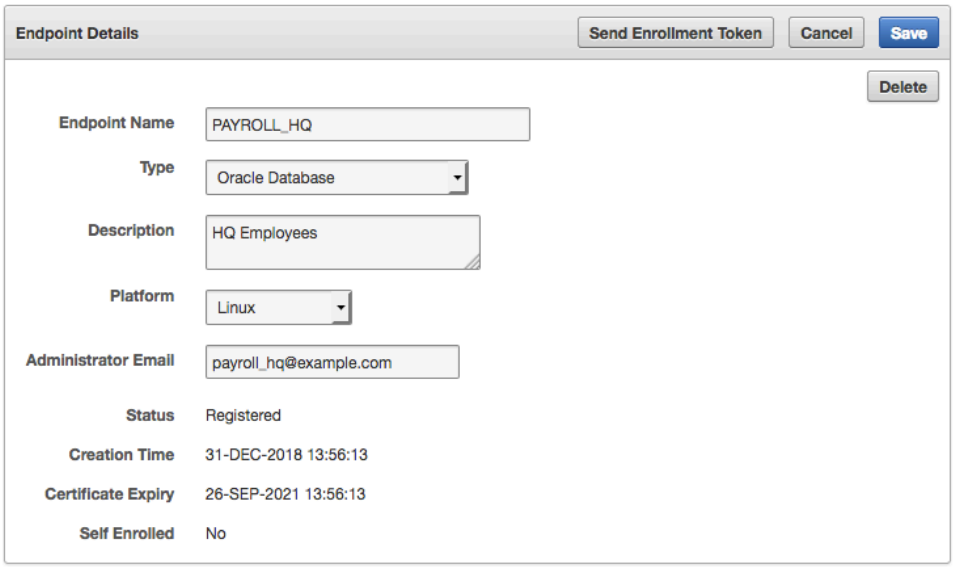
9.6.2 Modifying Endpoint Details

You can modify the endpoint name, endpoint type, description, platform, and email.

In a multi-master cluster, endpoint details can only be modified while the endpoint is in the `PENDING` state by the creator on the node on which it was created.

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. Click the name of the endpoint to display the **Endpoint Details** page.



4. Modify any of the following: endpoint name, endpoint type, description, platform, email as needed.
5. Click **Save**.

9.6.3 Global Endpoint Configuration Parameters

Oracle Key Vault provides endpoint-specific configuration parameters that you can set in the Oracle Key Vault management console.

- About Global Endpoint Configuration Parameters
  Users who have the System Administrator role can centrally update certain endpoint configuration parameters in the Oracle Key Vault management console.
- Setting Global Endpoint Configuration Parameters
  You can set global endpoint configuration parameters in the Oracle Key Vault management console.

9.6.3.1 About Global Endpoint Configuration Parameters

Users who have the 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 take precedence over global parameters. Global parameters take effect when endpoint-specific parameters are cleared. Oracle Key Vault uses the default system parameters if both global and endpoint specific parameters are cleared or not set from Oracle Key Vault management console.

The configuration parameter values set in the Oracle Key Vault management console are applied to endpoints dynamically. The next time that the endpoint contacts Oracle Key Vault server, the updated configuration parameters are applied to the endpoint. If there is an error, then the update is not applied. Both `okvutil` and the PKCS11 library can access and apply the endpoint configuration updates.

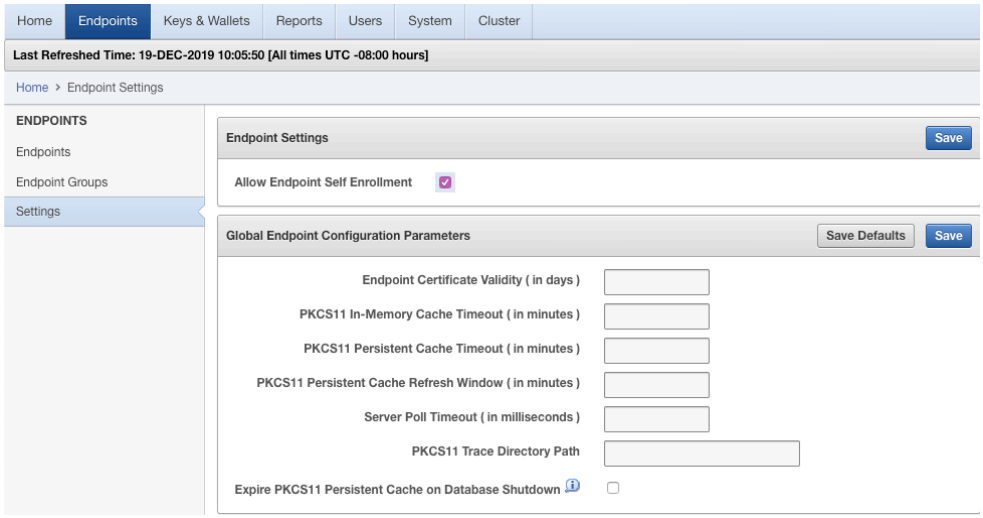
In a multi-master cluster, replication of configuration parameters depends on the replication lag. It is possible that an endpoint will not be able to get an update immediately because the node to which it is connected may not yet have received the new values of the parameters. The endpoint will refresh its configuration when it connects to a node that has new values or if it hasn't refreshed its configuration in the past hour.

9.6.3.2 Setting Global Endpoint Configuration Parameters

You can set global endpoint configuration parameters in the Oracle Key Vault management console.

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.



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 in minutes for which the master encryption key is available after it is cached in the in-memory cache. For more information about the PKCS 11 In-Memory Cache Timeout setting, see PKCS11_CACHE_TIMEOUT Parameter.
     PKCS 11 Cache Persistent Timeout: Specify the duration in minutes for which the master encryption key is available after it is cached in the persistent master encryption key cache. For more information about the **PKCS 11 Cache Persistent Timeout** setting, see PKCS11_PERSISTENT_CACHE_TIMEOUT Parameter.
   - **PKCS 11 Persistent Cache Refresh Window**: Specify the duration in minutes to extend the period of time for which the master encryption key is available after it is cached in the persistent master encryption key cache. For more information about the **PKCS 11 Persistent Cache Refresh Window** setting, see PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW Parameter.

- **Server Poll Timeout**: Specify a timeout in seconds 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.
- **Expire PKCS11 Persistent Cache on Database Shutdown**: Enables or disables the PKCS#11 persistent cache for a given endpoint database to automatically expire upon shutdown of the endpoint database. See EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN Parameter.

4. Click **Save**.

9.7 Upgrading Endpoints

You can perform endpoint upgrades from either the Oracle Key Vault management console login page or from the endpoint.

- Upgrading Endpoint Software from an Endpoint
  You can upgrade the endpoint software from the Oracle Key Vault management console login window.
- Upgrading Endpoint Software on an Enrolled Endpoint
  You should upgrade the endpoint software on an enrolled endpoint any time you upgraded to a new release of Oracle Key Vault.

9.7.1 Upgrading Endpoint Software from an Endpoint

You can upgrade the endpoint software from the Oracle Key Vault management console login window.

- Step 1: Prepare the Endpoint Environment
  Ensure that you have the correct privileges and that the endpoint has the correct configuration, such as Oracle environment variables.
- Step 2: Download the Oracle Key Vault Software onto the Endpoint
  You download the `okvclient.jar` file to local computer.
- Step 3: Install the Oracle Key Vault Software onto the Endpoint
  You must be the endpoint administrator to install the Oracle Key Vault software onto the endpoint.
- Step 4: Perform Post-Installation Tasks
  After you complete the installation, you can configure a TDE connection for the endpoint and verify that the endpoint software was installed correctly.

9.7.1.1 Step 1: Prepare the Endpoint Environment

Ensure that you have the correct privileges and that the endpoint has the correct configuration, such as Oracle environment variables.

These steps assume that the endpoint has already been enrolled in a previous release of Oracle Key Vault.

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, then 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.

9.7.1.2 Step 2: Download the Oracle Key Vault Software onto the Endpoint

You download the `okvclient.jar` file to local computer.

You can download the endpoint software without having to reenroll the endpoint.

1. Log in to the endpoint server as the endpoint administrator.

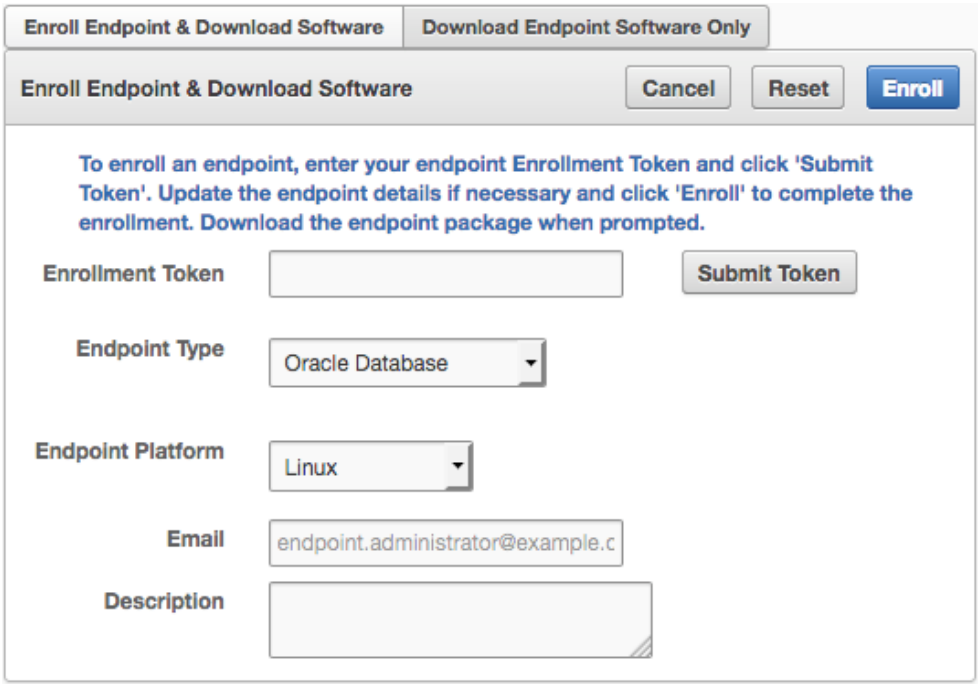
2. Connect to the Oracle Key Vault management console.

   For example:
   `https://192.0.2.254`

   The login page to the Oracle Key Vault management console appears. ***Do not log in.***

3. In the lower-right corner of the login page under **Login**, click **Endpoint Enrollment and Software Download**.

   The **Enroll Endpoint & Download Software** page appears.



4. At the top of the page, click the **Download Endpoint Software Only** tab.

5. In the Download Endpoint Software Only page, select the endpoint platform from the **Platform** drop down menu and click **Download**.
6. Save the file `okvclient.jar` to a desired location.

Related Topics

- Environment Variables and Endpoint Provisioning Guidance
  Environment variables such as `JAVA_HOME` and `OKV_HOME` must be correctly set so that Oracle Key Vault can access its utilities.
- Centralized Management of TDE Master Encryption Keys Using Online Master Keys
  You can use an online master key to centralize the management of TDE master encryption keys over a direct network connection.

9.7.1.3 Step 3: Install the Oracle Key Vault Software onto the Endpoint

You must be the endpoint administrator to install the Oracle Key Vault software onto the endpoint.

1. Ensure that you are logged in to the endpoint server as the endpoint administrator.
2. Navigate to the directory in which you saved the `okvclient.jar` file.
3. Confirm that the target directory exists, and that it is empty.
4. Run the `java` command to install the `okvclient.jar` file.

   ```
   java -jar okvclient.jar -d /home/oracle/okvutil -v
   ```

   In this specification:

   - `-d` specifies the directory location for the endpoint software and configuration files, in this case `/home/oracle/okvutil`.
   - `-v` writes the installation logs to the `/home/oracle/okvutil/log/okvutil.deploy.log` file at the server endpoint.

   `-o` is an optional argument that enables you to overwrite the symbolic link reference to `okvclient.ora` when `okvclient.jar` is deployed in a directory other than the original directory. This argument is used only when you re-enroll an endpoint.

   If you are installing the `okvclient.jar` file on a Windows endpoint system that has Oracle Database release 11.2.0.4 **only**, then include the `-db112` option. (This option is not necessary for any other combination of endpoint platform or Oracle Database version.) For example:

   ```
   java -jar okvclient.jar -d /home/oracle/okvutil -v -db112
   ```

5. When you are prompted for a password, then perform either of the following two steps.

   The optional password goes into two places: `okvutil` and in `ADMINISTER KEY MANAGEMENT`. With `okvutil`, only users who know that password can upload or download content to and from Oracle Key Vault. With `ADMINISTER KEY MANAGEMENT`, it becomes the password that you must use in the `IDENTIFIED BY` *`password`* clause. If you choose not to give a password, then `okvutil upload` and `download` commands will not prompt for a password, and the password for `ADMINISTER KEY MANAGEMENT` becomes `NULL.NULL` is used for an auto-login wallet.

The choices for handling the password are as follows:

- If you want to create a password-protected wallet, at minimum enter a password between 8 and 30 characters and then press **Enter**. For better security, Oracle recommends that you include uppercase letters, lowercase characters, special characters, and numbers in the password. The following special characters are allowed: (.), comma (,), underscore (_), plus sign (+), colon (:), space.

  ```
  Enter new Key Vault endpoint password (<enter> for auto-login):
  Key_Vault_endpoint_password
  Confirm new endpoint password: Key_Vault_endpoint_password
  ```

  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.

- Alternatively, enter no password and then press **Enter**.
  No password will be required when the endpoint connects to Oracle Key Vault with `okvutil`. With the `ADMINISTER KEY MANAGEMENT` statement, the password becomes `NULL`.

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.

Related Topics

- Environment Variables and Endpoint Provisioning Guidance
  Environment variables such as `JAVA_HOME` and `OKV_HOME` must be correctly set so that Oracle Key Vault can access its utilities.
- Centralized Management of TDE Master Encryption Keys Using Online Master Keys
  You can use an online master key to centralize the management of TDE master encryption keys over a direct network connection.

9.7.1.4 Step 4: Perform Post-Installation Tasks

After you complete the installation, you can configure a TDE connection for the endpoint and verify that the endpoint software was installed correctly.

1. Optionally, configure a TDE connection for the endpoint.

   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.

- **On Oracle Linux x86-64, Solaris, AIX, and HP-UX (IA) installations:** Log in as the root and then execute either of the following commands:

  ```
  $ sudo bin/root.sh
  ```

  Or:

  ```
  $ su -
  # bin/root.sh
  ```

  This command creates the directory tree /opt/oracle/extapi/64/hsm/oracle/1.0.0, changes ownership and permissions, then copies the PKCS#11 library into this directory.

- **On Windows installations:** Run the following command:

  ```
  bin\root.bat
  ```

  This command copies the liborapkcs.dll file to the C:\oracle\extapi\64\hsm\oracle\1.0.0 directory.

2. Use a command such as namei or ls -l to confirm that a softlink was created in $ORACLE_BASE/okv/$ORACLE_SID/okvclient.ora to point to the real file in the /conf subdirectory of the installation target directory.

   If the ORACLE_BASE environment variable has not been set, then the softlink was created in $ORACLE_HOME/okv/$ORACLE_SID.

3. 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.

   ```
   $ ./okvutil list
   ```

   If the endpoint is able to connect to Key Vault, then the No objects found message appears. If a Server connect failed message appears, then you must troubleshoot the installation for possible issues. Check that environment variables are correctly set. To get help on the endpoint software, execute the following command:

   ```
   java -jar okvclient.jar -h
   ```

   Output similar to the following appears:

   ```
   Production on Fri Apr 12 15:03:01 PDT 2019
   Copyright (c) 1996, 2019 Oracle. All Rights Reserved.
   Usage:
     java -jar okvclient.jar [-h | -help] [[-v | -verbose] [-d <destination
   directory>] [-o]]

   Options:
     -h or -help : Display command help.
     -v or -verbose : Turn on the verbose mode. Logs will be written to files
   under
                      <destination directory>/log/ directory.
     -d <destination directory> : Specify the software installation directory.
     -o : Overwrite the current symbolic link to okvclient.ora.
   ```

4. After you complete the installation, securely delete the `okvclient.jar` endpoint software file.

9.7.2 Upgrading Endpoint Software on an Enrolled Endpoint

You should upgrade the endpoint software on an enrolled endpoint any time you upgraded to a new release of Oracle Key Vault.

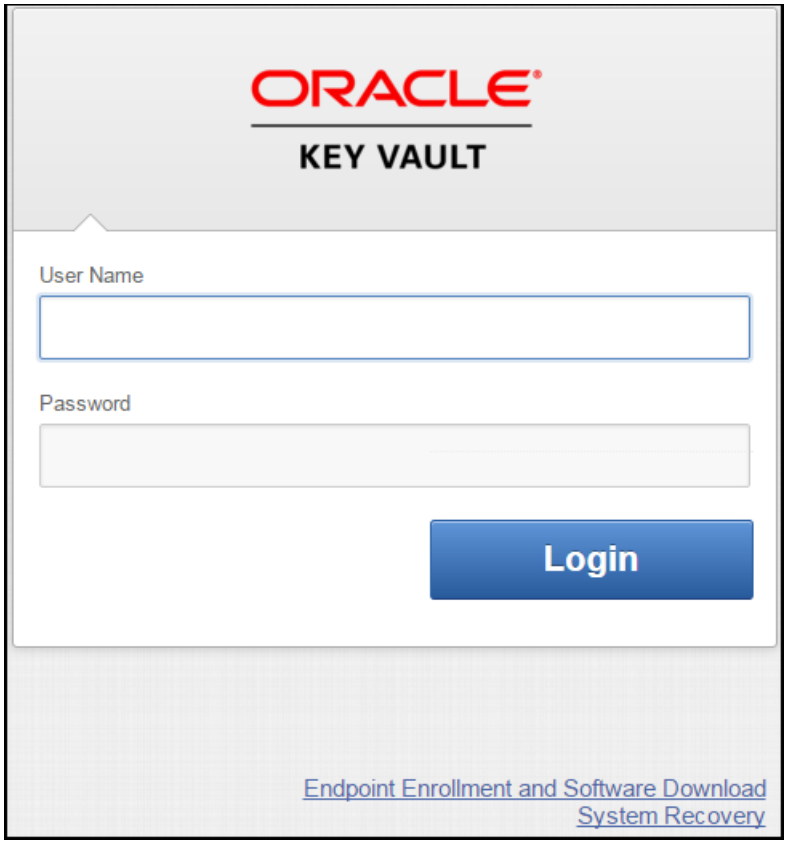
This ensures that you have the latest software on both the Oracle Key Vault server and the endpoint. Oracle highly recommends this for optimum performance. Oracle Key Vault servers can work with endpoint software from the previous major release, but may not work properly with endpoint software that is older. To upgrade the software on an already enrolled endpoint you can download and install the software `okvclient.jar` on the endpoint. You do not need to re-enroll the 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`

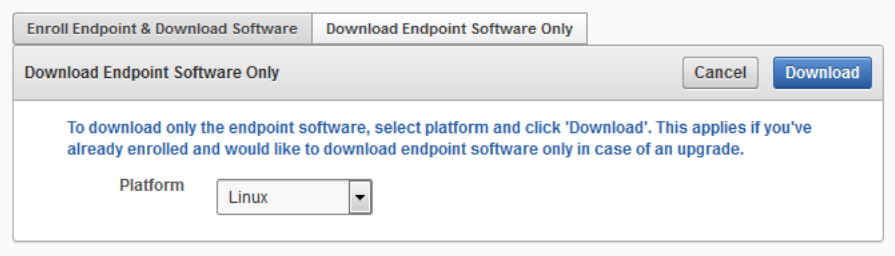
   The login page to the Oracle Key Vault management console appears. *Do not log in*.



3. In the lower-right corner of the login screen, under **Login**, click **Endpoint Enrollment and Software Download**.

4. In the Enroll Endpoint & Download Software page, click **Download Endpoint Software Only**.

   The Download Endpoint Software Only page appears.



5. Select the **Platform** from the drop-down list and then 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.

6. Save the file to an appropriate directory.

7. Verify that the file is downloaded.

After you complete these steps, you can install the Oracle Key Vault software on the endpoint, using the same steps that can be used for an unenrolled endpoint. Oracle recommends that you extract the jar file in the existing endpoint directory because the upgrade endpoint software will not work otherwise. For example:

```
java -jar /path/to/okvclient.jar -d /path/to/existing/ep/files -v
```

Related Topics

- Step 3: Install the Oracle Key Vault Software onto the Endpoint
  You must be the endpoint administrator to install the Oracle Key Vault software onto the endpoint.

10

Enrolling Endpoints for Oracle Key Vault

After an endpoint is registered in Oracle Key Vault, an endpoint administrator enrolls and provisions the endpoint to manage security objects in Key Vault.

- About Endpoint Enrollment and Provisioning
  Endpoints are Key Vault clients that use the server to store and manage security objects, share them with trusted peers, and retrieve them.
- Finalizing Enrollment and Provisioning
  To enroll and provision a registered endpoint an endpoint administrator must download and then install the `okvclient.jar` file.
- Environment Variables and Endpoint Provisioning Guidance
  Environment variables such as `JAVA_HOME` and `OKV_HOME` must be correctly set so that Oracle Key Vault can access its utilities.
- Endpoints That Do Not Use the Oracle Key Vault Client Software
  Third-party KMIP endpoints do not use the Oracle Key Vault software `okvutil` and `liborapkcs.so`.
- Transparent Data Encryption Endpoint Management
  Oracle Key Vault can manage TDE keys by using the same PKCS#11 interface that TDE uses to communicate with an external keystore.
- Endpoint okvclient.ora Configuration File
  Oracle Key Vault endpoint libraries and utilities use the `okvclient.ora` configuration file, which stores the configuration parameters associated with the endpoint.

10.1 About Endpoint Enrollment and Provisioning

Endpoints are Key Vault clients that use the server to store and manage security objects, share them with trusted peers, and retrieve them.

These clients can be systems like Oracle database servers, Oracle middleware servers, operating systems, and other information systems.

An Oracle Key Vault system administrator first adds (or registers) the endpoint to Key Vault, and 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.

Endpoint enrollment is a three-step process performed by two kinds of administrative users summarized in the following table.

Table 10-1 Summary of Endpoint Enrollment

Step #	Task	Performed by	Endpoint Status (as seen on Oracle Key Vault Management Console)
1.	1. System administrator creates an endpoint. 2. If this is an Oracle database, a key administrator creates a virtual wallet. 3. System administrator adds or registers the endpoint to Oracle Key Vault. An enrollment token for the endpoint is generated. 4. System administrator sends the enrollment token to the endpoint administrator to complete the enrollment process.	Users with the System Administrator role and Key Administrator role on Oracle Key Vault	Registered
2.	1. Verify the enrollment token. 2. Submit enrollment token to download endpoint software `okvclient.jar` to the endpoint.	Endpoint administrator using the Oracle Key Vault management console	Enrolled
3.	Install `okvclient.jar` on the endpoint.	Endpoint administrator on endpoint	Enrolled

Endpoint enrollment ensures that only authorized endpoints can communicate with Oracle Key Vault because the utilities needed to communicate are bundled with the `okvclient.jar` endpoint software file.

`okvclient.jar` contains the following:

- A Transport Layer Security (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 such as the Oracle 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. Each Oracle RAC-enabled database corresponds to one virtual wallet in Oracle Key Vault. Each Oracle RAC instance of that database corresponds to an endpoint in Oracle Key Vault. All endpoints for each database share the same wallet as their default wallet. You must download one distinct `okvclient.jar` for each instance.

Related Topics

- Types of Endpoint Enrollment
  The first step in enrolling an endpoint is to add the endpoint to Oracle Key Vault.
- Endpoint okvclient.ora Configuration File
  Oracle Key Vault endpoint libraries and utilities use the `okvclient.ora` configuration file, which stores the configuration parameters associated with the endpoint.
- Oracle Key Vault okvutil Endpoint Utility Reference
  The `okvutil` utility enables you to perform tasks uploading and downloading security objects.

10.2 Finalizing Enrollment and Provisioning

To enroll and provision a registered endpoint an endpoint administrator must download and then install the `okvclient.jar` file.

- Step 1: Enroll the Endpoint and Download the Software
  You must have the endpoint's enrollment token before you can download the endpoint software `okvclient.jar`.
- Step 2: Prepare the Endpoint Environment
  You must ensure that you have the right version of the Java Development Toolkit (JDK) and that the Oracle environment variables are set.
- Step 3: Install the Oracle Key Vault Software onto the Endpoint
  To upgrade to the latest endpoint software for an enrolled endpoint, you can download the endpoint software without having to reenroll the endpoint.
- Step 4: Perform Post-Installation Tasks
  The post-installation procedures include optionally configuring a TDE connection for the endpoint, checking the installation contents, and deleting the `okvclient.jar file`.

10.2.1 Step 1: Enroll the Endpoint and Download the Software

You must have the endpoint's enrollment token before you can 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.

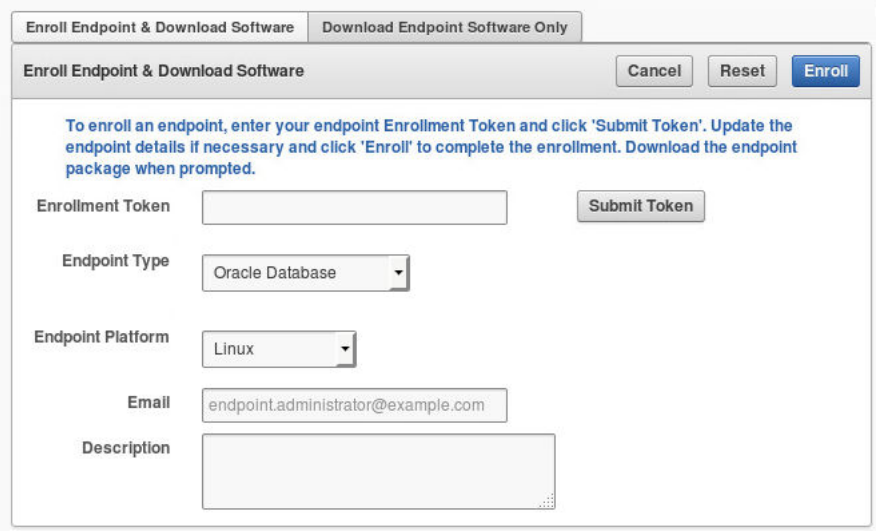
1. Log in to the endpoint server as the endpoint administrator.
2. Connect to the Oracle Key Vault management console.

   For example:

   ```
   https://192.0.2.254
   ```

   The login page to the Oracle Key Vault management console appears. ***Do not log in.***
3. In the lower right corner of the login page, click the **Endpoint Enrollment and Software Download** button, which is below the **Login** button.

   The Enroll Endpoint & Download Software page appears.



4. At the top of the page, click the **Enroll Endpoint & Download Software** tab.

   The next two steps depend on how the endpoint was added to or registered with Oracle Key Vault.

5. If the endpoint was registered by an Oracle Key Vault system administrator, then do the following:

   a. Enter the endpoint's enrollment token in **Enrollment Token**, and click **Submit Token**.

      If the token is valid, then a valid token message appears to the right of the **Submit Token** button. The **Endpoint Type**, **Endpoint Platform**, **Email** and **Description** fields are automatically populated with the values that were entered during endpoint registration.

      If the token is invalid, then an invalid token message appears. Check the token and retry the download procedure.

   b. Click **Enroll** at the top right corner of the page.

6. If the endpoint was registered by self-enrollment, then do the following:

   a. Bypass the step of validating the token because self-enrolled endpoints have no enrollment token.

   b. From the **Endpoint Type** list, select the type of endpoint: **Oracle Database**, **Oracle (non-database)**, or **Other**. If you are using Transparent Data Encryption (TDE), then you must enter Oracle Database.

   c. From the **Endpoint Platform** list, select the platform: **Linux**, **Solaris SPARC**, **Solaris x64**, **AIX**, **HPUX**, **Windows**.

   d. In the **Email** field, enter the email address of the endpoint administrator, for notification purposes. This field is optional but recommended.

   e. In the **Description** field, enter meaningful and identifying information for the endpoint. This field is also optional but strongly recommended.

   f. Click **Enroll** at the top right corner of the page.

7. In the directory window that appears, follow the prompt to save the `okvclient.ora` endpoint software file.

   You must navigate to the directory where you want to save the file.

8. Save the file to a secure directory with appropriate permissions in place so that it cannot be read or copied by others.
9. Verify that the file has been downloaded.

   If the download fails, then you must obtain a new enrollment token from the key administrator for the endpoint and repeat these steps, starting with Step 5. 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.

At this stage, you are ready to install the Oracle Key Vault `okvclient.jar` software file on the endpoint, starting with preparing the endpoint environment.

Related Topics

- Step 2: Prepare the Endpoint Environment
  You must ensure that you have the right version of the Java Development Toolkit (JDK) and that the Oracle environment variables are set.

10.2.2 Step 2: Prepare the Endpoint Environment

You must ensure that you have the right version of the Java Development Toolkit (JDK) and that the Oracle environment variables are set.

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, then 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.

10.2.3 Step 3: Install the Oracle Key Vault Software onto the Endpoint

To upgrade to the latest endpoint software for an enrolled endpoint, you can download the endpoint software without having to reenroll the endpoint.

1. Ensure that you are logged in to the endpoint server as the endpoint administrator.
2. Navigate to the directory in which you saved the `okvclient.jar` file.
3. Confirm that the target directory exists, and that it is empty.
4. Run the `java` command to install the `okvclient.jar` file.

   ```
   java -jar okvclient.jar -d /home/oracle/okvutil -v
   ```

   In this specification:

   - `-d` specifies the directory location for the endpoint software and configuration files, in this case `/home/oracle/okvutil`.

- `-v` writes the installation logs to the `/home/oracle/okvutil/log/okvutil.deploy.log` file at the server endpoint.

`-o` is an optional argument that enables you to overwrite the symbolic link reference to `okvclient.ora` when `okvclient.jar` is deployed in a directory other than the original directory. This argument is used only when you re-enroll an endpoint.

If you are installing the `okvclient.jar` file on a Windows endpoint system that has Oracle Database release 11.2.0.4 **only**, then include the `-db112` option. (This option is not necessary for any other combination of endpoint platform or Oracle Database version.) For example:

```
java -jar okvclient.jar -d /home/oracle/okvutil -v -db112
```

5. When you are prompted for a password, then perform either of the following two steps.

   The optional password goes into two places: `okvutil` and in `ADMINISTER KEY MANAGEMENT`. With `okvutil`, only users who know that password can upload or download content to and from Oracle Key Vault. With `ADMINISTER KEY MANAGEMENT`, it becomes the password that you must use in the `IDENTIFIED BY` *`password`* clause. If you choose not to give a password, then `okvutil upload` and `download` commands will not prompt for a password, and the password for `ADMINISTER KEY MANAGEMENT` becomes `NULL`.`NULL` is used for an auto-login wallet.

   The choices for handling the password are as follows:

   - If you want to create a password-protected wallet, at minimum enter a password between 8 and 30 characters and then press **Enter**. For better security, Oracle recommends that you include uppercase letters, lowercase characters, special characters, and numbers in the password. The following special characters are allowed: (.), comma (,), underscore (_), plus sign (+), colon (:), space.

     ```
     Enter new Key Vault endpoint password (<enter> for auto-login):
     Key_Vault_endpoint_password
     Confirm new endpoint password: Key_Vault_endpoint_password
     ```

     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.

   - Alternatively, enter no password and then press **Enter**.
     No password will be required when the endpoint connects to Oracle Key Vault with `okvutil`. With the `ADMINISTER KEY MANAGEMENT` statement, the password becomes `NULL`.

   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.

10.2.4 Step 4: Perform Post-Installation Tasks

The post-installation procedures include optionally configuring a TDE connection for the endpoint, checking the installation contents, and deleting the `okvclient.jar file`.

1. Optionally, configure a TDE connection for the endpoint.

   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.

   - **On Oracle Linux x86-64, Solaris, AIX, and HP-UX (IA) installations:** Log in as the root and then execute either of the following commands:

     ```
     $ sudo bin/root.sh
     ```

     Or:

     ```
     $ su -
     # bin/root.sh
     ```

     This command creates the directory tree `/opt/oracle/extapi/64/hsm/oracle/1.0.0`, changes ownership and permissions, then copies the PKCS#11 library into this directory.

   - **On Windows installations:** Run the following command:

     ```
     bin\root.bat
     ```

     This command copies the `liborapkcs.dll` file to the `C:\oracle\extapi\64\hsm\oracle\1.0.0` directory.

2. Use a command such as `namei` or `ls -l` to confirm that a softlink was created in `$ORACLE_BASE/okv/$ORACLE_SID/okvclient.ora` to point to the real file in the `/conf` subdirectory of the installation target directory.

   If the `ORACLE_BASE` environment variable has not been set, then the softlink was created in `$ORACLE_HOME/okv/$ORACLE_SID`.

3. 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.

   ```
   $ ./okvutil list
   ```

   If the endpoint is able to connect to Key Vault, then the `No objects found` message appears. If a `Server connect failed` message appears, then you must troubleshoot the installation for possible issues. Check that environment variables are correctly set. To get help on the endpoint software, execute the following command:

   ```
   java -jar okvclient.jar -h
   ```

   Output similar to the following appears:

   ```
   Production on Fri Apr 12 15:03:01 PDT 2019
   Copyright (c) 1996, 2019 Oracle. All Rights Reserved.
   ```

```
Usage:
  java -jar okvclient.jar [-h | -help] [[-v | -verbose] [-d <destination
directory>] [-o]]

Options:
  -h or -help : Display command help.
  -v or -verbose : Turn on the verbose mode. Logs will be written to files
under
                   <destination directory>/log/ directory.
  -d <destination directory> : Specify the software installation directory.
  -o : Overwrite the current symbolic link to okvclient.ora.
```

4. After you complete the installation, securely delete the `okvclient.jar` endpoint software file.

10.3 Environment Variables and Endpoint Provisioning Guidance

Environment variables such as `JAVA_HOME` and `OKV_HOME` must be correctly set so that Oracle Key Vault can access its utilities.

- How the Location of JAVA_HOME Location Is Determined
  The default location for the `okvclient.ora` file is the `$OKV_HOME/conf` directory.
- Location of the okvclient.ora File and Environment Variables
  `$OKV_HOME` is the destination directory for the endpoint software specified with the `-d` option during installation.
- Setting OKV_HOME for Non-Database Utilities to Communicate with Oracle Key Vault
  For non-database utilities, you must set the environment variable `OKV_HOME` to point to the destination directory for the endpoint software.
- Environment Variables in sqlnet.ora File
  You must consider several points while using the `srvctl` utility on Oracle Database endpoints.

10.3.1 How the Location of JAVA_HOME Location Is Determined

The default location for the `okvclient.ora` file is the `$OKV_HOME/conf` directory.

When you provision endpoints you must know how the installation process determines the location of Java home and the `okvclient.ora` file.

The endpoint software installation process uses the following rules to determine the Java home location:

- If a user-defined `JAVA_HOME` environment variable exists, the installation process uses this value.
- If `JAVA_HOME` is not set, then the installation process looks for it in the `java.home` system property of the Java Virtual Machine (JVM).

After the `JAVA_HOME` path is determined, the installation process adds it to the `okvclient.ora` configuration file to be used by all `okvutil` commands.

You can force `okvutil` to use a different `JAVA_HOME` setting by using one of the following 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
  ```

10.3.2 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 `okvclient.ora` file is a configuration file in the `$OKV_HOME/conf` directory .

In addition to the `$OKV_HOME/conf` file, the installation process creates a soft link to `okvclient.ora` for an existing database. The location of the soft link depends on the following:

- If the `$ORACLE_BASE` environment variable is set, then the installation process creates a symbolic link to the `okvclient.ora` configuration file (in `$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 `$ORACLE_HOME` environment variable is set but the `$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.

10.3.3 Setting OKV_HOME for Non-Database Utilities to Communicate with Oracle Key Vault

For non-database utilities, you must set the environment variable `OKV_HOME` to point to the destination directory for the endpoint software.

You must manually set `OKV_HOME` because the installation process does not set this variable automatically. Setting `OKV_HOME` enables utilities to communicate with Oracle 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 such as RMAN. For example, if you spawn a new `xterm` window, then you will need to set `OKV_HOME` in this environment before running RMAN.

10.3.4 Environment Variables in sqlnet.ora File

You must consider several 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.
- For Oracle Database endpoints, if you are using the `srvctl` utility and setting environment variables in `sqlnet.ora`, then you must set them in both the operating system and the `srvctl` environment.
- The operating system and `srvctl` utility should have `$ORACLE_SID`, `$ORACLE_HOME` and `$ORACLE_BASE` set to the same values.

10.4 Endpoints That Do Not Use the Oracle Key Vault Client Software

Third-party KMIP endpoints do not use the Oracle Key Vault software `okvutil` and `liborapkcs.so`.

In this case you must manually set the Transport Layer Security (TLS) authentication as follows:

1. Extract the `ssl` directory from the `okvclient.jar` file.

   ```
   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

10.5 Transparent Data Encryption Endpoint Management

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 Oracle Key Vault for storing and retrieving TDE master encryption 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, you can directly upload the keys in the wallet to Oracle 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 Oracle 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 TDE ignores the `WITH BACKUP` clause in an Oracle Key Vault online key deployment, even if it is required for the `ADMINISTER KEY MANAGEMENT` statement.

Oracle Database 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.

Example 10-1 shows examples of setting an encryption key.

Example 10-1 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 and later
```

Related Topics

- Centralized Management of TDE Master Encryption Keys Using Online Master Keys
  You can use an online master key to centralize the management of TDE master encryption keys over a direct network connection.

10.6 Endpoint okvclient.ora Configuration File

Oracle Key Vault endpoint libraries and utilities use the `okvclient.ora` configuration file, which stores the configuration parameters associated with the endpoint.

The `okvclient.ora` file consists of key-value pairs separated by an equal sign (=). You can set the following parameters in the endpoint configuration file:

- `SERVER=`*`node1_IP`*`:`*`node1_port`*`/`*`node1_DN`*`,node2_IP:`*`node2_port`*`/`*`node2_DN`*`,...`

  This parameter specifies the IP address and port number of the Oracle 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=`*`standby_server_IP`*`:`*`standby_server port`*

  This is the standby server. If primary-standby is configured, then this parameter shows the standby IP address.

- `READ_SERVER=`*`node1_IP`*`:`*`node1_port`*`/`*`node1_DN`*`,`*`node2_IP`*`:`*`node2_port`*`/`*`node2_DN`*`,...`
  This parameter specifies the list of read-only servers.

- `SSL_WALLET_LOC=`*`directory`*

  This parameter specifies the location of the wallet containing TLS credentials for the endpoint.

- `SERVER_POLL_TIMEOUT=`*`timeout_value`*

  You can use the `SERVER_POLL_TIMEOUT` parameter 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 clients first establish a non-blocking TCP connection to Oracle Key Vault to quickly detect unreachable servers.

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.

The `CONF_ID` value in an `okvclient.ora` file is a unique internal value that helps an Oracle database to find its virtual wallet in Oracle Key Vault. Do not modify this value.

11

Deploying Oracle Key Vault on an Oracle Cloud Infrastructure VM Compute Instance

You can install Oracle Key Vault on an Oracle Cloud Infrastructure (OCI) VM compute instance from Oracle Cloud Marketplace.

- About Deploying Oracle Key Vault on an Oracle Cloud Infrastructure Compute Instance
  Oracle Key Vault on Oracle Cloud Marketplace is the cloud-based version of Oracle Key Vault and provides flexible, continuous and scalable key management.
- Benefits of Using Oracle Key Vault in Oracle Cloud Infrastructure
  Quick deployments and ease of use are among the benefits of using an Oracle Key Vault Oracle Cloud Infrastructure (OCI) compute instance.
- Provisioning an Oracle Key Vault Compute Instance
  The provisioning process for an Oracle Key Vault compute instance entails launching the compute instance and performing post-launch and post-installation tasks.
- General Management of an Oracle Key Vault Compute Instance
  You can perform many of the Oracle Key Vault compute instance general management tasks in the Oracle Key Vault management console.
- Migrating Oracle Key Vault Deployments Between On-Premises and OCI
  You can migrate an Oracle Key Vault standalone, primary-standby or cluster deployment from an on-premises environment to OCI or back.

Related Topics

- Oracle Key Vault Available in the Oracle Cloud Marketplace
  Starting with this release, you can deploy Oracle Key Vault to run on an Oracle Cloud Infrastructure (OCI) VM compute instance.

11.1 About Deploying Oracle Key Vault on an Oracle Cloud Infrastructure Compute Instance

Oracle Key Vault on Oracle Cloud Marketplace is the cloud-based version of Oracle Key Vault and provides flexible, continuous and scalable key management.

Oracle Key Vault is quick and easy to launch on a VM compute instance of any shape or size in your OCI tenancy. This eliminates the need to procure hardware and drastically shortens the time to provision a fully functional Oracle Key Vault deployment. Oracle Key Vault deployed on an OCI VM compute instance (referred to as an Oracle Key Vault compute instance) is private to your tenancy and is managed by you. After the launch, an Oracle Key Vault compute instance has the same look and feel as an on-premises Oracle Key Vault installation, with the same flexibility in configuration.

An Oracle Key Vault server that is deployed on Oracle Cloud Infrastructure (OCI) VM compute instance can operate in the following situations:

- A standalone environment
- Be paired with another Oracle Key Vault server in OCI or on-premises to form a primary-standby configuration
- Be paired with other nodes in OCI or on-premises to form a multi-master cluster

The Oracle Key Vault multi-master cluster nodes could be entirely in OCI forming a cloud-only Oracle Key Vault cluster or some of the nodes can exist on-premises, thus forming a hybrid Oracle Key Vault cluster. This flexible deployment provides scalability regardless of whether Oracle Key Vault nodes are deployed in on-premises or cloud environments.

The Oracle Key Vault compute instance deployment enables the use of Oracle Key Vault to manage the encryption keys of your OCI-based database deployments. This enables you to maintain control over your encryption keys in a cloud environment. You can have up to 16 Oracle Key Vault compute instances in a multi-master cluster, distributed across any of the Oracle Cloud regions, to provide key management services to your globally distributed, on-premises, hybrid, or cloud-only Oracle database deployments.

When you enroll endpoints with the Oracle Key Vault compute instance, you must ensure that they are in the same VCN as the Oracle Key Vault compute instance itself. The endpoints will communicate with the Oracle Key Vault compute instance using the private IP of the instance. You can optionally configure the Oracle Key Vault compute instance to have a public IP address that can be used to access the Oracle Key Vault management console. You must configure the network to ensure that connectivity exists between Oracle Key Vault compute instances as well as between endpoints and the Oracle Key Vault compute instances.

11.2 Benefits of Using Oracle Key Vault in Oracle Cloud Infrastructure

Quick deployments and ease of use are among the benefits of using an Oracle Key Vault Oracle Cloud Infrastructure (OCI) compute instance.

- **Key management for OCI-based database environment:** The Oracle Key Vault compute instance deployment provides key management to your OCI-based database environments as well as on-premises and hybrid database environments. This enables you to own, manage, and maintain control over encryption keys of your database environments in the cloud.
- **Quick deployment:** You can launch the Oracle Key Vault compute instance within minutes and without the need to manage hardware or set up virtual machines. After it is launched, the Oracle Key Vault compute instance can run stand-alone, or be added to a multi-master cluster, or used in primary-standby configuration. You can enroll endpoints with an Oracle Key Vault compute instance. This way, you can quickly set up a production environment. You can also use Oracle Key Vault compute instances to quickly set up a test and development environment to validate and experiment with various use-cases and deployment scenarios of Oracle Key Vault.
- **Scaling out a production environment during peak load or hardware unavailability:** If you use FastConnect or IPSec VPN in OCI, then you can extend

the Oracle Key Vault cloud deployments to an on-premises environment. Using FastConnect or IPSec VPN, you can pair Oracle Key Vault nodes on-premises with Oracle Key Vault compute instances in OCI to form a hybrid cluster. You can use a hybrid cluster to run production Oracle Key Vault servers in OCI, or use them to expand the Oracle Key Vault cluster temporarily. Oracle Key Vault compute instances can be added quickly as new nodes to an on-premises, OCI or hybrid Oracle Key Vault cluster. This type of deployment provides spontaneous elasticity to the Oracle Key Vault cluster, and can be used to address any temporary increase of load on nodes of the Oracle Key Vault cluster.

- **Reduced latency for hybrid database environments:** For use cases where the data is shared between on-premises and cloud databases, managing the keys in a hybrid Oracle Key Vault cluster provides for locality of reference. Because the keys are available on all nodes of the cluster, the cluster subgroups can be setup in such a way that the databases in the cloud can primarily fetch the keys from the cluster nodes in OCI and the on-premises databases can primarily fetch the keys from cluster nodes that are provisioned on-premises.
- **Simplified transition of on-premises to OCI-based Oracle Key Vault clusters:** If you are connected to OCI using FastConnect or IPSec VPN, then you can extend your on-premises Oracle Key Vault cluster by adding Oracle Key Vault compute instances to that cluster. The IP addresses of the Oracle Key Vault nodes in OCI are added to the scan lists of your database endpoints. Once you have the appropriate number of Oracle Key Vault nodes in your OCI tenancy, you can remove the on-premises Oracle Key Vault nodes from the cluster. Following the same procedure, it is possible to seamlessly transition from an Oracle Key Vault cluster in OCI back to an on-premises Oracle Key Vault cluster.
- **Engaging OCI infrastructure and services:** You can take advantage of the unique benefits of the Oracle Cloud Infrastructure. If you install multiple Oracle Key Vault compute instances in the same region, you can choose to deploy them in different availability domains (fault domains are selected automatically, but can be changed) to guarantee the highest possible availability of your key management service. Services such as DNS and NTP are also natively available in OCI. You do not have to set them up, thereby simplifying Oracle Key Vault provisioning.

11.3 Provisioning an Oracle Key Vault Compute Instance

The provisioning process for an Oracle Key Vault compute instance entails launching the compute instance and performing post-launch and post-installation tasks.

- About Provisioning an Oracle Key Vault Compute Instance
  To provision the Oracle Key Vault compute instance, you choose an Oracle Key Vault image as your custom image.
- Launching the Oracle Key Vault Compute Instance
  The launching process for the Oracle Key Vault compute instance should take roughly two to five minutes.

11.3.1 About Provisioning an Oracle Key Vault Compute Instance

To provision the Oracle Key Vault compute instance, you choose an Oracle Key Vault image as your custom image.

You will launch this image from the OCI Marketplace on a compute shape. After you complete the process, the Oracle Key Vault compute image becomes unique to your environment. The disk size of this image is 2 TB.

After you complete the launch, you can begin to use the Oracle Key Vault compute image immediately. The steps that you must perform after the launch are similar to the steps that you would perform for an on-premises Oracle Key Vault installation.

Related Topics

- Installing and Configuring Oracle Key Vault
  You must download the Oracle Key Vault application software, and then you can perform the installation.

11.3.2 Launching the Oracle Key Vault Compute Instance

The launching process for the Oracle Key Vault compute instance should take roughly two to five minutes.

- About Launching the Oracle Key Vault Compute Instance
  The launch process requires some minor preparation work on your system.
- Step 1: Ensure That You Have Prerequisites in Place
  Before you can launch an Oracle Key Vault compute instance, you must ensure that you have prerequisites in place in the Oracle cloud.
- Step 2: Find the Oracle Key Vault Image
  The Oracle Key Vault image is available on the Oracle Cloud Marketplace web site.
- Step 3: Launch the Oracle Key Vault VM Compute Instance
  You perform the entire launching process in the Oracle Cloud Marketplace.
- Step 4: Perform Post-Launch and Post-Installation Tasks
  After you launch Oracle Key Vault in an OCI compute instance, you first perform the post-launch task, followed by post-installation tasks.

11.3.2.1 About Launching the Oracle Key Vault Compute Instance

The launch process requires some minor preparation work on your system.

Before you begin the launch process, ensure that the endpoints that you plan to use are in the same VCN as the Oracle Key Vault instance will be. The endpoints will communicate with Oracle Key Vault using the private IP of the compute instance. Optionally, the Oracle Key Vault compute instance can have a public IP that can be used to access the Oracle Key Vault management console. You will also set up the network and configure it to ensure that network connectivity will exist between the endpoints and the OCI compute instances.

11.3.2.2 Step 1: Ensure That You Have Prerequisites in Place

Before you can launch an Oracle Key Vault compute instance, you must ensure that you have prerequisites in place in the Oracle cloud.

Ensure that the following are in place:

- You have an Oracle cloud account.
- You have access to your assigned Oracle cloud tenant.

- You have sufficient compute node resources within the Oracle cloud tenant.

11.3.2.3 Step 2: Find the Oracle Key Vault Image

The Oracle Key Vault image is available on the Oracle Cloud Marketplace web site.

1. Log in to the Oracle Cloud Marketplace web site.

   https://cloudMarketplace.oracle.com/marketplace/oci

2. In the **Products** search field, enter `Oracle Key Vault` and then click **Go**.
3. Under the Search Results, click **Oracle Key Vault** to navigate to the Oracle Key Vault page.

11.3.2.4 Step 3: Launch the Oracle Key Vault VM Compute Instance

You perform the entire launching process in the Oracle Cloud Marketplace.

1. Click the **Get App** button.
2. If you already have an OCI account, select your home region, and then click **Sign In**. Otherwise, click **Sign Up** to create a new account.
3. In the **Get Version** menu, ensure that **Oracle Key Vault 18.3** is displayed.
4. From the **Compartment** menu, select your compartment.
5. Select the **I have reviewed the terms and conditions** check box.
6. In the Oracle Key Vault page, select **Launch Instance**.
7. In the page that appears, click **Change Instance**.
8. For the shape, select **VM.Standard2.2** or bigger. If you are using an older standard, then select **VM.Standard1.4** or bigger. Then click **Select Shape**.

   Next, you are ready to configure the network.
9. Upload your SSH public key.
10. Click **Advanced Options**, and then choose the **Network** tab.

    Here you can replace the default private address with another one. Both of these addresses must be within the range of your current subnet. In addition, you can change the host name to match your naming convention. Otherwise, the host name will be constructed from `okv|`*`MAC-address-of-NIC`*.
11. In the Boot Volume area, do not select any settings.
12. Click **Create** to complete the shape creation.

    In a moment, the Oracle Key Vault compute image starts and is made available as an Oracle Key Vault server.

At this stage, you must perform the post-launch and post-installation steps.

Related Topics

- Network Port Requirements
  Network port requirements includes requirements for SSH/SCP, SNMP, HTTPS, listeners, KMIP, and TCP ports.

11.3.2.5 Step 4: Perform Post-Launch and Post-Installation Tasks

After you launch Oracle Key Vault in an OCI compute instance, you first perform the post-launch task, followed by post-installation tasks.

The post-launch task is to set the installation passphrase. After you set this passphrase, you must perform the post-installation tasks, which are the same tasks that are required for an on-premises deployment. After you complete the post-installation tasks, you can start building your Oracle Key Vault cluster, set up the primary-standby configuration, or leave Oracle Key Vault in stand-alone mode.

1. Set the installation passphrase.
   a. In a command prompt, log in as the `opc` user.

      ```
      ssh opc@Oracle_Key_Vault_OCI_IP_address
      ```

   b. Set the installation passphrase by executing the following command:

      ```
      set_installation_passphrase
      ```

   c. When prompted, enter and confirm the installation passphrase.

      After you successfully enter the passphrase, the system deletes the `opc` account. After this deletion, logins to the Oracle Key Vault instance using SSH will be disabled.

      Only during upgrades, or when directed by Oracle Support, you can temporarily enable SSH from the Oracle Key Vault management console. You can then use SSH to log into the Oracle Key Vault server as the `support` user using the same SSH public key as the `opc` user.
2. Perform the following post-installation tasks:
   - Create the Oracle Key Vault administrator accounts, the recovery passphrase, and the `root` and `support` user passwords.
   - Enter the NTP and DNS addresses, using one of the following choices:
     – The NTP server address in Oracle Cloud Infrastructure, which is `169.254.169.254`, and then leave the remaining fields empty.
     – In all three fields, enter any external NTP servers. For example:

       ```
       0.north-america.pool.ntp.org

       1.north-america.pool.ntp.org

       2.north-america.pool.ntp.org
       ```

Related Topics

- Performing Post-Installation Tasks
  After you install Oracle Key Vault, you must complete a set of post-installation tasks.

11.4 General Management of an Oracle Key Vault Compute Instance

You can perform many of the Oracle Key Vault compute instance general management tasks in the Oracle Key Vault management console.

- Starting, Restarting, or Stopping an Oracle Key Vault Compute Instance
  Depending on the action you need, you can use the Oracle Key Vault management console or the OCI console.
- System Settings in an Oracle Key Vault Compute Instance
  Most system settings in an Oracle Key Vault compute instance are the same as an on-premises deployment, with a few exceptions.
- Backup and Restore Operations for Oracle Key Vault Compute Instances
  You can back up and restore Oracle Key Vault data between OCI environments and on-premises environments.
- Terminating an Oracle Key Vault Compute Instance
  You terminate an Oracle Key Vault compute instance from the OCI console.

11.4.1 Starting, Restarting, or Stopping an Oracle Key Vault Compute Instance

Depending on the action you need, you can use the Oracle Key Vault management console or the OCI console.

You can use the Oracle Key Vault management console or OCI console to restart and stop an Oracle Key Vault compute instance, but to start an already stopped instance, you must use the OCI console.

Select one of the following methods to restart or stop an Oracle Key Vault compute instance:

- From the Oracle Key Vault management console, you can restart or stop the Oracle Key Vault compute instance:
  1. Log into the Oracle Key Vault management console as a user with the System Administrator role.
  2. Select **System**, then **System Settings** from the left sidebar.
  3. In the Settings page, do one of the following:
     - To restart, click **Reboot**.
     - To stop, click **Power Off**.
- From the OCI console, you can start, restart, or stop the Oracle Key Vault compute instance:
  1. Open the navigation menu. Under Core Infrastructure, go to Compute and click **Instances**.
  2. Select the Oracle Key Vault compute instance that you want to stop or start.
  3. Click one of the following actions:
     - To start a stopped instance, click **Start**.
     - To gracefully shut down the instance by sending a shutdown command to the operating system, click **Stop**.
       If the Oracle Key Vault compute instance takes a long time to shut down, it could be improperly stopped, resulting in data corruption. To avoid this, shut down the instance using the commands available in the operating system before you stop the instance using the console.

– To gracefully reboot the Oracle Key Vault compute instance by sending a shutdown command to the operating system, and then power the instance back on, click **Reboot**.

11.4.2 System Settings in an Oracle Key Vault Compute Instance

Most system settings in an Oracle Key Vault compute instance are the same as an on-premises deployment, with a few exceptions.

Settings for system features such as auditing, email, RESTful services, integration with Oracle Audit Vault and Database Firewall are the same in both on-premises and OCI deployments.

- You can configure an Oracle Key Vault host name in either the OCI console or in the Oracle Key Vault management console. However, remember that if you set the IP address of the host in the OCI console, later on, you cannot change it in either the OCI console or the Oracle Key Vault management console.
- Oracle Cloud Infrastructure provides NTP and DNS services. For NTP, enter just one IP address into the first of three fields in the NTP section of the Oracle Key Vault management console System Settings page: `169.254.169.254`. For the DNS settings, consult with your network team because there are multiple options depending how DNS is configured in your subnet and tenancy.
- The SSH tunnel settings are used when on-premises Oracle Key Vault clusters provide key management services to Oracle databases that are deployed in OCI. Do not establish an SSH tunnel in OCI-based Oracle Key Vault deployments.

11.4.3 Backup and Restore Operations for Oracle Key Vault Compute Instances

You can back up and restore Oracle Key Vault data between OCI environments and on-premises environments.

You can back up an Oracle Key Vault compute instance that is stored in an on-premises host: this is the same backup that will be restored. Another on-premises Oracle Key Vault server can be a backup location for a server that is being restored into an Oracle Key Vault compute instance.

Requirements are as follows:

- If you are performing a backup or restore operation from Oracle Key Vault compute instances to an OCI compute instance, then persistent network connectivity to the OCI compute instance from Oracle Key Vault compute instance must exist.
- If you want to perform a backup or restore operation between an Oracle Key Vault compute instance and an on-premises host, ensure that the VCN can span the on-premises hosts.

11.4.4 Terminating an Oracle Key Vault Compute Instance

You terminate an Oracle Key Vault compute instance from the OCI console.

When you terminate the compute instance, all data, including keys that protect endpoints, are permanently lost and cannot be recovered except from a backup. Even

backups may not have the most recent keys. Terminating the instances can lead to loss of data for all endpoints. Exercise extreme caution before terminating an instance. Terminate the Oracle Key Vault compute instance only if you are sure that you have a copy of the keys in another, safe location or that you do not need them.

1. Log in to the OCI console.
2. Under Core Infrastructure, go to Compute, and then click **Instances**.
3. Select the name of the Oracle Key Vault compute instance that you want to remove.
4. Click **Terminate**, and then respond to the confirmation prompt.

Terminated instances temporarily remain in the list of instances with the status **Terminated**.

11.5 Migrating Oracle Key Vault Deployments Between On-Premises and OCI

You can migrate an Oracle Key Vault standalone, primary-standby or cluster deployment from an on-premises environment to OCI or back.

- About Performing Migrations with Oracle Key Vault Compute Instance Data
  You can transition an Oracle Key Vault deployment from on-premises to OCI, and from OCI back to on-premises.
- Migrating Oracle Key Vault Deployments into OCI Using Backup and Restore
  A user who has the System Administrator role can transition the Oracle Key Vault deployment from on-premises to OCI using backup and restore.
- Migrating Oracle Key Vault Deployments Out of OCI Using Backup and Restore
  A user who has the System Administrator role can transition the Oracle Key Vault deployment from OCI to on-premises.

11.5.1 About Performing Migrations with Oracle Key Vault Compute Instance Data

You can transition an Oracle Key Vault deployment from on-premises to OCI, and from OCI back to on-premises.

You can quickly set up a production Oracle Key Vault deployment in OCI to address your immediate key management needs and then transition to the on-premises deployment. Alternately, Oracle Key Vault compute instances require little to no overhead of hardware and VM management. To eliminate this overhead, you may want to transition your on-premises Oracle Key Vault deployment to OCI.

You can use the Oracle Key Vault backup and restore features to migrate an Oracle Key Vault cluster from on-premises to OCI, and back. You can transition an on-premises Oracle Key Vault cluster deployment to OCI by adding Oracle Key Vault compute instances to the cluster and removing on-premises Oracle Key Vault nodes from the cluster. The cluster is fully transitioned to OCI when no on-premises Oracle Key Vault node is left in the cluster. Similarly, you can also transition an Oracle Key Vault cluster in OCI to on-premises.

11.5.2 Migrating Oracle Key Vault Deployments into OCI Using Backup and Restore

A user who has the System Administrator role can transition the Oracle Key Vault deployment from on-premises to OCI using backup and restore.

1. Log in to the on-premises Oracle Key Vault server as a user who has the System Administrator role.
2. Configure an OCI compute instance as the backup destination.
3. Back up the on-premises Oracle Key Vault server to an OCI compute instance.
4. Launch an Oracle Key Vault compute instance with same Oracle Key Vault version as the on-premises Oracle Key Vault server.
5. Log in to the Oracle Key Vault compute instance as a user who has the System Administrator role.
6. Restore the backup from the OCI compute instance to the newly installed Oracle Key Vault compute instance.
7. To set up an Oracle Key Vault multi-master cluster, convert the restored Oracle Key Vault compute instance as the first (initial) node of the cluster.
8. Configure additional Oracle Key Vault compute instances and add them to the cluster as needed.

Related Topics

- Creating the First (Initial) Node of a Cluster
  To create a cluster, you must convert an existing standalone Oracle Key Vault server to become the first node in the cluster.
- Adding a Node to the Cluster
  You can create a read-write pair of nodes or a read-only node.
- Backup and Restore Operations
  You may configure automatic backups for continuous, reliable, and protected access to security objects with minimum downtime.

11.5.3 Migrating Oracle Key Vault Deployments Out of OCI Using Backup and Restore

A user who has the System Administrator role can transition the Oracle Key Vault deployment from OCI to on-premises.

1. Log in to the Oracle Key Vault compute instance as a user who has the System Administrator role.
2. Back up the Oracle Key Vault compute instance to an on-premises system.
3. Install a new Oracle Key Vault server on-premises with same Oracle Key Vault version as the Oracle Key Vault compute instance.
4. Log in to the on-premise Oracle Key Vault server as a user who has the System Administrator role.
5. Restore the backup from the on-premises backup destination to the newly installed on-premises Oracle Key Vault server.

6. To set up an Oracle Key Vault multi-master cluster, convert the restored on-premises Oracle Key Vault server as the first (initial) node of the cluster.
7. Configure additional Oracle Key Vault compute instances and add them to the cluster as needed.

Related Topics

- Creating the First (Initial) Node of a Cluster
  To create a cluster, you must convert an existing standalone Oracle Key Vault server to become the first node in the cluster.
- Adding a Node to the Cluster
  You can create a read-write pair of nodes or a read-only node.
- Backup and Restore Operations
  You may configure automatic backups for continuous, reliable, and protected access to security objects with minimum downtime.

12

Oracle Database Instances in Oracle Cloud Infrastructure

Oracle Key Vault deployed on-premises can manage the TDE master encryption keys for Oracle Database instances running in Oracle Cloud Infrastructure (OCI).

- About Managing Oracle Cloud Infrastructure Database Instance Endpoints
  This type of Oracle Key Vault server deployment meets compliance standards for the management of encryption keys.
- Preparing a Database Instance on OCI to be an Oracle Key Vault Endpoint
  Oracle Key Vault supports the use of Oracle database instances on Oracle Cloud Infrastructure (OCI).
- Using an SSH Tunnel Between Oracle Key Vault and Database as a Service
  An on-premises Oracle Key Vault communicates with an Oracle Cloud Database as a Service instance using a secure SSH tunnel.
- Registering and Enrolling a Database as a Service Instance as an Oracle Key Vault Endpoint
  You can use the command line and the Oracle Key Vault management console to complete this task.
- Suspending Database Cloud Service Access to Oracle Key Vault
  You can suspend one or more enrolled Database as a Service endpoints from access to Oracle Key Vault.
- Resuming Database Cloud Service Access to Oracle Key Vault
  You can reinstate the connection between suspended Database Cloud Service endpoints and Oracle Key Vault.
- Resuming a Database Endpoint Configured with a Password-Based Keystore
  Depending on the configuration, a Database as a Service endpoint can resume either automatically or must be manually resumed.

12.1 About Managing Oracle Cloud Infrastructure Database Instance Endpoints

This type of Oracle Key Vault server deployment meets compliance standards for the management of encryption keys.

The Oracle Database instances running in Oracle Cloud Infrastructure (OCI) can be deployed on VMshape, bare metal, or Exadata. This type of deployment 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.

Related Topics

- Managing a Reverse SSH Tunnel in a Multi-Master Cluster
  You can reverse an SSH tunnel in a multi-master cluster from more than one node to the cloud-based endpoint for redundancy.

12.2 Preparing a Database Instance on OCI to be an Oracle Key Vault Endpoint

Oracle Key Vault supports the use of Oracle database instances on Oracle Cloud Infrastructure (OCI).

- About Preparing a Database Instance on OCI to be an Oracle Key Vault Endpoint
  To prepare an Oracle database instance on OCI to be an Oracle Key Vault endpoint, you must first configure the instance, and then create a low-privileged user.
- Configuring a Database Cloud Service Instance
  An Database as a Service (DBaaS) instance must have the correct network configuration.
- Creating a Low Privileged Operating System User on Database as a Service
  The low privileged user account, `okv`, will be responsible for configuring an SSH tunnel and communicating with the DBaaS instances.

12.2.1 About Preparing a Database Instance on OCI to be an Oracle Key Vault Endpoint

To prepare an Oracle database instance on OCI to be an Oracle Key Vault endpoint, you must first configure the instance, and then create a low-privileged user.

Oracle databases on Oracle Cloud Infrastructure (OCI) provide 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.

12.2.2 Configuring a Database Cloud Service Instance

An Database as a Service (DBaaS) instance must have the correct network configuration.

You can find instructions for configuring a Database as a Service (DBaaS) instance in the Oracle Database Cloud Service (Database as a Service) documentation.

After you have configured the DBaaS instance, it should have the following default values:

- A public IP address
- Two users: `oracle` and `opc` (Oracle Public Cloud)
- SSH access to the `oracle` and `opc` users

12.2.3 Creating a Low Privileged Operating System User on Database as a Service

The low privileged user account, `okv`, will be responsible for configuring an SSH tunnel and communicating with the DBaaS instances.

By default, Database as a Service instances are provisioned with the `oracle` and `opc` users. These users have more privileges than necessary to create the SSH tunnel, so Oracle recommends that you create another low privileged operating system user named `okv` on the Database as a Service instance. Oracle Key Vault will use user `okv` to configure an SSH tunnel and communicate with the Database as a Service instances.

1. Log in to the Oracle Cloud Infrastructure (OCI) instance using public key authentication (default for Oracle OCI) as user `opc`.

   ```
   $ ssh -i private_key_file opc@node_ip_address
   ```

   In this specification:

   - *`private_key_file`* is the path to your private key file (`~/.ssh/id_rsa`). This key is the counterpart to the public key that you uploaded when you provisioned the Oracle Cloud Infrastructure instance.
   - *`node_ip_address`* is the public IP address of the Database as a Service compute node in *`x.x.x.x`* format.

   If this is the first time you are connecting to the compute node, the SSH utility prompts you to confirm the public key.

2. In response to the prompt asking you to confirm the public key, 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 `sshd_config` configuration file in the `/etc/ssh/` directory.

   ```
   $ sudo vi /etc/ssh/sshd_config
   ```

5. Add the following entry to the end of the file:

   ```
   AllowUsers oracle opc okv
   ```

6. Restart the SSH daemon:

   ```
   $ sudo /sbin/service sshd restart
   ```

7. Grant the Oracle Key Vault user `okv` permission to execute `/sbin/fuser` by following these steps:

   a. Change the file permission of the `/etc/sudoers` file.

      ```
      sudo chmod 740 /etc/sudoers
      ```

   b. Edit the `/etc/sudoers` file.

      ```
      sudo vi /etc/sudoers
      ```

   c. Add the following entry:

      ```
      okv     ALL=(root) NOPASSWD:/sbin/fuser
      ```

d. Save the `/etc/sudoers` file. Change the file permission of the `/etc/sudoers` file.

```
sudo chmod 440 /etc/sudoers
```

e. The `/etc/sudoers` would look similar to the following:

```
## Allow root to run any commands anywhere
root    ALL=(ALL)       ALL
okv     ALL=(root) NOPASSWD:/sbin/fuser
```

8. Become the `okv` user.

```
$ su okv
```

9. Create the `authorized_keys` file and then set appropriate permissions for this file.

```
$ cd $HOME
$ mkdir ./.ssh
$ chmod 700 ./.ssh
$ touch ./.ssh/authorized_keys
$ chmod 640 ./.ssh/authorized_keys
```

10. Log in to the Oracle Key Vault instance as the `support` user, and switch from `root`, and then switch to `oracle`.

11. Execute the following command to upload the Oracle Key Vault public key into the `authorized_keys` file in the Oracle Cloud Infrastructure that you just created.

```
ssh-copy-id ./.ssh/id_rsa.pub okv@node_ip_address
```

12. Confirm that the `oracle` user in Oracle Key Vault can log in to the OCI instance without providing a password:

```
$ ssh okv@node_ip_address
```

12.3 Using an SSH Tunnel Between Oracle Key Vault and Database as a Service

An on-premises Oracle Key Vault communicates with an Oracle Cloud Database as a Service instance using a secure SSH tunnel.

- Creating an SSH Tunnel Between Oracle Key Vault and a DBaaS Instance
  You can create a connection between Oracle Key Vault and a Database as a Service (DBaaS) instance by configuring an SSH tunnel.
- Managing a Reverse SSH Tunnel in a Multi-Master Cluster
  You can reverse an SSH tunnel in a multi-master cluster from more than one node to the cloud-based endpoint for redundancy.
- Managing a Reverse SSH Tunnel in a Primary-Standby Configuration
  A reverse SSH tunnel in a primary-standby configuration is similar to a reverse SSH tunnel on a standalone Oracle Key Vault server.
- Viewing SSH Tunnel Configuration Details
  The Oracle Key Vault management console provides information about SSH tunnels that have been configured for Oracle Key Vault.
- Disabling an SSH Tunnel Connection
  You can use the Oracle Key Vault management console to disable the Oracle Key Vault and Database as a Service instance connection.

- How the Connection Works if the SSH Tunnel Is Not Active
  The SSH tunnel is kept alive even if there is no activity between Oracle Key Vault and the Database as a Service instance.
- Deleting an SSH Tunnel Configuration
  You can use the Oracle Key Vault management console to delete the connection between Key Vault and a Database as a Service instance.

12.3.1 Creating an SSH Tunnel Between Oracle Key Vault and a DBaaS Instance

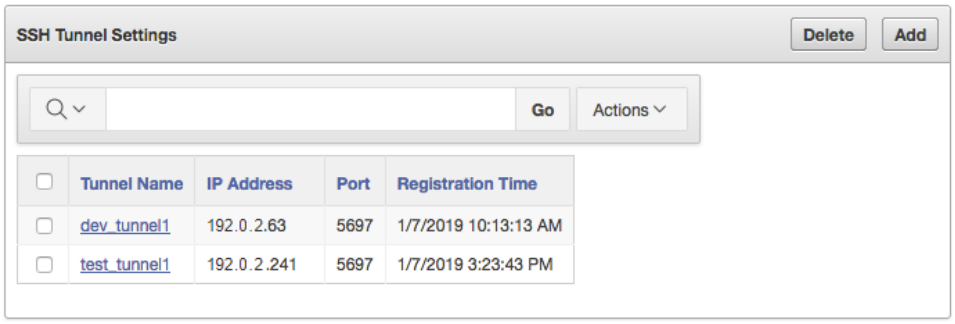
You can create a connection between Oracle Key Vault and a Database as a Service (DBaaS) instance by configuring an SSH tunnel.

You can configure the SSH tunnel only after you set up the Database as a Service instance. You must have the Database as a Service instance's public IP address and the name of the operating system user that you want to use to establish the tunnel.

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

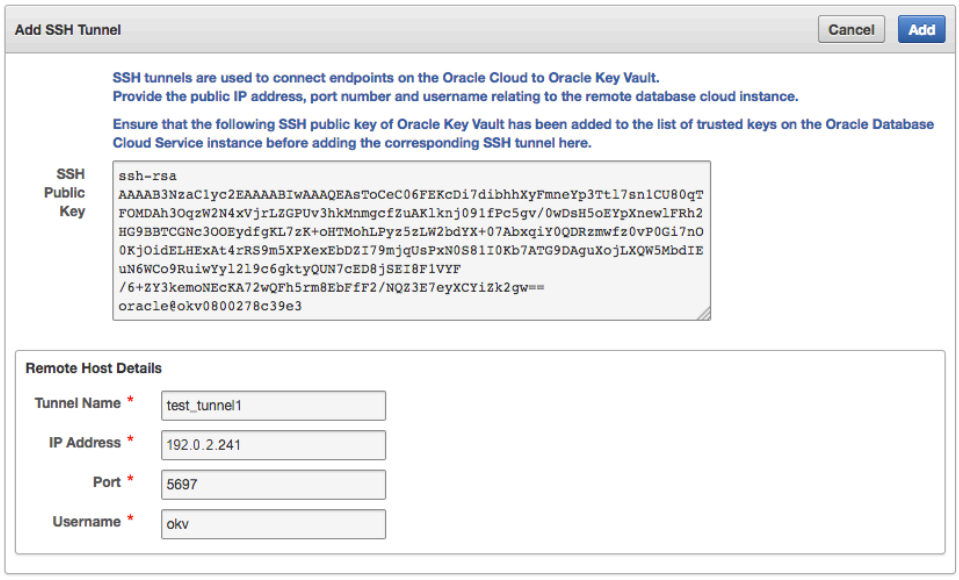
   The Status page appears.
3. Select **SSH Tunnel Settings** from the left side bar.

   The SSH Tunnel Settings page appears.



4. Click **Add**.

   The Add SSH Tunnel page appears.



5. Copy the text in **SSH Public Key** field and save it.

   Remember that this is the public key that was copied into the OCI instance for user `okv` and was uploaded when you created a low privileged operating system user the Database as a Service instance. 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`.

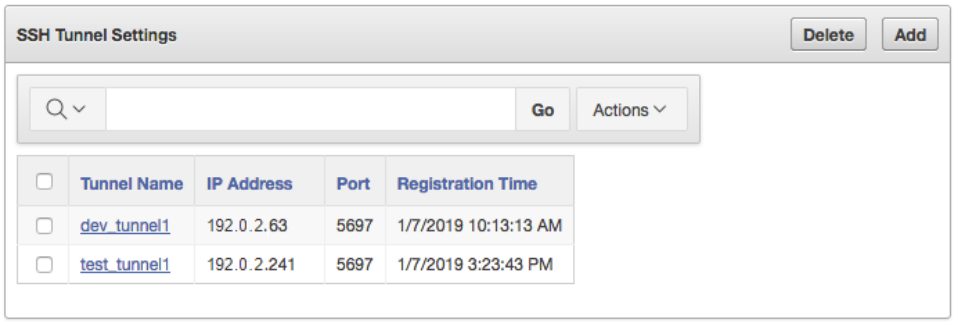
6. In the Remote Host Details page, enter information in the following fields:

   - **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**: Enter the 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 user name.

   You can complete these fields only after you set up the Database as a Service instance and obtained the public IP address and user name.

7. Click **Add**.

   The SSH Tunnel Settings page appears. It displays the SSH tunnel that you just created and any pre-existing SSH tunnels.

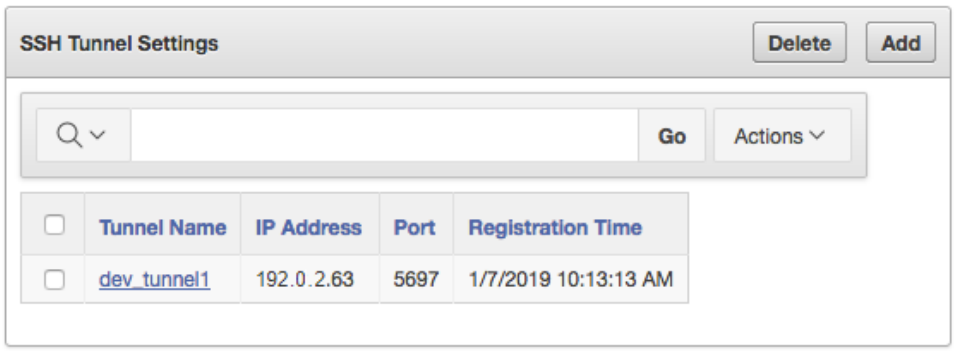


It lists the tunnels created with the name, IP address, port, and registration time of each.

8. Click a tunnel name to see the **SSH Tunnel Details** page.

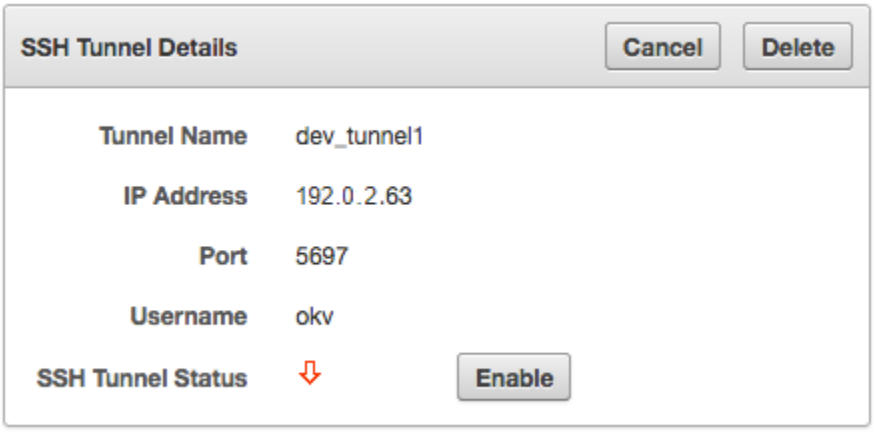
9. To delete a tunnel, check the box by the tunnel that you want to delete and then click **Delete.**

   You can delete more than one tunnel by selecting multiple boxes.



10. Click **Disable** to disable the tunnel.

    When you disable the tunnel, the endpoints that are associated with this tunnel will no longer be able to communicate with Oracle Key Vault.



11. In the confirmation dialog box, click **Yes**.

    The **Disable** button is replaced by an **Enable** button.

Related Topics

- Creating a Low Privileged Operating System User on Database as a Service
  The low privileged user account, `okv`, will be responsible for configuring an SSH tunnel and communicating with the DBaaS instances.

12.3.2 Managing a Reverse SSH Tunnel in a Multi-Master Cluster

You can reverse an SSH tunnel in a multi-master cluster from more than one node to the cloud-based endpoint for redundancy.

Oracle recommends that you configure three tunnels. Ideally, the cloud-based reverse SSH tunnels should be from different read-write pairs. Multiple SSH tunnels to the same endpoint are distinguished by the port number used. Oracle Key Vault suggests unique port numbers based on node ID. If you want to specify different port numbers, make port numbers for SSH tunnels from different nodes to the same endpoint unique.

In a multi-master cluster, multiple SSH tunnels are created from multiple nodes to the same endpoint. However, when you register and enroll endpoints, you will only see the tunnel from that node.

Be aware of the following:

- You should register and enroll the endpoint where there is a SSH tunnel created to that endpoint.
- You only see the tunnel from that node to endpoint in the following places:
  - During the registration, the option to select the SSH tunnel.
  - After registration, when you view endpoint details, only that tunnel is displayed.
  - When you submit the enrollment token and download the endpoint software, only that tunnel is displayed. However, the endpoint software downloaded has information about all tunnels to the endpoint. This means that the endpoint is able to use all the tunnels that were created before the endpoint is created.

All nodes which have an SSH tunnel created display their tunnel to the endpoint on the Endpoint Details page. They also list all tunnels that were created from that node on the SSH Tunnels page in the Oracle Key Vault management console.

12.3.3 Managing a Reverse SSH Tunnel in a Primary-Standby Configuration

A reverse SSH tunnel in a primary-standby configuration is similar to a reverse SSH tunnel on a standalone Oracle Key Vault server.

The SSH key of the primary and standby servers are the same after pairing. Tunnels created on an Oracle Key Vault server before primary-standby pairing as well as tunnels created on the primary after the primary-standby pairing are valid after primary-standby operations such as switchover, and failover, although the tunnels may be unavailable during the execution of these operations.

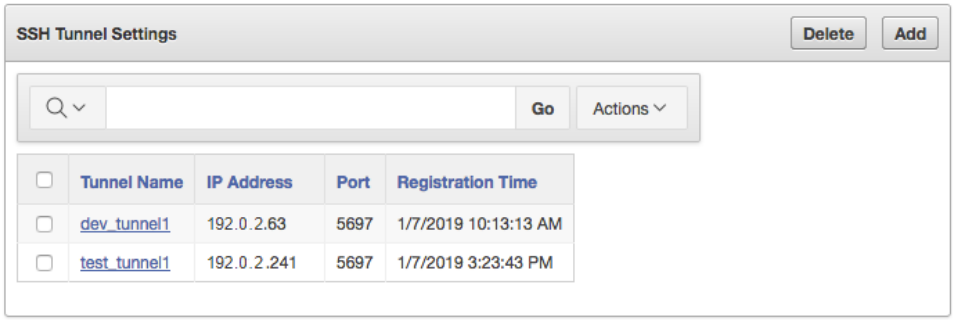
12.3.4 Viewing SSH Tunnel Configuration Details

The Oracle Key Vault management console provides information about SSH tunnels that have been configured for Oracle Key Vault.

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

   The Status page appears.
3. Select **SSH Tunnel Settings** from the left side bar.

   The SSH Tunnel Settings page appears.



4. Click a tunnel name to see the **SSH Tunnel Details** page.

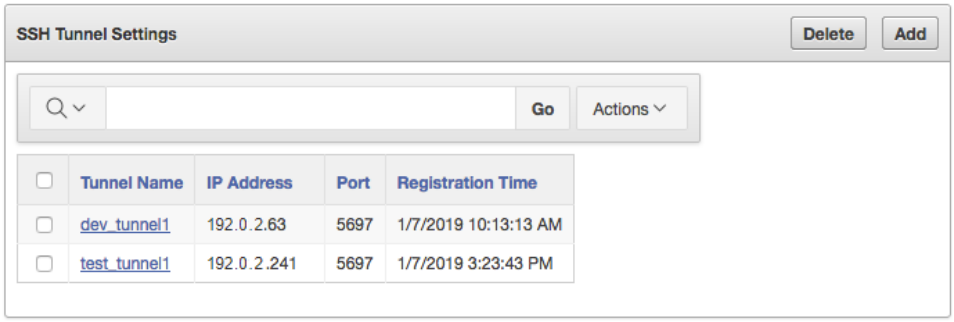
12.3.5 Disabling an SSH Tunnel Connection

You can use the Oracle Key Vault management console to disable the Oracle Key Vault and Database as a Service instance connection.

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

   The Status page appears.
3. Select **SSH Tunnel Settings** from the left side bar.

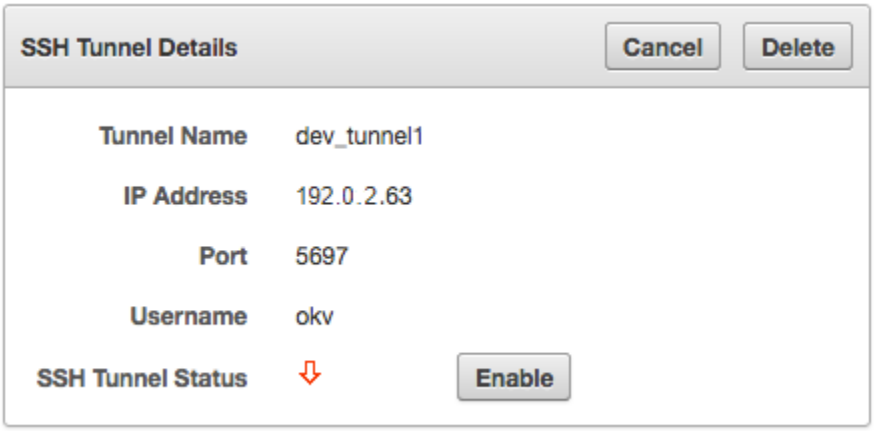
The SSH Tunnel Settings page appears.



4. Click a tunnel name to see the **SSH Tunnel Details** page.

5. Click **Disable** to disable the tunnel.

   When you disable the tunnel, the endpoints that are associated with this tunnel will no longer be able to communicate with Oracle Key Vault.



6. In the confirmation dialog box, click **Yes**.

   The **Disable** button is replaced by an **Enable** button.

12.3.6 How the Connection Works if the SSH Tunnel Is Not Active

The SSH tunnel is kept alive even if there is no activity between Oracle Key Vault and the Database as a Service instance.

If the tunnel stops, then it is automatically restarted. An alert will be sent 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 Oracle Key Vault.

Related Topics

- Configuring Email Settings
  You can configure the Simple Mail Transfer Protocol (SMTP) server properties to receive email notifications from Oracle Key Vault.

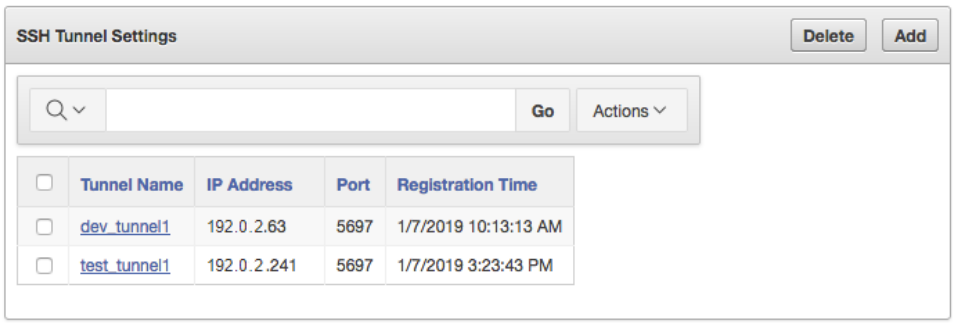
12.3.7 Deleting an SSH Tunnel Configuration

You can use the Oracle Key Vault management console to delete the connection between Key Vault and a Database as a Service instance.

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

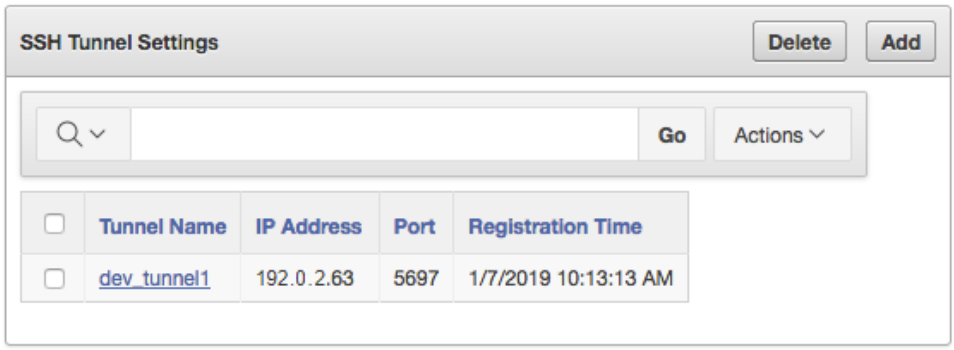
   The Status page appears.
3. Select **SSH Tunnel Settings** from the left side bar.

   The SSH Tunnel Settings page appears.



4. To delete a tunnel, check the box by the tunnel that you want to delete and then click **Delete.**

   You can delete more than one tunnel by selecting multiple boxes.



12.4 Registering and Enrolling a Database as a Service Instance as an Oracle Key Vault Endpoint

You can use the command line and the Oracle Key Vault management console to complete this task.

- About Registering and Enrolling a Database as a Service Instance as an Oracle Key Vault Endpoint
  You must enroll the Oracle Database as a Service instance before it can communicate with an Oracle Key Vault server.
- Step 1: Register the Endpoint in the Oracle Key Vault Management Console
  The endpoint registration process downloads an `okvclient.jar` file, which contains the Oracle Key Vault software that the endpoint needs, to the local system.
- Step 2: Prepare the Endpoint Environment
  The endpoint must have a compatible version of the Java Development Toolkit (JDK) and the Oracle Database environment variables must be set.
- Step 3: Install the Oracle Key Vault Software onto the Endpoint
  To install the Oracle Key Vault software installation, you run the `okvclient.jar` file on the endpoint.
- Step 4: Perform Post-Installation Tasks
  Post-installation tasks are important for a fully functioning Oracle Key Vault installation.

12.4.1 About Registering and Enrolling a Database as a Service Instance as an Oracle Key Vault Endpoint

You must enroll the Oracle Database as a Service instance before it can communicate with an 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.

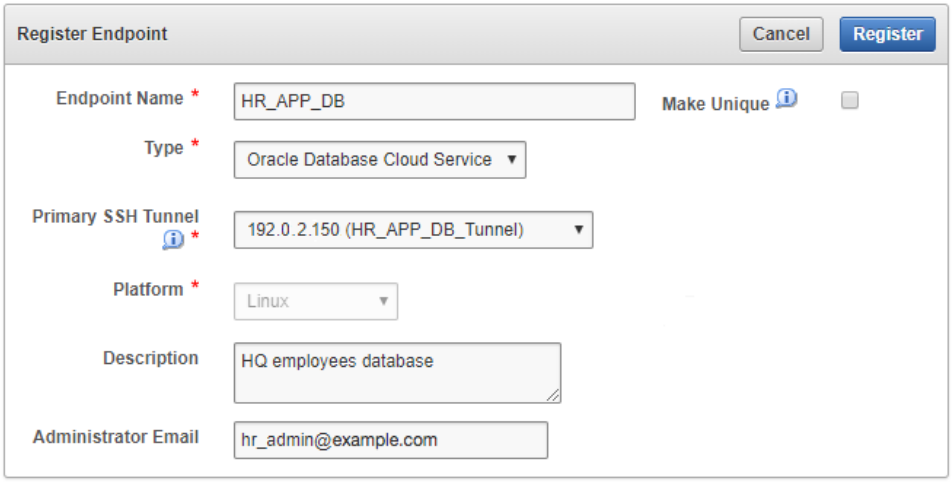
12.4.2 Step 1: Register the Endpoint in the Oracle Key Vault Management Console

The endpoint registration process downloads an `okvclient.jar` file, which contains the Oracle Key Vault software that the endpoint needs, to the local system.

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.
3. Click **Add**.

   The Register Endpoint page appears.



4. Enter the following endpoint details:
   - **Endpoint Name**: Enter a unique name for the endpoint.
   - **Make Unique**: If you are using a multi-master cluster, then choose whether to select the **Make Unique** checkbox. **Make Unique** helps to control naming conflicts with user names across the multi-master cluster environment.
     - If you select **Make Unique**, then the enspoing will be active immediately for Oracle Key Vault operations.
     - If you do not select **Make Unique**, then the user account will be created in the `PENDING` state. Oracle Key Vault will then begin a name resolution operation and may rename the endpoint name to a name that is unique across the clusters. If there is a naming collision, then you must recreate

the user with a unique name. An endpoint in the `PENDING` state cannot be used in any Oracle Key Vault operations.

- **Type**: Select **Oracle Database Cloud Service**.
- **Platform**: Linux is automatically selected.
- **Description**: Enter a meaningful description to identify the endpoint.
- **Administrator Email** Optionally, enter the email address of an administrator who should receive endpoint-related alerts.

5. Click **Register**.

   After a short delay the Endpoints page displays the new endpoint in the **Registered** state with an **Enrollment Token**.

6. Click **Endpoint Name**. The **Endpoint Details** page appears.

   Associate a default wallet with the registered endpoint now before enrolling the endpoint.

7. Copy the **Enrollment Token**.

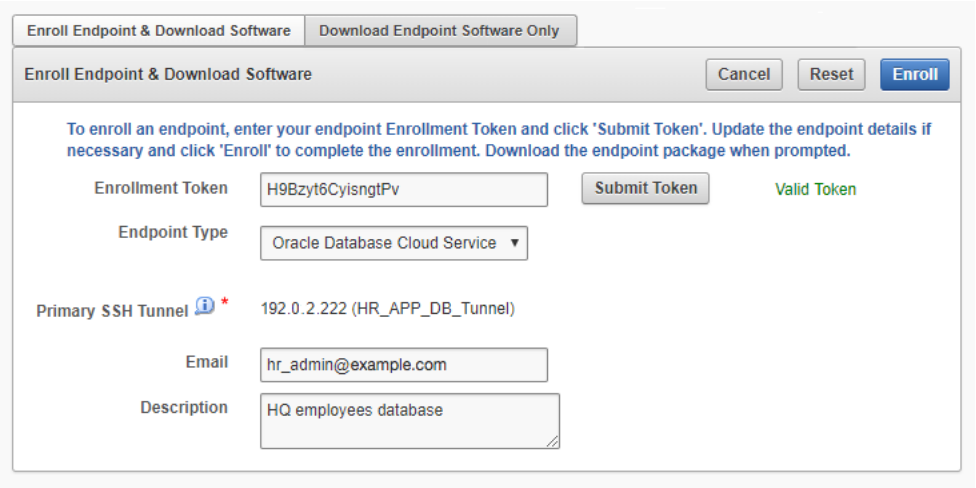
   You will need it to download the endpoint software and then enroll the endpoint (next step).

8. Log out of Oracle Key Vault and open a new session.

   The login page appears. ***Do not log in***.

9. Click **Endpoint Enrollment and Software Download** immediately below **Login**.

   The Enroll Endpoint & Download Software page appears.



   The fields are populated with the values that were chosen by the Oracle 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.

10. In the **Enrollment Token** field, enter the name of the endpoint token and then click **Submit Token** to validate the token.

11. Click **Enroll** to download the `okvclient.jar` file to your local system.

12. Move the `okvclient.jar` file 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_computer
    oracle@node_ip_address:path_to_okvclient.jar_on_cloud_db_instance
    ```

    In this specification:

    - *`path_to_okvclient.jar_on_local_computer`* refers to the location of `okvclient.jar` on an on-premises local computer.
    - *`path_to_okvclient.jar_on_cloud_db_instance`* refers to the location of `okvclient.jar` on the oracle cloud database as a service instance.

Related Topics

- Types of Endpoint Enrollment
  The first step in enrolling an endpoint is to add the endpoint to Oracle Key Vault.
- Setting the Default Wallet for an Endpoint
  Setting a default wallet for an endpoint automatically uploads the endpoint's security objects to the wallet if another wallet is not explicitly specified.

12.4.3 Step 2: Prepare the Endpoint Environment

The endpoint must have a compatible version of the Java Development Toolkit (JDK) and the Oracle Database environment variables must be set.

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, then 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.

12.4.4 Step 3: Install the Oracle Key Vault Software onto the Endpoint

To install the Oracle Key Vault software installation, you run the `okvclient.jar` file on the endpoint.

1. Ensure that you are logged in to the endpoint server as the endpoint administrator.
2. Navigate to the directory in which you saved the `okvclient.jar` file.
3. Confirm that the target directory exists, and that it is empty.
4. Run the `java` command to install the `okvclient.jar` file.

   ```
   java -jar okvclient.jar -d /home/oracle/okvutil -v
   ```

In this specification:

- `-d` specifies the directory location for the endpoint software and configuration files, in this case `/home/oracle/okvutil`.
- `-v` writes the installation logs to the `/home/oracle/okvutil/log/okvutil.deploy.log` file at the server endpoint.

`-o` is an optional argument that enables you to overwrite the symbolic link reference to `okvclient.ora` when `okvclient.jar` is deployed in a directory other than the original directory. This argument is used only when you re-enroll an endpoint.

If you are installing the `okvclient.jar` file on a Windows endpoint system that has Oracle Database release 11.2.0.4 **only**, then include the `-db112` option. (This option is not necessary for any other combination of endpoint platform or Oracle Database version.) For example:

```
java -jar okvclient.jar -d /home/oracle/okvutil -v -db112
```

5. When you are prompted for a password, then perform either of the following two steps.

   The optional password goes into two places: `okvutil` and in `ADMINISTER KEY MANAGEMENT`. With `okvutil`, only users who know that password can upload or download content to and from Oracle Key Vault. With `ADMINISTER KEY MANAGEMENT`, it becomes the password that you must use in the `IDENTIFIED BY` *`password`* clause. If you choose not to give a password, then `okvutil upload` and `download` commands will not prompt for a password, and the password for `ADMINISTER KEY MANAGEMENT` becomes `NULL`.`NULL` is used for an auto-login wallet.

   The choices for handling the password are as follows:

   - If you want to create a password-protected wallet, at minimum enter a password between 8 and 30 characters and then press **Enter**. For better security, Oracle recommends that you include uppercase letters, lowercase characters, special characters, and numbers in the password. The following special characters are allowed: (.), comma (,), underscore (_), plus sign (+), colon (:), space.

     ```
     Enter new Key Vault endpoint password (<enter> for auto-login):
     Key_Vault_endpoint_password
     Confirm new endpoint password: Key_Vault_endpoint_password
     ```

     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.

   - Alternatively, enter no password and then press **Enter**.
     No password will be required when the endpoint connects to Oracle Key Vault with `okvutil`. With the `ADMINISTER KEY MANAGEMENT` statement, the password becomes `NULL`.

   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.

12.4.5 Step 4: Perform Post-Installation Tasks

Post-installation tasks are important for a fully functioning Oracle Key Vault installation.

After you complete the installation, you can optionally configure a TDE connection for the endpoint, check the installation contents, and then delete the `okvclient.jar file`.

1. Optionally, configure a TDE connection for the endpoint.

   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.

   - **On Oracle Linux x86-64, Solaris, AIX, and HP-UX (IA) installations:** Log in as the root and then execute either of the following commands:

     ```
     $ sudo bin/root.sh
     ```

     Or:

     ```
     $ su -
     # bin/root.sh
     ```

     This command creates the directory tree `/opt/oracle/extapi/64/hsm/oracle/1.0.0`, changes ownership and permissions, then copies the PKCS#11 library into this directory.

   - **On Windows installations:** Run the following command:

     ```
     bin\root.bat
     ```

     This command copies the `liborapkcs.dll` file to the `C:\oracle\extapi\64\hsm\oracle\1.0.0` directory.

2. Use a command such as `namei` or `ls -l` to confirm that a softlink was created in `$ORACLE_BASE/okv/$ORACLE_SID/okvclient.ora` to point to the real file in the `/conf` subdirectory of the installation target directory.

   If the `ORACLE_BASE` environment variable has not been set, then the softlink was created in `$ORACLE_HOME/okv/$ORACLE_SID`.

3. 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.

   ```
   $ ./okvutil list
   ```

   If the endpoint is able to connect to Key Vault, then the `No objects found` message appears. If a `Server connect failed` message appears, then you must troubleshoot the installation for possible issues. Check that environment variables are correctly set. To get help on the endpoint software, execute the following command:

```
java -jar okvclient.jar -h
```

Output similar to the following appears:

```
Production on Fri Apr 12 15:03:01 PDT 2019
Copyright (c) 1996, 2019 Oracle. All Rights Reserved.
Usage:
  java -jar okvclient.jar [-h | -help] [[-v | -verbose] [-d <destination
directory>] [-o]]

Options:
  -h or -help : Display command help.
  -v or -verbose : Turn on the verbose mode. Logs will be written to files
under
                   <destination directory>/log/ directory.
  -d <destination directory> : Specify the software installation directory.
  -o : Overwrite the current symbolic link to okvclient.ora.
```

4. After you complete the installation, securely delete the `okvclient.jar` endpoint software file.

12.5 Suspending Database Cloud Service Access to Oracle Key Vault

You can suspend one or more enrolled Database as a Service endpoints from access to Oracle Key Vault.

- About Suspending Database Cloud Service Access to Oracle Key Vault
  When the DBaaS service is suspended, the Oracle Key Vault Server rejects all requests from the suspended endpoints.
- Suspending Access for a Database Cloud Service to Oracle Key Vault
  After you suspend the Database as a Service access to Oracle Key Vault, you can resume the access when needed.

12.5.1 About Suspending Database Cloud Service Access to Oracle Key Vault

When the DBaaS service is suspended, the Oracle Key Vault Server rejects all requests from the suspended endpoints.

When you use an on-premises Oracle Key Vault to manage the online master keys for Database as a Service endpoints, the master encryption keys are never stored persistently in Oracle Cloud. This way, the on-premises Oracle Key Vault administrator can control access to the encrypted data in the cloud.

An 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. Because the endpoint cannot request keys from the Oracle Key Vault server, its ability to access encrypted data is lost after the 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.

In a multi-master cluster, when a node is being enabled or disabled, the information may not yet have reached all nodes in the cluster. If an endpoint attempts to contact a node whose information has not yet propagated throughout the cluster, an error may be returned.

> **Caution:**
>
> The suspend operation is a disruptive operation as it results in operational discontinuity. Therefore, you should use it with care. Usually, you should suspend the database only if there is a strong indication of abnormal activity in the Database as a Service instance.

You can only suspend enrolled endpoints. You cannot suspend endpoints that are in the **Registered** state. If you try to suspend endpoints that are already suspended, no operation will be performed. The endpoints will continue to be in suspended state.

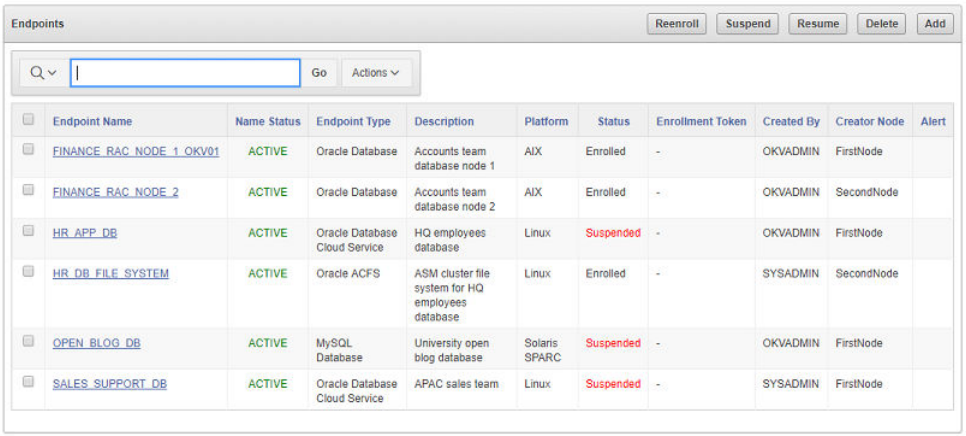
12.5.2 Suspending Access for a Database Cloud Service to Oracle Key Vault

After you suspend the Database as a Service access to Oracle Key Vault, you can resume the access when needed.

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

   The Endpoints page appears.
3. Check the boxes by the endpoints that you want to suspend.
4. Click **Suspend**.
5. In the confirmation dialog box, click **Yes**.
6. Click **Endpoints** to see the suspended endpoints.

   The status of suspended endpoints are highlighted in red.



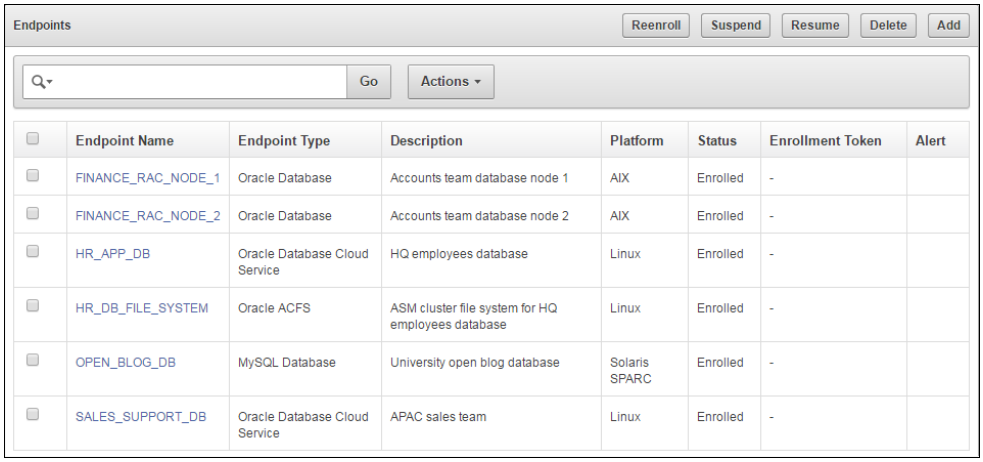
12.6 Resuming Database Cloud Service Access to Oracle Key Vault

You can reinstate the connection between suspended Database Cloud Service endpoints and Oracle Key Vault.

When you resume these endpoints, their status will change to **Enrolled**. Resuming enrolled endpoints does not change their enrolled status.

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**.
4. In the confirmation dialog box, click **Yes**.
5. Click **Endpoints** to see the re-enrolled endpoints. Their status is **Enrolled**.



12.7 Resuming a Database Endpoint Configured with a Password-Based Keystore

Depending on the configuration, a Database as a Service endpoint can resume either automatically or must be manually resumed.

A Database as a Service endpoint that is configured with auto-login keystore support will begin operations as soon as one of the nodes configured with reverse SSH access restores connectivity to the DBCS endpoint. 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 Oracle Key Vault suspended the endpoint. You should open the password-based keystore on the Database as a Service instance to resume operations.

13

Oracle Key Vault Administration and Key Management with RESTful Services

The Oracle Key Vault RESTful Services utility automates Oracle Key Vault administration tasks for a large distributed deployment.

- About RESTful Services
  The Oracle Key Vault tasks that you can automate using RESTful services include endpoint enrollment, virtual wallet management, and key management.
- Required Privileges for Using RESTful Services
  The required RESTful services privileges are consistent with the privileges required to perform the same task in the Oracle Key Vault management console.
- Enabling RESTful Services
  After checking the endpoint requirements, and enabling network services, you can enable RESTful services and then download the RESTful software utility.
- Managing the RESTful Services Configuration File
  You can use a configuration file to execute RESTful service commands either individually or in a group.
- Disabling RESTful Services
  You should enable RESTful Services for short periods during endpoint registration and enrollment only.
- Oracle Key Vault Administrative REST Client Tool Commands
  The RESTful services administrative commands are designed for administrators who manage endpoints and endpoint groups.
- Oracle Key Vault Key Management REST Client Tool Commands
  The RESTful services key management commands are designed for administrators who are responsible for managing keys that are uploaded to Oracle Key Vault.

13.1 About RESTful Services

The Oracle Key Vault tasks that you can automate using RESTful services include endpoint enrollment, virtual wallet management, and key management.

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. In an Oracle RAC environment, Oracle Key Vault virtual wallets must be shared between the Oracle RAC instances. In other words, each Oracle RAC-enabled database will have one virtual wallet in Oracle Key Vault. All endpoints that belong to that database will have that virtual wallet as their default wallet.

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.

13.2 Required Privileges for Using RESTful Services

The required RESTful services privileges are consistent with the privileges required to perform the same task in the Oracle Key Vault management console.

For example, if you want to add and manage endpoints, then you must have the System Administrator role. If you want to work with wallets and keys, then you must have the Key Administrator role. The System Administrator and Key Administrator privileges are required for connecting Oracle databases to Oracle Key Vault. For MySQL databases, you only need the Key Administrative privilege because MySQL does not use wallets. You do not need to have endpoint administrator privileges to use RESTful services.

The RESTful commands require either the System Administrator role or the Key Administrator role. To simplify the use of RESTful services, you can create a user who has both of these roles. Typically, this user is an administrator who must self-register their databases with Oracle Key Vault by using scripts that will need to perform the actions that need both of these privileges.

13.3 Enabling RESTful Services

After checking the endpoint requirements, and enabling network services, you can enable RESTful services and then download the RESTful software utility.

- Step 1: Check the Endpoint System Requirements
  Before you can provision endpoints with the REST API, you must have the tools to transfer data securely across the network.
- Step 2: Enable Network Services
  You must configure web access for RESTful clients by their IP addresses to access the Oracle Key Vault server.
- Step 3: Enable RESTful Services
  After you have enabled the network services, you can enable the RESTful services.
- Step 4: Download the RESTful Software Utility
  The RESTful software utility is in the `okvrestservices.jar` file.

13.3.1 Step 1: Check the Endpoint System Requirements

Before you can provision endpoints with the REST API, you must have the tools to transfer data securely across the network.

1. Log in to the endpoint as an endpoint administrator.
2. Ensure that you have the following tools:
   - cURL version that supports Transport Layer Security (TLS) 1.2 or later
   - OpenSSL 1.0.1p or later
   - Java 1.7.0.21 or later (the Java Runtime Environment (JRE) is provided with Oracle Database release 12.2, so you do not need to install the JRE if you already have Oracle Database release 12.2 or later). If you plan to deploy RESTful services on a database server with Oracle Database release 12.2.0.1 or later, then you can use the embedded Java Runtime Environment (JRE) in `$ORACLE_HOME/jdk/jre`.
     For database installations from Oracle release 12.2.0.1 and later, set `JAVA_HOME` to `$ORACLE_HOME/jdk/jre`, and add `JAVA_HOME/bin` to the `PATH`.
     For earlier database releases, download and install a JRE version 1.7.0.21 or later, and then set `JAVA_HOME` and `PATH` appropriately. OpenJDK is not supported.
   - For the provision command, a soft link from `/usr/bin/java` either to the extra JRE installation (for older databases before Oracle Database release 12.2.0.1) or `$ORACLE_HOME/jdk/jre/bin/java` for Oracle Database release 12.2.0.1 and later. You can confirm by executing `namei /usr/bin/java` at the command line.

13.3.2 Step 2: 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.

1. Log in to the Oracle Key Vault management console as a user the System Administrator role.
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:
   - **All** to allow all IP addresses.
   - **IP address(es)** to designate a set of IP addresses. After you select this option, enter the IP addresses in the next field, separating each IP address by a space.
4. Click **Save**.

13.3.3 Step 3: Enable RESTful Services

After you have enabled the network services, you can enable the RESTful services.

1. Log in to the Oracle 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. Go to the RESTful Services section.
3. Check the box to the right of **Enable**.
4. Click **Save**.

13.3.4 Step 4: Download the RESTful Software Utility

The RESTful software utility is in the `okvrestservices.jar` file.

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

> **Note:**
>
> - If you install a third-party certificate, then 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 Oracle Key Vault appliance with new software or a backup.
> - In a multi-master cluster environment, you must download `okvrestservice.jar` from a read-write node. If you download it from a read-only node, then you cannot connect to Oracle Key Vault.

13.4 Managing the RESTful Services Configuration File

You can use a configuration file to execute RESTful service commands either individually or in a group.

- About Managing the RESTful Services Configuration File
  The RESTful services key management commands are designed for administrators who are responsible for managing keys that are uploaded to Oracle Key Vault.
- Configuration File Creation Guidelines
  You should follow the recommended guidelines to avoid script execution errors.
- Creating the RESTful Services Configuration File
  You must set properties in the configuration file that the RESTful service utility will use to run commands.
- Examples of Configuration Files
  You can create a configuration file that uses an IP address and a host name.
- Executing a Single RESTful Command
  If you only want to run a few commands, then run them individually from the command line using the `-r` or `--service` option.
- Executing Multiple RESTful Administrative Commands Using a Script
  To save time and effort, as well as ensuring accuracy, you can use a script to run a sequence of commands.
- Creating a Script to Automatically Enroll Oracle Databases as Endpoints
  You can create a script that database administrators can run to automatically enroll Oracle Database endpoints in Oracle Key Vault.

13.4.1 About Managing the RESTful Services Configuration File

The RESTful services key management commands are designed for administrators who are responsible for managing keys that are uploaded to Oracle Key Vault.

You must use a configuration file to run the RESTful service utility, `okvrestservices.jar`, whether you run the utility from the command line or from a script. You can use the `okvclient.ora` file that was created when you deployed the Oracle Key Vault client software, or you can create a new configuration file. In a

multi-master cluster environment, you can use one of the read-write nodes having its own configuration file. However, Oracle recommends that you use `okvclient.ora` as an configuration file to take advantage of the Oracle Key Vault multi-master cluster capabilities.

> **Note:**
>
> Script mode is not available for the KMIP RESTful service.

Related Topics

- Required Privileges for Using RESTful Services
  The required RESTful services privileges are consistent with the privileges required to perform the same task in the Oracle Key Vault management console.

13.4.2 Configuration File Creation Guidelines

You should follow the recommended guidelines to avoid script execution errors.

- Be aware that the commands and syntax in the script are identical to those used on the command line.
- Ensure that each line in the script is either a command or a line starting with the character #.
- Put each command on its own line.
- Start each line that does not have a command with the # character.
- Use the # character for comments and blank lines.
- Remember that the order in which command options appear do not matter.
- Ensure that all required options have valid values.
- Specify the `-i` or `--script` option when you execute the script.
- Enclose descriptions that are used for the `-d` or `--desc` option in double quotation marks if they contain spaces.

13.4.3 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.

   The default location is the ssl directory where you deployed the okvclient.jar file.

   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 13-1 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. Not used by KM REST API.
`log_level`	string	Required	Specifies one of the following log levels: • `all` logs every message • `severe` logs critical errors • `warning` logs non-critical errors that might pose problems • `info` logs general information • `fine` logs detail; is useful for debugging • `finest` logs the most detailed logging information
`ssl_wallet_loc`	string	Optional (required only if the configuration file is not `okvclient.ora` in KMIP REST)	Specifies the path to the Secure Sockets Layter (SSL) deployment directory for `okvlient.ora`. It is the location of`ewallet.p12` and `cwallet.sso` which are deployed from the `okvclient.jar` file.
`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. The `usr` property is not used by KM REST API as key access is determined by the endpoint privileges.
`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. Not used by KM REST API.

Table 13-1 (Cont.) Properties to Set in the Configuration File

Property	Value	Option	Description
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 usr option, the command will pick up the user's credentials from the wallet to establish connection with the Key Vault server. Not used by KM REST API.

3. Save the configuration file to a secure location.

13.4.4 Examples of Configuration Files

You can create a configuration file that uses an IP address and a host name.

Example 13-1 Configuration File Using IP Address

```
server=192.0.2.254
usr=okvadmin
log=/absolute_path_to_log_file/log_file_name
log_level=warning
client_wallet=/path_to_wallet_that_contains_credentials_for_RESTadmin
```

Example 13-2 Configuration File Using Host Name

```
server=Prod-OKV-07.example.com
usr=okvadmin
log=/absolute_path_to_log_file/log_file_name
log_level=warning
client_wallet=/path_to_wallet_that_contains_credentials_for_RESTadmin
```

13.4.5 Executing a Single RESTful Command

If you only want to run a few commands, then run them individually from the command line using the -r or --service option.

1. Log in to the endpoint where you want to execute the command.
2. Ensure that the script property has no value in the configuration file
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@example.com
   User: Key_Vault_user_name
   Password: Key_Vault_user_password
   ```

   In this specification:

   - -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@example.com`.

In a multi-master cluster environment, you must use the `okvclient.ora` configuration file in the default location to take advantage of multi-master cluster features. You can still use your own configuration file instead of using `okvclient.ora`. However, it will connect in standalone mode, you you cannot take advantage of the multi-master cluster environment.

> **Note:**
>
> Command line options have priority over options that are specified in the configuration file or script. For example, if the property `usr` is specified in the configuration file and the command line, then the command line option will override the one in the configuration file.

13.4.6 Executing Multiple RESTful Administrative Commands Using a Script

To save time and effort, as well as ensuring accuracy, you can use a script to run a sequence of commands.

You can run a sequence of Oracle Key Vault administration 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. This does not apply to executing key management commands using the KM REST API. You can define the `script` property in the configuration file to avoid entering it in the command line. You enter the `script` parameter only once, either in the configuration file or the command line.

1. Log in to the endpoint where you want to execute the commands.
2. Create the script file and add the service commands that 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@example.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, such as `create_hr_endpoint_group.txt`.
4. Edit the configuration file named `conf_file` and add the `script` property, and then set the property to the name of the script file and its full path.

   The configuration file should look similar to the following:

```
server=192.0.2.254
usr=okvadmin
log=/logs/okvrestservices.log
log_level=warning
script=/scripts/create_hr_endpoint_group.txt
client_wallet=/path_to_wallet_that_contains_credentials_for_RESTuser
```

5. If the the script property was defined in the configuration file, then only specify the configuration file to run the RESTful service utility:

   ```
   java -jar okvrestservices.jar -c conf_file
   User: Oracle_Key_Vault_user
   Password: Oracle_Key_Vault_user_password
   ```

   If you did not set the `script` property in the configuration file (Step 4) then 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: Oracle_Key_Vault_user
   Password: Oracle_Key_Vault_user_password
   ```

The RESTful Services utility executes one command at a time. If a command fails, then 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.

Related Topics

- Error Reporting
  The RESTful Service utility has robust error reporting to debug in order to run RESTful service commands quickly and successfully.

13.4.7 Creating a Script to Automatically Enroll Oracle Databases as Endpoints

You can create a script that database administrators can run to automatically enroll Oracle Database endpoints in Oracle Key Vault.

An Oracle Key Vault administrator can create a set of scripts and files that database administrators can later download from a shared location, and execute on their database servers to automatically on-board their databases into Oracle Key Vault, without any further intervention by the Oracle Key Vault administrators. As the Oracle Key Vault administrator, you will package the following:

- configuration file (for example, `config.ini`)
- `ewallet.p12` and `cwallet.sso` wallet files
- `run-me.sh` script

The following procedure explains how to create these components.

1. Log in to the Oracle Key Vault server.
2. Download the RESTful services package and store it in your working directory, where you will also create the other files.
3. If you have not done so already, create a user who has both the System Administrator and Key Administrator roles.

Use the Oracle Key Vault management console to create this user. For the procedure in this topic, this user will be named `restadmin`. A user with the System Administrator role creates the `restadmin` account and then grants the user the System Administrator role. Then a user who has the Key Administrator role will grant the `restadmin` user the Key Administrator role. Finally, the `restadmin` user must log in and change the one-time password to a permanent password.

4. Create the configuration file (for example `config.ini`) to have the following settings:

   - `server` is the IP address of the Oracle Key Vault server from which you downloaded the `okvrestservices.jar` file.
   - `usr` is the user name of the Oracle Key Vault user who has both the System Administrator and Key Administrator roles. Ideally, this should be a special user who exists along with the administrative users who have the System Administrator, Key Administrator, and Audit Manager roles.
   - `log` is the log file location, including the file name.
   - `log_level` is the log level for the configuration file. The setting finest is good because it logs detailed messages and it good for debugging.
   - `client_wallet` is an absolute path to a wallet that will contain the permanent password of the `restadmin` user. Because you are including the `usr` option, the command will pick up the user's credentials from the wallet to establish a connection with the Oracle Key Vault server.
   - `script` is an absolute path to the script file that you plan to run. This script contains the commands in the correct syntax that the RESTful services execute in Oracle Key Vault. But since all database administrators download that same collection of files, the script would use the same names for virtual wallets and endpoints in Oracle Key Vault. For each database that you want to configure as an endpoint, you must create a script that includes unique names for the virtual wallet and endpoints for this database instance in Oracle Key Vault. Hence, you must create and run this script on the server itself. One way to create unique names for virtual wallets and endpoints is to create them from the local environment variables of that database instance. You may need to find another method or naming convention to match your environment. A later step in this procedure provides an example of this script.

   For example, to create a `config.ini` file for a server whose IP address is `192.0.2.181`:

   ```
   server=192.0.2.181
   usr=restadmin
   log=/home/oracle/OKVrest.log
   log_level=finest
   client_wallet=/home/oracle
   script=/home/oracle/script.txt
   ```

   In this specification:

   - `usr=restadmin` refers to a user named `restadmin`, who has both the System Administrator and Key Administrator roles.
   - `client_wallet` points to the location where the RESTful services can find the password for the `restadmin` user. An Oracle Key Vault administrator must

create this wallet, which the `restadmin` account can do because this account is granted the Key Administrator role.

5. Execute the following command, which creates a wallet and insert the password of the `restadmin` user into it.

```
java -jar okvrestservices.jar -c config.ini --client_wallet /home/
oracle --add restadmin
Password: restadmin_password
```

This command creates the password-protected wallet `ewallet.p12` and the auto-login wallet `cwallet.sso`.

6. Create the script that will be referred to in the `script` parameter in the `config.ini` file.

Use the naming convention that your site normally uses for names of wallets and other components.
The following shell script, called `run-me.sh` is part of the package that an Oracle Key Vault administrator creates for the database administrators to download. It creates a file `script.txt` that is referenced in the `config.ini` file and contains unique names for the virtual wallet and the associated endpoints.

```
$ more run-me.sh

#!/bin/bash
export EP_NAME=${ORACLE_SID^^}_on_${HOSTNAME/.*}
cat > /home/oracle/script.txt << EOF
create_wallet -w ${DB_NAME^^}
create_endpoint -e $EP_NAME -t ORACLE_DB -q LINUX64 -d "$HOSTNAME,
$(hostname -i)"
set_default_wallet -e $EP_NAME -w ${DB_NAME^^}
provision -v temporary_password -e $EP_NAME -y /u01/opt/oracle/
product/okv
EOF
more script.txt
```

In this specification:

a. `export ...` creates the endpoint name from `uppercase(ORACLE_SID)_on_`*`short_hostname`*.

b. `create_wallet ...` creates a (shared) virtual wallet in Oracle Key Vault. Here, it has the `uppercase(database name)`.

c. `create_endpoint ...` creates an endpoint, named after the endpoint created in the `export` command in line a, with `type=ORACLE_DB, OS=LINUX64`, free text *`fully_qualified_hostname, IP address`*.

d. `set_default_wallet ...` sets the default wallet, associating the endpoint (from (line c) with the shared wallet (line b).

e. `provision ...` download and install with a temporary password into the `wallet_root/okv` existing directory.

7. Duplicate the `run-me.sh` script so that you will have a primary script and a secondary script, to be used for different situations.

The primary script will be used for single-instance databases and the first Oracle RAC instance. The secondary script will be used for the remaining Oracle RAC nodes of a primary database and all nodes of the corresponding standby Oracle RAC database. This secondary script will associate the endpoints with the shared wallet that was created on the first instance.

a. Rename the `run-me.sh` script to `primary-run-me.sh`.

b. Copy `primary-run-me.sh` to a new file named `secondary-run-me.sh`.

c. Open `secondary-run-me.sh` and remove the following line:

```
create_wallet -w ${DB_NAME^^}
```

8. Make the scripts executable.

```
$ chmod +x primary-run-me.sh
$ chmod +x secondary-run-me.sh
```

9. Test each of the scripts to ensure that they can create a `script.txt` file.

```
$ ./primary-run-me.sh
$ ./secondary-run-me.sh
```

10. Confirm that the names for the virtual wallets and endpoints follow your naming convention by executing the following command:

```
$ more script.txt
```

11. Create two `.zip` file packages for each of the scripts.

    Each package must have the following contents:

    - `primary.zip` contains `primary-run-me.sh`, `okvrestservices.jar`, `ewallet.p12`, `cwallet.sso`, and `config.ini`.
    - `secondary.zip` contains `secondary-run-me.sh`, `okvrestservices.jar`, `ewallet.p12`, `cwallet.sso`, and `config.ini`.

12. Make these two `.zip` files available to the database administrators for them to download from a shared file server.

13. Instruct the database administrators where to download and execute the scripts:

    - Execute the `primary-run-me.sh` script on single-instance databases and the first Oracle RAC instance. For an Oracle Data Guard environment, execute the script on the primary server.
    - Execute the `secondary-run-me.sh` script on all the remaining Oracle RAC nodes of a primary database and all nodes of the corresponding standby Oracle RAC database. For an Oracle Data Guard environment, execute the script on the standby server.

Related Topics

- Step 4: Download the RESTful Software Utility
  The RESTful software utility is in the `okvrestservices.jar` file.

- Creating an Oracle Key Vault User Account
  A user with the System Administrator role can create user accounts from the Oracle Key Vault management console.

13.5 Disabling RESTful Services

You should enable RESTful Services for short periods during endpoint registration and enrollment only.

RESTful Services are disabled by default. After you have enrolled the endpoints, you should disable RESTful services.

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

13.6 Oracle Key Vault Administrative REST Client Tool Commands

The RESTful services administrative commands are designed for administrators who manage endpoints and endpoint groups.

- RESTful Services Command Syntax
  The RESTful services command syntax provides for both long and short styles.
- RESTful Services Wallet Command Syntax
  The RESTful services wallet command syntax provides both long and short styles.
- Commands to Add and Enroll Endpoints
  You must have the System Administrator role use RESTful commands to manage endpoints.
- Commands to Modify Endpoint Details
  You must have the System Administrator role to modify endpoint details.
- Endpoint Group Commands
  You must have the System Administrator role to use RESTful commands to create and manage endpoint groups.
- Virtual Wallet Commands
  Virtual wallet commands manage the virtual wallet lifecycle and define access control mappings between virtual wallets and endpoints or endpoint groups.
- Error Reporting
  The RESTful Service utility has robust error reporting to debug in order to run RESTful service commands quickly and successfully.
- Help Information
  You can find information about valid options and the available commands that the RESTful services utility `okvrestservices.jar` provides.

13.6.1 RESTful Services Command Syntax

The RESTful services command syntax provides for both long and short styles.

You must be an endpoint administrator to use these commands.

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 13-2 List of RESTful Administrative Service Commands

Command Name	Description
`add_epg_member`	Adds an endpoint to an endpoint group. The endpoint must already exist.
`add_wallet_access_ep`	Sets access mappings on a virtual wallet for an endpoint
`add_wallet_access_epg`	Sets access mappings on a virtual wallet for an endpoint group
`check_object_status`	Checks the status of an endpoint, endpoint group, or wallet
`create_endpoint`	Adds an endpoint to Oracle Key Vault. When added, the endpoint is in the **Registered** state.
`create_endpoint_group`	Adds a new endpoint group
`create_unique_endpoint`	Adds a unique endpoint to Oracle Key Vault.
`create_unique_wallet`	Adds a unique virtual wallet to Oracle Key Vault
`create_wallet`	Adds a virtual wallet to Oracle Key Vault
`delete_endpoint`	Removes an endpoint from Oracle Key Vault
`delete_endpoint_group`	Deletes an endpoint group
`delete_wallet`	Removes the virtual wallet from Oracle Key Vault
`download`	Downloads the endpoint software `okvclient.jar` to install it manually at the endpoint.
`drop_epg_member`	Removes an endpoint from an endpoint group
`drop_wallet_access_ep`	Removes access mappings on a virtual wallet for an endpoint
`drop_wallet_access_epg`	Removes access mappings on a virtual wallet for an endpoint group
`get_default_wallet`	Gets the default wallet for an endpoint
`get_enrollment_token`	Gets the enrollment token to download the endpoint software for the registered endpoint
`get_wallets`	Gets all virtual wallets for an endpoint
`modify_endpoint_desc`	Changes the endpoint description
`modify_endpoint_email`	Changes the endpoint's email
`modify_endpoint_name`	Changes the endpoint name
`modify_endpoint_platform`	Changes the endpoint platform
`modify_endpoint_type`	Changes the endpoint type
`modify_wallet_access_ep`	Changes access mappings on a virtual wallet for an endpoint

Table 13-2 (Cont.) List of RESTful Administrative Service Commands

Command Name	Description
`modify_wallet_access_epg`	Changes access mappings on a virtual wallet for an endpoint group
`modify_wallet_desc`	Changes the virtual wallet description
`provision`	Downloads and installs the endpoint software `okvclient.jar`. After this, the endpoint is in the **Enrolled** state.
`re_enroll`	Reenrolls an endpoint
`re_enroll_all`	Reenrolls all endpoints
`set_default_wallet`	Sets the default wallet for an endpoint

Example 13-3 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 13-4 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]
```

13.6.2 RESTful Services Wallet Command Syntax

The RESTful services wallet command syntax provides both long and short styles.

The following example shows RESTful service commands that pertain to Oracle wallets specified by the `--client_wallet` option. This wallet stores 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 13-3 Wallet Command Options

Option	Required?	Description
`-A, --add`	Optional	Adds a user to wallet
`-D, --delete`	Optional	Deletes a user from a wallet
`-f, --force`	Optional	Performs the operation without prompting for confirmation
`-j, --client_wallet` *arg*	Required	Stands for the absolute path to the wallet location
`-L, --listuser`	Optional	Lists the users who have access to a wallet
`-M, --modify`	Optional	Modifies a user's password
`-w, --wallet_name` *arg*	Required	Stands for the wallet name

Example 13-5 Wallet Command Syntax

Add the user's password into the `client_wallet`, so that the password can be hidden from the endpoint database administrator who runs the RESTful script:

```
java -jar okvrestservices.jar -c path_to_configuration file/rest.init --
client_wallet absolute_path_to_wallet_location --add user
```

Change the credentials and password in the `client_wallet`, after ensuring that this password matches the password that was created for that user in the **Users** tab in the Oracle Key Vault management console:

```
java -jar okvrestservices.jar -c path_to_configuration file/rest.init --
client_wallet absolute_path_to_wallet_location --modify user
```

Lists all Oracle Key Vault users whose credentials are stored in the client_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 credentials from the `client_wallet`:

```
java -jar okvrestservices.jar --config path_to_configuration file/rest.init --
client_wallet absolute_path_to_wallet_location --delete user
```

13.6.3 Commands to Add and Enroll Endpoints

You must have the System Administrator role use RESTful commands to manage endpoints.

- create_endpoint Command
  The `create_endpoint` command adds a new endpoint to Oracle Key Vault.
- create_unique_endpoint Command
  The `create_unique_endpoint` command adds a new unique endpoint to Oracle Key Vault.
- delete_endpoint Command
  The `delete_endpoint` command removes an endpoint from Oracle Key Vault.
- download Command
  The `download` command downloads the endpoint software (`okvclient.jar`) to a directory that you name.
- get_enrollment_token Command
  The `get_enrollment_token` command retrieves an enrollment token for a registered endpoint.
- provision Command
  The `provision` command downloads and installs the endpoint software in the specified directory, which must exist.
- re_enroll Command
  The `re_enroll` command re-enrolls a previously enrolled endpoint in order to upgrade the endpoint software.
- re_enroll_all Command
  The `re_enroll_al` command re-enrolls all previously enrolled endpoints in order to upgrade the endpoint software.

13.6.3.1 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: • `linux64` • `solaris64` • `solaris_sparc` • `aix` • `hpux` • `windows`
`-t, --ep_type`	Required	Type of the endpoint. Allowed values are: • `oracle_db` • `oracle_non_db` • `other`
`-m, --ep_email`	Optional	Email address of the endpoint administrator. Enclose this value in double quotation marks.
`-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. If you omit this option, then you will be prompted to enter the password interactively. For greater security, omit this option.

Parameter	Required?	Description
`-r`, `--service` *`arg`*	Required	Specifies the RESTful service that you want to execute listed in RESTful Services Command Syntax
`-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
```

13.6.3.2 create_unique_endpoint Command

The `create_unique_endpoint` command adds a new unique endpoint to Oracle Key Vault.

This command is only used in a multi-master cluster environment.

When you create the endpoint, a unique ID is returned. You an use this ID to check the status of the endpoint creation, whether it is in progress (`PENDING`) or complete (`ACTIVE`). If the status is `PENDING`, then it is not yet usable, so any actions performed on the endpoint will fail. If the status is `ACTIVE`, then the endpoint is usable. To check the status, execute the `check_object_status` command, specifying this unique ID by including the `-x` or `--uid` parameter. Next, if the status is `ACTIVE`, execute the `get_object_name` command to confirm the name of the endpoint after Oracle Key Vault performs name resolution for this name. If the name that you provided is already used in another node, then the name for this endpoint will have `_OKV`*`xx`* appended to it. For example, if you named the endpoint `ep12`, and there is a naming conflict, the name could be `EP12_OKV01`.

Syntax

Short form:

```
create_unique_endpoint -e endpoint_name -d "description" -q platform -m
email_address -t type
```

Long form:

```
create_unique_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, then you must enclose it within double quotation marks.
`-q, --platform`	Required	The endpoint platform. Allowed values are: • `linux64` • `solaris64` • `solaris_sparc` • `aix` • `hpux` • `windows`
`-t, --ep_type`	Required	Type of the endpoint. Allowed values are: • `oracle_db` • `oracle_non_db` • `other`
`-m, --ep_email`	Required	Email address of the endpoint administrator. Enclose this value in double quotation marks.
`-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. 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
`-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 `my_db_ep` is added, with an optional identifying description `'My 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_unique_endpoint -e my_db -d
"My database endpoint" -q solaris64 -t oracle_db -m psmith@example.com
```

Long Form Example

```
java -jar okvrestservices.jar --config conf_file --
service create_unique_endpoint --ep_name my_db --desc "My database endpoint" --
platform solaris64 --ep_type oracle_db --ep_email psmith@example.com
```

13.6.3.3 delete_endpoint Command

The `delete_endpoint` command removes an endpoint from Oracle Key Vault.

A confirmation message appears asking if you are sure you want to delete the endpoint. You can use the `-f` or `--force` option to remove the endpoint without a confirmation message. Use the `-f` or `--force` option carefully, because 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
`-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. 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
`-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 removes the `sales_db_ep` endpoint from Oracle 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
```

13.6.3.4 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.

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. For example, if you specify `-o /tmp`, then the endpoint software is downloaded to `/tmp/endpoint_name/okvclient.jar`.
`-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. If you omit this option, then you will be prompted to enter the password interactively. For greater security, omit this option.

Parameter	Required ?	Description
`-r`, `--service` *`arg`*	Required	Specifies the RESTful service that you want to execute listed in RESTful Services Command Syntax
`-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, the endpoint software, `okvclient.jar`, is downloaded to `/home/oracle/downloads/hr_db_ep/okvclient.jar` for an endpoint called `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/
```

13.6.3.5 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 the **Registered** state.

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
`-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.

Parameter	Required?	Description
`-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. 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
`-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, a registered endpoint called `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
```

13.6.3.6 provision Command

The `provision` command downloads and installs the endpoint software in the specified directory, which must exist.

This directory should have read, write and execute permissions for the owner and its group. For example, if the Oracle Key Vault endpoint software is installed in an Oracle Database server, then this endpoint installation directory should have read, write, and execute permissions by the `oracle` user and the `oinstall` group. This ensures that processes can access directories appropriately at run time.

You must meet the following prerequisites to run this command:

- You must be a user with system administrative privileges
- You must ensure that the soft link`/usr/bin/java` points to `$ORACLE_HOME/jdk/jre/bin/java`.
- You must know how the installation process determines the location of the `okvclient.ora` file.

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.
`-o`, `--dir`	Required	Existing directory in which to download and install the endpoint software
`-y`, `--okv_home`	Required if `-o`, `--dir` option is not used	Oracle Key Vault home directory to be used for the wallet root on the client side. Choose between this parameter and `-o`, `--dir`. The only difference between using `-o`, `--dir` and `-y`, `--okv_home` is that `-y`, `--okv_home` does not create an endpoint directory.
`-v`, `--endpoint_password`	Optional	Endpoint password. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons. If you omit this option (recommended for better security), 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 Oracle Key Vault server over mutually authenticated Transport Layer Security (TLS). If you created an auto-login wallet without a password during the endpoint software installation, then the endpoint credentials are stored in an Oracle wallet.
`-a`, `--autologin`	Required	The endpoint credentials to connect to the Oracle Key Vault server are stored in an auto-login 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.

Parameter	Required?	Description
`-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. 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
`-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 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 -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, the password 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 --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/
```

13.6.3.7 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
`-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. 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
`-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, the endpoint `hr_db_ep` is 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
```

13.6.3.8 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. 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
`-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
```

13.6.4 Commands to Modify Endpoint Details

You must have the System Administrator role to modify endpoint details.

- modify_endpoint_email Command
  The `modify_endpoint_email` command changes the email address for the endpoint.

- modify_endpoint_desc Command
  The `modify_endpoint_desc` command changes the description of an endpoint.
- modify_endpoint_name Command
  The `modify_endpoint_name` command changes the name of an endpoint.
- modify_endpoint_platform Command
  The `modify_endpoint_platform` command changes the platform for an endpoint.
- modify_endpoint_type Command
  The `modify_endpoint_type` command changes the endpoint type.

13.6.4.1 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. Enclose this value in double quotation marks.
`-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. 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
`-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 email of endpoint `hr_db` to `tjones@example.com`.

```
java -jar okvrestservices.jar -c conf_file -r modify_endpoint_email -e hr_db -m
tjones@example.com
```

Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service modify_endpoint_email
--ep_name hr_db --ep_email tjones@example.com
```

13.6.4.2 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 quotation marks
`-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. 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
`-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 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"
```

13.6.4.3 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
`-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. 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

Parameter	Required?	Description
`-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 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 -n
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
```

13.6.4.4 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: • `linux64` • `solaris64` • `solaris_sparc` • `aix` • `hpux` • `windows`
`-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.

Parameter	Required?	Description
`-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. 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
`-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
```

13.6.4.5 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: • `oracle_db` • `oracle_non_db` • `other`
`-c, --config`	Required	Specifies the absolute path to the configuration file

Parameter	Required?	Description
`-i, --script arg`	Required for multiple RESTful service commands	Specifies the absolute path to the script file. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons. 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. 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
`-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 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
```

13.6.5 Endpoint Group Commands

You must have the System Administrator role to use RESTful commands to create and manage endpoint groups.

- add_epg_member Command
  The `add_epg_member` command adds an existing endpoint to an endpoint group.
- create_endpoint_group Command
  The `create_endpoint_group` command creates a new endpoint group.
- create_unique_endpoint_group Command
  The `create_unique_endpoint_group` command creates a new endpoint group.
- delete_endpoint_group Command
  The `delete_endpoint_group` command removes an endpoint group from Oracle Key Vault.
- drop_epg_member Command
  The `drop_epg_member` command removes an endpoint from an endpoint group.

- modify_endpoint_group_desc Command
  The `modify_endpoint_group_desc` command changes the description of an endpoint group.
- modify_endpoint_group_name Command
  The `modify_endpoint_group_name` command changes the name of an endpoint group.

13.6.5.1 add_epg_member Command

The `add_epg_member` command adds an existing endpoint to an endpoint group.

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
`-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. 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
`-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 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
```

13.6.5.2 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_unique_endpoint_group --epg_name endpoint_group_name --desc "endpoint
grouop 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 quotation marks
`-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. 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

Parameter	Required?	Description
`-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 shows an endpoint group called `epg_hr` being created with the description `HR endpoint group`.

```
java -jar okvrestservices.jar -c conf_file -r create_unique_endpoint_group --
epg_name EPG3 --desc "EPG test"
```

Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service create_endpoint_group
--epg_name epg_hr --desc "HR endpoint group"
```

13.6.5.3 create_unique_endpoint_group Command

The `create_unique_endpoint_group` command creates a new endpoint group.

This command is only used in a multi-master cluster environment.

When you create the endpoint group, a unique ID is returned. You an use this ID to check the status of the endpoint group creation, whether it is in progress (`PENDING`) or complete (`ACTIVE`). If the status is `PENDING`, then it is not yet usable, so any actions performed on the endpoint group will fail. If the status is `ACTIVE`, then the endpoint group is usable. To check the status, execute the `check_object_status` command, specifying this unique ID by including the `-x` or `--uid` parameter. Next, if the status is `ACTIVE`, execute the `get_object_name` command to confirm the name of the endpoint group after Oracle Key Vault performs name resolution for this name. If the name that you provided is already used in another node, then the name for this endpoint group will have `_OKV`*xx* appended to it. For example, if you named the endpoint group `epg12`, and there is a naming conflict, the name could be `EPG12_OKV01`.

Syntax

Short form:

```
create_unique_endpoint_group -g unique_endpoint_group_name -d "unique_endpoint
group description"
```

Long form:

```
create_unique_endpoint_group --epg_name endpoint_group_name --desc "endpoint
group description"
```

Parameters

Parameter	Required?	Description
`-g`, `--epg_name`	Required	Name of the endpoint group

Parameter	Required?	Description
`-d, --desc`	Optional	A user-friendly description of the endpoint group enclosed within double quotation marks
`-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. 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
`-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 shows an endpoint group called `epg_hr` being created with the description `HR endpoint group`.

```
java -jar okvrestservices.jar -c conf_file -r create_unique_endpoint_group --
epg_name EPG3 --desc "EPG test"
```

Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service
create_unique_endpoint_group --epg_name epg_hr --desc "HR endpoint group"
```

13.6.5.4 delete_endpoint_group Command

The `delete_endpoint_group` command removes an endpoint group from Oracle Key Vault.

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
`-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. 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
`-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 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
```

13.6.5.5 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
`-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. 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
`-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 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
```

13.6.5.6 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 quotation marks
-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. 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
-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 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"
```

13.6.5.7 modify_endpoint_group_name Command

The modify_endpoint_group_name command changes the name of an endpoint group.

Syntax

Short form:

```
modify_endpoint_group_name -g endpoint_group_name -z new_endpoint_group_name
```

Long form:

```
modify_endpoint_group_name  --epg_name epg endpoint_group_name --new_name
new_endpoint_group_name
```

Parameters

Parameter	Required?	Description
`-g, --epg_group_name`	Required	Current name of the endpoint group
`-z, --new_name`	Required	New endpoint group name. The name can include letters, numbers, and underscores. The endpoint group name is case sensitive. The maximum length is 30 characters.
`-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. 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
`-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 endpoint group name `hr_db` to `hr_db_west`.

```
java -jar okvrestservices.jar modify_endpoint_name -g hr_db -z hr_db_west
```

Long Form Example

```
java -jar okvrestservices.jar modify_endpoint_name --epg_name hr_db --new_name
hr_db_west
```

13.6.6 Virtual Wallet Commands

Virtual wallet commands manage the virtual wallet lifecycle and define access control mappings between virtual wallets and endpoints or endpoint groups.

You must have the Key Administrator role to execute the wallet commands.

- add_wallet_access_ep Command
  The `add_wallet_access_ep` command grants an endpoint a level of access to a virtual wallet.
- add_wallet_access_epg Command
  The `add_wallet_access_epg` command grants an endpoint group a level of access to a virtual wallet.
- check_object_status Command
  The `check_object_status` command checks the status of an endpoint, an endpoint group, or a wallet.
- create_unique_wallet Command
  The `create_unique_wallet` command creates a unique virtual wallet.
- create_wallet Command
  The `create_wallet` command creates a virtual wallet.
- delete_wallet Command
  The `delete_wallet` command deletes a wallet from Oracle Key Vault.
- drop_wallet_access_ep Command
  The `drop_wallet_access_ep` command removes an endpoint's access to a wallet.
- drop_wallet_access_epg Command
  The `drop_wallet_access_epg` command removes an endpoint group's access to virtual wallet.
- get_default_wallet Command
  The `get_default_wallet` command gets the default wallet associated with an endpoint.
- get_object_name Command
  The `get_object_name` command retrieves the name of a managed object if the object status is `ACTIVE`.
- get_wallets Command
  The `get_wallets` command gets all the virtual wallets associated with an endpoint.
- modify_wallet_access_ep Command
  The `modify_wallet_access_ep` command changes the virtual wallet access level to an endpoint.
- modify_wallet_access_epg Command
  The `modify_wallet_access_epg` command modifies the virtual wallet access level to an endpoint group.
- modify_wallet_desc Command
  The `modify_wallet_desc` command modifies the description of an existing virtual wallet.
- modify_wallet_name Command
  The `modify_wallet_name` command modifies the virtual wallet name.
- set_default_wallet Command
  The `set_default_wallet` command sets the default wallet for an endpoint.

13.6.6.1 add_wallet_access_ep Command

The `add_wallet_access_ep` command grants 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: • `ro`: Read only • `rm`: Read and modify • `ro_mw`: Read only and manage virtual wallet • `rm_mw`: Read and modify and manage virtual 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. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons. 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
`-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 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
```

13.6.6.2 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: • `ro`: Read only • `rm`: Read and modify • `ro_mw`: Read only and manage virtual wallet • `rm_mw`: Read and modify and manage virtual 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. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons. If you omit this option, then you will be prompted to enter the password interactively. For greater security, omit this option.

Parameter	Required?	Description
`-r, --service arg`	Required	Specifies the RESTful service that you want to execute listed in RESTful Services Command Syntax
`-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 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
```

13.6.6.3 check_object_status Command

The `check_object_status` command checks the status of an endpoint, an endpoint group, or a wallet.

This command is only used in a multi-master cluster environment.

Syntax

Short form:

```
check_object_status -b EP|EPG|WALLET -x uuid
```

Long form:

```
check_object_status --type EP|EPG|WALLET --uuid uuid
```

Parameters

Parameter	Required?	Description
`-b, --type`	Required	Object type to check
`-x, --uid`	Required	UUID (universally unique ID) of the managed object
`-b, --type arg`	Required	Specifies the object type to check. Valid values include: • `EP` • `EPG` • `WALLET`
`-c, --config`	Required	Specifies the absolute path to the configuration file

Parameter	Required?	Description
`-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. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons. 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
`-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 shows the status for a wallet with the specified UUID.

```
java -jar okvrestservices.jar -c path_to_okvclient.ora -r check_object_status -b
WALLET -x 7C3DC1FF-213A-4FBE-BF4A-98A04F8D05DF
```

Long Form Example

```
java -jar okvrestservices.jar --config path_to_okvclient.ora --
service check_object_status --type WALLET --uid 7C3DC1FF-213A-4FBE-
BF4A-98A04F8D05DF
```

13.6.6.4 create_unique_wallet Command

The `create_unique_wallet` command creates a unique virtual wallet.

This command is only used in a multi-master cluster environment.

In order to use this command, the `okvclient.ora` file must be the configuration file that is referenced. When you create the wallet, a unique ID is returned. You an use this ID to check the status of the wallet creation, whether it is in progress (`PENDING`) or complete (`ACTIVE`). If the status is `PENDING`, then it is not yet usable, so any actions performed on the wallet will fail. If the status is `ACTIVE`, then the wallet is usable. To check the status, execute the `check_object_status` command, specifying this unique ID by including the `-x` or `--uid` parameter. Next, if the status is `ACTIVE`, execute the `get_object_name` command to confirm the name of the wallet after Oracle Key Vault performs name resolution for this name. If the name that you provided is already used in another node, then the name for this wallet will have `_OKV`*xx* appended to it. For example, if you named the wallet `WALLET12`, and there is a naming conflict, the name could be `wallet12_OKV01`.

Syntax

Short form:

```
create_unique_wallet -w wallet_name -d "wallet_description"
```

Long form:

```
create_unique_wallet --wallet_name wallet_name --desc "wallet_description"
```

Parameters

Parameter	Required?	Description
`-w, --wallet_name`	Required	The name of the unique wallet you want to create
`-d, --desc`	Optional	A descriptive name for the unique wallet enclosed within double quotation marks
`-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. 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
`-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 creates a unique virtual wallet named `my_wallet` with the description `My unique wallet`.

```
java -jar okvrestservices.jar -c conf_file -r create_unique_wallet -w my_wallet
-d "My unique wallet"
```

Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service create_unique_wallet
--wallet  my_wallet --desc "My unique wallet"
```

13.6.6.5 create_wallet Command

The `create_wallet` command creates 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 quotation marks
`-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. 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
`-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 creates a wallet named `hr_wallet` with the description `Virtual wallet for HR endpoint`.

```
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"
```

13.6.6.6 delete_wallet Command

The `delete_wallet` command deletes a wallet from Oracle 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
`-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. 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
`-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 deletes the wallet `hr_wallet` 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
```

13.6.6.7 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
`-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. 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
`-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 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
```

13.6.6.8 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
`-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. 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
`-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 grants the endpoint group `epg_hr` the read, modify, and manage access privileges 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
```

13.6.6.9 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
`-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. 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
`-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

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
```

13.6.6.10 get_object_name Command

The `get_object_name` command retrieves the name of a managed object if the object status is `ACTIVE`.

This command is only used in a multi-master cluster environment.

Syntax

Short form:

```
get_object_name -b EP|EPG|WALLET -x uid
```

Long form:

```
get_object_name --type EP|EPG|WALLET --uid uid
```

Parameters

Parameter	Required?	Description
`-x, --uid`	Required	Unique identifier
`-b, --type arg`	Required	Specifies the object type to check. Valid values include: • `EP` • `EPG` • `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. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons. 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

Parameter	Required?	Description
`-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 get_object_name -b WALLET -
u 7C3DC1FF-213A-4FBE-BF4A-98A04F8D05DF
```

Long Form Example

```
java -jar okvrestservices.jar --config conf_file --service get_object_name --
type WALLET --uid 7C3DC1FF-213A-4FBE-BF4A-98A04F8D05DF
```

13.6.6.11 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
`-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.

Parameter	Required?	Description
`-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. 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
`-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

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
```

13.6.6.12 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
`-l, --access_level`	Required	Level of access for the virtual wallet. Values are as follows: • `ro`: Read only • `rm`: Read and modify • `ro_mw`: Read only, and manage virtual wallet • `rm_mw`: Read, modify, and manage virtual 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. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons. 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
`-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 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
```

13.6.6.13 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
`-l, --access_level`	Required	Level of access for the virtual wallet. Values are as follows: • `ro`: Read only • `rm`: Read and modify • `ro_mw`: Read only and manage virtual wallet • `rm_mw`: Read and modify and manage virtual 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. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons. 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
`-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 grants the read, modify, and manage privileges on the endpoint group `epg_hr` for 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
```

13.6.6.14 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 quotation marks
`-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. 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

Parameter	Required?	Description
`-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 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"
```

13.6.6.15 modify_wallet_name Command

The `modify_wallet_name` command modifies the virtual wallet name.

Syntax

Short form:

```
modify_wallet_name -w current_wallet_name -z new_wallet_name
```

Long form:

```
modify_wallet_name --wallet_name current_wallet_name --new_name new_wallet_name
```

Parameters

Parameter	Required?	Description
`-w`, `--wallet_name`	Required	Current name of the wallet
`-z`, `--new_name`	Required	New virtual wallet name. The name can include letters, numbers, and underscores. The endpoint group name is case sensitive. The maximum length is 30 characters.
`-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.

Parameter	Required?	Description
`-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. 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
`-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 change the name of the virtual wallet `hr_db_wallet` to `hr_db_west_wallet`.

```
java -jar okvrestservices.jar -r modify_wallet_name -w hr_db_wallet -z
hr_db_west_wallet
```

Long Form Example

```
java -jar okvrestservices.jar --service modify_wallet_name --wallet_name
hr_db_wallet --new_name hr_db_west_wallet
```

13.6.6.16 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
`-c, --config`	Required	Specifies the absolute path to the configuration file

Parameter	Required?	Description
`-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. 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
`-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 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
```

13.6.7 Error Reporting

The RESTful Service utility has robust error reporting to debug in order to run RESTful service commands quickly and successfully.

- About Error Reporting
  The status of command execution, passed and failed, is reported promptly on the command line and written to the log file.
- Command Line Error Reporting
  Error reporting captures both faulty actions, such as incorrect passwords, and successful command executions.
- 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.

13.6.7.1 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.

13.6.7.2 Command Line Error Reporting

Error reporting captures both faulty actions, such as incorrect passwords, and successful command executions.

Example 13-6 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 13-7 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 13-8 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 13-9 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]
```

13.6.7.3 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 13-10 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]
```

13.6.8 Help Information

You can find information about valid options and the available commands that the RESTful services utility `okvrestservices.jar` provides.

For a list of valid options, you can use the `-h` or `--help` option with the RESTful services utility `okvrestservices.jar`.

```
-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
```

To see the list of RESTful service commands, include `-H` or `--list` at the command line. For example:

```
-bash-4.1$ java -jar okvrestservices.jar -H
```

13.7 Oracle Key Vault Key Management REST Client Tool Commands

The RESTful services key management commands are designed for administrators who are responsible for managing keys that are uploaded to Oracle Key Vault.

- About Oracle Key Vault Key Management REST Client Tool Commands
  The Oracle Key Vault key management REST client tool provides a simplified interface to Key Management Interoperability Protocol (KMIP) operations using `okvrestservice.jar` commands.
- Oracle Key Vault Key Management REST Client API Using OKVRESTSERVICE
  To call the KM REST client API using `OKVRESTSERVICE` (`okvrestservice.jar`), you must include the `kmip` option.

- List of Key Management REST Client Tool Commands
  The RESTful key management client tools perform tasks such as creating and activating keys.
- Key Creation and Registration Commands
  The key creation and registration commands perform tasks such as registering different types of security objects.
- Key Attribute Management Commands
  The key attribute management commands perform tasks such as adding, modifying, and deleting standard and custom attributes.
- Key Life Cycle Management Commands
  The key life management commands perform tasks such as activating and revoking managed cryptographic objects.
- Wallet Commands
  The wallet commands control wallet memberships.

13.7.1 About Oracle Key Vault Key Management REST Client Tool Commands

The Oracle Key Vault key management REST client tool provides a simplified interface to Key Management Interoperability Protocol (KMIP) operations using `okvrestservice.jar` commands.

You can embed this tool in client shell scripts or in any programming language. Before you use KM REST client, you must download and deploy the endpoint software, `okvclient.jar`. The Oracle Key Vault key management REST server will reject the connection from any endpoint that was not deployed. The Oracle Key Vault key management REST API requires JRE 1.8 or later. You must have the Key Administrator role to execute the commands that this interface provides.

If you are using a multi-master cluster environment, then you must download the `okvrestservices.jar` file while connected in a read/write node. Oracle Key Vault does not allow `okvrestservices.jar` to be downloaded in read-only mode. Once it is downloaded in read/write mode, the client tool can connect in read/write mode or read-only mode.

13.7.2 Oracle Key Vault Key Management REST Client API Using OKVRESTSERVICE

To call the KM REST client API using `OKVRESTSERVICE` (`okvrestservice.jar`), you must include the `kmip` option.

The `drop_wallet_access_epg` command removes an endpoint group's access to virtual wallet.

Syntax

```
java -jar okvrestservices.jar kmip --config conf_file --service
options_for_the_KMIP_API
```

KMIP REST Service Options

Option	Description
`-a, --algorithm`	Cryptographic algorithm to use. Choices are: • `3DES` • `AES`
`-c, --config`	Specifies the absolute path to the configuration file
`-e, --user`	User name
`-f , --attr`	The attribute list file. The file format is: *`name1:value1`* *`name2:value2`* *`name3:value3`*
`-g, --group`	Group member
`-i, --index`	Attribute index
`-l, --length`	Key lengths supported: • 3DES: 112, 168 • AES : 128, 192, 256
`-m, --mask`	Cryptographic usage mask. Valid values include the following settings, which are described in the Key Management Interoperability Protocol (KMIP) Specification version 1.1. • `ENCRYPT` • `DECRYPT` • `WRAP_KEY` • `UNWRAP_KEY` • `EXPORT` • `DERIVE_KEY` • `GENERATE_CRYPTOGRAM` • `VALIDATE_CRYPTOGRAM` • `TRANSLATE_ENCRYPT` • `TRANSLATE_DECRYPT` • `TRANSLATE_WRAP` • `TRANSLATE_UNWRAP`
`-n, --attribute`	Attribute name
`-o, --object`	Path to object file to register
`-r, --code`	Revoke code values include: • `UNSPECIFIED` • `KEY_COMPROMISE` • `CA_COMPROMISE` • `AFFILIATION_CHANGED` • `SUPERSEDED` • `CESSATION_OF_OPERATION` • `PRIVILEGE_WITHDRAWN`
`-s, --reason`	Revoke reason message
`-t, --type`	Object type to register
`-u, --uid`	UUID (Universally Unique ID) of the managed object
`-v, --value`	Attribute value

Option	Description
`-w, --wallet`	Wallet name
`-x, --max`	Maximum number of UIDs to display

13.7.3 List of Key Management REST Client Tool Commands

The RESTful key management client tools perform tasks such as creating and activating keys.

Table 13-4 RESTful Key Management Commands

Command Name	Description
`activate`	Activates a managed cryptographic object
`add_attr`	Adds a new attribute for an object and sets its value
`add_custom_attr`	Adds a custom attribute specifying name, type, and value
`add_member`	Adds a user to a wallet
`all_attr`	Gets all attribute names and values
`create_key`	Creates a new key
`del_attr`	Deletes an attribute of a managed object
`del_custom_attr`	Deletes a custom attributes
`del_member`	Deletes a user from a wallet
`destroy`	Requests the server to destroy the key data for the managed object
`get_attr`	Retrieves one or more attribute values of an object
`get_cert`	Requests a digital certificate from the server
`get_key`	Retrieves an encryption key from the server
`get_secret`	Retrieves all available attribute names and values
`list_attr`	Lists all available attribute names only
`list_wallet`	Lists virtual wallets
`locate`	Searches for one or more managed objects
`mod_attr`	Modifies the value of an attribute for a managed object
`mod_custom_attr`	Modifies a custom attribute
`query`	Queries the server regarding its supported capabilities
`reg_cert`	Registers a digital certificate
`reg_key`	Registers a key

Table 13-4 (Cont.) RESTful Key Management Commands

Command Name	Description
`reg_opaque`	Registers an object containing data that the server may not be able to interpret
`reg_secret`	Registers an object containing secret data
`revoke`	Revokes a managed cryptographic object.

13.7.4 Key Creation and Registration Commands

The key creation and registration commands perform tasks such as registering different types of security objects.

You must have the Key Administrator role to use these commands.

- create_key Command
  The `create_key` command creates a new key.
- reg_key Command
  The `reg_key` command registers a key.
- get_cert Command
  The `get_cert` command retrieves a digital certificate.
- get_key Command
  The `get_key` command retrieves an encryption key.
- get_opaque Command
  The `get_opaque` command retrieves an object that contains secret data.
- get_secret Command
  The `get_secret` command retrieves an object that contains secret data.
- reg_cert Command
  The `reg_cert` command registers a certificate.
- reg_opaque Command
  The `reg_opaque` command registers opaque objects.
- reg_secret Command
  The `reg_secret` command registers secret data such as passwords or random seeds.

13.7.4.1 create_key Command

The `create_key` command creates a new key.

You must provide a cryptographic algorithm, a key length, and a usage mask.

Syntax

Short form:

```
-s create_key -a  cryptographic_algorithm -l key_length -m
cryptographic_usage_mask
```

Long form:

```
--service create_key --algorithm  cryptographic_algorithm --length key_length
--mask crypographic_usage_mask
```

Parameters

Parameter	Required?	Description
-a, --algorithm	Required	Cryptographic algorithm
-l, --length	Required	Key length
-m, --mask	Required	Cryptographic usage mask, enclosed in double quotation marks

Short Form Example

This example creates an Advanced Encryption Standard (AES) key.

```
java -jar okvrestservices.jar kmip -c conf_file -s create_key -a AES -l 128
-m "ENCRYPT,DECRYPT,EXPORT"
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service create_key --
algorithm AES --length 128  --mask "ENCRYPT,DECRYPT,EXPORT"
```

13.7.4.2 reg_key Command

The reg_key command registers a key.

Syntax

Short form:

```
-s reg_key -a algorithm -l key_length -m crypto_usage_mask -o
path_to_object_file  [-f path_to_object_attr_file>]
```

Long form:

```
--service reg_key --algorithm algorithm --length key_length
--mask crypto_usage_mask --object path_to_object_file  [-
attr path_to_object_attr_file]
```

Parameters

Parameter	Required?	Description
-l, --length	Required	Key length
-a, --algorithm	Required	Algorithm
-m, --mask	Required	Cryptographic usage mask
-o, --object	Required	Path to key file
-f, --attr	Optional	Attribute list file

Short Form Example

These examples locate an object based on the specified search criteria.

```
java -jar okvrestservices.jar kmip -c conf_file -s reg_key -a AES -l 128 -m
"EXPORT" -o ./object.key  -f ./obj_key_attr_file.txt
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service_reg_key --
algorithm AES --length 128 --mask "EXPORT" --object ./object.key  --attr ./
obj_key_attr_file.txt
```

13.7.4.3 get_cert Command

The `get_cert` command retrieves a digital certificate.

Syntax

Short form:

```
-s get_cert -u UUID
```

Long form:

```
--service get_cert --uid UUID
```

Parameters

Parameter	Required?	Description
`-u`, `--uid`	Required	Universally unique ID (UUID) of the managed object

Short Form Example

```
java -jar okvrestservices.jar kmip -c conf_file -s get_cert -u
D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service get_cert --uid
D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

13.7.4.4 get_key Command

The `get_key` command retrieves an encryption key.

Syntax

Short form:

```
-s get_key -u UUID
```

Long form:

```
--service get_key --uid UUID
```

Parameters

Parameter	Required?	Description
`-u, --uid`	Required	Universally unique ID (UUID) of the managed object

Short Form Example

```
java -jar okvrestservices.jar kmip -c conf_file -s get_key -u D69D2F32-2DBB-4FF3-
BF52-95487526E6EC
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service get_key --uid
D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

13.7.4.5 get_opaque Command

The `get_opaque` command retrieves an object that contains secret data.

Syntax

Short form:

```
-s get_opaque -u UUID
```

Long form:

```
--service get_opaque --uid UUID
```

Parameters

Parameter	Required?	Description
`-u, --uuid`	Required	Universally unique ID (UUID) of the managed object

Short Form Example

```
java -jar okvrestservices.jar kmip -c conf_file -s get_opaque -u
D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service get_opaque --uid
D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

13.7.4.6 get_secret Command

The `get_secret` command retrieves an object that contains secret data.

Syntax

Short form:

```
-s get_secret -u UUID
```

Long form:

```
--service get_secret --uid UUID
```

Parameters

Parameter	Required?	Description
-u, --uid	Required	Universally unique ID (UUID) of the managed object

Short Form Example

```
java -jar okvrestservices.jar kmip -c conf_file -s get_secret -u
D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service get_secret --uid
D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

13.7.4.7 reg_cert Command

The `reg_cert` command registers a certificate.

Syntax

Short form:

```
-s reg_cert -t object_type -a algorithm -l key_length -m crypto_usage_mask -
o path_to_cert file  [-f path_to_object_attr_file]
```

Long form:

```
--service reg_cert --type object_type --algorithm algorithm --
length key_length --mask crypto_usage_mask --object path_to_cert_file  [-attr
path_to_cert_attr_file]
```

Parameters

Parameter	Required?	Description
-t, --type	Required	Object type
-l, --length	Required	Key length
-a, --algorithm	Required	Algorithm
-m, --mask	Required	Cryptographic usage mask
-o, --object	Required	Path to object attributes file
-f, --attr	Optional	Attribute list file

Short Form Example

```
java -jar okvrestservices.jar kmip -c conf_file -s reg_cert -t X_509 -a AES -
l 128 -m ENCRYPT -o ./my_cert -f ./cert_attr_file.txt
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service reg_cert -t
X_509 --algorithm AES --length 128 --mask ENCRYPT --object ./my_cert  --attr ./
cert_attr_file.txt
```

13.7.4.8 reg_opaque Command

The `reg_opaque` command registers opaque objects.

Objects containing opaque data are not necessarily interpreted by the server.

Syntax

Short form:

```
-s reg_opaque -m crypto_usage_mask -o path_to_object_file [-
f path_to_object_attr_file]
```

Long form:

```
--service reg_opaque --mask crypto_usage_mask --object path_to_object_file [--
attr path_to_object_attr_file]
```

Parameters

Parameter	Required?	Description
`-m, --mask`	Required	Cryptographic usage mask
`-o, --object`	Required	Path to object file
`-f, --attr`	Optional	Attribute list file

Short Form Example

```
java -jar okvrestservices.jar kmip -c conf_file -s reg_opaque -m ENCRYPT -o ./
my.opaque -f ./obj_attr_file.txt
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service reg_opaque --
mask ENCRYPT --object ./my.opaque  --attr ./obj_attr_file.txt
```

13.7.4.9 reg_secret Command

The `reg_secret` command registers secret data such as passwords or random seeds.

Syntax

Short form:

```
-s reg_secret -t object_type -m crypto_usage_mask -o path_to_object_file  [-
f path_to_object_attr_file]
```

Long form:

```
--service reg_secret --type object_type --mask crypto_usage_mask --
object path_to_object_file  [--attr path_to_object_attr_file]
```

Parameters

Parameter	Required?	Description
`-t`, `--type`	Required	Object type
`-m`, `--mask`	Required	Cryptographic usage mask
`-o`, `--object`	Required	Path to object attributes file
`-f`, `--attr`	Optional	Attribute list file

Short Form Example

```
java -jar okvrestservices.jar kmip -c conf_file -s reg_secret -t PASSWORD -m
ENCRYPT -o ./secret.password -f ./obj_attr_file.txt
```

Example Contents of `obj_attr_file.txt`

```
CONTACT_INFO=admin@example.com
NAME=John
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service reg_secret
--type PASSWORD --mask ENCRYPT --object ./secret.password  --attr ./
obj_attr_file.txt
```

13.7.5 Key Attribute Management Commands

The key attribute management commands perform tasks such as adding, modifying, and deleting standard and custom attributes.

You must have the Key Administrator role to use these commands.

- add_attr Command
  The `add_attr` command adds an attribute to a managed object.
- add_custom_attr Command
  The `add_custom_attr` command adds a custom attribute to an object.
- all_attr Command
  The `all_attr` command retrieves all attributes of an object.
- del_attr Command
  The `del_attr` command deletes an attribute from a managed object.
- del_custom_attr Command
  The `del_custom_attr` command deletes a custom attribute.
- get_attr Command
  The `get_attr` command retrieves an attribute or list of attributes of an object.
- list_attr Command
  The `list_attr` command retrieves a list of attributes of an object.
- mod_attr Command
  The `mod_attr` command modifies an object's attributes.
- mod_custom_attr Command
  The `mod_custom_attr` command modifies a custom attribute.

13.7.5.1 add_attr Command

The `add_attr` command adds an attribute to a managed object.

Syntax

Short form:

```
-s add_attr -u UUID -n attribute_name -v attribute_value
```

Long form:

```
--service add_attr --uid UUID --attribute attribute_name --value attribute_value
```

Parameters

Parameter	Required?	Description
`-u`, `--uuid`	Required	Universally unique ID (UUID) of the managed object
`-n`, `--attribute`	Required	Attribute name. Modifiable attributes include: • `ACTIVATION_DATE` • `CONTACT_INFO` • `DEACTIVATION_DATE` • `NAME` • `PROCESS_START_DATE` • `PROTECT_STOP_DATE`
`-v`, `--value`	Required	Attribute value. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons.

Short Form Example

These examples assign the value `okv@example.com` for the attribute `CONTACT_INFO` for the object specified by the UUID.

```
java -jar okvrestservices.jar kmip -c conf_file -s add_attr -u
7D767505-2FBE-4F5E-BF81-F95A4FE88E03 -n CONTACT_INFO -v "okv@example.com"
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service add_attr
--uid 7D767505-2FBE-4F5E-BF81-F95A4FE88E03 --attribute CONTACT_INFO --value
"okv@example.com"
```

13.7.5.2 add_custom_attr Command

The `add_custom_attr` command adds a custom attribute to an object.

Syntax

Short form:

```
-s add_custom_attr -u UUID -n {X-|Y-}attribute_name -t attribute_type -v
attribute_value
```

Long form:

```
--service add_custom_attr --uid UUID --attribute {X-|Y-}attribute_name --type
attribute_type --value attribute_value
```

Parameters

Parameter	Required?	Description
`-u`, `--uid`	Required	Universally unique ID (UUID) of the managed object
`-n`, `--attribute`	Required	Custom attribute name. Must include the prefix `X-` or `Y-`. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons.
`-t`, `--type`	Required	Object type. Can be `NUMBER` or `TEXT`.
`-v`, `--value`	Required	Custom attribute value. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons.

Short Form Example

This example assigns the `TEXT` value `OGG_MASTER_KEY_NAME` to the custom attribute named `x-OGG_MK_NAME` for the object specified by the UUID. Enclose these values in double quotation marks if they contain a space.

```
java -jar okvrestservices.jar kmip -c conf_file -s add_custom_attr
-u D69D2F32-2DBB-4FF3-BF52-95487526E6EC -n x-OGG_MK_NAME -t TEXT -v
OGG_MASTER_KEY_NAME
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service add_custom_attr
--uid D69D2F32-2DBB-4FF3-BF52-95487526E6EC --attribute x-OGG_MK_NAME --type TEXT
--value OGG_MASTER_KEY_NAME
```

13.7.5.3 all_attr Command

The `all_attr` command retrieves all attributes of an object.

Syntax

Short form:

```
-s all_attr -u UUID
```

Long form:

```
--service all_attr --uid UUID
```

Parameters

Parameter	Required?	Description
`-u`, `--uid`	Required	Universally unique ID (UUID) of the managed object

Short Form Example

These examples list all attribute values for the object specified by the UUID.

```
java -jar okvrestservices.jar kmip -c conf_file -s all_attr -u
D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service all_attr --uid
D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

13.7.5.4 del_attr Command

The `del_attr` command deletes an attribute from a managed object.

Syntax

Short form:

```
-s del_attr -u UUID -n attribute_name
```

Long form:

```
--service del_attr --uid UUID --attribute attribute_name
```

Parameters

Parameter	Required?	Description
`-u`, `--uuid`	Required	Universally unique ID (UUID) of the managed object
`-n`, `--attribute`	Required	Attribute name. Modifiable attributes include: • `ACTIVATION_DATE` • `CONTACT_INFO` • `DEACTIVATION_DATE` • `NAME` • `PROTECT_STOP_DATE`

Short Form Example

These examples delete the attribute `CONTACT_INFO` for the object specified by the UUID.

```
java -jar okvrestservices.jar kmip -c conf_file -s del_attr -u
7D767505-2FBE-4F5E-BF81-F95A4FE88E03 -n CONTACT_INFO
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service del_attr --uid
7D767505-2FBE-4F5E-BF81-F95A4FE88E03 --attribute CONTACT_INFO
```

13.7.5.5 del_custom_attr Command

The `del_custom_attr` command deletes a custom attribute.

Syntax

Short form:

```
-s del_custom_attr -u UUID -n attribute_name -i attribute_index
```

Long form:

```
--service del_custom_attr --uid UUID --attribute attribute_name --index
attribute_index
```

Parameters

Parameter	Required?	Description
`-u`, `--uid`	Required	Universally unique ID (UUID) of the managed object
`-n`, `--attribute`	Required	Attribute name. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons.
`-i`, `--index`	Required	Attribute index

Short Form Example

These examples delete the value for the attribute named `x-OGG_MK_NAME` identified by the index `1` for the object specified by the UUID.

```
java -jar okvrestservices.jar kmip -c conf_file -s del_custom_attr -u
D69D2F32-2DBB-4FF3-BF52-95487526E6EC -n x-OGG_MK_NAME -i 1
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service del_custom_attr
--uid D69D2F32-2DBB-4FF3-BF52-95487526E6EC --attribute x-OGG_MK_NAME --index 1
```

13.7.5.6 get_attr Command

The `get_attr` command retrieves an attribute or list of attributes of an object.

To find the available attributes, use the `list_attr` command.

Syntax

Short form:

```
-s get_attr -u UUID -n attribute name or list
```

Long form:

```
--service get_attr --uid UUID --attribute attribute name or list
```

Parameters

Parameter	Required?	Description
`-u, --uid`	Required	Universally unique ID (UUID) of the managed object
`-n, --attribute`	Required	Attribute name or a list of attributes in quotes separated by commas

Short Form Example

These examples list the attribute values for `CRYPTO_USAGE_MASK`, `CONTACT_INFO` , and `NAME` for the object specified by the UUID.

```
java -jar okvrestservices.jar kmip -c conf_file -s get_attr -u
D69D2F32-2DBB-4FF3-BF52-95487526E6EC -n "CRYPTO_USAGE_MASK,CONTACT_INFO,NAME"
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --
service get_attr --uid D69D2F32-2DBB-4FF3-BF52-95487526E6EC --attribute
"CRYPTO_USAGE_MASK,CONTACT_INFO,NAME"
```

13.7.5.7 list_attr Command

The `list_attr` command retrieves a list of attributes of an object.

Syntax

Short form:

```
-s list_attr -u UUID
```

Long form:

```
--service list_attr --uid UUID
```

Parameters

Parameter	Required?	Description
`-u, --uuid`	Required	Universally unique ID (UUID) of the managed object

Short Form Example

These examples list all of the attribute values for the object specified by the UUID.

```
java -jar okvrestservices.jar kmip -c conf_file -s list_attr -u
D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service list_attr --uid
D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

13.7.5.8 mod_attr Command

The `mod_attr` command modifies an object's attributes.

Syntax

Short form:

```
-s mod_attr -u UUID -n attribute_name -v attribute_value
```

Long form:

```
--service mod_attr --uid UUID --attribute attribute_name --value
attribute_value
```

Parameters

Parameter	Required?	Description
`-u`, `--uid`	Required	Universally unique ID (UUID) of the managed object
`-n`, `--attribute`	Required	Attribute name. Modifiable attributes include: • `ACTIVATION_DATE` is the date and time when the managed object can first be used. • `CONTACT_INFO` is used for contact purposes only and not policy enforcement. • `DEACTIVATION_DATE` is the date and time when the managed object can no longer be used for any purpose. • `NAME` is a user-friendly name provided to identify and locate an object. • `PROCESS_START_DATE` is the date and time when a key object can first be used to process cryptographically protected data. • `PROTECT_STOP_DATE` is the date and time after which a key object will be used for applying cryptographic protection.
`-v`, `--value`	Required	Attribute value. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons.

Short Form Example

These examples modify the value for attribute `PROCESS_START_DATE` for the object specified by the UUID.

```
java -jar okvrestservices.jar kmip -c conf_file -s mod_attr
-u 7D767505-2FBE-4F5E-BF81-F95A4FE88E03 -n PROCESS_START_DATE -v
"2030/10/1010:10:10"
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service mod_attr --
uid 7D767505-2FBE-4F5E-BF81-F95A4FE88E03 --attribute PROCESS_START_DATE -value
"2030/10/1010:10:10"
```

13.7.5.9 mod_custom_attr Command

The `mod_custom_attr` command modifies a custom attribute.

Syntax

Short form:

```
-s mod_custom_attr -u UUID -n attribute_name -i attribute_index -v
attribute_value
```

Long form:

```
--service mod_custom_attr --uid UUID --attribute attribute_name --index
attribute_index --value attribute_value
```

Parameters

Parameter	Required?	Description
`-u`, `--uuid`	Required	Universally unique ID (UUID) of the managed object
`-n`, `--attrubute`	Required	Attribute name. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons.
`-i`, `--index`	Required	Attribute index. The attribute index is a unique position for each attribute. This enables you to distinguish between attributes that have the same name with the index.
`-v`, `--value`	Required	Attribute value. Enclose this value in double quotation marks if the value contains spaces, slashes, or colons.

Short Form Example

This examples changes the value for the custom attribute named `x_OGG_MK_NAME` to `OGG_MASTER_KEY_NAME2` for the object specified by the UUID.

```
java -jar okvrestservices.jar kmip -c conf_file -s mod_custom_attr
-u D69D2F32-2DBB-4FF3-BF52-95487526E6EC -n x-OGG_MK_NAME -i 1 -v
OGG_MASTER_KEY_NAME2
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service mod_custom_attr
--uid D69D2F32-2DBB-4FF3-BF52-95487526E6EC --attribute x-OGG_MK_NAME --index 1
--value OGG_MASTER_KEY_NAME2
```

13.7.6 Key Life Cycle Management Commands

The key life management commands perform tasks such as activating and revoking managed cryptographic objects.

You must have the Key Administrator role to use these commands.

- activate Command
  The `activate` command activates a managed cryptographic object.

- destroy Command
  The `destroy` command destroys a managed object.
- locate Command
  The `locate` command locates managed objects.
- revoke Command
  The `revoke` command revokes a managed object such as a key or a certificate.
- query Command
  The `query` command identifies supported operations and objects.

13.7.6.1 activate Command

The `activate` command activates a managed cryptographic object.

Syntax

Short form:

```
-s activate -u unique_identifier
```

Long form:

```
--service activate --uid  unique_identifier
```

Parameters

Parameter	Required?	Description
`-u`, `--uid`	Required	Unique identifier

Short Form Example

This example activates a managed object.

```
java -jar okvrestservices.jar kmip -c conf_file -s activate --uid
7D767505-2FBE-4F5E-BF81-F95A4FE88E03
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service activate --uid
7D767505-2FBE-4F5E-BF81-F95A4FE88E03
```

13.7.6.2 destroy Command

The `destroy` command destroys a managed object.

Syntax

Short form:

```
-s destroy -u unique_identifier
```

Long form:

```
--service destroy --uid unique_identifier
```

Parameters

Parameter	Required?	Description
`-u, --uid`	Required	Unique identifier

Short Form Example

This example destroys a managed object.

```
java -jar okvrestservices.jar kmip -c conf_file -s destroy -u D0ABC5A5-BB30-4F20-
BFE2-54E3044F5296
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service destroy --uid
D0ABC5A5-BB30-4F20-BFE2-54E3044F5296
```

13.7.6.3 locate Command

The `locate` command locates managed objects.

Syntax

Short form:

```
-s locate [-x max_uid_# -g group_member –y object_state]
```

Long form:

```
--service locate [--max  max_uid_#  --group value –-state object_state]
```

Parameters

Parameter	Required?	Description
`-x, --max`	Optional	Unique identifier
`-g, --group`	Optional	Group member
`-y, --state`	Optional	Object state. For a list of valid object state values, see KMIP REST service options that are available when you use `OKVRESTSERVICE`.

Short Form Example

These examples locate an object based on the specified search criteria.

```
java -jar okvrestservices.jar kmip -c conf_file -s locate

java -jar okvrestservices.jar kmip -c conf_file -s locate -x 50
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service locate --state
ACTIVE --group FRESH

java -jar okvrestservices.jar kmip --config conf_file --service locate --state
ACTIVE --group DEFAULT --max 1
```

Related Topics

- Oracle Key Vault Key Management REST Client API Using OKVRESTSERVICE
  To call the KM REST client API using `OKVRESTSERVICE` (`okvrestservice.jar`), you must include the `kmip` option.

13.7.6.4 revoke Command

The `revoke` command revokes a managed object such as a key or a certificate.

Syntax

Short form:

```
-s revoke -u unique_identifier -r revoke_code [-s revoke_reason]
```

Long form:

```
--service revoke --uid unique_identifier --code revoke_code [--reason
revoke_reason]
```

Parameters

Parameter	Required?	Description
`-u`, `--uid`	Required	Unique identifier
`-q`, `--code`	Required	Revoke code
`-r`, `--reason`	Optional	Reason to revoke. When the revoke code is `KEY_COMPROMISE`, then you must provide compromise occurrence date by using the `--compromise_date` parameter. For a complete list of valid reasons, see the KMIP REST service options that are available when you use `OKVRESTSERVICE`.

Short Form Example

This example revokes a managed object.

```
java -jar okvrestservices.jar kmip -c conf_file -s revoke -u 7D767505-2FBE-4F5E-
BF81-F95A4FE88E03 -r CA_COMPROMISE -s "security problem"
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service revoke --
uid 7D767505-2FBE-4F5E-BF81-F95A4FE88E03 --code CA_COMPROMISE --reason "security
problem"
```

Related Topics

- Oracle Key Vault Key Management REST Client API Using OKVRESTSERVICE
  To call the KM REST client API using `OKVRESTSERVICE` (`okvrestservice.jar`), you must include the `kmip` option.

13.7.6.5 query Command

The `query` command identifies supported operations and objects.

Syntax

Short form:

```
-s query
```

Long form:

```
--service query
```

Short Form Example

This example queries the server for supported operations and objects.

```
java -jar okvrestservices.jar kmip -c conf_file -s query
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service query
```

13.7.7 Wallet Commands

The wallet commands control wallet memberships.

- add_member Command
  The `add_member` command add a wallet membership.
- del_member Command
  The `del_member` command deletes a wallet membership.
- list_wallet Command
  The `list_wallet` command lists wallets that are managed by the endpoint used to connect to Oracle Key Vault.

13.7.7.1 add_member Command

The `add_member` command add a wallet membership.

Syntax

Short form:

```
-s add_member -w wallet -u UUID
```

Long form:

```
--service add_member --wallet wallet --uid UUID
```

Parameters

Parameter	Required?	Description
`-u, --uid`	Required	Universally unique ID (UUID) of the managed object
`-w, --wallet`	Required	Wallet name

Short Form Example

```
java -jar okvrestservices.jar kmip -c conf_file -s add_member -w WALLET -u
D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service add_member --
wallet WALLET --uid D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

13.7.7.2 del_member Command

The `del_member` command deletes a wallet membership.

Syntax

Short form:

```
-s del_member -w wallet -u UUID
```

Long form:

```
--service del_member --wallet wallet --uid UUID
```

Parameters

Parameter	Required?	Description
`-u, --uid`	Required	Universally unique ID (UUID) of the managed object
`-w, --wallet`	Required	Wallet name

Short Form Example

```
java -jar okvrestservices.jar kmip -c conf_file -s del_member -w WALLET -u
D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service del_member --
wallet WALLET --uid D69D2F32-2DBB-4FF3-BF52-95487526E6EC
```

13.7.7.3 list_wallet Command

The `list_wallet` command lists wallets that are managed by the endpoint used to connect to Oracle Key Vault.

Syntax

Short form:

```
-s list_wallet
```

Long form:

```
--service list_wallet
```

Short Form Example

```
java -jar okvrestservices.jar kmip -c conf_file -s list_wallet
```

Long Form Example

```
java -jar okvrestservices.jar kmip --config conf_file --service list_wallet
```

14

Backup and Restore Operations

You may configure automatic backups for continuous, reliable, and protected access to security objects with minimum downtime.

- About Backing Up and Restoring Data in Oracle Key Vault
  You can use Oracle Key Vault to back up and restore Oracle Key Vault data.
- Oracle Key Vault Backup Destinations
  A backup destination is the location where Oracle Key Vault data will be copied to and stored.
- Backup Schedules and States
  Oracle Key Vault provides backup schedule types depending on the backup destination, and different states that indicate the progress of the backup activity.
- Scheduling and Managing Oracle Key Vault Backups
  You can schedule Oracle Key Vault backups to specific backup destinations and times.
- Restoring Oracle Key Vault Data
  Oracle Key Vault data from a remote backup destination can be restored onto a another Oracle Key Vault server.
- Backup and Restore Best Practices
  Oracle provides best practices to keep backups current so that you can recover from catastrophic failures with minimum down time and data loss.

14.1 About Backing Up and Restoring Data in Oracle Key Vault

You can use Oracle Key Vault to back up and restore Oracle Key Vault data.

You should 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 server from a backup.

Backup and restore operations may be performed from the Oracle Key Vault management console. You must be a user who has the System Administrator role to back up and restore Oracle Key Vault data. You can schedule backups at periodic intervals to run automatically at designated times. You also can run these operations on-demand to save a current snapshot of the system.

Oracle strongly recommends that you back up Oracle Key Vault data regularly on a schedule. This practice ensures that backups are current and hold the most recent data. You can use this backup to restore a new or existing Oracle Key Vault server and be fully operational with minimum downtime and data loss.

Oracle 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.

In an Oracle Key Vault multi-master cluster environment, the replication intrinsically creates copies of the data in the cluster. You can perform backups and backup related operations on individual Oracle Key Vault servers, or on any read-write node in a multi-master cluster. These operations are, however, not allowed on the primary in a primary-standby environment. In addition, you cannot restore a backup to a node in the cluster. Therefore, backups in a cluster are taken for disaster recovery in case of a complete cluster failure, and should be normally kept remote from the cluster nodes.

14.2 Oracle Key Vault Backup Destinations

A backup destination is the location where Oracle Key Vault data will be copied to and stored.

- About the Oracle Key Vault Backup Destination
  The backup destination enables the backup data to be available in a location other than the Oracle Key Vault server itself.
- Creating a Remote Backup Destination
  You can use the Oracle Key Vault management console to create a remote backup destination.
- Changing Settings on a Remote Backup Destination
  After you have created the backup destination, you can only change the SCP port number and details of the user account.
- Deleting a Remote Backup Destination
  You can delete a remote backup destination to stop future backups to that destination server.

14.2.1 About the Oracle Key Vault Backup Destination

The backup destination enables the backup data to be available in a location other than the Oracle Key Vault server itself.

This ensures that you have all the relevant data to recover in case of a catastrophic failure with the Oracle 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.

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.

Oracle Key Vault provides two types of backup destinations: local and remote. The local backup destination resides on the Oracle 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.

Local and remote backup destinations have the following characteristics:

- **Local backup destinations:** The local backup destination, `LOCAL`, is present by default and cannot be removed.

  Backups to `LOCAL` are useful to save a current state of Oracle Key Vault. Since these backups are stored in Key Vault, they will be lost in case of a failover or switchover in a primary-standby deployment. Therefore, you should 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.

- **Remote backup destinations:** 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 Oracle Key Vault maintains at the backup destination. The `okvbackup.mgr` file catalogs the backups performed and is used to restore data.

> **Note:**
>
> You cannot use another Oracle Key Vault server 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 Oracle Key Vault servers as backup destinations, because backups that happen concurrently from different Oracle Key Vault servers will overwrite each other's catalog file, with the result that Oracle Key Vault may not be able to locate the backups correctly.
> - After you restore a backup that contains a remote backup destination, do not continue to use that remote backup destination. Delete any backup jobs that are configured to send backups to that destination. Continuing to use this backup destination could corrupt the backup catalog file. Oracle Key Vault may not be able to locate backups correctly.
> - Configure each node in a multi-master cluster to send their backups to a different backup destination.

Related Topics

- Types of Oracle Key Vault Backups
  Oracle Key Vault provides two types of backup jobs that can be scheduled: one-time backups, and periodic backups.

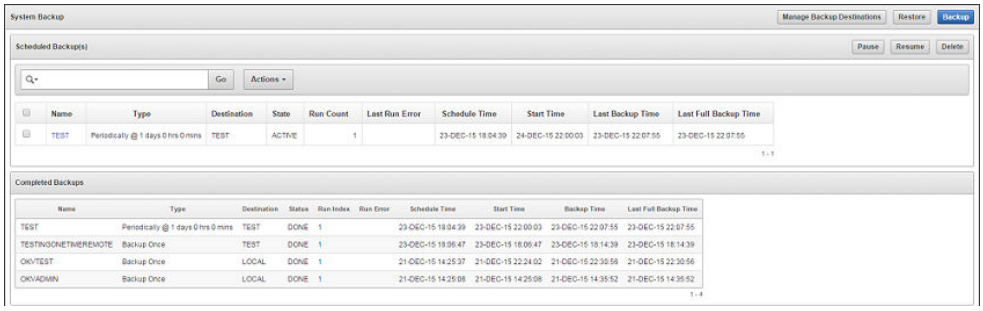
14.2.2 Creating a Remote Backup Destination

You can use the Oracle Key Vault management console to 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.

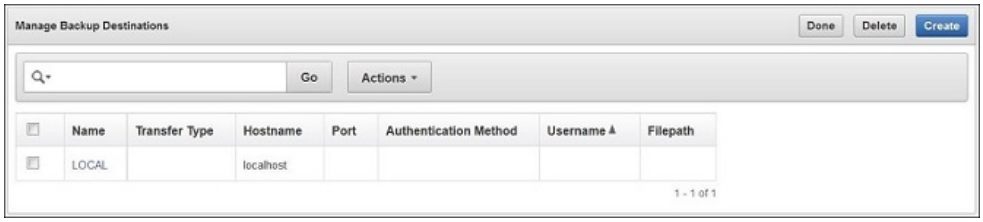
1. Log in to the Oracle Key Vault management console as a user with the System Administrator role.
2. Select the **System** tab, then click **System Backup** from the left sidebar.

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



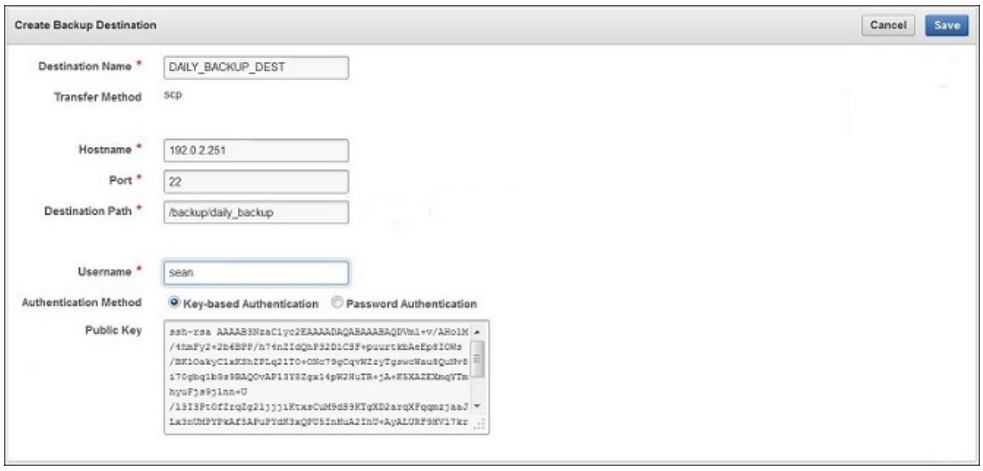
3. Click **Manage Backup Destinations**.

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



4. Click **Create**.

   The Create Backup Destination page appears.



5. Enter the following information for the backup location:

   - **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 host name or IP address of the remote server for the backup. If you enter the host name, then ensure that DNS is configured to translate the host name to its corresponding IP address. Do not include spaces, single quotation marks, or double quotation marks in a host name that is in a remote backup destination.
   - **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. You cannot modify this directory location after the backup destination is created. This path must not be the destination for backups from another Oracle Key Vault server. Do not include spaces, single quotation marks, or double quotation marks destination path that is in a remote backup destination.
   - **Username** Enter the user name 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. Do not include spaces, single quotation marks, or double quotation marks in a user name that is in a remote backup destination.
   - **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. Be aware that certain events may trigger a change of the public key, which means that Oracle Key Vault cannot use the backup destination until the new public key is re-copied from Oracle Key Vault to the appropriate configuration file. These events include but are not limited to certificate rotation, changing the IP address, and conversion to a cluster node.

6. Click **Save**.

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

14.2.3 Changing Settings on a Remote Backup Destination

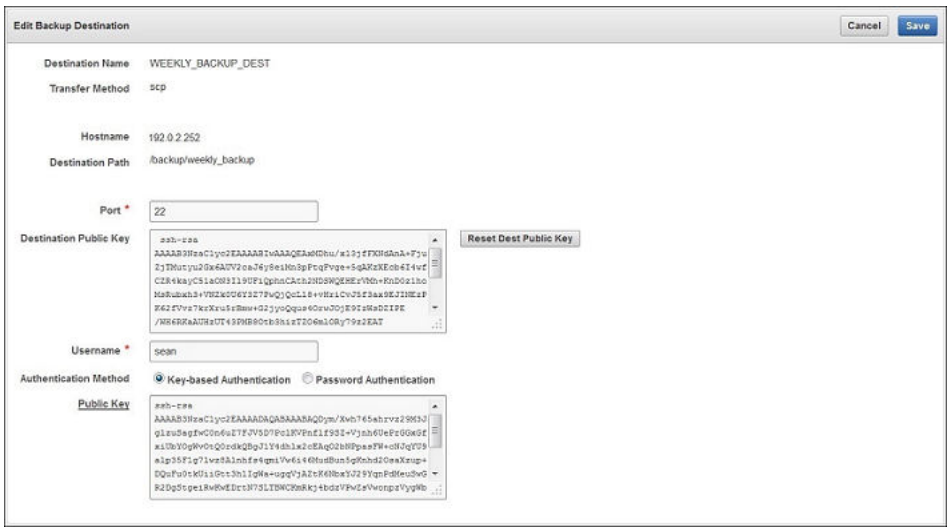
After you have created the backup destination, you can only change the SCP port number and details of the user account.

You cannot change any other setting.

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:
   - **Port:** Change the default port number running SCP on the external server.
   - **Username:** Enter the user name of the user account on the remote server. Ensure that the new user has write permissions on the directory that is specified in **Destination Path**, because this path cannot 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. Be aware that certain events may trigger a change of the public key, which means that Oracle Key Vault cannot use the backup destination until the new public key is re-copied from Oracle Key Vault to the appropriate configuration file. These events include but are not limited to certificate rotation, changing the IP address, and conversion to a cluster node.

6. Click **Save**.

Oracle Key Vault validates the input that you supplied to update the backup destination. If the validation fails, then the backup destination is not created. If this happens, then check values for the user account on the remote server (user name and password) and ensure that the directory has write permissions for the user. Finally, ensure that the remote server is active.

14.2.4 Deleting 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.

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 the remote backup destinations.

4. Check the boxes for the backup destinations that you want to delete.
5. Click **Delete**.

14.3 Backup Schedules and States

Oracle Key Vault provides backup schedule types depending on the backup destination, and different states that indicate the progress of the backup activity.

- About Backup Schedule Types and States
  You can schedule backups in Oracle Key Vault for specific times and backup destinations.
- Types of Oracle Key Vault Backups
  Oracle Key Vault provides two types of backup jobs that can be scheduled: one-time backups, and periodic backups.

- Scheduled Backup States in Oracle Key Vault
  Scheduled backups have four states, which indicate whether the backup is scheduled, in progress, completed, or paused.

14.3.1 About Backup Schedule Types and States

You can schedule backups in Oracle 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 Oracle Key Vault while the backup is in progress.

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

14.3.2 Types of Oracle Key Vault Backups

Oracle Key Vault provides two types of backup jobs that can be scheduled: one-time backups, and periodic backups.

- **One-time backup:** A one-time backup makes a full backup of the Oracle Key Vault system. You can schedule multiple one-time backup jobs, each with its own start time.

  You should make one-time local backups before making significant configuration changes to Oracle 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.

- **Periodic backup:** The periodic backup process first makes a full backup of the Oracle Key Vault system and puts the backup schedule in active state. 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.

  You should schedule periodic backups 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.

Related Topics

- Scheduled Backup States in Oracle Key Vault
  Scheduled backups have four states, which indicate whether the backup is scheduled, in progress, completed, or paused.

14.3.3 Scheduled Backup States in Oracle Key Vault

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

- **ACTIVE:** The backup is scheduled and will be processed at the specified start time or period.
- **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 Oracle Key Vault and the remote destination is broken
  - If the remote destination is unavailable or inactive
  - If you want to defer the backup

  You can delete the scheduled backups that have not completed.
- **ONGOING:** The backup is in progress.
- **DONE:** The backup is complete.

14.4 Scheduling and Managing Oracle Key Vault Backups

You can schedule Oracle Key Vault backups to specific backup destinations and times.

You must create the backup destinations that you will use beforehand, and you can modify or delete backup schedules.

- Scheduling a Backup for Oracle Key Vault
  You can schedule a one-time or a periodic backup to a local or remote backup destination.
- Changing a Backup Schedule for Oracle Key Vault
  You cannot change the schedule of a backup in progress.
- Deleting a Backup Schedule from Oracle Key Vault
  You can delete a backup schedule from the Oracle Key Vault management console.
- How Primary-Standby Affects Oracle Key Vault Backups
  In a primary-standby deployment, you must perform backups on the primary server.
- Protecting the Backup Using the Recovery Passphrase
  Oracle Key Vault uses the recovery passphrase to control who can restore user and system data.

14.4.1 Scheduling a Backup for Oracle Key Vault

You can schedule a one-time or a periodic backup to a local or remote backup destination.

You can start a one-time backup to start immediately without setting a time. However, do not schedule backup operations if a certificate rotation operation is in progress.

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. If you want to perform a periodic backup, then do the following:
   a. In the **Start Time** field, use the Calendar icon to specify the start time for the backup.
   b. For **Type**, select **PERIODIC**.

      The additional fields **Days**, **Hours**, and **Mins** appear.
   c. In the **Days**, **Hours**, and **Mins** fields, enter the times for the periodic backups to occur.
   d. For **Destination**, select the destination backup from the list.
   e. Click **Schedule** to add the scheduled backup to the Scheduled Backup(s) page.
6. If you want to perform a one-time backup, then do the following:
   a. In the **Start Time** field, use the Calendar icon to specify the start time for the backup. If you want the backup to perform immediately after you click **Schedule**, then select **Now**.
   b. For **Type**, select **ONE-TIME**.
   c. For **Destination**, select the destination backup from the list.
   d. Click **Schedule** to add the scheduled backup to the Scheduled Backup(s) page.

14.4.2 Changing a Backup Schedule for Oracle Key Vault

You cannot change the schedule of a backup in progress.

To change the backup schedule the state must be active or paused.

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.

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

   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**.

14.4.3 Deleting a Backup Schedule from Oracle Key Vault

You can delete a backup schedule from the Oracle Key Vault management console.

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 boxes of scheduled backups listed in **Scheduled Backup(s)**.
4. Click **Delete** to delete the selected backup schedules.

14.4.4 How Primary-Standby Affects Oracle Key Vault Backups

In a primary-standby deployment, you must perform backups on the primary server.

Because the standby synchronizes its state with the primary, you do not need to back up the standby.

Be aware of the following behavior for failover or switchover operations in a primary-standby deployment:

- Any backups in progress will terminate if there is a failover or a primary-standby switchover. Backups to `LOCAL` are private to the Oracle Key Vault server and therefore the local backup on the primary server 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.

14.4.5 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.

Related Topics

- 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.

14.5 Restoring Oracle Key Vault Data

Oracle Key Vault data from a remote backup destination can be restored onto a another Oracle Key Vault server.

This restore operation minimizes downtime and data loss.

- About the Oracle Key Vault Restore Process
  The restore process replaces all data on the new server except the `root` and `support` user passwords.
- Procedure for Restoring Oracle Key Vault Data
  You can store Oracle Key Vault data using the Oracle Key Vault management console.
- Multi-Master Cluster and the Restore Operation
  In a multi-master cluster deployment, you must consider several factors before you restore data to Oracle Key Vault.
- Primary-Standby and the Restore Operation
  In a primary-standby deployment, you must consider several factors before you restore data to Oracle Key Vault.
- 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 server from this backup.
- Changes Resulting from a System State Restore
  Restoring an Oracle Key Vault server brings the system state back to the time when the backup last performed.

14.5.1 About the Oracle Key Vault Restore Process

The restore process replaces all 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 Oracle Key Vault data to a server only after ensuring that all scheduled backups on the server are completed.

Restoring data to an Oracle Key Vault server replaces the data in the server with that of the backup. Any changes made since the last backup will be lost. Backups can only be restored to the same version of Oracle Key Vault at which the backup was taken.

The maximum life of a backup is 1 year.

> **Note:**
>
> 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 Oracle Key Vault, then you must use the recovery passphrase that you created during the post-installation process.

Restoring data in Oracle Key Vault entails the following general steps:

1. Setting up the backup environment, which includes, after install Oracle Key Vault, configuring backup destinations.
2. Performing the restore operation by determining the backup to use from a local or remote backup destination, and then providing the recovery passphrase to begin the restore process. You create the recovery passphrase as part of the post-installation tasks for Oracle Key Vault.

Related Topics

- Performing Post-Installation Tasks
  After you install Oracle Key Vault, you must complete a set of post-installation tasks.

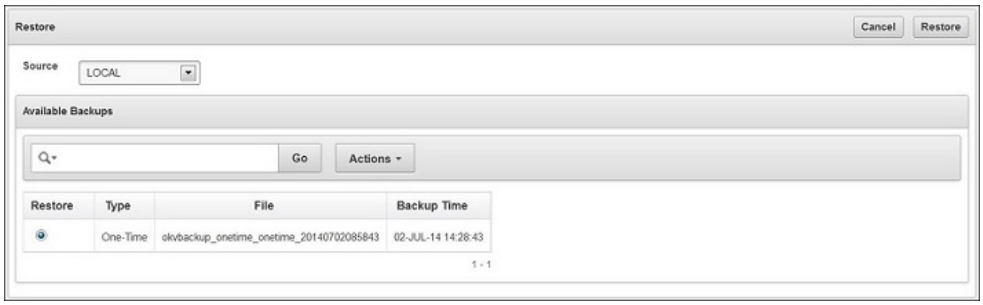
14.5.2 Procedure for Restoring Oracle Key Vault Data

You can store Oracle Key Vault data using the Oracle Key Vault management console.

Before you restore, ensure that you have the correct recovery passphrase. You will need to enter this passphrase during the restore process. In addition, do not perform a restore operation while a certificate rotation process is in progress.

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 the names of configured remote destinations.
5. Select **Restore** next to the backup you want to restore from.
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 restart.
8. Delete any paused periodic backup jobs and then re-create them, using a new backup destination.

   Oracle recommends that you delete such jobs in the event that the backup manager file may be corrupted.
9. If your site uses the Commercial National Security Algorithm (CNSA) suite, then re-install these algorithms on the Oracle Key Vault server after the restore operation is complete.

Related Topics

- Performing Backup and Restore Operations with CNSA
  After you back up and restore Oracle Key Vault, use `/usr/local/okv/bin/okv_cnsa` to use the enhanced Commercial National Security Algorithm (CNSA) Suite.

14.5.3 Multi-Master Cluster and the Restore Operation

In a multi-master cluster deployment, you must consider several factors before you restore data to Oracle Key Vault.

- You must restore only if all nodes in the cluster are lost.
- You must restore the backup to a standalone Oracle Key Vault server that has the same IP address as the node from which the backup was taken. Not doing so may affect the ability of endpoints to connect to the restored backup.
- The data restored is only as current as the backup.

- After the restore operation, you must now use the restored server as the first node of a new cluster.

14.5.4 Primary-Standby and the Restore Operation

In a primary-standby deployment, you must consider several factors before you restore data to Oracle Key Vault.

- You must perform the restore operation only if both the primary and standby data are lost.
- You must restore the backup on a standalone Oracle Key Vault server only, even if the backup was taken from the primary.
- The restore operation replaces the Oracle Key Vault server with the backup. This means that some data can be lost. You might need to restore the endpoint database.
- If you restore a backup taken from the primary node, then you must discard (or reinstall) the standby server and configure a new standby.
- 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.
- If your site uses the Commercial National Security Algorithm (CNSA) suite, then you must re-install these algorithms on the Oracle Key Vault server after the restore operation is complete.

Related Topics

- Performing Backup and Restore Operations with CNSA
  After you back up and restore Oracle Key Vault, use `/usr/local/okv/bin/okv_cnsa` to use the enhanced Commercial National Security Algorithm (CNSA) Suite.

14.5.5 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 server from this backup.

You must re-install the third-party certificate on the new server in order to use it.

If you perform a certificate rotation in your configuration (so that the endpoints have been updated to using new certificates as well), but then restore the Oracle Key Vault server from a backup that was taken **before** the certificate rotation was performed, then the endpoints will not be able to connect to the restored Oracle Key Vault system.

Related Topics

- Managing Console Certificates
  You can use the Oracle Key Vault management console to manage console certificates.

14.5.6 Changes Resulting from a System State Restore

Restoring an Oracle Key Vault server brings the system state back to the time when the backup last performed.

Therefore, any changes that were 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 Oracle Key Vault server has been restored and is functional, you can continue to back up Oracle 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 and backup the Oracle Key Vault.

14.6 Backup and Restore Best Practices

Oracle provides best practices to keep backups current so that you can recover from catastrophic failures with minimum down time and data loss.

- Ensure that the recovery passphrase at the time of backup is accessible because you will need it to restore data from a backup.
- Back up data any time you change the recovery passphrase.
- Ensure that you create at least one remote backup destination in a primary-standby deployment. Because the local backup resides on the Oracle Key Vault server itself, it will be lost in a failover or switchover situation.
- Do not edit or delete the backup catalog file that is 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.
- If you use the same remote server for multiple backup destinations, then 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, then the backup catalog file will be overwritten during subsequent backups and become unusable.
- When you restore a backup, do so to a standalone Oracle Key Vault server that has the same IP address as the Oracle Key Vault server on which the backup was taken. Failing to do so may result in endpoints not being able to connect to the restored backup.

- Before you restore data, ensure that all scheduled backups are complete.
- To create remote backup destinations successfully:
  - Ensure that the servers used as remote backup destinations are enabled and active.
  - Ensure that there is connectivity between Oracle Key Vault and remote server that you plan to use as a backup destination.
  - Ensure that the remote server designated as a backup destination supports the secure copy protocol (SCP).
  - Validate the user account credentials on the remote server before you create the backup destination on Oracle Key Vault.
  - Ensure that the destination directory has write permissions.
  - Create more than one remote backup destination on multiple servers for redundancy.
  - Ensure that the 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.
- Perform a one-time backup once every seven days.
- Schedule a periodic backup with a period of one day. This ensures that you have a full backup once in seven days.
- Perform a local one-time backup before system changes. You can use this backup as a restore point.
- Backup before and after upgrading Oracle Key Vault server software.
- Change the backup destination after each upgrade. If at all possible do not reuse the backup destination.

15

Oracle Key Vault General System Administration

General system administration refers to system management tasks for the Oracle Key Vault system, such as configuring network details and services.

- Overview of Oracle Key Vault General System Administration
  System administrators can perform most general administration tasks in the Oracle Key Vault management console, including finding the current status of the overall system.
- Configuring Oracle Key Vault in a Non-Multi-Master Cluster Environment
  On the system Settings page, you can configure the network settings for either a standalone environment or a primary-standby environment.
- Configuring Oracle Key Vault in a Multi-Master Cluster Environment
  When you configure Oracle Key Vault in a multi-master cluster environment, you can configure either individual nodes or the entire multi-master cluster environment.
- Managing System Recovery
  System recovery includes tasks such as recovering lost administrative passwords.
- Support for a Primary-Standby Environment
  To ensure that Oracle Key Vault can always access security objects, you can deploy Oracle Key Vault in a primary-standby (highly available) configuration.
- Commercial National Security Algorithm Suite Support
  You can use scripts to perform Commercial National Security Algorithm (CNSA) operations for Oracle Key Vault HSM backup and upgrade operations.
- Minimizing Downtime
  Business-critical operations require data to be accessible and recoverable with minimum downtime.

15.1 Overview of Oracle Key Vault General System Administration

System administrators can perform most general administration tasks in the Oracle Key Vault management console, including finding the current status of the overall system.

- About Oracle Key Vault General System Administration
  System administrators configure the Oracle Key Vault system settings.
- Viewing the Oracle Key Vault Dashboard
  The dashboard presents the current status of the Oracle Key Vault at a high level and is visible to all users.

- Using the Status Panes in the Dashboard
  The status panes on the dashboard provide useful high level information, such as links to alerts and an overview of current user activity.

15.1.1 About Oracle Key Vault General System Administration

System administrators configure the Oracle Key Vault system settings.

The Oracle Key Vault system settings include administration, local and remote monitoring, email notification, backup and recovery operations, and auditing. You must have the appropriate role for performing these tasks. Users who have the System Administrator role can perform most of the administrative tasks, and users with the Audit Manager role can configure audit settings and export audit records. In most cases, you will perform these tasks in the Oracle Key Vault management console.

To quickly find information about the current status of the Oracle Key Vault system, you can view the Oracle Key Vault dashboard.

Related Topics

- 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.
- Managing Oracle Key Vault Endpoints
  Oracle Key Vault endpoints are computer systems like database or application servers, where keys and credentials are used to access data.
- Managing Oracle Key Vault Virtual Wallets and Security Objects

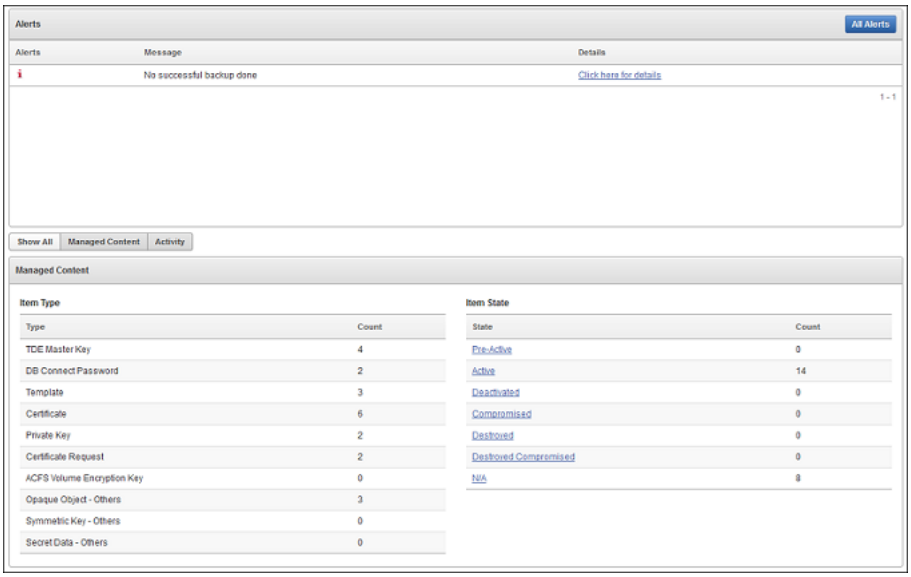
15.1.2 Viewing the Oracle Key Vault Dashboard

The dashboard presents the current status of the Oracle Key Vault at a high level and is visible to all users.

The **Home** tab of the management console displays the dashboard when you log into the management console.

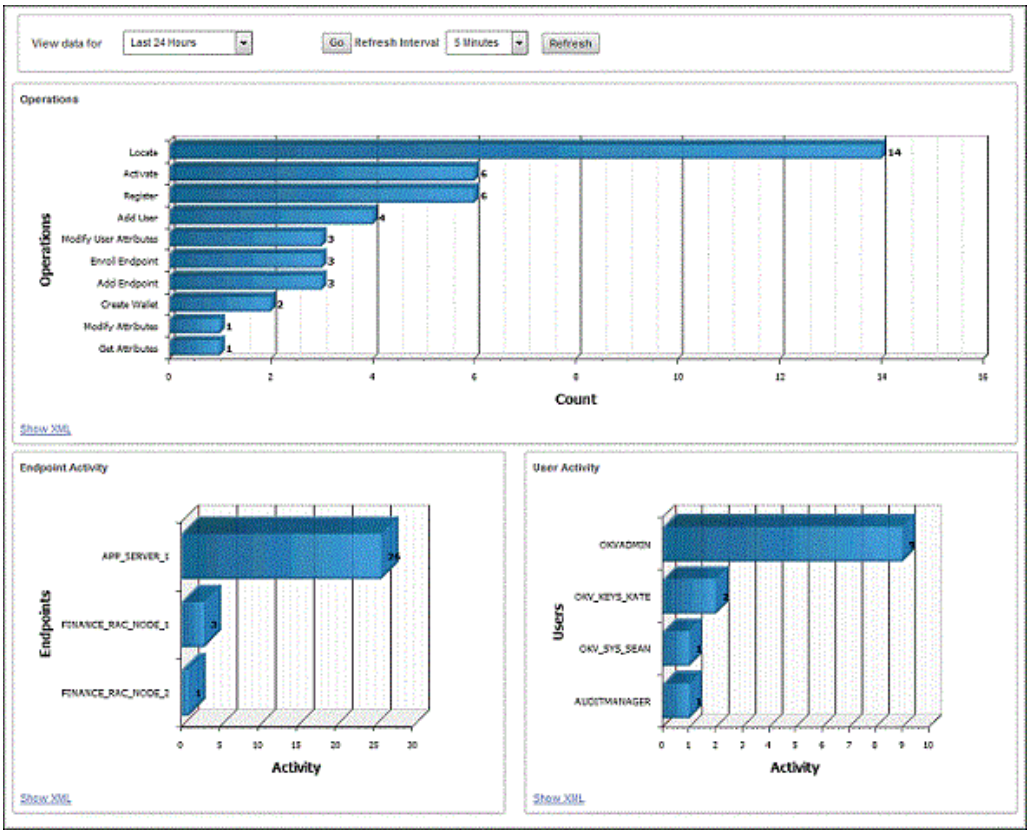
Alerts and **Managed Content** are the first sections you will see on logging in.

Figure 15-1 Alerts and Managed Content Panes



The **Data Interval**, **Operations**, **Endpoint Activity**, and **User Activity** panes of the **Home** page follow **Alerts** and **Managed Content**.

Figure 15-2 Data Interval, Operations, Endpoint Activity, and User Activity Panes



15.1.3 Using the Status Panes in the Dashboard

The status panes on the dashboard provide useful high level information, such as links to alerts and an overview of current user activity.

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

   The dashboard appears in the **Home** tab.

2. To take corrective action on a particular alert:

   a. Click the link in the Details column that corresponds to the alert. The appropriate page appears.

   b. Take the corrective action for the alert as necessary.

3. To configure the alerts that you want to see on the dashboard:

   a. Click the **Reports** tab, and then click **Alerts** from the left side bar to display the Alerts page.

   b. Click **Configure** from the top right, or **Configure Alerts** from the left sidebar under **ALERTS**, to display the Configure Alerts page.

   c. Select the **Alert Type** and then click **Save**.

4. To view managed content, click the **Managed Content** button, which appears below the Alerts pane, along with the **Show All** and **Activity** buttons.

   The **Managed Content** pane of the dashboard displays aggregated information about security objects that are 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 encryption 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.

5. To view information about from a specific time (data interval) about operations and endpoint activity, click the **Show All** button.

   Data Interval: This pane shows the length of the time period. You can set the time period to **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 panes.

   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.

Related Topics

- Searching for Security Object Items
  You can search for individual security objects if you have privileges to view these objects.

15.2 Configuring Oracle Key Vault in a Non-Multi-Master Cluster Environment

On the system Settings page, you can configure the network settings for either a standalone environment or a primary-standby environment.

- Configuring the Network Details
  In a non-multi-master cluster environment, you can configure the network details from any Oracle Key Vault management console.
- Configuring the Network Services
  In a non-multi-master cluster, you can configure the network services from any Oracle Key Vault management console.
- Configuring the System Time
  In a non-multi-master cluster environment, you can set the system time for the Oracle Key Vault system.
- Configuring DNS
  In a non-multi-master cluster environment, you can enter up to three DNS server IP addresses.
- Configuring FIPS Mode
  In a non-multi-master cluster environment, you can enable or disable FIPS mode for Oracle Key Vault.
- Configuring Syslog
  In a non-multi-master cluster environment, you can enable syslog for specific destinations and transmit the records either using Transmission Control Protocol (TCP) or User Datagram Protocol (UDP).
- Configuring RESTful Services
  In a non-multi-master cluster environment, you can enable or disable RESTful services.
- Configuring Oracle Audit Vault Integration
  In a non-multi-master cluster environment, you can enable Oracle Key Vault to send data to Oracle Audit Vault for centralized audit reporting and alerting.
- Configuring the Oracle Key Vault Management Console Web Session Timeout
  In a non-multi-master cluster environment, you can configure a timeout value in minutes for the Oracle Key Vault management console Web session.
- Restarting or Powering Off Oracle Key Vault
  You can manually restart or power off Oracle Key Vault as required for maintenance or for patch and upgrade procedures.

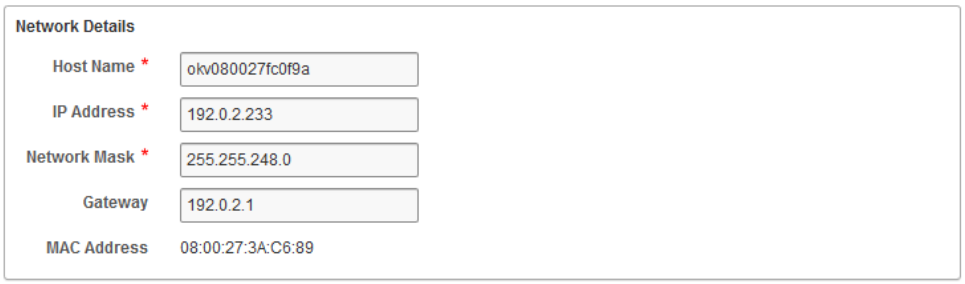
15.2.1 Configuring the Network Details

In a non-multi-master cluster environment, you can configure the network details from any Oracle Key Vault management console.

If you have a high availability configuration, then you must unpair the primary and standby Oracle Key Vault servers before changing the IP address. After you have changed the IP address of the primary or standby Oracle Key Vault server, pair the two servers again. After you complete the pairing process, re-enroll the Oracle Key

Vault endpoints to ensure that they are updated with the new IP addresses for both the primary and the standby Oracle Key Vault servers.

1. Log into the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. In the Settings page, go to the Network Details area.



4. Update the values for the following fields:
   - **Host Name**: Enter the name of the server.
   - **IP Address**: Enter the IP address of the server.
   - **Network Mask**: Enter the network mask of the server.
   - **Gateway**: Enter the network gateway of the server.

   The fields in this pane are automatically populated with the IP address and host name of your Oracle Key Vault server. But if you want, then you can update the **Host Name**, **IP Address**, **Network Mask**, and the **Gateway** for the Oracle Key Vault installation. You cannot change the **MAC Address** setting, because this is the hard-wired address of the network interface.
5. Click **Save**.

15.2.2 Configuring the Network Services

In a non-multi-master cluster, you can configure the network services from any Oracle Key Vault management console.

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.

1. Log into the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. In the Settings page, go to the Network Services area.

4. For Web Access, select one of the following options:
   - **All** to select all IP addresses
   - **IP Address(es)** to select a set of IP addresses that you specify in the next field, separating each IP address by a space. The **IP address(es)** web access option enables 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 Oracle 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 that you created during installation. SSH access is used only under the direction of Oracle Support, or when you upgrade.
   As a best practice, enable SSH access for short durations, solely for diagnostic, troubleshooting, or upgrade purposes, and then disable it as soon as you are done.

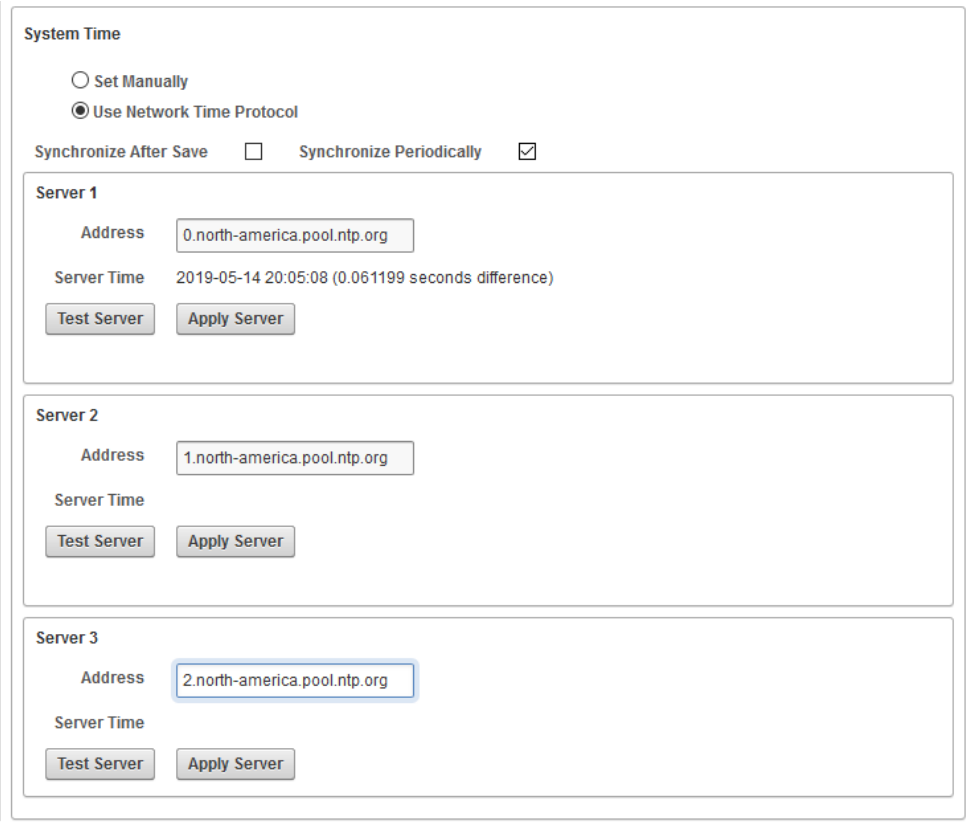
   Enabling or disabling SSH access will enable or disable the **inbound** SSH connection to the Oracle Key Vault server. Enabling or disabling SSH access in this manner has no bearing on the SSH Tunnel settings or any other outbound SSH connections that the Oracle Key Vault server itself establishes. SSH connections can still be established by the Oracle Key Vault to other servers as in the case of SSH Tunnel settings.
5. Click **Save**.

15.2.3 Configuring the System Time

In a non-multi-master cluster environment, you can set the system time for the Oracle Key Vault system.

You can configure Oracle Key Vault to use an NTP server to remain synchronized with the current time. (Fields for up to three servers are provided.) 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 primary-standby deployment, you must set the primary and standby servers to the same time. If you want to save an NTP server using its host name, then ensure that you have already configured DNS.

1. Log into the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. In the Settings page, go to the System Time area.



4. Enter up to three NTP server IP addresses, and then click **Save**.

   If you plan to use host names for the NTP servers, then you must configure DNS beforehand.

5. If necessary, return to the System Settings page by select **System Settings** under the **System** tab.
6. Choose **Use Network Time Protocol**.
7. Enter values for the following fields:
   - **Synchronize After Save**: This setting immediately synchronizes the system time to one of the given NTP servers after you save the settings.
   - **Synchronize Periodically**: This setting synchronizes the system time at a predetermined interval.
   - **Server 1**: Enter the IP address of a NTP server. You must supply an address for Server 1. To test the NTP server, click **Test Server**. To immediately synchronize the system time with this server, click **Apply Server**.
   - **Server 2**: Enter the IP address of a second NTP server. This value is optional. To test the NTP server, click **Test Server**. To immediately synchronize the system time with this server, click **Apply Server**.
   - **Server 3**: Enter the IP address of a third NTP server. This value is optional. To test the NTP server, click **Test Server**. To immediately synchronize the system time with this server, click **Apply Server**.
8. Click **Save**.

Related Topics

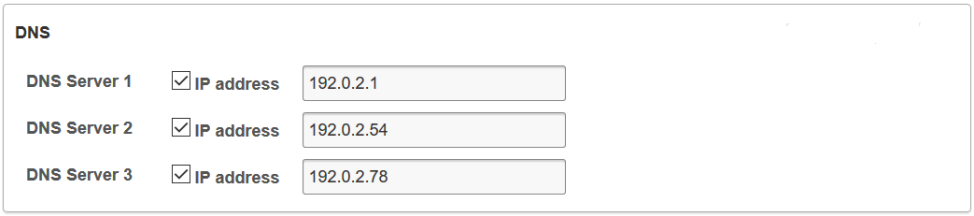
- Configuring DNS
  In a non-multi-master cluster environment, you can enter up to three DNS server IP addresses.

15.2.4 Configuring DNS

In a non-multi-master cluster environment, you can enter up to three DNS server IP addresses.

You can configure up to three domain name service (DNS) servers with IP addresses that Oracle Key Vault will use to resolve host names. This is useful if you only know the host name 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.

1. Log into the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. In the Settings page, go to the DNS area.



4. Enter up to three IP addresses for DNS servers.

   You must at minimum configure the DNS setting for Server 1.
5. Click **Save**.

Related Topics

- Configuring the System Time
  In a non-multi-master cluster environment, you can set the system time for the Oracle Key Vault system.

15.2.5 Configuring FIPS Mode

In a non-multi-master cluster environment, you can enable or disable FIPS mode for Oracle Key Vault.

In a primary-standby environment, ensure that both servers are consistent in their FIPS mode setting: either both are enabled, or both are disabled.

1. Log into the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. In the Settings page, go to the FIPS mode section.

4. Do one of the following:
   - To enable FIPS mode, select the **Enable** check box.
   - To disable FIPS mode, clear the **Enable** check box.

   Enabling or disabling FIPS mode will take a few minutes and will also restart the system automatically.
5. Click **Save**.

   After you click **Save**, a confirmation dialog box will appear.
6. In the confirmation dialog box, click **OK** to apply the changes and restart the Oracle Key Vault system.

   If you click **OK**, be aware that the operation cannot be canceled. The restart operation takes place immediately.

15.2.6 Configuring Syslog

In a non-multi-master cluster environment, you can enable syslog for specific destinations and transmit the records either using Transmission Control Protocol (TCP) or User Datagram Protocol (UDP).

All system related alerts are sent to syslog.

1. Log into the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. In the Settings page, go to the Syslog area.

4. Select one of the following protocols:
   - **TCP**: Enables syslog using the TCP protocol.
   - **UDP**: Enables syslog using the UDP protocol.
5. Enter the syslog destination IP addresses and port numbers in the **Syslog Destinations** field, in the format *`IP_address`*`:`*`port`*.

   You can enter multiple destinations, separated by a space.
6. Click **Save**.

15.2.7 Configuring RESTful Services

In a non-multi-master cluster environment, you can enable or disable RESTful services.

RESTful services allow you to automate endpoint enrollment and provisioning. RESTful services also support regular key management activities.

1. Log into the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. In the Settings page, go to the RESTful Services area.

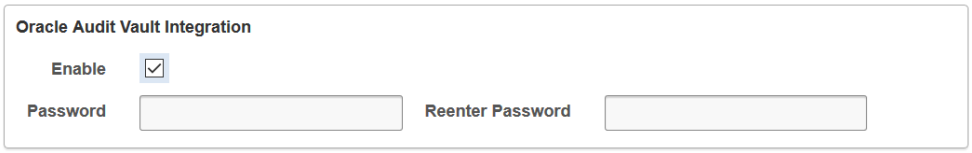
4. Select the **Enable** checkbox in the RESTful Services section.

   To disable RESTful services, clear the **Enable** checkbox.
5. Click **Save**.

15.2.8 Configuring Oracle Audit Vault Integration

In a non-multi-master cluster environment, you can enable Oracle Key Vault to send data to Oracle Audit Vault for centralized audit reporting and alerting.

1. Log into the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. In the Settings page, go to the Oracle Audit Vault Integration area.



4. Select the **Enable** check box.
5. When prompted, enter the Oracle Audit Vault Super Administrator password.
6. Click **Save**.

15.2.9 Configuring the Oracle Key Vault Management Console Web Session Timeout

In a non-multi-master cluster environment, you can configure a timeout value in minutes for the Oracle Key Vault management console Web session.

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

2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. In the System Settings page, navigate to **Management Console Timeout**.
4. Enter the value in minutes for the timeout.

   The default value is `10`. The range you can enter is `1` through `100`.
5. Click **Save**.

   After you click **Save**, the setting takes effect in the currently active user Web session. For other active sessions, this setting takes effect when the session is extended, the user refreshes the page, or the user navigates to another page. The user session remains active as long as the user clicks a button, moves the mouse or presses a key, or is performing other activities. If the user session is idle for more than the management console timeout duration, then the user is logged out and the login screen appears.

Just before the Web session ends, the user will be notified, starting earlier if the timeout value is larger, and is given the option to extend the session for the same length of time that was set for timeout value. For example, if the timeout was set to `20` minutes, then the user can extend the session for another 20 minutes.

Related Topics

- Ability to Set a Timeout Value for the Oracle Key Vault Management Console

Related Topics

- 18.4 NF: Management Console Idle Session Timeout

15.2.10 Restarting or Powering Off Oracle Key Vault

You can manually restart or power off Oracle Key Vault as required for maintenance or for patch and upgrade procedures.

1. Log into the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. Go to the top of the Settings page.

4. Do one of the following:
   - To restart, click **Reboot**.
   - To power off, click **Power Off**.

15.3 Configuring Oracle Key Vault in a Multi-Master Cluster Environment

When you configure Oracle Key Vault in a multi-master cluster environment, you can configure either individual nodes or the entire multi-master cluster environment.

- Configuring System Settings for Individual Multi-Master Cluster Nodes
  You can set or change settings that apply to the cluster node.

- Managing Oracle Key Vault Multi-Master Clusters
  You can create, configure, manage, and administer an Oracle Key Vault multi-master cluster by using the Oracle Key Vault management console.

15.3.1 Configuring System Settings for Individual Multi-Master Cluster Nodes

You can set or change settings that apply to the cluster node.

Examples of these settings are the network details, network services, system time, DNS, FIPS mode, syslog, and Oracle Audit Vault integration. Values set for the node override the cluster setting. However, you can clear any individual node setting to revert to the cluster setting.

- Configuring the Network Details for the Node
  In a multi-master cluster, you can change the host name for a node.
- Configuring the Network Services for the Node
  In a multi-master cluster, you can configure the network services for a node.
- Configuring the System Time for the Node
  In a multi-master cluster, you can set the system time for a node.
- Configuring DNS for the Node
  When you configure the DNS for a multi-master cluster node, you should enter more than one DNS IP address.
- Configuring the FIPS Mode for the Node
  All multi-master cluster nodes must use the same FIPS mode setting or you will receive an alert.
- Configuring Syslog for the Node
  In a node, you can enable syslog for specific destinations and transmit the records either using Transmission Control Protocol (TCP) or User Datagram Protocol (UDP).
- Configuring Oracle Audit Vault Integration for the Node
  You can configure the integration of Oracle Audit Vault (but not the Database Firewall component) for a node.
- Restarting or Powering Off Oracle Key Vault from a Node
  You can manually restart or power off an Oracle Key Vault node as required for maintenance or for patch and upgrade procedures.

15.3.1.1 Configuring the Network Details for the Node

In a multi-master cluster, you can change the host name for a node.

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. In the **Host Name** field, enter the name of the host name for the node.

   You cannot modify the **IP Address**, **Network Mask**, **Gateway**, and **MAC Address** fields, which are automatically populated.
4. Click **Save**.

15.3.1.2 Configuring the Network Services for the Node

In a multi-master cluster, you can configure the network services for a node.

1. Log into the Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. In the Settings page, go to the Network Services area.

4. For Web Access, select one of the following options:
   - **All** to select all IP addresses
   - **IP Address(es)** to select a set of IP addresses that you specify in the next field, separating each IP address by a space. The **IP address(es)** web access option enables 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 Oracle 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 that you created during installation. SSH access is used only under the direction of Oracle Support, or when you upgrade.
   As a best practice, enable SSH access for short durations, solely for diagnostic, troubleshooting, or upgrade purposes, and then disable it as soon as you are done.

   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.
5. Click **Save**.

15.3.1.3 Configuring the System Time for the Node

In a multi-master cluster, you can set the system time for a node.

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar. (Alternatively, you can select **Cluster System Settings**.)

3. If you want to configure NTP servers using their host names, then ensure that DNS is configured first.
4. If necessary, return to the System Settings page by select **System Settings** under the **System** tab.
5. Choose **Use Network Time Protocol**.

   In cluster mode, you must use the Network Time Protocol (NTP), so you cannot change the selection from **Use Network Time Protocol** to **Set Manually**.
6. Enter values for the following fields:
   - **Synchronize After Save**: This setting immediately synchronizes the system time for the node to one of the given NTP servers after you save the settings.
   - **Synchronize Periodically**: This setting synchronizes the system time for the node at a predetermined interval.
   - **Server 1**: Enter the IP address of a NTP server. You must supply an address for Server 1. To test the NTP server, click **Test Server**. To immediately synchronize the system time with this server, click **Apply Server**.
   - **Server 2**: Enter the IP address of a second NTP server. This value is optional. To test the NTP server, click **Test Server**. To immediately synchronize the system time with this server, click **Apply Server**.
   - **Server 3**: Enter the IP address of a third NTP server. This value is optional. To test the NTP server, click **Test Server**. To immediately synchronize the system time with this server, click **Apply Server**.
7. Click **Save** (or **Save to Cluster**).

Related Topics

-

15.3.1.4 Configuring DNS for the Node

When you configure the DNS for a multi-master cluster node, you should enter more than one DNS IP address.

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar. (Alternatively, you can select **Cluster System Settings**.)
3. In the **DNS** section of the System Settings pages, enter up to three DNS server IP addresses.

   While only the first value is required, two entries are recommended for fault tolerance.
4. Click **Save** (or **Save to Cluster**).

15.3.1.5 Configuring the FIPS Mode for the Node

All multi-master cluster nodes must use the same FIPS mode setting or you will receive an alert.

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

2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. In the FIPS Mode section, do one of the following:
   - To enable FIPS mode, select the **Enable** check box.
   - To disable FIPS mode, clear the **Enable** check box.

   Enabling or disabling FIPS mode will take a few minutes and will also restart the system automatically.
4. Click **Save**.

   After you click **Save**, Oracle Key Vault will restart automatically.

15.3.1.6 Configuring Syslog for the Node

In a node, you can enable syslog for specific destinations and transmit the records either using Transmission Control Protocol (TCP) or User Datagram Protocol (UDP).

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar. (Alternatively, you can select **Cluster System Settings**.)
3. In the **Syslog** section, select one of the following protocols:
   - **TCP**: Enables syslog using the TCP protocol.
   - **UDP**: Enables syslog using the UDP protocol.
4. Enter the syslog destination IP addresses and port numbers in the **Syslog Destinations** field, in the format *`IP_address`*`:`*`port`*.

   You can enter multiple destinations, separated by a space.
5. Click **Save** (or **Save to Cluster**).

15.3.1.7 Configuring Oracle Audit Vault Integration for the Node

You can configure the integration of Oracle Audit Vault (but not the Database Firewall component) for a node.

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. Select the **Enable** check box to Oracle Audit Vault Integration for the node.
4. In the **Password** and **Reenter password** fields that appear after you click **Enable**, enter the password of the user in the database that Audit Vault and Database Firewall will use to extract the audit records.
5. Click **Save**.

15.3.1.8 Restarting or Powering Off Oracle Key Vault from a Node

You can manually restart or power off an Oracle Key Vault node as required for maintenance or for patch and upgrade procedures.

When you restart or power-off Oracle Key Vault nodes, only the current node is restarted or powered-off. The other nodes in the cluster are unable to send information

to and from the nodes that are powered off. When the nodes are restarted, there will be a period needed for the restarted nodes to catch up on activities that took place while they were down.

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. Go to the top of the Settings page.

4. Do one of the following to restart or power off the node:
   - To restart, click **Reboot**.
   - To power off, click **Power Off**.

15.3.2 Managing Oracle Key Vault Multi-Master Clusters

You can create, configure, manage, and administer an Oracle Key Vault multi-master cluster by using the Oracle Key Vault management console.

- About Configuring Cluster System Settings
  You can set or change settings that apply to an entire multi-master cluster.
- Configuring the System Time for the Cluster
  When you configure the system time, you can set it for multiple servers and also set the synchronization.
- Configuring DNS for the Cluster
  When you configure the DNS for a cluster, you can enter up to three DNS server IP addresses.
- Configuring Maximum Disable Node Duration for the Cluster
  You can set the Configuring Maximum Node Duration time for the cluster in hours.
- Configuring RESTful Services for the Cluster
  You can enable or disable RESTful Services for the cluster.
- Configuring Syslog for the Cluster
  In a multi-master cluster environment, you can enable syslog for specific destinations and transmit the records either using Transmission Control Protocol (TCP) or User Datagram Protocol (UDP).
- Configuring SNMP Settings for the Cluster
  You can enable or disable SNMP access for a multi-master cluster.
- Configuring the Oracle Key Vault Management Console Web Session Timeout for the Cluster
  You can configure a timeout value in minutes for the Oracle Key Vault management console for all nodes in a multi-master cluster.

15.3.2.1 About Configuring Cluster System Settings

You can set or change settings that apply to an entire multi-master cluster.

You can set the system time, DNS, the maximum time a server can be disabled before it is evicted from the cluster, enable RESTful services, the protocol to use for syslog,

the syslog destination, and monitoring settings for the cluster. Any values that are set and saved to an individual node will not be overridden by cluster settings. It may take several minutes for changes to propagate to other nodes.

15.3.2.2 Configuring the System Time for the Cluster

When you configure the system time, you can set it for multiple servers and also set the synchronization.

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **Cluster System Settings** from the left navigation bar.
3. Choose the **User Network Time Protocol** option.

   Only the first value is required.
4. Enter values for the following fields:
   - **Synchronize After Save**: This synchronizes the time across the cluster after you save the settings.
   - **Synchronize Periodically**: This synchronizes the time across the cluster at a predetermined interval. Once selected and applied, this option cannot be deselected.
   - **Server 1**: Enter the IP address of a NTP server. To test the NTP server, click **Test Server**. To immediately synchronize the system time with this server, click **Apply Server**.
   - **Server 2**: Enter the IP address of a second NTP server. To test the NTP server, click **Test Server**. To immediately synchronize the system time with this server, click **Apply Server**.
   - **Server 3**: Enter the IP address of a third NTP server. To test the NTP server, click **Test Server**. To immediately synchronize the system time with this server, click **Apply Server**.
5. In the System Time section, click **Save to Cluster**.

15.3.2.3 Configuring DNS for the Cluster

When you configure the DNS for a cluster, you can enter up to three DNS server IP addresses.

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **Cluster System Settings** from the left navigation bar.
3. In the **DNS** section of the Cluster System Settings page, enter up to three DNS Server IP addresses.
4. In the DNS section, click **Save to Cluster**.

15.3.2.4 Configuring Maximum Disable Node Duration for the Cluster

You can set the Configuring Maximum Node Duration time for the cluster in hours.

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **Cluster System Settings** from the left navigation bar.
3. In the **Maximum Disable Node Duration** section, enter a value, in hours, for the duration that a node can be disabled before it is evicted from the cluster.
4. In the Maximum Disable Node Duration section, click **Save to Cluster**.

15.3.2.5 Configuring RESTful Services for the Cluster

You can enable or disable RESTful Services for the cluster.

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **Cluster System Settings** from the left navigation bar.
3. Select the **Enable** checkbox in the RESTful Services section.
4. In the RESTful Services section, click **Save to Cluster**.

15.3.2.6 Configuring Syslog for the Cluster

In a multi-master cluster environment, you can enable syslog for specific destinations and transmit the records either using Transmission Control Protocol (TCP) or User Datagram Protocol (UDP).

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **Cluster System Settings** from the left navigation bar.
3. In the **Syslog** section, select one of the following protocols:
   - **TCP**: Enables syslog using the TCP protocol.
   - **UDP**: Enables syslog using the UDP protocol.
4. Enter the syslog destination IP addresses and port numbers in the **Syslog Destinations** field, in the format *`IP_address`*`:`*`port`*.

   You can enter multiple destinations, separated by a space.
5. In the Syslog section, click **Save to Cluster**.

15.3.2.7 Configuring SNMP Settings for the Cluster

You can enable or disable SNMP access for a multi-master cluster.

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **Monitoring Settings** from the left navigation bar.
3. For **Scope**, select **Cluster**.
4. Select who has SNMP access to the multi-master cluster by choosing one of the options:

- **All**: Allows SNMP access from all IP addresses.
- **Disabled**: Allows no SNMP access.
- **IP address(es)**: Allows SNMP access from the list of IP addresses supplied in the address box. Enter a space-separated list of IP addresses.

5. Enter values for the following fields:
   - **Username**: Enter the SNMP user name.
   - **Password**: Enter the SNMP password.
   - **Reenter Password**: Enter the SNMP password again.
6. Click **Save to Cluster**.

15.3.2.8 Configuring the Oracle Key Vault Management Console Web Session Timeout for the Cluster

You can configure a timeout value in minutes for the Oracle Key Vault management console for all nodes in a multi-master cluster.

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **Cluster System Settings** from the left navigation bar.
3. In the Cluster System Settings page, navigate to **Management Console Timeout**.
4. Enter the value in minutes for the timeout.

   The default value is `10`. The range you can enter is `1` through `100`.
5. Click **Save to Cluster**.

   After you click **Save to Cluster**, the setting takes effect in the currently active user Web session and is applied to all nodes in the cluster. For other active sessions, this setting takes effect when the session is extended, the user refreshes the page, or the user navigates to another page. The user session remains active as long as the user clicks a button, moves the mouse or presses a key, or is performing other activities. If the user session is idle for more than the management console timeout duration, then the user is logged out and the login screen appears.

Just before the Web session ends, the user will be notified, starting earlier if the timeout value is larger, and is given the option to extend the session for the same length of time that was set for timeout value. For example, if the timeout was set to `20` minutes, then the user can extend the session for another 20 minutes.

Related Topics

- Ability to Set a Timeout Value for the Oracle Key Vault Management Console

Related Topics

- 18.4 NF: Management Console Idle Session Timeout

15.4 Managing System Recovery

System recovery includes tasks such as recovering lost administrative passwords.

- About Managing System Recovery
  To perform system recovery, you use the recovery passphrase.
- Recovering Credentials for Administrators
  You can recover the system by adding credentials for administrative users.
- Changing the Recovery Passphrase in a Non-Clusters Environment
  Periodically changing the recovery passphrase is a good security practice.
- Changing the Recovery Passphrase in a Multi-Master Cluster
  Changing the recovery passphrase in a multi-master cluster is a two-step process.
- Changing the Installation Passphrase
  You can change the installation passphrase from the system console.

15.4.1 About Managing System Recovery

To perform system recovery, you use the recovery passphrase.

In an emergency when no administrative users are available, or you must change the password of administrative users, you can recover the system with the recovery passphrase that was created during Oracle Key Vault installation. In addition, you can change the recovery passphrase to keep up with security best practices.

15.4.2 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
  After you install Oracle Key Vault, you must complete a set of post-installation tasks.

15.4.3 Changing the Recovery Passphrase in a Non-Clusters Environment

Periodically changing the recovery passphrase is a good security practice.

A user with the System Administrator role should 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.

1. Perform a server backup.
2. From a web browser, enter the IP address of your Oracle Key Vault installation.
3. In the Oracle Key Vault login page, *do not log in*.
4. Click the **System Recovery** link.

   A new login page appears with a single field: **Recovery Passphrase**.
5. Enter the recovery passphrase and then click **Login**.

   The **Administrator Recovery** page appears with two tabs above it: **Administrator Recovery** and **Recovery Passphrase**.
6. Click **Recovery Passphrase**.

   The **Recovery Passphrase** page appears with two fields to enter and reenter the new passphrase.
7. Enter the new recovery passphrase in the two fields.
8. Click **Submit**.

Related Topics

- *Oracle Database Backup and Recovery User's Guide*

15.4.4 Changing the Recovery Passphrase in a Multi-Master Cluster

Changing the recovery passphrase in a multi-master cluster is a two-step process.

To change the recovery passphrase for a multi-master cluster, you must first initiate the change throughout the nodes in the multi-master cluster environment before changing the recovery passphrase.

- Step 1: Initiate the Recovery Passphrase Change Across the Nodes
  A user with the System Administrator role should perform a new backup whenever the recovery passphrase changes.
- Step 2: Change the Recovery Passphrase
  After the multi-master cluster nodes have been notified of the impending recovery passphrase change, you can change the recovery passphrase.

15.4.4.1 Step 1: Initiate the Recovery Passphrase Change Across the Nodes

A user with the System Administrator role should perform a new backup whenever the recovery passphrase changes.

This is 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. First, you must initiate the change for the recovery passphrase so that all nodes in the multi-master cluster will be notified of the impending change.

1. Perform a server backup.
2. Ensure that all nodes are in the `ACTIVE` state and replication has been verified between all nodes. Ensure that there are no cluster operations going on (such as adding a node).

3. From a web browser, enter the IP address of the Oracle Key Vault installation that is not in read-only restricted mode.
4. In the Oracle Key Vault login page, *do not log in*.
5. Click the **System Recovery** link at the lower right corner of the login page.

   A new login page appears with a single field: **Recovery Passphrase**.
6. Enter the recovery passphrase and click **Login**.

   The **Administrator Recovery** page appears with two tabs above it: **Administrator Recovery** and **Recovery Passphrase**.
7. Click the **Recovery Passphrase** tab.

8. Click the **Initiate Change** button.
9. Log out.
10. Wait 3 to 4 minutes before continuing.

    During this time, all nodes will be notified that a passphrase change will be performed. To cancel a passphrase change, click the **Reset** button.

    All nodes will determine if more than one passphrase change has been initiated. If more than one passphrase change has been initiated, conflict resolution will be performed.

    After you cancel a passphrase change by using the **Reset** button, Oracle recommends that you remedy the issue and again initiate a passphrase change, making sure to then change the passphrase on every node in the cluster.

Related Topics

- *Oracle Database Backup and Recovery User's Guide*
- Step 2: Change the Recovery Passphrase
  After the multi-master cluster nodes have been notified of the impending recovery passphrase change, you can change the recovery passphrase.

15.4.4.2 Step 2: Change the Recovery Passphrase

After the multi-master cluster nodes have been notified of the impending recovery passphrase change, you can change the recovery passphrase.

1. From a Web browser, enter the IP address of a multi-master cluster node in the Oracle Key Vault installation.

You can find a list of available nodes in the Oracle Key Vault management console by selecting the Clusters tab and then checking the Cluster Details section.

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 login page.

   A new login page appears with a single field: **Recovery Passphrase**.
4. Enter the recovery passphrase and click **Login**.

   The **Administrator Recovery** page appears with two tabs above it: **Administrator Recovery** and **Recovery Passphrase**.
5. Click the **Recovery Passphrase** tab.

   The **Recovery Passphrase** page appears with two fields to enter and re-enter the new passphrase.
6. Enter the new recovery passphrase in the two fields.
7. Click **Submit**.
8. Repeat these steps for each node in the cluster.

> **Note:**
>
> HSM reverse migrate cannot run when the recovery passphrase is being changed.

> **Caution:**
>
> It is your responsibility to keep the recovery passphrase the same on all nodes in the cluster. If you set the recovery passphrase differently on cluster nodes it will negatively impact cluster functionality, such as adding nodes and HSM-enabling nodes. In addition to the addition of nodes and nodes being HSM-enabled, certificate rotation in a multi-master cluster depends on all nodes having the same recovery passphrase.

Related Topics

- Step 1: Initiate the Recovery Passphrase Change Across the Nodes
  A user with the System Administrator role should perform a new backup whenever the recovery passphrase changes.

15.4.5 Changing the Installation Passphrase

You can change the installation passphrase from the system console.

- About Changing the Installation Passphrase
  You can only change the installation passphrase during a specific window of time.
- Changing an Installation Passphrase
  You must change the installation passphrase in the system console.

15.4.5.1 About Changing the Installation Passphrase

You can only change the installation passphrase during a specific window of time.

The installation passphrase is specified during installation. You must use the installation passphrase to log in to Oracle Key Vault and complete the post-installation tasks. The installation passphrase can only be changed on the console after installation but before post-installation. After the post-installation tasks are completed, this option no longer appears on the console.

If you forget the installation passphrase, then you can create a new installation passphrase. As with all Oracle Key Vault passphrases, it is important to store the installation passphrase securely.

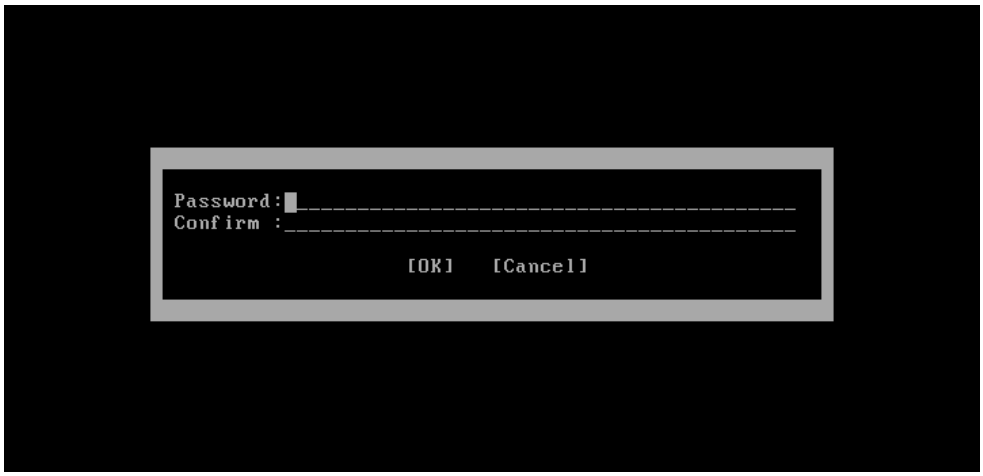
15.4.5.2 Changing an Installation Passphrase

You must change the installation passphrase in the system console.

1. Access the system console of the server where Oracle Key Vault is installed.

2. Select **Change Installation Passphrase** and press **Enter**.

   The **New Passphrase** screen appears.

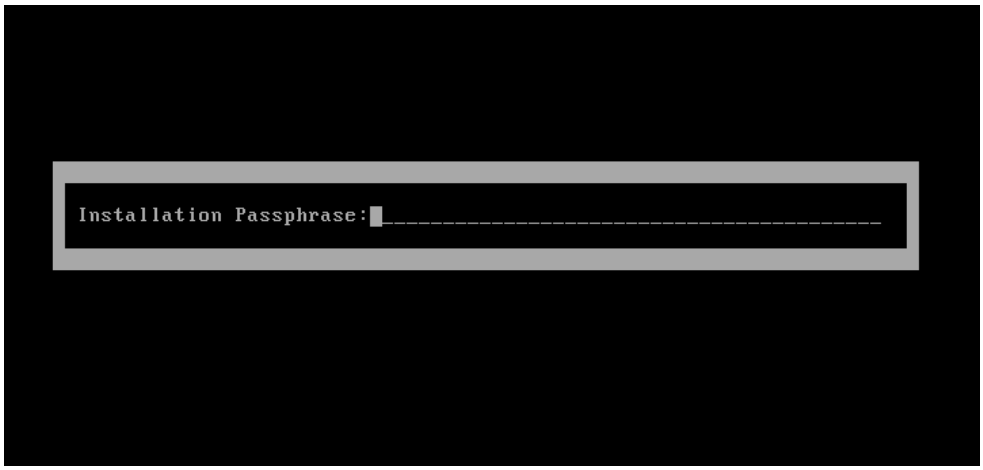


3. Enter the new installation passphrase in the **New Passphrase** and **Confirm** fields.

   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.

4. Select **OK** and then press **Enter**.

   The **Installation Passphrase** screen appears.



5. Enter the old installation passphrase and then press **Enter**.

15.5 Support for a Primary-Standby Environment

To ensure that Oracle Key Vault can always access security objects, you can deploy Oracle Key Vault in a primary-standby (highly available) configuration.

This configuration also supports disaster recovery scenarios.

You can deploy two Oracle Key Vault servers in a primary-standby configuration. The primary server services the requests that come from endpoints. If the primary server fails, then the standby server 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.

The primary-standby configuration was previously known as the high availability configuration. The primary-standby configuration and the multi-master cluster configuration are mutually exclusive.

Oracle Key Vault supports primary-standby read-only restricted mode. When the primary server is affected by server, hardware, or network failures, primary-standby read-only restricted mode ensures that an Oracle Key Vault server is available to service endpoints, thus ensuring operational continuity. However, key and sensitive operations, such as generation of keys are disabled, while operations such as generation of audit logs are unaffected.

When an unplanned shutdown makes the standby server unreachable, the primary server is still available to the endpoints in read-only mode.

Related Topics

- About the Oracle Key Vault Primary-Standby Configuration
  You configure a primary-standby environment by providing the primary and standby servers with each other's IP address and certificate, and then pairing them.

15.6 Commercial National Security Algorithm Suite Support

You can use scripts to perform Commercial National Security Algorithm (CNSA) operations for Oracle Key Vault HSM backup and upgrade operations.

- About Commercial National Security Algorithm Suite Support
  You can configure Oracle Key Vault for compliance with the Commercial National Security Algorithm (CNSA) Suite.
- Running the Commercial National Security Algorithm Scripts
  The Commercial National Security Algorithm (CNSA) scripts update the `okv_security.conf` file.
- Performing Backup and Restore Operations with CNSA
  After you back up and restore Oracle Key Vault, use `/usr/local/okv/bin/okv_cnsa` to use the enhanced Commercial National Security Algorithm (CNSA) Suite.
- Upgrading a Standalone Oracle Key Vault Server with CNSA
  You can upgrade a standalone Oracle Key Vault while using Commercial National Security Algorithm (CNSA) compliance by upgrading and then executing the `okv_cnsa` script.
- Upgrading Primary-Standby Oracle Key Vault Servers to Use CNSA
  You can upgrade Oracle Key Vault primary-standby servers while using Commercial National Security Algorithm (CNSA) compliance by upgrading and then executing the `okv_cnsa` script.

15.6.1 About Commercial National Security Algorithm Suite Support

You can configure Oracle Key Vault for compliance with the Commercial National Security Algorithm (CNSA) Suite.

This compliance applies to TLS connections to and from the Oracle Key Vault appliance.

The CNSA suite is a list of strong encryption algorithms and key lengths, that offer greater security and relevance into the future.

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

- `/usr/local/okv/bin/okv_cnsa` makes configuration file changes to update as many components as possible to use the enhanced algorithms.
- `/usr/local/okv/bin/okv_cnsa_cert` regenerates CNSA compliant public key pairs and certificates.

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

> **Limitations:**
>
> - CNSA compliance is not supported for all components in the Oracle Key Vault infrastructure (for example, SSH or Transparent Data Encryption (TDE)).
> - The Firefox browser is not supported for use with the Oracle Key Vault management console when CNSA is enabled. This is because the Firefox browser does not support CNSA-approved cipher suites.

15.6.2 Running the Commercial National Security Algorithm Scripts

The Commercial National Security Algorithm (CNSA) scripts update the `okv_security.conf` file.

1. Back up Oracle Key Vault.
2. If necessary, enable SSH access.

Log in to the Oracle Key Vault management console as a user who has the System Administrator role. Select the **System Settings** tab, then under Network Services, select **SSH Access**. Select **IP address(es)** and then enter only the IP addresses that you need. Click **Save**.

3. SSH into the Oracle Key Vault server as the `support` user, entering the `support` user password that was created during post-installation, when prompted.

```
$ ssh support@okv_instance
```

4. Change to the `root` user:

```
$  su root
```

5. Run the scripts as follows:

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

6. Disable SSH access and then restart the Oracle Key Vault server.

   Log in to the Oracle Key Vault management console as a user who has the System Administrator role. Select the **System Settings** tab, then under Network Services, select **Disabled**. Click **Save**. Restart the Oracle Key Vault server by clicking **Reboot** on the top right.

The scripts update the `/usr/local/okv/etc/okv_security.conf` with the following line:

```
USE_ENHANCED_ALGORITHMS_ONLY="1"
```

Related Topics

- Backup and Restore Operations
  You may configure automatic backups for continuous, reliable, and protected access to security objects with minimum downtime.

15.6.3 Performing Backup and Restore Operations with CNSA

After you back up and restore Oracle Key Vault, use `/usr/local/okv/bin/okv_cnsa` to use the enhanced Commercial National Security Algorithm (CNSA) Suite.

1. Perform the backup and restore operation.
2. Wait until the restore operation is complete and the system has restarted.

   Do not proceed without completing this step.

3. SSH into the Oracle Key Vault server as the `support` user:

```
$ ssh support@okv_instance
```

4. Switch to the `root` user:

```
$ su root
```

5. Run the following CNSA script :

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

Related Topics

- Backup and Restore Operations
  You may configure automatic backups for continuous, reliable, and protected access to security objects with minimum downtime.

15.6.4 Upgrading a Standalone Oracle Key Vault Server with CNSA

You can upgrade a standalone Oracle Key Vault while using Commercial National Security Algorithm (CNSA) compliance by upgrading and then executing the `okv_cnsa` script.

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

3. If necessary, enable SSH access.

   Log in to the Oracle Key Vault management console as a user who has the System Administrator role. Select the **System Settings** tab, then under Network Services, select **SSH Access**. Select **IP address(es)** and then enter only the IP addresses that you need. Click **Save**.

4. Ensure you have enough space in the destination directory for the upgrade ISO files.

5. Log in to the Oracle Key Vault server through SSH as user support, then switch user `su` to `root`.

6. 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-18.4.0.0.0.iso /var/lib/oracle/
   destination_directory_for_iso_file
   ```

   In this specification:

   - `remote_host` is the IP address of the computer containing the ISO upgrade file.
   - `remote_host` is the IP address of the computer containing the ISO upgrade file.

7. Make the upgrade accessible by using the mount command:

   ```
   root# /bin/mount -o loop,ro /var/lib/oracle/okv-upgrade-disc-18.4.0.0.0.iso /
   images
   ```

8. Clear the cache using the `clean all` command:

   ```
   root# yum -c /images/upgrade.repo clean all
   ```

9. Execute the following upgrade ruby script:

   ```
   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.

10. Run the first CNSA script, which is available from the Oracle Key Vault ISO files location:

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

11. Restart the Oracle Key Vault database server:

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

    On the first restart of the computer after the upgrade, the system will apply the necessary changes. This can take a few hours. Do not shut down the system during this time.

    The upgrade is completed when the screen with heading: `Oracle Key Vault Server 18.4.0.0.0` appears. The revision should reflect the upgraded release. Following 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.

12. Confirm that Oracle Key Vault has been upgraded to the correct version.

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

    b. Select the **System** tab, and then select **Status**.

    c. Verify that the version displayed is 18.4.0.0.0.

       The release number is also at the bottom of each page, to the right of the copyright information.

13. Disable SSH access.

    Log in to the Oracle Key Vault management console as a user who has the System Administrator role. Select the **System Settings** tab, then under Network Services, select **Disabled**. Click **Save**.

Related Topics

- https://apps.nsa.gov/iaarchive/programs/iad-initiatives/cnsa-suite.cfm
- https://cryptome.org/2016/01/CNSA-Suite-and-Quantum-Computing-FAQ.pdf

15.6.5 Upgrading Primary-Standby Oracle Key Vault Servers to Use CNSA

You can upgrade Oracle Key Vault primary-standby servers while using Commercial National Security Algorithm (CNSA) compliance by upgrading and then executing the `okv_cnsa` script.

You must perform the upgrade standby and primary servers in one session with as little time between the standby and primary upgrade as possible. 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 longer than several hours.

1. Prepare for the upgrade.

    - 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 shut down during

the upgrade, if no persistent cache has been configured. With persistent cache enabled, endpoints will continue to be operational during the upgrade process.

- Ensure that both the primary and standby systems have 8 GB memory.

2. Ensure that you have backed up the server you are upgrading so your data is safe and recoverable.

   You can use Oracle Backup and Recovery (Oracle RMAN) to perform this backup. Ensure that in the time between the backup and shutting down the Oracle Key Vault servers for upgrade, that no databases perform a set or rekey operation (for example, using the `ADMINISTER KEY MANAGEMENT` statement), since these new keys will not included in the backup.

   Do not proceed without completing this step.

3. First, upgrade the standby server while the primary server is running.

   Follow Steps 2 through Step 11 of the standalone server upgrade process for CNSA.

4. Ensure that the upgraded standby Oracle Key Vault server is restarted and running.

5. Upgrade the primary Oracle Key Vault server following Steps 1 through 11 of the standalone server upgrade.

   After both the standby and primary Oracle Key Vault servers are upgraded, the two servers will automatically synchronize.

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

7. Select the **System** tab, and then **Status**.

8. Verify that the **Version** field displays the new software version 18.4.0.0.0.

Related Topics

- Upgrading a Standalone Oracle Key Vault Server with CNSA
  You can upgrade a standalone Oracle Key Vault while using Commercial National Security Algorithm (CNSA) compliance by upgrading and then executing the `okv_cnsa` script.
- *Oracle Database Backup and Recovery User's Guide*

15.7 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 a multi-master cluster:** You can configure a multi-master cluster by adding redundancy in the form of additional nodes. The client can access any available node. In the event of a failure of any node, a client will automatically connect to another node in the endpoint node scan list. This reduces and potentially eliminates downtime.
- **Configuring a primary-standby environment:** A primary-standby environment 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.

- **Enabling read-only restricted mode:** Primary-standby 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 primary-standby read-only restricted mode is disabled, then 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 encryption key cache:** The persistent master encryption 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.

- **Apply the TDE heartbeat database patch on endpoints:** Apply the database patch for Bug 22734547 to tune the Oracle Key Vault heartbeat.

Oracle strongly recommends that you back up Oracle Key Vault data regularly on a schedule. This practice ensures that backups are current and hold the most recent data. You can use this backup to restore a new or existing Oracle Key Vault server and enable it to be fully operational with minimum downtime and data loss.

If the Oracle Key Vault 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.

Related Topics

- Using the Persistent Master Encryption Key Cache
  The persistent master encryption key cache feature enables databases to be operational when the Oracle Key Vault server is unavailable.

16
Managing Certificates

In addition to Oracle Key Vault-generated certificates, you can manage third-party certificates.

- Rotating Certificates
  You can rotate both Oracle Key Vault-generated certificates or third-party certificates.
- Managing Console Certificates
  You can use the Oracle Key Vault management console to manage console certificates.

16.1 Rotating Certificates

You can rotate both Oracle Key Vault-generated certificates or third-party certificates.

- About Rotating Certificates
  The certificate rotation process captures all certificates in the Oracle Key Vault server. This operation does not rotate the console certificates.
- Advice for Managing Certificate Rotations
  Oracle Key Vault provides advice on the best ways to rotate certificates.
- Factors That May Affect the Certificate Rotation Process
- Rotating All Certificates
  You can use the Oracle Key Vault management console to rotate certificates.
- Checking the Certificate Rotation Status
  You can use the Oracle Key Vault management console to check the status of a certificate rotation.

16.1.1 About Rotating Certificates

The certificate rotation process captures all certificates in the Oracle Key Vault server. This operation does not rotate the console 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.

In addition to standalone environments, you can rotate certificates in primary-standby and multi-master cluster environments. In both, Oracle Key Vault automatically synchronizes the certificates in both systems in a primary-standby configuration, and

in all nodes in a multi-master cluster configuration. You do not have to perform any extra configuration.

Related Topics

- Configuring Alerts
  You can configure alerts in the Reports page of the Oracle Key Vault management console.

16.1.2 Advice for Managing Certificate Rotations

Oracle Key Vault provides advice on the best ways to rotate certificates.

- Do not initiate a certificate rotation while a node addition is in progress.
- Do not try node operations (such as adding or disabling nodes) while a certificate rotation is in process.
- You cannot initiate certificate rotation unless all nodes in the cluster are active. You can check if a node is active by checking the Cluster Monitoring page. (Click the **Cluster** tab, and then select **Monitoring** from the left navigation bar.)
- 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.

16.1.3 Factors That May Affect the Certificate Rotation Process

- Each cluster node only generates certificates for a small set of endpoints. These endpoints are those whose creator node (the node on which the certificates are generated) it is. (You can find an endpoint's creator node in the Oracle Key Vault management console by going to the Endpoints page, and then looking for the creator node for each endpoint.) If all endpoints were created before an upgrade from Oracle Key Vault release 12.2, then it is possible that they may all be associated with one single cluster node. This can make the rotation process slower than if the endpoints had been created on different cluster nodes.
- During the rotation process, Oracle Key Vault rotates endpoints in batches on each node of the cluster, with a maximum number of endpoints that are allowed to be in the rotated state at any one time. At least one of those rotated endpoints must receive its new certificates and acknowledge receipt (involving at least two communications with the server) before the server moves on to processing another endpoint. If all endpoints are considered to have been created on a single Oracle Key Vault cluster node, then the rotation process may degenerate to rotating a few endpoints at a time across the cluster.

- In order to receive the new certificates, the endpoint must reach out to the node on which its certificates have been generated (that is, the creator node). In a multi-master cluster configuration, whenever the endpoint attempts to make a connection to Oracle Key Vault, it performs the following actions:
  - First, it obtains the list of server IPs from its configuration file (`okvclient.ora`).
  - Next, it picks one at random from those in the cluster subgroup to which the endpoint's creator node belongs.
    The endpoint reaches out to a random Oracle Key Vault cluster node, and not necessarily to its creator node. This means that even if the Oracle Key Vault management console shows that the endpoint has had its certificates rotated, the endpoint may not receive the new certificates for some considerable period of time, despite making repeated attempts to reach out to the Oracle Key Vault cluster.
- If a given endpoint does not receive its rotated certificates due to network or other issues, or is in the "Suspended" state, Oracle recommends that you re-enenroll the endpoint, or even delete 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.

Related Topics

- Deleting, Suspending, or Reenrolling Endpoints
  When endpoints no longer use Oracle Key Vault to store security objects, you can delete them, and then re-enroll when they are needed.

16.1.4 Rotating All Certificates

You can use the Oracle Key Vault management console to rotate certificates.

Before beginning certificate rotation, ensure that the recovery passphrase is the same across all multi-master cluster nodes.

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

   In a primary-standby environment, log in to the primary Oracle Key Vault server. In a multi-master cluster environment, you can log in to any node in the cluster. Oracle recommends that you initiate the rotation from one node at a time. Do not perform multiple rotations on different nodes at once.
3. Select the **System** tab.
4. Select **Manage Server Certificate**.
5. In the Manage Server Certificate page, click **Generate System Certificate**.
6. 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.

   In a mult-master cluster environment:

- After the certification rotation process is initiated, the details of the new certificate that was generated are shown on the node on which you initiated the rotation. After a few minutes, if you refresh the Manage Server Certificate page on all of the other nodes, this page should show that a message saying that the new certificate is being propagated to that node.
- The certificate will be propagated to all nodes, but not activated. Depending on the number of nodes in the cluster, it may take some time to complete the propagation process.
- You can cancel the certificate rotation only up to the point that 1) all nodes in the cluster have received the certificates, and 2) each node has notified the other nodes that it has received the certificate. At this point, the **Abort** button will disappear and only **Activate Certificate** remains. The certificate activation process can only take place when all nodes in the cluster no longer have the **Abort** button appearing.
- Periodically refresh the Manage Server Certificate page, in case there have been changes to the status. For example, you should refresh this page if you want to determine that the **Abort** button is no longer showing and the **Activate Certificate** button has appeared. To access this page, select the **System** tab and then select **Manage Server Certificate** from the left menu.

7. When the **Activate Certificate** button appears and is enabled, click it.

   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**.

   In a multi-master cluster environment, **Activate Certificate** applies the certificate to all nodes in the cluster. The certificate activation process can only take place when all nodes in the cluster no longer have the **Abort** button appearing. It takes a few minutes for the remaining nodes to be updated. Ensure that you click **Activate Certificate** on only one node before you refresh the Manage Server Certificates page on the other nodes. Wait a few minutes for the screen to refresh. (You only need to click Activate Certificate on one node, not multiple nodes.) Note that the Manage Server Certificates page on all nodes other than the one that you clicked **Activate Certificate** on may show no change in status for a few minutes, until the process starts to take effect on those nodes.

8. 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 over 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. You should periodically check the status of replication across the cluster by viewing either the Cluster Monitoring page or the Cluster Management page. (To access either of these pages, click the **Cluster** tab, and then select either **Management** or **Monitoring** in the left navigation bar.) When the endpoint successfully receives the credentials, 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.

9. If you had previously downloaded the Oracle Key Vault RESTful services software utility (`okvrestservices.jar`), then download it again so that you can continue to use the RESTful services utility.

   If you are using KMIP REST, then you do not need to perform this step because the `okvutil` endpoint that contains the `okvclient.ora` has received the updates.

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. In a multi-master cluster environment, you can initiate another certificate rotation only after all the nodes have completed their certification rotation process.
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

- Backup and Restore Operations
  You may configure automatic backups for continuous, reliable, and protected access to security objects with minimum downtime.
- Configuring Alerts
  You can configure alerts in the Reports page of the Oracle Key Vault management console.
- Step 4: Download the RESTful Software Utility
  The RESTful software utility is in the `okvrestservices.jar` file.

16.1.5 Checking the Certificate Rotation Status

You can use the Oracle Key Vault management console to check the status of a certificate rotation.

You should also check the Manage Server Certificates page.

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 for the certificate (**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.

16.2 Managing Console Certificates

You can use the Oracle Key Vault management console to manage console certificates.

- About Managing Console Certificates
  Oracle Key Vault enables you to install a certificate signed by a Certificate Authority (CA) for more secure connections.
- Step 1: Download the Certificate Request
  When you request the console certificate, you can suppress warning messages.
- Step 2: Have the Certificate Signed
  After you download the Oracle Key Vault `certificate.csr` file, you can have it signed.
- Step 3: Upload the Signed Certificate to Oracle Key Vault
  In addition to uploading the signed certificate, you can optionally choose to deactivate and re-activate the certificate.
- Console Certificates in Special Use Case Scenarios
  Depending on the situation, you must perform additional steps when you use console certificates.

16.2.1 About Managing Console Certificates

Oracle Key Vault enables you to install a certificate signed by a Certificate Authority (CA) for more secure connections.

You can upload upload a certificate that was signed by a third-party CA to Oracle Key Vault to prove its identity, encrypt the communication channel, and protect the data that is exchanged throughout the Oracle Key Vault system.

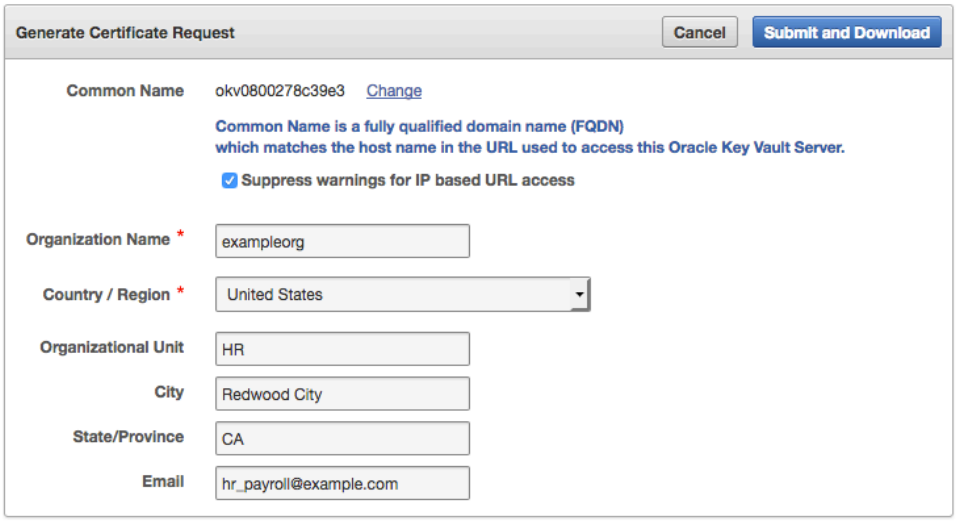
To install a console certificate, you must generate a certificate request, get it signed by a CA, and then upload the signed certificate back to Oracle Key Vault.

16.2.2 Step 1: Download the Certificate Request

When you request the console certificate, you can suppress warning messages.

These warning messages 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.

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 **Console Certificate** from the **System** menu to display the **Console Certificate** page.
3. Click **Generate Certificate Request** on the top right to display the **Generate Certificate Request** page.



4. If you need to change the host name of the Oracle Key Vault server, which appears next to **Common Name**, then click **Change**.

   The System Settings page appears. Change the host name in the Network page.

5. 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.

6. 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 rest of the optional fields as needed.

7. Click **Submit and Download** to the top right.

   A directory window appears, where you can save the `certificate.csr` file. Select a directory and save the file to a secure location.

16.2.3 Step 2: Have the Certificate Signed

After you download the Oracle Key Vault `certificate.csr` file, you can have it signed.

To have the certificate signed, you can use any out-of-band method to have it signed by a CA of your choice.

Afterward, you can then upload the signed certificate back to Oracle Key Vault using the management console.

16.2.4 Step 3: Upload the Signed Certificate to Oracle Key Vault

In addition to uploading the signed certificate, you can optionally choose to deactivate and re-activate the certificate.

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

2. Click the **System** tab and then click **Console Certificate** in the left **System** menu to display the **Console Certificate** page.

3. Click **Upload Certificate** at the top right to display the **Upload Certificate** page.

4. Click **Choose File** to display a directory window on your local system.
5. Navigate to the directory where you stored the signed certificate and select it. When you are done, you will see the file name to the right of text **Choose File**.

   After you select the certificate, you will see the file name to the right of **Choose File**.
6. Click **Upload**.

   If the certificate is installed with no errors, then you will see its details appear in a new Uploaded Certificate Details panel just below **Console Certificate**.

At this stage, if you need to, you can deactivate the certificate by clicking **Deactivate** on the top right of the **Uploaded Certificate Details** section. When you deactivate the certificate, the **Deactivate** button is replaced by an **Apply Certificate** button. You can click this button to re-activate the certificate.

16.2.5 Console Certificates in Special Use Case Scenarios

Depending on the situation, you must perform additional steps when you use console certificates.

- **Primary-standby environments:** If you want to use a console certificate in a primary-standby configuration, then you must install it on the primary and standby servers first, and then pair them.
- **RESTful services:** When you install a console certificate, you must download the RESTful software utility again before you can use the new certificate.
- **Restored data from a backup:** If you install a console certificate, perform a backup, and then restore another Oracle Key Vault appliance from that backup, you must re-install the console certificate on the new server before you can use it. The restore process does not copy the console certificate.

17
Monitoring and Auditing Oracle Key Vault

Oracle Key Vault administrators can monitor and audit the Oracle Key Vault system, configure alerts and use reports.

- Managing System Monitoring
  System monitoring refers to tasks such as configuring SNMP connections, email notifications, the syslog destination, and system diagnostics.
- Configuring Oracle Key Vault Alerts
  You can select the type of alerts that you want to see in the Oracle Key Vault dashboard.
- Managing System Auditing
  Auditing entails tasks such as capturing audit records in a syslog file or downloading the audit records to a local file.
- Using Oracle Key Vault Reports
  Oracle Key Vault collects statistical information on a range of activities that impact Key Vault operations.

17.1 Managing System Monitoring

System monitoring refers to tasks such as configuring SNMP connections, email notifications, the syslog destination, and system diagnostics.

- Configuring Remote Monitoring to Use SNMP
  With Simple Network Management Protocol (SNMP) enabled, system administrators can remotely monitor the Oracle Key Vault appliance and its services.
- Configuring Email Notification
  You can use email notifications to directly notify administrators of Key Vault status changes without logging into the Oracle Key Vault management console.
- Configuring the Syslog Destination for Individual Multi-Master Cluster Nodes
  On each node, you can forward syslog entries to a remote service such as Splunk or SIEM.
- Capturing System Diagnostics
  To troubleshoot problems that may arise, you can generate a system diagnostics file.
- Configuring Oracle Audit Vault Integration for the Node
  You can configure the integration of Oracle Audit Vault (but not the Database Firewall component) for a node.

17.1.1 Configuring Remote Monitoring to Use SNMP

With Simple Network Management Protocol (SNMP) enabled, system administrators can remotely monitor the Oracle 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
  You can use the Simple Network Management Protocol (SNMP) to monitor devices on a network for resource usage.
- Granting SNMP Access to Users
  You can grant any user, including users who are not Oracle Key Vault administrators, access to SNMP data.
- Changing the SNMP User Name and Password
  You can change the SNMP user name and password for a node at any time.
- Changing SNMP Settings on the Standby Server
  You change the SNMP settings from the command line on the standby server.
- 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.
- 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.
- 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.

17.1.1.1 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 an Oracle Key Vault multi-master cluster. The types of resource usage that you should monitor include memory, CPU utilization, and processes. Even though Oracle Key Vault provides continuous key availability by allowing up to 16 (geographically distributed) instances to be connected to a single cluster, the health of each individual node contributes to the performance and availability of the entire cluster.

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.

In a multi-master cluster, the SNMP account with a user name and password can be set for all nodes of the cluster at once. It can also be set for each individual node.

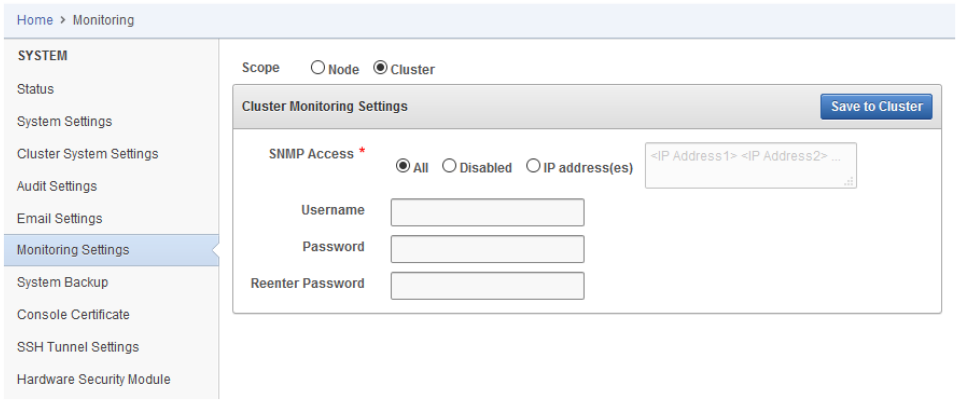
> **Note:**
>
> You must ensure that the SNMP username and password is *not* the same username and password as any of the Oracle Key Vault administrative user accounts with the System Administrator, Key Administrator, or Audit Manager role.

17.1.1.2 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**.

17.1.1.3 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**.

17.1.1.4 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"
```

17.1.1.5 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
   ```

   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
   iso.3.6.1.4.1.2021.2.1.2.5 = STRING: "ServiceManager"       <== Golden Gate
   ```

```
Service Manager (Monitors other processes and reports status)
iso.3.6.1.4.1.2021.2.1.2.6 = STRING: "adminsrvr"           <== Golden Gate
Admin Server (Communicates with the DB to perform certain maintenance/admin
tasks)
iso.3.6.1.4.1.2021.2.1.2.7 = STRING: "distsrvr"            <== Golden Gate
Distribution Server (Sends the OGG changes to other nodes)
iso.3.6.1.4.1.2021.2.1.2.8 = STRING: "recvsrvr"            <== Golden Gate
Receiver Server
```

Related Topics

- 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.

17.1.1.6 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 17-1 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: • 1: Instance is running • 2: Instance is down • 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
`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

Table 17-1 (Cont.) MIBs That SNMP Tracks for Oracle Key Vault

MIB Variable	Object ID	Description
`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`

17.1.1.7 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
   UCD-SNMP-MIB::prNames.5 = STRING: ServiceManager
   UCD-SNMP-MIB::prNames.6 = STRING: adminsrvr
   UCD-SNMP-MIB::prNames.7 = STRING: distsrvr
   UCD-SNMP-MIB::prNames.8 = STRING: recvsrvr
   ```

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
      5  ServiceManager     1     1       1     noError
noError
      6       adminsrvr     1     1       1     noError
noError
      7        distsrvr     1     1       1     noError
noError
      8        recvsrvr     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   7247856    12779404     36%
/usr/local/dbfw/tmp         6932408     15764     6916644      0%
/var/log                    5932616     19932     5912684      0%
/tmp                        1999184      3072     1996112      0%
/var/lib/oracle           143592160  35023900   108568260     24%
```

17.1.2 Configuring Email Notification

You can use email notifications to directly notify administrators of Key Vault status changes without logging into the Oracle Key Vault management console.

- About Email Notification
  Email notifications alert users of status changes and are used to complete the processes of endpoint enrollment and user password reset operations.
- Configuring Email Settings
  You can configure the Simple Mail Transfer Protocol (SMTP) server properties to receive email notifications from Oracle Key Vault.
- Testing the Email Configuration
  Oracle Key Vault management console enables you to send test emails to test the email configuration.
- Disabling Email Notifications for a User
  You can use the Oracle Key Vault management console to enable or disable email notifications.

17.1.2.1 About Email Notification

Email notifications alert users of status changes and are used to complete the processes of endpoint enrollment and user password reset operations.

To enable email notification you must set your email preferences in Oracle Key Vault. You can choose the events that you want updates to. The events include Oracle Key Vault system status like disk utilization, backup, and primary-standby, or user and endpoint status like expiration of user passwords, endpoint certificates, and keys, or cluster status like the heartbeat lag, naming conflicts, cluster-wide HSM status, and others.

Oracle Key Vault supports anonymous and insecure connections to the SMTP server. By default, Oracle Key Vault uses the default Java `truststore` packaged with Oracle Key Vault's Java library 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.

You can modify the SMTP server configuration at any time. If a custom SMTP certificate was used initially, and you later decide to use the default, you can modify the trust store setting to default, instead of custom.

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 connection between Oracle Key Vault and the SMTP server.

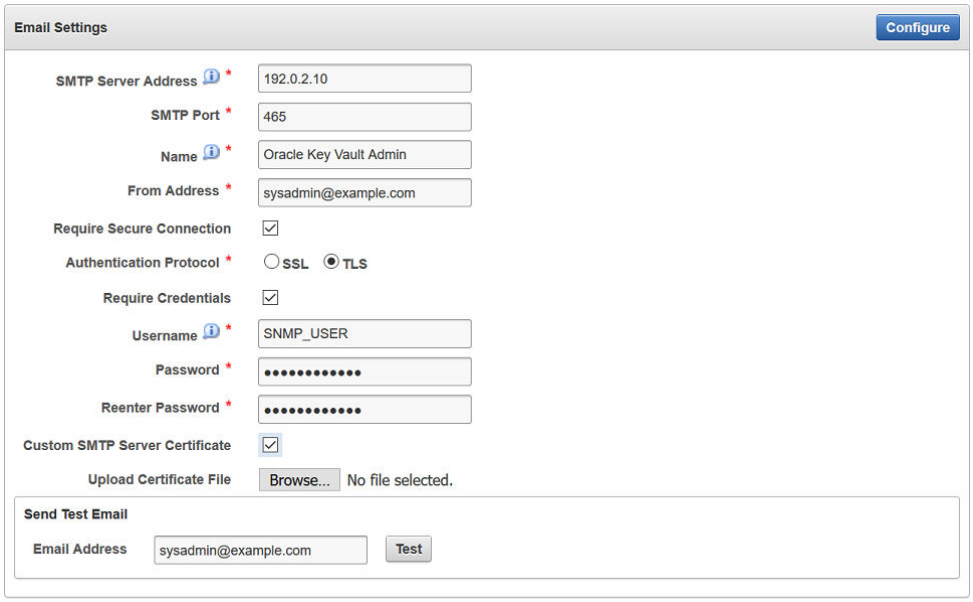
You can disable email notifications at any time.

17.1.2.2 Configuring Email Settings

You can configure the Simple Mail Transfer Protocol (SMTP) server properties to receive email notifications from Oracle Key Vault.

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.



3. In the **Email Settings** page, enter the following values:
   - **SMTP Server Address:** Enter a valid SMTP server address or host name 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 Oracle Key Vault. If you enter the SMTP hostname, you must configure DNS from the **System Settings** menu, so the host name 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 that you want to provide as a sender.
   - If the SMTP server requires a secure connection, select **Require Secure Connection**. 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, then 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:**
>
> Oracle strongly recommends that you have a secure connection to the SMTP server, because auto-generated tokens are sent over email for operations such as the creation of administrative users and Oracle Key Vault system alerts.

Do not check **Require Credentials** for non-secure connections.

- If **Custom SMTP Server Certificate** is checked, then the field **Upload Certificate File** appears with the **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 you can add a custom truststore in cases where the default Java truststore does not contain a necessary certificate. After Upload Certificate File, click **Browse** to upload a custom certificate file.

4. Click **Configure**.

   On successful configuration, a `SMTP successfully configured` message is displayed.

   If the configuration fails, then check that the SMTP server settings of the user email account are correct. Error messages highlight the field where the error has occurred to help isolate the problem.

17.1.2.3 Testing the Email Configuration

Oracle Key Vault management console enables you to send test emails to test the email configuration.

You can test the email configuration of the SMTP user account any time *after* you save the configuration. If you change an existing SMTP configuration, then you must save the configuration before you can test it.

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 before you can 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.

Related Topics

- Configuring Email Settings
  You can configure the Simple Mail Transfer Protocol (SMTP) server properties to receive email notifications from Oracle Key Vault.

17.1.2.4 Disabling Email Notifications for a User

You can use the Oracle Key Vault management console to enable or disable email notifications.

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.

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 **Save**.

17.1.3 Configuring the Syslog Destination for Individual Multi-Master Cluster Nodes

On each node, you can forward syslog entries to a remote service such as Splunk or SIEM.

- Setting the Syslog Destination Setting for the Node
  You can set the syslog destination to use either Transmission Control Protocol (TCP) or User Datagram Protocol (UDP).
- Clearing the Syslog Destination Setting for the Node
  You can clear the syslog destination setting for the node and then reset the node to the cluster setting.

17.1.3.1 Setting the Syslog Destination Setting for the Node

You can set the syslog destination to use either Transmission Control Protocol (TCP) or User Datagram Protocol (UDP).

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.

3. In the **Syslog** section, select one of the following options:
   - **TCP**: Enables syslog using the TCP protocol.
   - **UDP**: Enables syslog using the UDP protocol.
4. Enter the syslog destination IP addresses and port numbers in the **Syslog Destinations** field, in the format *`IP_address`*`:`*`port`*.

   You can enter multiple destinations, each separated by a space.
5. In the Syslog section, click **Save**.

17.1.3.2 Clearing the Syslog Destination Setting for the Node

You can clear the syslog destination setting for the node and then reset the node to the cluster setting.

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. In the Syslog section, click the **Use Cluster Settings** button.

   Clicking **Use Cluster Settings** is immediate for this setting. You do not need to click **Save** afterward.

17.1.4 Capturing System Diagnostics

To troubleshoot problems that may arise, you can generate a system diagnostics file.

- About Capturing System Diagnostics
  The Oracle Key Vault diagnostics file provides advanced debug and troubleshooting information for problems that you may encounter while using Oracle Key Vault.
- Installing the Diagnostics Generation Utility
  You can use the Oracle Key Vault management console to download instructions for installing and using the diagnostics generation utility.
- Generating a System Diagnostics File
  The system diagnostics file that you download is in a `.zip` file.
- Removing the Diagnostic Generation Utility Temporary Files
  Removing the diagnostic generation utility temporary files frees up space on your server.
- Removing the Diagnostic Generation Utility
  If you no longer need to generate system diagnostic reports, then you can remove the diagnostic generation utility.

17.1.4.1 About Capturing System Diagnostics

The Oracle Key Vault diagnostics file provides advanced debug and troubleshooting information for problems that you may encounter while using Oracle Key Vault.

You can download this file and provide it to Oracle support for further analysis and debugging. The diagnostics file includes information about free space and disk usage reported is space available to Oracle Key Vault and not based on total disk size.

Diagnostics reporting is not enabled by default. You must enable the feature to generate diagnostics reports. After you have enabled diagnostics, you can configure the necessary information to be captured in diagnostics reports. You then can customize and package diagnostics reports with flexibility. Be aware that the first time you run the diagnostic utility or after the Oracle Key Vault system's internal database has been restarted, it can take longer that it will in future runs because it must gather all the diagnostic information in the system.

If you plan to perform an upgrade of Oracle Key Vault, then you must remove the diagnostic generation utility before performing the upgrade.

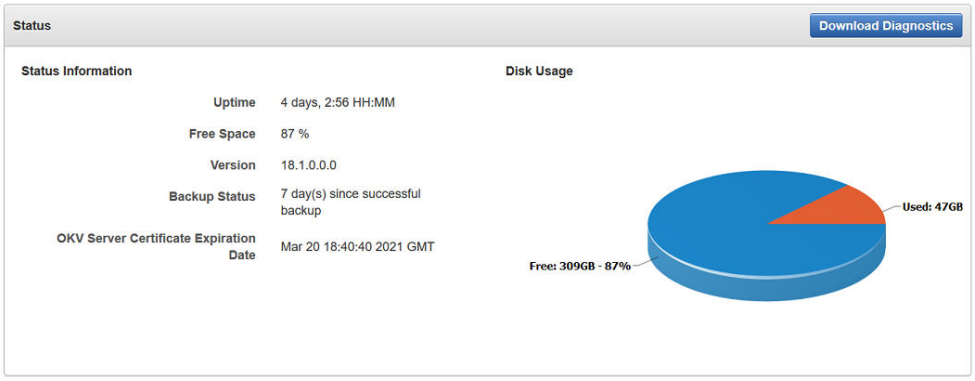
17.1.4.2 Installing the Diagnostics Generation Utility

You can use the Oracle Key Vault management console to download instructions for installing and using the diagnostics generation utility.

The instructions also explain how you can customize the output in the reports to accommodate different categories.

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

   The Status page appears.



3. Click **Download Diagnostics**.

   If the diagnostics generation utility is not installed, then you will be prompted to download the `diagnostics-not-enabled.readme` file.
4. Save the `diagnostics-not-enabled.readme` file to a local directory.
5. Follow the directions in this readme file to install and run the diagnostics generation utility, and to customize the report output.

   The readme file includes the following instructions, but you should double-check this file in case these instructions have changed:

   a. Use SSH as to connect as user `support` , then switch user (`su`) to `root` .

   b. Install the diagnostics generation utility:

   ```
   /usr/local/dbfw/bin/priv/dbfw-diagnostics-package.rb --install
   ```

   c. Enable the collection of diagnostics:

   ```
   /usr/local/dbfw/bin/priv/dbfw-diagnostics-package.rb --enable ALL
   ```

17.1.4.3 Generating a System Diagnostics File

The system diagnostics file that you download is in a `.zip` file.

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

   The Status page appears.
3. Click **Download Diagnostics**.

   You should be prompted to download a `.zip` file, which will contain the diagnostics reports. If you are prompted to download the `diagnostics-not-enabled.readme` file, then the diagnostics generation utility has not been installed and you will need to install it.
4. Download the `.zip` file that contains the diagnostic reports to a secure location.

Related Topics

- Installing the Diagnostics Generation Utility
  You can use the Oracle Key Vault management console to download instructions for installing and using the diagnostics generation utility.

17.1.4.4 Removing the Diagnostic Generation Utility Temporary Files

Removing the diagnostic generation utility temporary files frees up space on your server.

After you have run diagnostic reports, temporary files will accumulate. You should periodically remove these files. You can execute the command to remove these files from any directory.

1. Log in to the server where you downloaded and installed the diagnostic generation utility.
2. Use SSH as to connect as user `support` , then switch user (`su`) to `root` .
3. Execute the following command:

   ```
   /usr/local/dbfw/bin/priv/dbfw-diagnostics-package.rb --clean
   ```

   This command removes any `.zip` files that are found in the `/usr/local/dbfw/tmp` directory.

17.1.4.5 Removing the Diagnostic Generation Utility

If you no longer need to generate system diagnostic reports, then you can remove the diagnostic generation utility.

If you plan to upgrade Oracle Key Vault, then you must remove the diagnostic generation utility before you perform the upgrade. Removing this utility does not remove its temporary files.

1. Log in to the server where you downloaded and installed the diagnostic generation utility.
2. Use SSH as to connect as user `support` , then switch user (`su`) to `root` .

3. Execute the following command:

```
/usr/local/dbfw/bin/priv/dbfw-diagnostics-package.rb --remove
```

17.1.5 Configuring Oracle Audit Vault Integration for the Node

You can configure the integration of Oracle Audit Vault (but not the Database Firewall component) for a node.

1. Log into any Oracle Key Vault management console as a user who has the System Administrator role.
2. Select the **System** tab, and then **System Settings** from the left navigation bar.
3. Select the **Enable** check box to Oracle Audit Vault Integration for the node.
4. In the **Password** and **Reenter password** fields that appear after you click **Enable**, enter the password of the user in the database that Audit Vault and Database Firewall will use to extract the audit records.
5. Click **Save**.

17.2 Configuring Oracle Key Vault Alerts

You can select the type of alerts that you want to see in the Oracle Key Vault dashboard.

- About Configuring Alerts
  System administrators can configure alerts from the Oracle Key Vault dashboard, but all users can see alerts for their security objects.
- Configuring Alerts
  You can configure alerts in the Reports page of the Oracle Key Vault management console.
- Viewing Open Alerts
  Users can view alerts depending on their privileges.

17.2.1 About Configuring Alerts

System administrators can configure alerts from the Oracle Key Vault dashboard, but all users can see alerts for their security objects.

The Oracle Key Vault 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.

Oracle Key Vault has 17 alerts, including alerts for an HSM-enabled Oracle Key Vault server, that you can configure with appropriate thresholds according to your requirements.

You can configure the following alerts:

Table 17-2 Available Alerts

Alert Type	Applicability	Purpose
`Cluster FIPS Not Consistent`	Cluster-wide	Raised when at least one, but not all, `ACTIVE` nodes in the cluster are in FIPS mode
`Cluster Heartbeat Lag`	Node specific	Raised when a node has not received a heartbeat from another `ACTIVE` node in the cluster for over the threshold value (default 5 minutes)
`Cluster HSM Not Consistent`	Cluster-wide	Raised when at least one, but not all, `ACTIVE` nodes in the cluster are HSM-enabled
`Cluster Naming Conflict`	Cluster-wide	Raised when a naming conflict is resolved
`Cluster Redo Shipping Status`	Node specific	Raised when a read-write node is unable to ship redo to its read-write peer, and as a result, is in read-only restricted mode
`Disk Utilization`	Node specific	Raised when the free disk space percentage of the `/var/lib/oracle` partition is lower than the threshold value (default 25 percent)
`Endpoint Certificate Expiration`	Cluster-wide	Raised when an endpoint's certificate is expiring within the threshold value (default 30 days)
`Failed System Backup`	Node specific	Raised when the last backup did not complete successfully
`Primary-Standby Data Guard Broker Status`	Primary-Standby Only	Raised when the Oracle Data Guard Broker status is not `ENABLED`
`Primary-Standby Data Guard Fast-Start Failover Status`	Primary-Standby Only	Raised when the fast-start failover status is not `SYNCHRONIZED`
`Primary-Standby Destination Failure`	Primary-Standby Only	Raised when the switchover status is `FAILED DESTINATION`
`Primary-Standby Restricted Mode`	Primary-Standby Only	Raised when in primary-standby environment and the primary is running in read-only restricted mode
`Primary-Standby Role Change`	Primary-Standby Only	Raised when there is a role change
`Key Rotations`	Cluster-wide	Raised when a key's deactivation date is within the threshold value (default 7 days)
`OKV Server Certificate Expiration`	Node specific	Raised when the Oracle Key Vault server certificate is expiring within the threshold value (default 30 days)
`SSH Tunnel Failure`	Node specific	Raised when an SSH tunnel is not available
`System Backup`	Node specific	Raised when the last successful backup is over the threshold value (default 14 days)

Table 17-2 (Cont.) Available Alerts

Alert Type	Applicability	Purpose
`User Password Expiration`	Cluster-wide	Raised when a user's password will expire within the threshold value (default 14 days)
`OKV Server Certificate Expiration`	Node specific	Raised when the Oracle Key Vault server certificate is expiring within the threshold value (default 30 days)
`Invalid HSM Configuration`	Node specific	Raised when there is an error in the HSM configuration (checked by default every 5 minutes)
`Cluster Replication Lag`	Node specific	Raised when incoming replication lag is greater than the threshold value (default 60 seconds)

Related Topics

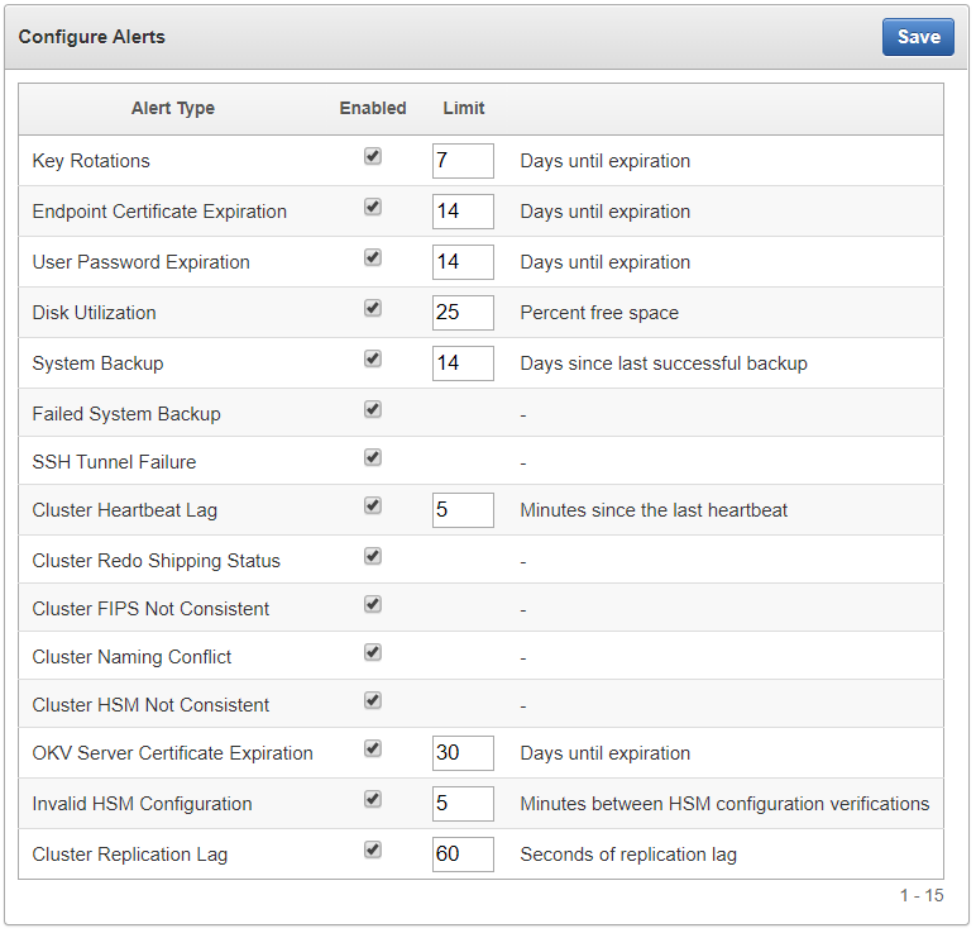
- *Oracle Key Vault Root of Trust HSM Configuration Guide*

17.2.2 Configuring Alerts

You can configure alerts in the Reports page of the Oracle Key Vault management console.

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, listing various alert types and for some, information such as the days until expiration (such as a user password expiration). If you are using a multi-master cluster, then the Configure Alerts page will provide cluster-specific alerts, such as the cluster heartbeat lag, redo shipping status, or whether naming conflicts resolution is enabled. The following image shows how the Configure Alerts page appears in a non-multi-master cluster environment.



4. Check the boxes in the **Enabled** column to the right of the alert types to enable the alert.

   Then set the threshold value in the box under **Limit**. This value determines when the alert will be sent. You can uncheck the boxes by alerts that you do not want to appear in the dashboard.

5. Click **Save**.

Related Topics

- Viewing the Oracle Key Vault Dashboard
  The dashboard presents the current status of the Oracle Key Vault at a high level and is visible to all users.

17.2.3 Viewing Open Alerts

Users can view alerts depending on their privileges.

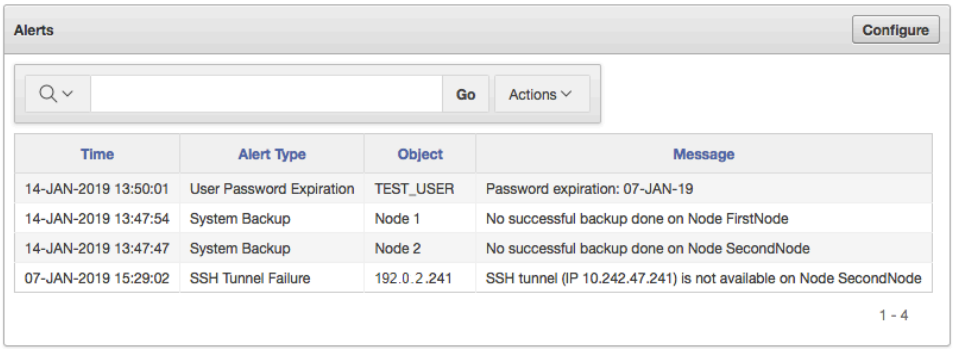
Users with the System Administrator role can view all alerts. Users without system administrator privileges can only view alerts related to objects they can access.

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. You cannot explicitly remove them.



Oracle Key Vault sends all system alerts to the syslog. The following is an example of a system alert in syslog:

```
Mar 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
Disk utilization	`Free disk space is below threshold value (currently current value)`
Endpoint certificate expiration	`Endpoint endpoint_name certificate expiration date`
Failed system backup	`Most recent backup failed!`
Key rotations	`Key unique_ID expiration: <date>`
Primary-standby destination failure	`One or more standy servers are in an error state. HA destination failure.`
Primary-standby Oracle Data Guard Broker status	`Data Guard Broker is disabled`
Primary-standby Oracle Data Guard fast-start failover status	`HA FSFO is not synchronized. FSFO status is HA_status`
Primary-standby restricted mode	`HA running in read-only restricted mode`
Primary-standby role change	`HA role changed. Primary IP Address: IP_address`
SSH tunnel failure	`SSH tunnel (IP IP_address) is not available`
System backup	`No successful backup for number day(s)`

Condition	System Alert Message
User password expiration	`User user_name password expiration: date`
Invalid HSM Configuration	`HSM configuration error. Please refer to the HSM Alert section in the Oracle Key Vault HSM Integration Guide`
Cluster Replication Lag	`Replication lag from node node_name to node node_name is greater than number seconds.`

17.3 Managing System Auditing

Auditing entails tasks such as capturing audit records in a syslog file or downloading the audit records to a local file.

- About Auditing in Oracle Key Vault
  Oracle Key Vault records and time-stamps all endpoint and user activity.
- Configuring Syslog to Store Audit Records
  You can configure the Oracle Key Vault syslog to store audit records if the System Administrator has enabled this functionality.
- Configuring Audit Settings for a Multi-Master Cluster
  You can enable or disable auditing for a multi-master cluster.
- Viewing Audit Records
  To view audit records, access the Oracle Key Vault management console Audit Trail page.
- Exporting and Deleting Audit Records
  Oracle Key Vault audit records are stored in a `.csv` file.
- 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.

17.3.1 About Auditing in Oracle Key Vault

Oracle Key Vault records and time-stamps all endpoint and user activity.

The audit records include 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. The audit trail captures details on who initiated which action, with what keys and tokens, and the result of the action. In addition, it records the success or failure of each action.

Only a user who has the Audit Manager role can manage the audit trail for Oracle Key Vault activity. Each user can see audit records of the objects that the user can access.

Auditing happens in the background, and is always enabled in Oracle Key Vault. It cannot be disabled.

A user with the Audit Manager role can see and manage all the audit records. Other users can see only those audit records that pertain to security objects that they have created, or have been granted access to.

The audit manager can export audit records to view system activity off line. After exporting the records, the audit manager can delete them from the system to free up resources.

Related Topics

- Audit Manager Role Duties
  The Oracle Key Vault Audit Manager is responsible for audit-related tasks.

17.3.2 Configuring Syslog to Store Audit Records

You can configure the Oracle Key Vault syslog to store audit records if the System Administrator has enabled this functionality.

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.
3. Click the **Audit Settings** button.

   The Audit Settings page appears.

4. Enter the following settings:
   - **Scope**: Select **Node** to restrict the audit records to those that were generated in the current node, or select **Cluster** to capture audit records for the entire multi-master cluster environment.
   - **Send Audit Records To Syslog**: Click **Yes**.
5. Click **Save**.

   If syslog is not configured, then the `Syslog forwarding to remote machines not enabled` error message appears. If this error appears, dismiss the error dialog and go to the next step.
6. If syslog is configured, then do the following:
   a. Select the **System** tab, and then select **System Settings**.
   b. In the Settings page, go to the Syslog pane.
   c. Select the protocol to use to transfer syslog files: **TCP** or **UDP**.

d. Enter the IP address of the remote system where the syslog files will be stored.

7. Click **Save**.

Related Topics

- Configuring Syslog
  In a non-multi-master cluster environment, you can enable syslog for specific destinations and transmit the records either using Transmission Control Protocol (TCP) or User Datagram Protocol (UDP).

17.3.3 Configuring Audit Settings for a Multi-Master Cluster

You can enable or disable auditing for a multi-master cluster.

You can also enable or disable replicating the audit records to other nodes in the cluster and saving the syslog to its configured destination.

1. Log into any Oracle Key Vault management console as a user who has the Audit Manager role.
2. Select the **System** tab, and then **Audit Settings** from the left navigation bar.
3. For **Scope**, select **Cluster**.
4. In the Cluster Audit Settings section, select the **Yes** or **No** option for each of these actions:
   - **Enable Auditing**: Enables or disables auditing for all nodes in the cluster.
   - **Replicate Audit Records**: Enables or disables replication of audit records to all nodes in the cluster.
   - **Send Audit Records to Syslog**: Enables or disables sending of audit records to the configured syslog location, as configured by a user with the System Administrator role.
5. Click **Save**.

   Saving settings in the **Node** scope overrides the cluster settings for this node.

17.3.4 Viewing Audit Records

To view audit records, access the Oracle Key Vault management console Audit Trail page.

The reports page shows the Audit Trail page by default. The Audit Trail page lists all system activity with details on who performed an operation, when the operation was performed, what object was used to perform the operation, and the result.

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. Optionally, filter records by selecting the table column heads, and from the drop-down list, select the type of sort order that you want.

17.3.5 Exporting and Deleting Audit Records

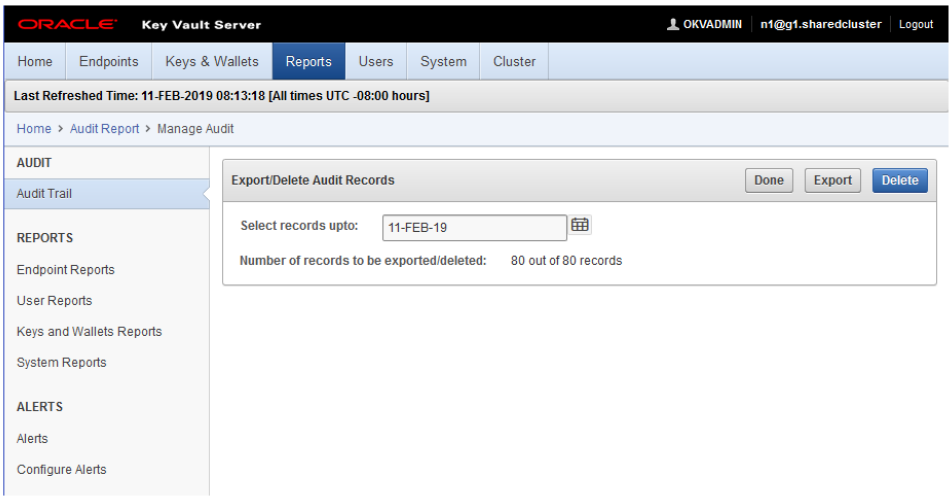
Oracle Key Vault audit records are stored in a `.csv` file.

A user with the Audit Manager role can export the audit trail to 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. The timestamp in the `.csv file` reflects the time zone of the particular Oracle Key Vault server whose records were exported. After you export the records, you can delete them from the Oracle Key Vault server to free up space.

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 appears.
3. Click **Export/Delete Audit Records** on the top right.

   The **Export/Delete Audit Records** page appears.



4. Select the date by clicking the calendar icon.

   Based on the date that you select, the number of records appears after the **Number of records to be exported/deleted** label.
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 Oracle Key Vault to free up resources.
6. Click **Delete** to remove the audit records.
7. Click **OK** to delete or **Cancel** to stop.

17.3.6 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.

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.

   Store this password in a safe place. You will need it when you create a secured target on the Audit Vault and Database FirewallAVDF side.

Related Topics

- Integrating Oracle Key Vault with Oracle Audit Vault and Database Firewall
  You can consolidate audits between Oracle Audit Vault and Database Firewall (AVDF) with Oracle Key Vault.

17.4 Using Oracle Key Vault Reports

Oracle Key Vault collects statistical information on a range of activities that impact Key Vault operations.

- About Oracle Key Vault Reports
  The reports cover system activity, certificate expiration, keys, passwords, entitlement status, and metadata.
- Viewing Endpoint Reports
  You must have the Audit Manager role to view the four categories of endpoint reports.
- Viewing User Reports
  You must have the Audit Manager role to view the four categories of user reports.
- Viewing Keys and Wallets Reports
  You must have the Audit Management role to view the two categories of keys and wallets reports.
- Viewing System Reports
  You must have the Audit Manager role to view the three categories of system reports.

17.4.1 About Oracle Key Vault Reports

The reports cover system activity, certificate expiration, keys, passwords, entitlement status, and metadata.

Oracle Key Vault provides four types of reports for endpoints, users, keys and wallets, and system. In a multi-master cluster, some reports contain additional information, such as the node ID, node name, and IP address.

The four report types are as follows:

- 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 encryption keys managed by Oracle Key Vault.
- System reports contain a history of system backups taken and scheduled, details of remote restoration points, and RESTful API usage.

A user who has the Audit Manager role can view all reports, including reports that are accessible from the Audit Trail pages in the Oracle Key Vault management console. A user with the Key Administrator role can view user reports and keys and wallets reports. Users with the System Administrator role can view endpoint, user, and system reports.

Related Topics

- Managing System Auditing
  Auditing entails tasks such as capturing audit records in a syslog file or downloading the audit records to a local file.

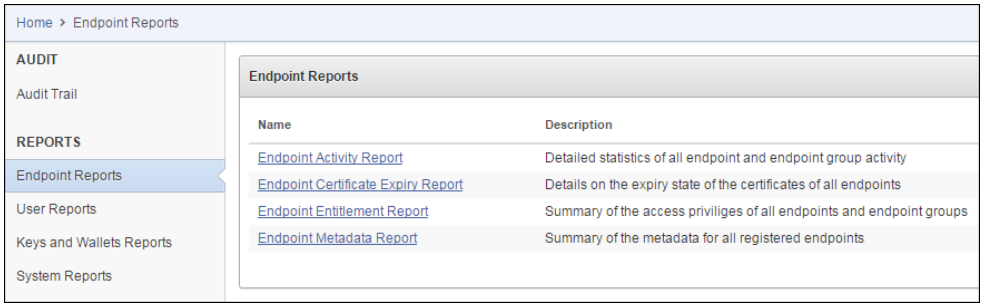
17.4.2 Viewing Endpoint Reports

You must have the Audit Manager role to view the four categories of endpoint reports.

Oracle Key Vault offers four endpoint reports: Endpoint Activity, Endpoint Certificate Expiry, Endpoint Entitlement, and Endpoint Metadata.

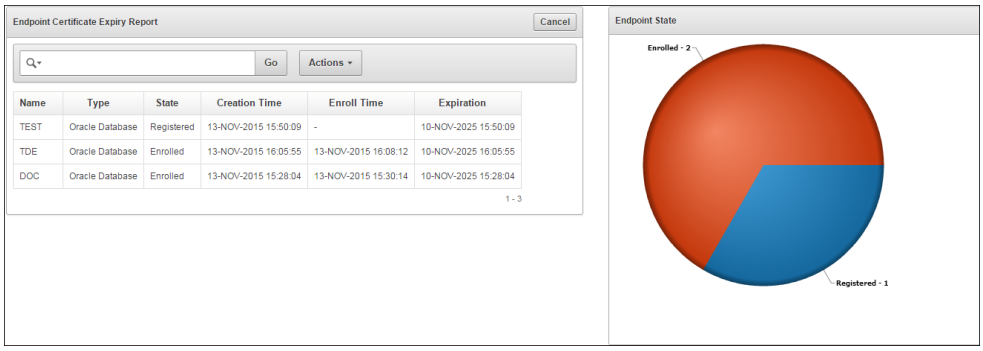
1. Log in to the Oracle Key Vault management console as a user who has the Audit Manager role.
2. Click the **Reports** tab to display the **Reports** page.
3. Click **Endpoint Reports** under **Reports** in the left sidebar.

   The **Endpoint Reports** page appears displaying four endpoint report types.



4. Click the link under **Name** to view the report that you want.

   For example, the Endpoint Certificate Expiry report appears similar to the following:



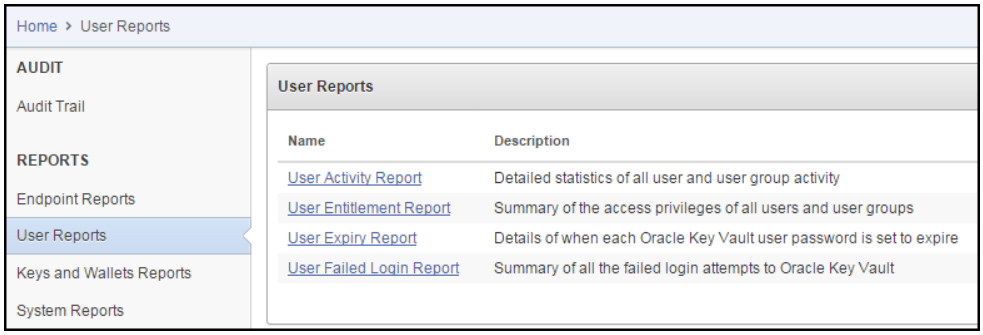
17.4.3 Viewing User Reports

You must have the Audit Manager role to view the four categories of user reports.

Oracle Key Vault offers four user reports: User Activity, User Entitlement, User Expiry, and User Failed Login.

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.



4. Click the report name to see the corresponding user report.

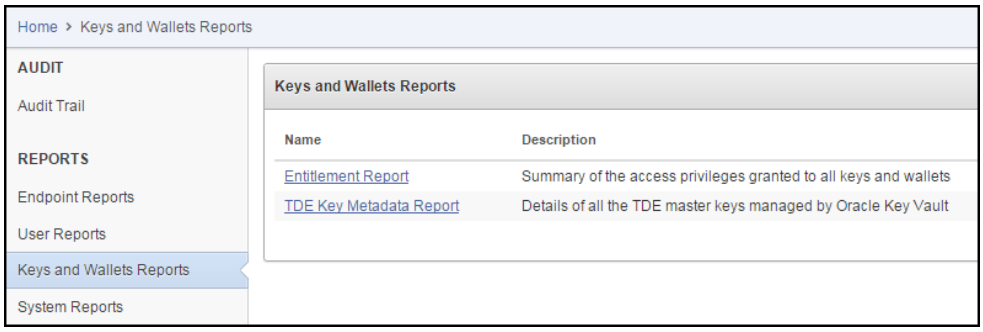
17.4.4 Viewing Keys and Wallets Reports

You must have the Audit Management role to view the two categories of keys and wallets reports.

Oracle Key Vault offers two reports for keys and wallets: Entitlement and TDE Key Metadata.

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.



4. Click the report name to see the corresponding report.

17.4.5 Viewing System Reports

You must have the Audit Manager role to view the three categories of system reports.

Oracle Key Vault offers three system reports: Backup History, Backup Restoration Catalog, and RESTful API Usage.

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.

4. Click the report type to see the corresponding system report.

18

Managing Security Objects in Oracle Key Vault

Managing security objects includes uploading and downloading security objects, managing the persistent master encryption key cache, and using user-defined TDE keys.

- Configuring an Oracle Key Vault-to-New TDE-Enabled Database Connection
  You can configure a connection between Oracle Key Vault and a database that has not yet been configured for Transparent Data Encryption.
- Migrating Existing TDE Wallets to Oracle Key Vault
  A migrated TDE wallet can be used to restore database contents that were previously encrypted by TDE.
- Using the Persistent Master Encryption Key Cache
  The persistent master encryption key cache feature enables databases to be operational when the Oracle Key Vault server is unavailable.
- Uploading and Downloading Oracle Wallets
  To store and share Oracle wallets, you must upload them to Oracle Key Vault.
- Uploading and Downloading JKS and JCEKS Keystores
  The `okvutil upload` and `okvutil download` commands can upload and download JKS and JCEKS keystores.
- Uploading and Downloading Credential Files
  The `okvutil upload` and `okvutil download` commands can upload and download credential files.
- Using a User-Defined Key as the TDE Master Encryption Key
  You can import a generated key to be used as the Transparent Data Encryption (TDE) master encryption key in Oracle Key Vault.

18.1 Configuring an Oracle Key Vault-to-New TDE-Enabled Database Connection

You can configure a connection between Oracle Key Vault and a database that has not yet been configured for Transparent Data Encryption.

- About Configuring an Oracle Key Vault-to-New TDE-Enabled Database Connection
  You can configure a connection between Oracle Key Vault and a database that has not yet been configured for TDE.
- Limitations to Transparent Data Encryption Endpoint Integration
  This type of Transparent Data Encryption (TDE) endpoint integration can have problems if the versions are incompatible.

- Step 1: Configure the Oracle Key Vault Server Environment
  Before you can configure the connection, you must ensure that Oracle settings are correct.
- Step 2: Integrate Transparent Data Encryption with Oracle Key Vault
  This integration enables Oracle Key Vault to directly manage the TDE master encryption keys.

18.1.1 About Configuring an Oracle Key Vault-to-New TDE-Enabled Database Connection

You can configure a connection between Oracle Key Vault and a database that has not yet been configured for TDE.

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` must be set to the same values in `svrctl` and operating system environment variables. This also applies if you are using the Oracle Key Vault RESTful services utility to enroll endpoints.

For information about configuring Oracle Data Guard so that it can work with TDE and Oracle Key Vault:

- *Oracle Database Advanced Security Guide*, release 19c
- *Oracle Database Advanced Security Guide*, release 18c
- *Oracle Database Advanced Security Guide*, release 12.2.0.1

For information about configuring Oracle Real Application Clusters (Oracle RAC) to work with TDE and Oracle Key Vault:

- *Oracle Database Advanced Security Guide*, release 19c
- *Oracle Database Advanced Security Guide*, release 18c
- *Oracle Database Advanced Security Guide*, release 12.2.0.1

18.1.2 Limitations to Transparent Data Encryption Endpoint Integration

This type of Transparent Data Encryption (TDE) endpoint integration can have problems if the versions are incompatible.

The limitations to TDE endpoint integration are as follows:

- All endpoints on the same computer must use the same version of the Oracle Key Vault library. There is only one location per computer for the `liborapkcs.so` file, which is `/opt/oracle/expapi/64/hsm/oracle/1.0.0/liborapkcs.so`.
- On the same computer, you should use the same external key manager for the Oracle database, either Oracle Key Vault or an HSM. Using Oracle Key Vault for one Oracle database and an 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, which is `/opt/oracle/expapi/64/hsm/`.

> **Caution:**
>
> Oracle strongly recommends that you never remove keys from a wallet or the wallet itself after TDE is configured. Loss of keys will result in the loss of encrypted data and hamper the normal functioning of the database. This is true even in the folowing scenarios:
>
> - 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

18.1.3 Step 1: Configure the Oracle Key Vault Server Environment

Before you can configure the connection, you must ensure that Oracle settings are correct.

These settings include Oracle environment variables and the Oracle `COMPATIBILITY` parameter.

1. Log in to the server where the database endpoint is installed.
2. Ensure that the `ORACLE_BASE` environment variable is set before you start the `oracle` process manually.

   This step is very important. If the `ORACLE_BASE` environment variable is not present, then 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, you must perform this step on all database instances.
3. Ensure that the `COMPATIBILITY` initialization parameter is set to `11.2.0.0` or later.
4. Enroll and provision the endpoint for the Transparent Data Encryption (TDE)-enabled database that contains the TDE data.

   When you initially enroll the endpoint, select `ORACLE_DB` as the endpoint type for integration with TDE.
5. Ensure that the endpoint has access to the virtual wallet that you want to use by defining the wallet as the default wallet for that endpoint.

   In Oracle Real Application Clusters (Oracle RAC) and Oracle Data Guard environments, the endpoints of all instances share the same default wallet. The endpoint must have the read, modify, and manage wallet access.
6. Depending on the Oracle Database release, configure the encryption wallet location.
   - For release 12.2.0.1 and earlier, onfigure the `sqlnet.ora` file on this database to point to Oracle Key Vault as follows:

     ```
     ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=OKV))
     ```

     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.
   - For release 18c and later, configure the following parameters:

- Set `WALLET_ROOT` to define the installation directory of the Oracle Key Vault client.
- Set `TDE_CONFIGURATION` to `OKV` for Oracle Key Vault.

Endpoints use the PKCS#11 library support to manage TDE master encryption keys. Execute the `root.sh` script (in the *`okvclient_installation_directory`*`/bin` directory) to deploy the PKCS#11 library that facilitates the communication between the endpoint and Oracle Key Vault.

At this stage, the Oracle database will use Oracle Key Vault for centralized key management. For all TDE commands and statements, use the Oracle Key Vault endpoint password that was specified during the endpoint enrollment process.

7. 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 connect again.

8. Query the `V$ENCRYPTION_WALLET` dynamic view to ensure that the `METHOD_DATA` setting in the `sqlnet.ora file` changed.

   ```
   SELECT * FROM V$ENCRYPTION_WALLET;
   ```

   The output of the query should now show `OKV`.

The TDE configuration is complete at this stage. You can now encrypt tables or create encrypted tablespaces in the database. Oracle does not recommend using TDE column encryption with an external key manager. (This applies to not only Oracle Key Vault but to all third-party key managers and HSMs.) If you have configured the `sqlnet.ora` file correctly along with the rest of the TDE configuration, then a TDE master encryption key is created in Oracle Key Vault when you set the encryption key by using one of the following SQL statements:

- `ALTER SYSTEM SET [ENCRYPTION] KEY IDENTIFIED BY "`*`password`*`";`
- `ADMINISTER KEY MANAGEMENT SET [ENCRYPTION] KEY IDENTIFIED BY "`*`password`*`" WITH BACKUP;`

With both of these SQL statements, the password was defined when the Oracle Key Vault client software was installed. If no password was defined at that time, then the password for these two statements is `NULL`.

18.1.4 Step 2: Integrate Transparent Data Encryption with Oracle Key Vault

This integration enables Oracle Key Vault to directly manage the TDE master encryption keys.

1. Ensure that you are logged into the server where the Oracle Key Vault client is installed.
2. Execute the `root` script from *`okvclient_installation_directory`*`/bin`.

   This script implements the persistent master encryption key cache feature in the Oracle Key Vault PKCS#11 library, which improves the availability of the database during intermittent network disruptions or Oracle Key Vault upgrade.

- **UNIX:** The root script copies the PKCS#11 library into `/opt/oracle/extapi/64/hsm/oracle/1.0.0` and minimizes the file access privileges.
- **Windows:** Run the `root.bat` script 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.

3. For password-protected wallets on the database, open the wallet. (Auto-login wallets are automatically opened.)
   - **For Oracle Database 11g release 2:** As a user who has been granted the `ALTER SYSTEM` system privilege:

     ```
     ALTER SYSTEM SET ENCRYPTION WALLET OPEN IDENTIFIED BY
     "Oracle_Key_Vault_endpoint_password";
     ```

   - **For Oracle Database 12c or later:** As a user who has been granted the `SYSKM` administrative privilege:

     ```
     ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN IDENTIFIED BY
     "Oracle_Key_Vault_endpoint_password" WITH BACKUP;
     ```

4. Set the master encryption key.
   - **For Oracle Database 11g release 2:**

     ```
     ALTER SYSTEM SET ENCRYPTION KEY IDENTIFIED BY
     "Key_Vault_endpoint_password";
     ```

   - **For Oracle Database 12c and later:**

     ```
     ADMINISTER KEY MANAGEMENT SET ENCRYPTION KEY IDENTIFIED BY
     "Key_Vault_endpoint_password" WITH BACKUP;
     ```

Related Topics

- Enrolling Endpoints for Oracle Key Vault
  After an endpoint is registered in Oracle Key Vault, an endpoint administrator enrolls and provisions the endpoint to manage security objects in Key Vault.
- Endpoint Database Requirements
  For endpoints, Oracle Key Vault supports Oracle Database release 10 and later.
- Managing Endpoints
  You can enroll, reenroll, suspend, and delete endpoints.
- Granting Access to Users, User Groups, Endpoints, and Endpoint Groups
  You can grant the **Read Only**, **Read and Modify**, and **Manage Wallet** access levels to users, user groups, endpoints, and endpoint groups.

18.2 Migrating Existing TDE Wallets to Oracle Key Vault

A migrated TDE wallet can be used to restore database contents that were previously encrypted by TDE.

- About Migrating Existing TDE Wallets to Oracle Key Vault
  The `sqlnet.ora` file enables the migration of existing TDE wallets to Oracle Key Vault.
- 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.

- Restoring Database Contents Previously Encrypted by TDE Using an Oracle Wallet
  You perform the restoration process on the endpoint where you downloaded the Oracle wallet.

18.2.1 About Migrating Existing TDE Wallets to Oracle Key Vault

The `sqlnet.ora` file enables the migration of existing TDE wallets to Oracle Key Vault.

When the Transparent Data Encryption (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 Oracle Key Vault.

Along with the current TDE master encryption key, Oracle wallets maintain historical TDE master encryption keys that are replaced by each rekey operation that rotates the TDE master encryption key. These historical TDE master encryption keys help to restore Oracle Database backups that were previously made using one of the historical TDE master encryption keys. During the TDE migration from an Oracle wallet file to Oracle Key Vault, Key Vault generates new master encryption keys. After this master encryption key generation, Oracle Key Vault maintains all new keys.

Oracle recommends that you upload the Oracle wallet to Oracle 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 Real Application Clusters (Oracle RAC) configuration, then you must restart the database so that the new TDE master encryption key is updated to all the endpoint database nodes in the shared server configuration.

Related Topics

- Restoring Database Contents Previously Encrypted by TDE Using an Oracle Wallet
  You perform the restoration process on the endpoint where you downloaded the Oracle wallet.

18.2.2 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.

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 use the `okvutil upload` command to upload current and historical master encryption keys from the TDE wallet into the virtual wallet in Oracle Key Vault.

   ```
   $ OKV_HOME/bin/okvutil upload -t WALLET -l /path/to/tde-wallet -g
   name_of_wallet_in_Oracle_Key_Vault -v 4
   ```

   This step ensures that Oracle Key Vault has a copy of the wallet that contains all of the historical TDE master encryption keys.
4. Configure the wallet location.

- **For Oracle Database releases 12.2.0.1 and earlier:** Edit the `METHOD` setting of the Oracle Database `sqlnet.ora` file for the migration to Oracle Key Vault:

  ```
  ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=OKV)
  (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.

- **For Oracle Database release 18c and later:** Set the `TDE_CONFIGURATION` parameter.

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

  The `WALLET_ROOT` parameter is already so, so you do not need to change it.

5. Reconnect to the database instance (or restart an Oracle RAC database) to capture the changes in the `sqlnet.ora` file.

   If you are reconnecting to the database instance, then changes to the `TDE_CONFIGURATION` parameter in Oracle Databae 18c and later are applied immediately, and no restart is necessary.

6. Query the `V$ENCRYPTION_WALLET` dynamic view to ensure that the `METHOD_DATA` setting in the `sqlnet.ora file` changed.

   ```
   SELECT * FROM V$ENCRYPTION_WALLET;
   ```

   The output of the query should now show `METHOD=OKV`.

7. If the endpoint is a an Oracle Release 11g release 2 database, then close the local Oracle wallet and open the HSM wallet as follows:

8. Close the local Oracle wallet using these steps:

   a. Close the local Oracle wallet as follows:

      If the auto-login wallet is open, execute the following commands:

      ```
      oracle$ cd wallet_location
      oracle$ mv cwallet.sso cwallet.sso.bak
      sqlplus sys as sysdba
      Enter password: password
      SQL> ALTER SYSTEM SET WALLET CLOSE;
      ```

      If the password-protected wallet is open, then execute the following statement in SQL*Plus:

      ```
      ALTER SYSTEM SET WALLET CLOSE IDENTIFIED BY "wallet_password";
      ```

   b. In SQL*Plus, open the HSM wallet, open the connection into Oracle Key Vault, where *Oracle_Key_Vault_password* is the password that was provided when the `okvclient.jar` file was installed:

      ```
      ALTER SYSTEM SET WALLET OPEN IDENTIFIED BY "Oracle_Key_Vault_password";
      ```

9. Migrate from TDE wallets to Oracle Key Vault.

   - **For Oracle Database 11g release 2:** If you entered a password for the wallet while installing the endpoint client software, then in SQL*Plus, execute this statement:

```
ALTER SYSTEM SET ENCRYPTION KEY
IDENTIFIED BY "Oracle_Key_Vault_password"
MIGRATE USING "wallet_password";
```

If you chose the auto-login option while installing the endpoint client software, then execute this statement:

```
ALTER SYSTEM SET ENCRYPTION KEY
IDENTIFIED BY "NULL"
MIGRATE USING "wallet_password";
```

- **For Oracle Database 12c and later:** As a user who has been granted the `SYSKM` administrative privilege:

```
ADMINISTER KEY MANAGEMENT SET ENCRYPTION KEY IDENTIFIED BY
"Oracle_Key_Vault_password" MIGRATE USING "wallet_password" WITH BACKUP;
```

Though the `WITH BACKUP` clause is required for the `ADMINISTER KEY MANAGEMENT` statement, it is ignored by TDE in Oracle Key Vault.

10. Open the wallet.
    - **For Oracle Database 11g release 2:** If the endpoint requires a password to connect to Oracle Key Vault, then enter the password.

```
ALTER SYSTEM SET ENCRYPTION WALLET OPEN
IDENTIFIED BY "Oracle_Key_Vault_endpoint_password";
```

    - **For Oracle Database 12c and later:**

```
ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN
IDENTIFIED BY "Oracle_Key_Vault_endpoint_password";
```

11. 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`.

Related Topics

- *Oracle Database Backup and Recovery User's Guide*
- About Endpoint Enrollment and Provisioning
  Endpoints are Key Vault clients that use the server to store and manage security objects, share them with trusted peers, and retrieve them.
- okvutil download Command
  The `okvutil download` command downloads security objects from Oracle Key Vault to the endpoint

18.2.3 Restoring Database Contents Previously Encrypted by TDE Using an Oracle Wallet

You perform the restoration process on the endpoint where you downloaded the Oracle wallet.

When an Oracle database endpoint is converted from a local Oracle wallet file to using Oracle Key Vault, you may need to restore backups that were encrypted by 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, then 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.

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, edit the `sqlnet.ora` file to have the following setting:

   ```
   ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=FILE)
   (METHOD_DATA=(DIRECTORY=wallet_file_path)))
   ```

   Put the `ENCRYPTION_WALLET_LOCATION` setting on one line.
3. Open the downloaded wallet using the password you specified.
   - For Oracle Database 11g 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 12c and later, 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.

Related Topics

- okvutil download Command
  The `okvutil download` command downloads security objects from Oracle Key Vault to the endpoint

18.3 Using the Persistent Master Encryption Key Cache

The persistent master encryption key cache feature enables databases to be operational when the Oracle Key Vault server is unavailable.

- About the Persistent Master Encrption Key Cache
  The persistent master encryption key cache ensures the availability of TDE master encryption keys.
- About Oracle Key Vault Persistent Master Encyrption Key Cache Architecture
  The Oracle Key Vault persistent master encryption key cache is implemented in Oracle Key Vault's `PKCS#11` library.
- Caching Master Encryption Keys in the In-Memory and Persistent Master Encryption Key Cache
  After a master encryption key is created or fetched from a different location, it is stored in an Oracle Key Vault cache.
- Storage Location of Persistent Master Encryption Key Cache
  The persistent master encryption key cache is created in the same location as the configuration file `okvclient.ora`.

- Persistent Master Encryption Key Cache Modes of Operation
  The persistent master encryption key cache operates in two modes.
- Persistent Master Encryption Key Cache Refresh Window
  The persistent master encryption key cache refresh window helps to extend the availability of the master encryption key.
- Persistent Master Encryption Key Cache Parameters
  Oracle Key Vault provides parameters to configure the persistent master encryption key cache.
- Listing the Contents of the Persistent Master Encryption Key Cache
  The `okvutil list` command lists the master encryption keys that are cached in the persistent master encryption key cache.
- Oracle Database Deployments and Persistent Master Encryption Key Cache
  The persistent master encryption key cache affects the integration of other Oracle features with Oracle Key Vault.

18.3.1 About the Persistent Master Encrption Key Cache

The persistent master encryption key cache ensures the availability of TDE master encryption keys.

It accomplishes this by reducing dependence on the state of the Oracle Key Vault server.

The TDE master encryption key is cached in the persistent master encryption key cache in addition to the in-memory cache, to make the master encryption 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 encryption keys:

- Continuous operation of endpoints during upgrade, primary-standby configuration, switchover, failover, and other procedures that require an Oracle Key Vault restart operation
- 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

18.3.2 About Oracle Key Vault Persistent Master Encyrption Key Cache Architecture

The Oracle Key Vault persistent master encryption key cache is implemented in Oracle Key Vault's `PKCS#11` library.

When the persistent master encryption key cache feature is configured, the Oracle Key Vault `PKCS#11` library will create the persistent master encryption key cache when the first master encryption key is retrieved from Oracle Key Vault.

The persistent master encryption 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 encryption key cache is a password-based wallet.

- If Oracle Key Vault is installed without a password specified, then the persistent master encryption key cache is an auto-login wallet.

The PKCS#11 library also implements an in-memory master encryption key cache. When the in-memory master encryption key cache feature is configured, the master encryption key is cached in the process memory of the process that loaded the library into memory. The in-memory and persistent master encryption key caches are independent of each other. You can enable and disable these caches independently.

For operations that involve encryption and decryption, PKCS#11 attempts to look up the master encryption key in the in-memory master encryption key cache. If it does not find it, PKCS#11 then it looks up the master encryption key in the persistent master encryption key cache. If the master encryption key is not found in the in-memory or the persistent master encryption key cache, then it is retrieved from the Oracle Key Vault server, if the server is online.

18.3.3 Caching Master Encryption Keys in the In-Memory and Persistent Master Encryption Key Cache

After a master encryption key is created or fetched from a different location, it is stored in an Oracle Key Vault cache.

When the master encryption key is first fetched from the Oracle Key Vault server, or created in the Oracle Key Vault server, the master encryption key is stored in the in-memory master encryption key cache and in the persistent master encryption key cache.

Master encryption keys stored in the in-memory master encryption key cache are available for a limited time from the moment the key is placed into the persistent cache. The duration is defined by the PKCS11_CACHE_TIMEOUT parameter in the okvclient.ora file.

If the persistent cache exists, then it will be used. If the persistent cache does not exist, then Oracle Key Vault creates it. When the key is created, all future sessions will retrieve it from the in-memory master encryption key cache or persistent master encryption key cache.

Persistent master encryption keys that are stored in the persistent master encryption key cache are available for a limited time from the moment the key is placed into the persistent cache. You can define this time by setting the PKCS11_PERSISTENT_CACHE_TIMEOUT parameter in the okvclient.ora file.

When the endpoint deletes the master encryption key, the key will be removed from the in-memory master encryption key cache and persistent master encryption key cache.

18.3.4 Storage Location of Persistent Master Encryption Key Cache

The persistent master encryption key cache is created in the same location as the configuration file okvclient.ora.

The default location for the okvclient.ora file is the directory $OKV_HOME/conf.

It is important that the ORACLE_HOME, ORACLE_BASE, and OKV_HOME environment variables are consistently set across the deployment. If they are not consistent, then

operations requiring the persistent cache may fail, and the persistent cache may be created in multiple locations.

If the environment variable `OKV_HOME` is set, then the persistent cache is created in `$OKV_HOME/conf`.

If `OKV_HOME` is not set, but `ORACLE_BASE` is set, then the persistent cache is created in `$ORACLE_BASE/okv/$ORACLE_SID`.

If neither `OKV_HOME` nor `ORACLE_BASE` is set, but `ORACLE_HOME` is set, then the persistent cache is created in `$ORACLE_HOME/okv/$ORACLE_SID`.

> **Note:**
>
> Ensure that the directory in which the persistent cache is created is secure and has restricted permissions.

18.3.5 Persistent Master Encryption Key Cache Modes of Operation

The persistent master encryption key cache operates in two modes.

The difference between the two modes is the order in which the persistent master encryption key cache and Oracle Key Vault are looked up to retrieve the master encryption key.

- 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.
- Persistent Master Encryption Key Cache First Mode
  In persistent master encryption key cache first mode, the endpoints retrieve the master encryption key from the persistent master encryption key cache.

18.3.5.1 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 encryption key cache.

The endpoints must determine the status of the Oracle Key Vault server, and if it is offline, then the endpoints attempt to retrieve the master encryption key from the persistent master encryption key cache. Hence, database operations that require access to master encryption keys will experience a delay.

18.3.5.2 Persistent Master Encryption Key Cache First Mode

In persistent master encryption key cache first mode, the endpoints retrieve the master encryption key from the persistent master encryption key cache.

If the master encryption key is not available in the persistent master encryption 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 encryption key cache.

18.3.6 Persistent Master Encryption Key Cache Refresh Window

The persistent master encryption key cache refresh window helps to extend the availability of the master encryption key.

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

The refresh window feature of the persistent master encryption key cache therefore extends the duration for which the master encryption key is available after it is cached in the persistent master encryption 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 during the time when the Oracle Key Vault is unavailable, such as when a primary-standby switchover is in progress. The refresh window terminates and then the cache period begins as soon as the key is refreshed.

In the `okvclient.ora` file, you can use the `PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW` parameter to extend the duration for which the master encryption key is available after it is cached in the persistent master encryption key cache. This value reflects the amount of time it takes for the Oracle Key Vault server to recover and return online. You must specify this value in minutes. The default value for `PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW` is `30` (minutes).

Related Topics

- PKCS11_PERSISTENT_CACHE_TIMEOUT Parameter
  The `PKCS11_PERSISTENT_CACHE_TIMEOUT` parameter sets how long the master encryption is available in the persistent cache.

18.3.7 Persistent Master Encryption Key Cache Parameters

Oracle Key Vault provides parameters to configure the persistent master encryption key cache.

- PKCS11_CACHE_TIMEOUT Parameter
  The `PKCS11_CACHE_TIMEOUT` parameter sets how long a master encryption key is available in the in-memory cache.
- PKCS11_PERSISTENT_CACHE_TIMEOUT Parameter
  The `PKCS11_PERSISTENT_CACHE_TIMEOUT` parameter sets how long the master encryption is available in the persistent cache.
- PKCS11_PERSISTENT_CACHE_FIRST Parameter
  The `PKCS11_PERSISTENT_CACHE_FIRST` parameter sets the persistent master encryption key cache operation mode.
- PKCS11_CONFIG_PARAM_REFRESH_INTERVAL Parameter
  The `PKCS11_CONFIG_PARAM_REFRESH_INTERVAL` parameter describes the frequency at which a long-running process will re-read the `okvclient.ora` configuration file.

- PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW Parameter
  The PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW parameter extends time the master encryption key is available after it is cached in the persistent master encryption key cache.
- EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN Parameter
  The EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN parameter ensures that the PKCS#11 persistent cache for a given endpoint database automatically expires upon shutdown of the endpoint database.

18.3.7.1 PKCS11_CACHE_TIMEOUT Parameter

The PKCS11_CACHE_TIMEOUT parameter sets how long a master encryption key is available in the in-memory cache.

You set the PKCS11_CACHE_TIMEOUT parameter in the okvclient.ora file. You must specify the value minutes. When the specified duration of time elapses, the master encryption key expires. Expired master encryption keys are not deleted from the in-memory cache.

The default value for PKCS11_CACHE_TIMEOUT is 60 (minutes). Oracle recommends that you set the PKCS11_CACHE_TIMEOUT parameter in the Oracle Key Vault management console, where it is called PKCS11 In-Memory Cache Timeout.

Related Topics

- Setting Global Endpoint Configuration Parameters
  You can set global endpoint configuration parameters in the Oracle Key Vault management console.

18.3.7.2 PKCS11_PERSISTENT_CACHE_TIMEOUT Parameter

The PKCS11_PERSISTENT_CACHE_TIMEOUT parameter sets how long the master encryption is available in the persistent cache.

You set the PKCS11_PERSISTENT_CACHE_TIMEOUT parameter in the okvclient.ora file. This time starts when the database retrieves the key from the Oracle Key Vault server and puts it in the cache. After this duration has elapsed, the master encryption key expires. At this time, the endpoint will attempt to contact the Oracle Key Vault server in order to retrieve the key, and if it succeeds, the key remains available for another duration specified by this parameter. If it is unable to retrieve the key, the key remains available for the amount of time dictated by the PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW parameter, after which the database can no longer use the key without successfully retrieving it again from the Oracle Key Vault server. Expired master encryption keys are not deleted from persistent master encryption 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 encryption key cache by setting both the PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW and the PKCS11_PERSISTENT_CACHE_TIMEOUT parameters to 0 (zero).

Oracle recommends that you set this global parameter in the Oracle Key Vault management console.

> **Note:**
>
> The parameter `PKCS11_PERSISTENT_CACHE_TIMEOUT` and its default value are included by default in the `okvclient.ora` file.

Related Topics

- Setting Global Endpoint Configuration Parameters
  You can set global endpoint configuration parameters in the Oracle Key Vault management console.

18.3.7.3 PKCS11_PERSISTENT_CACHE_FIRST Parameter

The `PKCS11_PERSISTENT_CACHE_FIRST` parameter sets the persistent master encryption key cache operation mode.

You set the `PKCS11_PERSISTENT_CACHE_FIRST` parameter in the `okvclient.ora` file.

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 Encryption Key Cache First Mode:** Persistent master encryption key cache first mode is the default mode.

  To enable persistent master encryption key cache first mode, set the value of the `PKCS11_PERSISTENT_CACHE_FIRST` parameter to `1`.

Related Topics

- 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.
- Persistent Master Encryption Key Cache First Mode
  In persistent master encryption key cache first mode, the endpoints retrieve the master encryption key from the persistent master encryption key cache.

18.3.7.4 PKCS11_CONFIG_PARAM_REFRESH_INTERVAL Parameter

The `PKCS11_CONFIG_PARAM_REFRESH_INTERVAL` parameter describes the frequency at which a long-running process will re-read the `okvclient.ora` configuration file.

When the process cannot use a key from the in-memory cache and instead reaches out to the persistent cache or the Oracle Key Vault server, if it has been longer than the value specified by `PKCS11_CONFIG_PARAM_REFRESH_INTERVAL` since `okvclient.ora` was last read, the process will re-read `okvclient.ora` and start using any changed parameters. Note that if the parameter for the in-memory cache, `PKCS11_CACHE_TIMEOUT`, is larger than `PKCS11_CONFIG_PARAM_REFRESH_INTERVAL`, then `okvclient.ora` will be re-read at intervals described by the `PKCS11_CACHE_TIMEOUT` parameter instead.

You set the PKCS11_CONFIG_PARAM_REFRESH_INTERVAL parameter in okvclient.ora. You must specify this value in minutes. The default value for PKCS11_CONFIG_PARAM_REFRESH_INTERVAL is 10 (minutes).

You can disable this parameter by setting the PKCS11_CONFIG_PARAM_REFRESH_INTERVAL parameter to 0 (zero).

18.3.7.5 PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW Parameter

The PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW parameter extends time the master encryption key is available after it is cached in the persistent master encryption key cache.

You set the PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW parameter in the okvclient.ora. 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 encryption key cache by setting the PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW and PKCS11_PERSISTENT_CACHE_TIMEOUT parameters to 0 (zero). Oracle recommends that you set these global parameters in the Oracle Key Vault management console.

Related Topics

- Setting Global Endpoint Configuration Parameters
  You can set global endpoint configuration parameters in the Oracle Key Vault management console.

18.3.7.6 EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN Parameter

The EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN parameter ensures that the PKCS#11 persistent cache for a given endpoint database automatically expires upon shutdown of the endpoint database.

When enabled, the EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN protects the persistent cache by using a system-generated random password that is independent of the password that was set when an endpoint database was enrolled in Oracle Key Vault, even if an auto-login wallet was used. Having the persistent cache password protected provides better security.

Before you can use the EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN parameter, ensure that the endpoint database has had the patch for bug 29869906: AUTO-LOGIN OKV NEEDS PERSISTENT CACHE PROTECTION KEY FROM RDBMS applied to it. This patch applies to Oracle Database releases 12.1 through 19c. Contact Oracle Support for more information.

You can set EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN for individual endpoint databases that have been enrolled with Oracle Key Vault, or globally for all endpoint databases that have been enrolled in Oracle Key Vault. This parameter is not available in the okvclient.ora configuration file for the database endpoint. To set this parameter, use the Oracle Key Vault management console.

After you have enabled EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN, the PKCS#11 persistent cache is created when keys are fetched from Oracle Key Vault. The cache remains available to the endpoint database only as long as the database instance is mounted or open. When the endpoint database is shut down, the

PKCS#11 persistent cache is no longer available, but is recreated the next time the endpoint database is started. The persistent cache does, however, remain available when an endpoint pluggable database (PDB) is closed and then re-opened.

Be aware that after you have enabled `EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN` for a given endpoint database, you can no longer use the `okvutil list -t okv_persistent_cache` command to view the contents of the persistent cache. In addition, you must ensure that Oracle Key Vault is available when keys are fetched after the endpoint database is started.

Setting EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN for Individual Endpoint Databases

1. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
2. Click the **Endpoints** tab.
3. On the Endpoints page, select the endpoint for which you want to set `EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN`.
4. On the Endpoint Details page, scroll to the bottom and then set the **Expire PKCS11 Persistent Cache on Database Shutdown** checkbox.
5. Click **Save**.

Setting EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN Globally

The following procedure will apply the `EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN` to all current and future endpoint databases that have been enrolled with Oracle Key Vault.

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..
3. In the Global Endpoint Configuration Parameters page, set the **Expire PKCS11 Persistent Cache on Database Shutdown** checkbox.
4. Click **Save**.

18.3.8 Listing the Contents of the Persistent Master Encryption Key Cache

The `okvutil list` command lists the master encryption keys that are cached in the persistent master encryption 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 following example shows how to list the master encryption keys cached in the persistent master encryption key cache:

```
$ ./okvutil list -t okv_persistent_cache -l $ORACLE_HOME/okv/$ORACLE_SID
Enter Oracle Key Vault endpoint password: password
```

Output similar to the following appears:

```
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 describes the columns in the `OKV Persistent Cache entries` list:

Column Name	Description
`Version`	Persistent master encryption key cache version
`Unique ID`	KMIP identifier assigned to the master encryption key
`TDE Master Key Identifier`	Database ID assigned to the master encryption key
`Cache Start Time`	Time at which the master encryption key was cached
`Maximum Use Time`	Time until the master encryption key expires, in seconds, from the moment that the key was placed into the persistent master encryption key cache
`Maximum Refresh Window`	Extended duration for which the master encryption key is available after it is cached in the persistent master encryption key cache
`Status`	Indicates whether the master encryption key is available, refreshing or expired

18.3.9 Oracle Database Deployments and Persistent Master Encryption Key Cache

The persistent master encryption 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 encryption 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 encryption key cache and the Oracle Key Vault endpoint wallet are synchronized.

> **Note:**
>
> The persistent master encryption key cache must be deleted when the endpoint wallet credentials are modified.

- **Using the persistent master encryption 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 a unique persistent master encryption key cache.

  In an Oracle RAC Environment, you must query the database from each Oracle RAC node to cache the most recent version of the master encryption key in the persistent master encryption key cache of each Oracle RAC node.

- **Using persistent master encryption key cache in an Oracle Data Guard Environment:** Rotation of the master encryption key in the primary server's database caches the master encryption key in the persistent master encryption key cache of the primary server's database.

  The standby server retrieves and caches the new master encryption key in the persistent master encryption 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.

18.4 Uploading and Downloading Oracle Wallets

To store and share Oracle wallets, you must upload them to Oracle Key Vault.

- About Uploading and Downloading Oracle Wallets
  You use the `okvutil` utility to upload and download Oracle wallets.
- Uploading Oracle Wallets
  The `okvutil upload` command uploads wallets to Oracle Key Vault.
- Downloading Oracle Wallets
  The `okvutil download` command downloads an Oracle wallet from the Oracle Key Vault server to an endpoint.
- Guidelines for Uploading and Downloading Oracle Wallets
  Oracle provides guidelines for uploading and downloading wallets to and from Oracle Key Vault.

18.4.1 About Uploading and Downloading Oracle Wallets

You use the `okvutil` utility to upload and download Oracle wallets.

After you upload a wallet to Oracle 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. You can use the

`okvutil upload` and `okvutil download` commands to upload and download Oracle wallets between Oracle Key Vault and its endpoints. The `okvutil` utility is packaged with the endpoint software that you install at the endpoint.

The Oracle Key Vault `okvutil` 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.

Related Topics

- Oracle Key Vault okvutil Endpoint Utility Reference
  The `okvutil` utility enables you to perform tasks uploading and downloading security objects.

18.4.2 Uploading Oracle Wallets

The `okvutil upload` command uploads wallets to Oracle Key Vault.

Uploading the contents of a TDE wallet into Oracle Key Vault is a unique feature of Oracle Key Vault: If you plan to migrate the database to use Oracle Key Vault in online master key mode, then you must upload the wallet content before the migration step. That allows you to delete the file-based wallet after a successful migration from the database server, which is a requirement of the PCI-DSS. The upload operation uploads everything in the Oracle wallet, including security objects and their metadata so that the wallet can be reconstructed during the download process. The Oracle wallet typically contains TDE master encryption keys, historical TDE master encryption keys, SSL or TLS certificates and their metadata (stored in Oracle Key Vault as opaque objects), wallet metadata, as well as keys that you have explicitly added.

1. Ensure that the server containing the Oracle wallet has been enrolled and provisioned as an Oracle 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 Oracle Key Vault virtual wallet that was configured in Step 2, so that this wallet can be part of that virtual wallet.

At this point, the upload is complete. You can now share the virtual wallet with other users and endpoints.

Related Topics

- Managing Endpoints
  You can enroll, reenroll, suspend, and delete endpoints.
- Granting Access to Users, User Groups, Endpoints, and Endpoint Groups
  You can grant the **Read Only**, **Read and Modify**, and **Manage Wallet** access levels to users, user groups, endpoints, and endpoint groups.
- okvutil upload Command
  The `okvutil upload` command uploads security objects to Oracle Key Vault.
- How Password Prompts for okvutil Work

18.4.3 Downloading Oracle Wallets

The `okvutil download` command downloads an Oracle wallet from the Oracle Key Vault server to an endpoint.

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.

   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. Close and then reopen the wallet.

   Closing and reopening the wallet makes the wallet content, including TDE master keys, available on a database encrypted with TDE, and loads the wallet contents

into the TDE database. (Auto-login wallets are automatically opened the next time that they are accessed.)

- For Oracle Database 11g 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 12c or later:

```
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.

Related Topics

- okvutil download Command
  The `okvutil download` command downloads security objects from Oracle Key Vault to the endpoint
- Granting Access to Users, User Groups, Endpoints, and Endpoint Groups
  You can grant the **Read Only**, **Read and Modify**, and **Manage Wallet** access levels to users, user groups, endpoints, and endpoint groups.

18.4.4 Guidelines for Uploading and Downloading Oracle Wallets

Oracle provides guidelines for uploading and downloading wallets to and from Oracle Key Vault.

- 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.
- Use care if you plan to use the `okvutil upload` and `okvutil download` commands, which provide an overwrite (`-o`) option. This option 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.

Related Topics

- Managing Endpoint Groups
  An endpoint group is a named group of endpoints that share a common set of wallets.

18.5 Uploading and Downloading JKS and JCEKS Keystores

The `okvutil upload` and `okvutil download` commands can upload and download JKS and JCEKS keystores.

- About Uploading and Downloading JKS and JCEKS Keystores
  You use the `okvutil` utility to upload and download JKS and JCEKS keystores.
- Uploading JKS or JCEKS Keystores
  The `okvutil upload` command can upload a JKS or JCEKS to the Oracle Key Vault server.
- Downloading JKS or JCEKS Keystores
  The `okvutil download` command can download an uploaded JKS or JCEKS keystore.
- Guidelines for Uploading and Downloading JKS and JCEKS Keystores
  Oracle provides recommendations for when you upload and download JKS and JCEKS keystores.

18.5.1 About Uploading and Downloading JKS and JCEKS Keystores

You use the `okvutil` utility to upload and download JKS and JCEKS keystores.

You can upload 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.

18.5.2 Uploading JKS or JCEKS Keystores

The `okvutil upload` command can upload a JKS or JCEKS to the Oracle Key Vault server.

1. Ensure that the server containing the Java keystore has been enrolled and provisioned as an Oracle 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 virtual wallet.
3. Run the `okvutil upload` command to upload the keystore.

   The following examples show how to upload the JKS and JCEKS 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.

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.

Related Topics

- download Command
  The `download` command downloads the endpoint software (`okvclient.jar`) to a directory that you name.
- Granting Access to Users, User Groups, Endpoints, and Endpoint Groups
  You can grant the **Read Only**, **Read and Modify**, and **Manage Wallet** access levels to users, user groups, endpoints, and endpoint groups.

18.5.3 Downloading JKS or JCEKS Keystores

The `okvutil download` command can download an uploaded JKS or JCEKS 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
   -g FinanceGrp
   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.
   - `-g` is the name of the object to download, which in this case is the JKS keystore `FinanceGrp`.

Related Topics

- okvutil download Command
  The `okvutil download` command downloads security objects from Oracle Key Vault to the endpoint

- Granting Access to Users, User Groups, Endpoints, and Endpoint Groups
  You can grant the **Read Only**, **Read and Modify**, and **Manage Wallet** access levels to users, user groups, endpoints, and endpoint groups.

18.5.4 Guidelines 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.
- Use care if you plan to use the `okvutil upload` and `okvutil download` commands, which provide an overwrite (`-o`) option. This option overwrites data in the file. Before you use the `-o` option, you should create backups of the keystore files 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.

Related Topics

- Managing Endpoint Groups
  An endpoint group is a named group of endpoints that share a common set of wallets.

18.6 Uploading and Downloading Credential Files

The `okvutil upload` and `okvutil download` commands can upload and download credential files.

- About Uploading and Downloading Credential Files
  You use the `okvutil` utility to upload and download credential files.
- Uploading a Credential File
  The `okvutil upload` command can upload credential files.
- Downloading a Credential File
  The `okvutil download` command can download credential files.
- Guidelines for Uploading and Downloading Credential Files
  Oracle provides recommendations for when you upload and download credential files.

18.6.1 About Uploading and Downloading Credential Files

You use the `okvutil` utility to upload and download credential files.

Credential files are uploaded and stored as opaque objects in Oracle Key Vault, which means that Oracle 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 servers. Oracle Key Vault supports credential files up to 128 KB in size.

You can place the credential file anywhere in your server infrastructure (which includes database servers and application servers) that is accessible by an Oracle Key Vault endpoint.

18.6.2 Uploading a Credential File

The `okvutil upload` command can upload credential files.

1. Ensure that the server that contains the credential file has been enrolled and provisioned as an Oracle 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, which 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 must already exist. 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 Oracle Key Vault virtual wallet to which access control was configured in Step 2.
   - `-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.

Related Topics

- okvutil upload Command
  The `okvutil upload` command uploads security objects to Oracle Key Vault.
- Managing Endpoints
  You can enroll, reenroll, suspend, and delete endpoints.
- Granting Access to Users, User Groups, Endpoints, and Endpoint Groups
  You can grant the **Read Only**, **Read and Modify**, and **Manage Wallet** access levels to users, user groups, endpoints, and endpoint groups.

18.6.3 Downloading a Credential File

The `okvutil download` command can download credential files.

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 an endpoint administrator 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 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.

Related Topics

- okvutil upload Command
  The `okvutil upload` command uploads security objects to Oracle Key Vault.

18.6.4 Guidelines for Uploading and Downloading Credential Files

Oracle provides recommendations for when you upload and download credential files.

- After you complete the upload, upload the 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.
- Use care if you use the `okvutil upload` and `okvutil download` commands, which provide an overwrite (`-o`) option. This option 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. Afterward, all the members of the group will have access to that wallet.

18.7 Using a User-Defined Key as the TDE Master Encryption Key

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

- About Using a User-Defined Key as the TDE Master Encryption Key
  Key administrators can upload a user-defined key to the groups to which they have write access.
- Step 1: Upload the User-Defined Key
  Use the `okvutil upload` command to upload user-defined master encryption keys to Oracle Key Vault.
- Step 2: Activate the User-Defined Key as a TDE Master Encryption Key
  After you upload the user-defined key, you are ready to activate the key as a TDE master encryption key.

18.7.1 About Using a User-Defined Key as the TDE Master Encryption Key

Key administrators can upload a user-defined key to the groups to which they have write access.

This enables it can be used as the Transparent Data Encryption (TDE) master encryption key. This feature provides key administrators with more control on creation of the master encryption key used to encrypt TDE data encryption keys.

The `type` parameter of the `okvutil upload` command includes the option `TDE_KEY_BYTES`, which enables you to upload user-defined key bytes to Oracle Key

Vault to be used as the TDE master encryption key. You must then activate the key as a TDE master encryption key by running the `ADMINISTER KEY MANAGEMENT` SQL statement on the database.

Related Topics

- *Oracle Database Advanced Security Guide*

18.7.2 Step 1: Upload the User-Defined Key

Use the `okvutil upload` command to upload user-defined master encryption keys to Oracle Key Vault.

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`. `TDE Master Key Identifier` represents the master encryption 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` values are stored as managed KMIP objects with a symmetric key as a KMIP object type.

1. Create a text file containing the raw bytes data of the user-defined key.

   Use the following format:

   ```
   TDE Master Encryption Key Identifier:
   contiguous_TDE_Master_Encryption_Key_Identifier_bytes_encoded_in_32_
   hex_characters_(16_bytes_long)
   TDE Master Encryption Key:
   contiguous_TDE_Master_Encryption_Key_bytes_encoded_in_64_hex_charact
   ers_(32_bytes_long)
   ```

   For example:

   ```
   TDE Master Encryption Key Identifier:
   1F1E1D1C1B1A10191817161514131210
   TDE Master Encryption 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 Mar 2019"
```

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.

When `-t` is `TDE_KEY_BYTES`, the description specified for `–d` is displayed as the tag in the `V$ENCRYPTION_KEYS` dynamic view.

4. Specify the required parameters and then 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 Mar 2019"
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.

Related Topics

- Step 2: Activate the User-Defined Key as a TDE Master Encryption Key
  After you upload the user-defined key, you are ready to activate the key as a TDE master encryption key.

18.7.3 Step 2: Activate the User-Defined Key as a TDE Master Encryption Key

After you upload the user-defined key, you are ready to activate the key as a TDE master encryption key.

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.

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 **Keys & Wallets** tab.

   The Wallets page appears.
3. Click **All Items** in the left sidebar.

The All Items page appears displaying all the security objects in Oracle Key Vault.

4. 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.

5. Click **Advanced** to view the cryptographic attributes of the key.

   The required Key ID is displayed. The Key ID is prefixed with `ORACLE.TDE.HSM.MK`.

   For example:

   ```
   ORACLE.TDE.HSM.MK.061F1E1D1C1B1A10191817161514131210
   ```

   The TDE master encryption key identifiers contain the user defined raw bytes data prefixed by additional metadata.

6. Copy and store the key ID displayed after the prefix `ORACLE.TDE.HSM.MK`

   For example:

   ```
   061F1E1D1C1B1A10191817161514131210
   ```

7. Connect to the database as a user who has privileges to run the `ADMINISTER KEY MANAGEMENT` SQL statement.

   For example:

   ```
   sqlplus / as sysdba
   ```

8. Activate the key as a TDE master encryption key using the `ADMINISTER KEY MANAGEMENT` command.

   ```
   ADMINISTER KEY MANAGEMENT USE KEY
   '061F1E1D1C1B1A10191817161514131210' IDENTIFIED BY "password";
   ```

   You can query the `TAG` column of the `V$ENCRYPTION_KEYS` view for the identifier of the newly created key.

19

Using Oracle Key Vault with Other Features

You can use Oracle Key Vault with other Oracle features and products, such as Oracle GoldenGate or Oracle Data Guard.

- Using a TDE-Configured Oracle Database in an Oracle RAC Environment
  Each Oracle RAC database has its own Oracle virtual wallet in Oracle Key Vault.
- Using a TDE-Configured Oracle Database in an Oracle GoldenGate Environment
  Oracle Key Vault supports the use of Oracle wallets with Oracle GoldenGate shared secrets.
- Using a TDE-Configured Oracle Database in an Oracle Data Guard Environment
  You can perform the activities such as upoading Oracle wallets or using online master keys in an Oracle Data Guard environment.
- Uploading Keystores 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.
- MySQL Integration with Oracle Key Vault
  You can manage TDE encryption keys in MySQL with Oracle Key Vault.
- Other Oracle Database Features That Oracle Key Vault Supports
  You can deploy Transparent Data Encryption (TDE) in multiple topologies with other database features that move or use clustered deployments.

19.1 Using a TDE-Configured Oracle Database in an Oracle RAC Environment

Each Oracle RAC database has its own Oracle virtual wallet in Oracle Key Vault.

In an Oracle Real Application Clusters (Oracle RAC) environment, each Oracle RAC instance has its own endpoint in Oracle Key Vault; these endpoints share the same virtual wallet in Oracle Key Vault as their default wallet.

You can enable the cluster to share the virtual wallet by using either of the following approaches:

- If the Oracle RAC database is using TDE with individual wallets, then confirm that these wallets have the identical content. Execute the `mkstore -wrl /directory/to/TDE-wallet -list` command to compare the content of each wallet. If they all contain the same keys, then upload the content of one of them into the shared virtual wallet in Oracle Key Vault.
- If the Oracle RAC database is using TDE with a shared wallet (which is the recommended deployment), then upload that wallet to Oracle Key Vault.
- Establish an auto-open connection with Oracle Key Vault.

- Migrate the Oracle RAC database to Oracle Key Vault.

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, then you must explicitly open the wallets on each of these nodes.

Each Oracle RAC node is a different endpoint of the database and has its own individual persistent cache. For Oracle RAC databases, you should initiate a query from each Oracle RAC node to cache the latest master encryption key in the Oracle RAC node for uninterrupted operations

Related Topics

- About the Persistent Master Encrption Key Cache
  The persistent master encryption key cache ensures the availability of TDE master encryption keys.
- Uploading Oracle Wallets
  The `okvutil upload` command uploads wallets to Oracle Key Vault.

19.2 Using a TDE-Configured Oracle Database in an Oracle GoldenGate Environment

Oracle Key Vault supports the use of Oracle wallets with Oracle GoldenGate shared secrets.

You can upload or migrate Oracle wallets that contain Oracle GoldenGate shared secrets and TDE master encryption keys to the Oracle Key Vault server.

- Oracle Wallets in an Oracle GoldenGate Environment
  An Oracle GoldenGate shared secret can be in the same Oracle wallet where master encryption keys are stored.
- Online Master Keys in an Oracle GoldenGate Deployment
  There are two configuration steps to using the oline master key in an Oracle GoldenGate deployment.
- Migration of TDE Wallets in Oracle GoldenGate to Oracle Key Vault
  Oracle wallets can contain both a TDE master encryption key and an Oracle GoldenGate shared secret.

19.2.1 Oracle Wallets in an Oracle GoldenGate Environment

An Oracle GoldenGate shared secret can be in the same Oracle wallet where master encryption keys are stored.

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 encryption 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 encryption 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 encryption 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 encryption 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. The two Oracle TDE-enabled databases, source and target, do not need to have the same master encryption key in the Oracle wallet. If you configure this target database as a new Oracle Key Vault endpoint, then you can upload and download wallets to and from Oracle Key Vault as you normally would with any independent Oracle Key Vault endpoint. No additional configuration is necessary.

19.2.2 Online Master Keys in an Oracle GoldenGate Deployment

There are two configuration steps to using the oline 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 steps 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 encryption 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 you perform this procedure. The other Oracle GoldenGate shared secrets are stored in Oracle Key Vault.

See Also:

- Step 1: Configure the Oracle Key Vault Server Environment for instructions to connect a source database in GoldenGate to Oracle Key Vault.
- *Oracle Database Advanced Security Guide* for more information on configuring the storage of Oracle GoldenGate secrets in the source database.

19.2.3 Migration of TDE Wallets in Oracle GoldenGate to Oracle Key Vault

Oracle wallets can contain both a TDE master encryption key and an Oracle GoldenGate shared secret.

In an Oracle GoldenGate environment with a TDE-configured database, an Oracle wallet contains both the TDE master encryption keys and the Oracle GoldenGate shared secret.

You can also configure target Oracle TDE-enabled databases that are used in this Oracle GoldenGate environment to use Oracle Key Vault or continue to use an 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 encryption 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.

Related Topics

- Migration of TDE Wallets in Oracle GoldenGate to Oracle Key Vault
  Oracle wallets can contain both a TDE master encryption key and an Oracle GoldenGate shared secret.

19.3 Using a TDE-Configured Oracle Database in an Oracle Data Guard Environment

You can perform the activities such as upoading Oracle wallets or using online master keys in an Oracle Data Guard environment.

- About Uploading Oracle Wallets in an Oracle Data Guard Environment
  The upload operation enables both a primary and standby to benefit from the use of Oracle wallets.
- Uploading Oracle Wallets in an Oracle Data Guard Environment
  You can upload an Oracle wallet to an Oracle Data Guard environment.
- Performing an Online Master Key Connection in an Oracle Data Guard Environment
  The procedure for performing a TDE direct connection in an Oracle Data Guard environment is the same as in a standard Oracle Database environment.
- Migrating Oracle Wallets in an Oracle Data Guard Environment
  You can migrate an Oracle wallet in an Oracle Data Guard environment by using `okvutil` and SQL*Plus.
- Reverse Migrating Oracle Wallets in an Oracle Data Guard Environment
  You can use `okvutil` and SQL*Plus to reverse migrate an Oracle wallet in an Oracle Data Guard environment.
- Migrating an Oracle TDE Wallet to Oracle Key Vault for a Logical Standby Database
  You can migrate a TDE wallet to Oracle Key Vault to a logical standby database using Oracle Database release 12c or 18c.
- Checking the Oracle TDE Wallet Migration for a Logical Standby Database
  You use SQL*Plus to check the migration.

19.3.1 About Uploading Oracle Wallets in an Oracle Data Guard Environment

The upload operation enables both a primary and standby to benefit from the use of Oracle wallets.

In an Oracle Data Guard environment with a TDE-enabled primary and standby databases using an Oracle wallet, you must physically copy the Oracle wallet file from the primary database to the standby and restart the managed recovery process after the initial TDE configuration and later, when you rekey the master encryption key on the primary database.

Whereas, when using Oracle Key Vault with a TDE-enabled Oracle Data Guard database, you must register the primary and standby databases in Oracle Key Vault as endpoints. You must ensure that the endpoints for the primary and all standby databases share the same virtual wallet.

This way, the primary and standby databases can benefit from centralized key management without the need of a manual copy of the wallet file from the primary database to the standby database.

In an Oracle Data Guard environment, for a persistent cache, a rekey operation on the primary database will cache the master encryption 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 Oracle Key Vault 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. Oracle recommends that you synchronize the primary and standby as soon as possible after the rekey operation. In addition, you should confirm the content of the persistent cache on the primary and standby databases with the following command:

```
$ okvutil list -t okv_peristent_cache -l /path_to_persistent_cache/
```

Related Topics

- About the Persistent Master Encrption Key Cache
  The persistent master encryption key cache ensures the availability of TDE master encryption keys.

19.3.2 Uploading Oracle Wallets in an Oracle Data Guard Environment

You can upload an Oracle wallet to an Oracle Data Guard environment.

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.

Related Topics

- Managing Endpoints
  You can enroll, reenroll, suspend, and delete endpoints.

19.3.3 Performing an Online Master Key Connection in an Oracle Data Guard Environment

The procedure for performing a TDE direct connection in an Oracle Data Guard environment is the same as in a standard Oracle Database environment.

Related Topics

- Centralized Management of TDE Master Encryption Keys Using Online Master Keys
  You can use an online master key to centralize the management of TDE master encryption keys over a direct network connection.

19.3.4 Migrating Oracle Wallets in an Oracle Data Guard Environment

You can migrate an Oracle wallet in an Oracle Data Guard environment by using `okvutil` and SQL*Plus.

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, as described in Migrating an Existing TDE Wallet to Oracle Key Vault.
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 12c or later, 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 12c or 18c, 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, as described in *Oracle Data Guard Concepts and Administration*.

Related Topics

- okvutil upload Command
  The `okvutil upload` command uploads security objects to Oracle Key Vault.
- Migrating Existing TDE Wallets to Oracle Key Vault
  A migrated TDE wallet can be used to restore database contents that were previously encrypted by TDE.
- *Oracle Data Guard Concepts and Administration*

19.3.5 Reverse Migrating Oracle Wallets in an Oracle Data Guard Environment

You can use `okvutil` and SQL*Plus to reverse migrate an Oracle wallet in an Oracle 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.
2. Perform a reverse migration, as described in *Oracle Database Advanced Security Guide*.
3. Close the existing Oracle wallet on the standby database.
   - For Oracle Database 11g release 2:

```
ALTER SYSTEM SET ENCRYPTION WALLET CLOSE
IDENTIFIED BY "Key_Vault_endpoint_password";
```

   - For Oracle Database 12c or 18c:

```
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, as described in *Oracle Database Advanced Security Guide*.
5. Open the Oracle wallet on the standby database.
   - For Oracle Database 11g release 2:

```
ALTER SYSTEM SET ENCRYPTION WALLET OPEN
IDENTIFIED BY "Key_Vault_endpoint_password";
```

   - For Oracle Database 12c or 18c:

```
ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN
IDENTIFIED BY "Key_Vault_endpoint_password";
```

6. Start the apply process on the standby database, as described in *Oracle Data Guard Concepts and Administration*.

If the endpoint password and the local TDE wallet password are different, then use the auto-login HSM feature.

Related Topics

- okvutil upload Command
  The `okvutil upload` command uploads security objects to Oracle Key Vault.
- *Oracle Database Advanced Security Guide*
- *Oracle Database Advanced Security Guide*
- *Oracle Data Guard Concepts and Administration*

19.3.6 Migrating an Oracle TDE Wallet to Oracle Key Vault for a Logical Standby Database

You can migrate a TDE wallet to Oracle Key Vault to a logical standby database using Oracle Database release 12c or 18c.

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, as described in *Oracle Data Guard Concepts and Administration*.
5. To check that the status the that migration was successful, query the `V$ENCRYPTION_WALLET` dynamic view.

Related Topics

- 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.
- *Oracle Data Guard Concepts and Administration*Applying Redo Data to Logical Standby Databases

19.3.7 Checking the Oracle TDE Wallet Migration for a Logical Standby Database

You use SQL*Plus to check the migration.

In an Oracle Database Release 12c environment, after you have migrated an Oracle TDE wallet in a logical standby configuration, you can check the configuration.

1. In the standby database instance, log in to SQL*Plus.

   For example:

   ```
   sqlplus / as sysdba
   ```

2. Query the `WRL_TYPE` and `WALLET_ORDER` columns of the `V$ENCRYPTION_WALLET` dynamic view.

   The `V$ENCRYPTION_WALLET` view tracks 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 encryption key is located.

For example, in the following, only a single wallet is configured:

```
SELECT WRL_TYPE, WALLET_ORDER FROM V$ENCRYPTION_WALLET;

WRL_TYPE             WALLET_ORDER
-------------------- ------------
FILE                 SINGLE
```

In this query in a logical standby configuration, the active master encryption key has been migrated to an Oracle Key Vault virtual wallet:

```
SELECT WRL_TYPE, WALLET_ORDER FROM V$ENCRYPTION_WALLET;

WRL_TYPE             WALLET_ORDER
-------------------- ------------
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.

19.4 Uploading Keystores 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.

- About Uploading Keystores from Automatic Storage Management to Oracle Key Vault
  Uploading a keystore from Oracle Automatic Storage Management (ASM) to Oracle Key Vault is a two-step process.
- Uploading a Keystore from Automatic Storage Management to Oracle Key Vault
  You can use the `ADMINISTER KEY MANAGEMENT` statement to move a software keystore out of Automatic Storage Management (ASM).
- Copying a Keystore from Oracle Key Vault to Automatic Storage Management
  You use both `okvutil download` and SQL*Plus to complete the copy process.

19.4.1 About Uploading Keystores from Automatic Storage Management to Oracle Key Vault

Uploading a keystore from Oracle Automatic Storage Management (ASM) to Oracle Key Vault is a two-step process.

1. Copy the keystore from ASM to the file system.
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.

19.4.2 Uploading a Keystore from Automatic Storage Management to Oracle Key Vault

You can use the `ADMINISTER KEY MANAGEMENT` statement to move a software keystore out of Automatic Storage Management (ASM).

1. Initialize a target keystore on the file system with the following SQL statement:

   ```
   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";
   ```

   In this specification, */etc/ORACLE/KEYSTORE/DB1/* is the path to the target keystore in the file system and *`destination_password`* is the keystore password.

   You now can copy the keystore from ASM 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 by using the `okvutil upload` command.

   ```
   $ okvutil upload -l location -t type
   ```

   In this specification:

   - `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
   ```

19.4.3 Copying a Keystore from Oracle Key Vault to Automatic Storage Management

You use both `okvutil download` and SQL*Plus to complete the copy process.

To copy a keystore from Oracle Key Vault to Automatic Storage Management (ASM), use the reverse procedure from copying the keystore from ASM to Oracle Key Vault.

1. Initialize a target keystore on the file system, *if the keystore does not exist*.

   If the keystore does exist on the file system, then bypass this step.

2. Copy the keystore from Oracle Key Vault to the target keystore on the file system using the `okvutil download` command.

   ```
   $ okvutil download -l location -t type
   ```

   In this specifcation:

   - `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. Initialize a keystore on the ASM instance by using the following SQL statement:

   ```
   ADMINISTER KEY MANAGEMENT CREATE KEYSTORE asmKeystorePath
   IDENTIFIED BY asmKeystorePassword;
   ```

   In this specification:

   - *`asmKeystorePath`* is the directory path for the keystore on the ASM file system.
   - *`asmKeystorePassword`* is a password that you create for the keystore.

4. Copy the keystore to the initialized ASM keystore that you just created.

   This step requires that you merge the keystore from the file system to ASM as follows:

   ```
   ADMINISTER KEY MANAGEMENT MERGE KEYSTORE srcKeystorePath
   IDENTIFIED BY srcKeystorePassword
   INTO EXISTING KEYSTORE asmKeystorePath
   IDENTIFIED BY asmKeystorePassword
   WITH BACKUP USING backupIdentifier;
   ```

   In this specification:

- *srcKeystorePath* is the directory path to the source keystore.
- *srcKeystorePassword* is the source keystore password.
- *asmKeystorePath* is the path to the ASM keystore.
- *asmKeystorePassword* is the ASM keystore password.
- *backupIdentifier* is the backup identifier to be added to the backup file name.

19.5 MySQL Integration with Oracle Key Vault

You can manage TDE encryption keys in MySQL 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.

19.6 Other Oracle Database Features That Oracle Key Vault Supports

You can deploy Transparent Data Encryption (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 the following:

- Oracle Recovery Manager (RMAN) backup and recovery operations
- Oracle Data Pump
- Transportable tablespaces (Oracle Database 12c and later)

For clustered deployments, Oracle Key Vault supports the following:

- Oracle Data Guard
- Oracle Real Application Clusters (Oracle RAC)
- Oracle GoldenGate

A

Oracle Key Vault Multi-Master Cluster Operations

There are restrictions and conditions for Oracle Key Vault multi-master cluster operations on cluster nodes.

Table A-1 Oracle Key Vault Multi-Master Cluster Operations on Cluster Nodes

Management Console Tab and Operation	Read-Only Node	Read-Write Node in Read-Only Restricted Mode	Read-Write Node in normal (Read-Write) Mode
Home tab	No restrictions	No restrictions	No restrictions
Endpoints tab Endpoints • Add, Delete • Suspend, Resume • Reenroll Endpoint Groups • Create Group • Delete Group Update Endpoint Settings	Updated only via replication from a read-write node	Updated only via replication from a read-write node	Directly updated using client tools on this node Also updated by replication from other read-write nodes
Keys and Wallets tab Wallets • Create, Delete, Edit All Items • Delete • Edit – Update – Revoke, Destroy – Change Wallet Membership	Updated only via replication from a read-write node	Updated only via replication from a read-write node	Updated using client tools on this node. Also updated by replication from other read-write nodes.

Table A-1 (Cont.) Oracle Key Vault Multi-Master Cluster Operations on Cluster Nodes

Management Console Tab and Operation	Read-Only Node	Read-Write Node in Read-Only Restricted Mode	Read-Write Node in normal (Read-Write) Mode
Reports tab Audit • Generate audit report • Export audit records • Delete audit records Reports • Generate any report Alerts • View alerts • Configure alerts	No restrictions	No restrictions	No restrictions
Users tab Users • Create, Delete • Check Conflict Status Manage Access (User Groups) • Update • Add, Remove Wallet Access • Add, Remove Members Change Password	Updated only with replication from a read-write node	Updated only with replication from a read-write node	Updated using client tools on this node Also updated by replication from other read-write nodes There are additional considerations and restrictions based on the status of the user name and user group name.
System tab System Settings • Reboot, Poweroff • Edit Network Details • Edit Network Services • Edit System Time • Edit DNS • Enable FIPS Mode • Configure Syslog • Enable Audit Vault Integration	Node is used to update these settings. The updates are local to the node.	Node is used to update these settings. The updates are local to the node.	Node is used to update these settings. The updates are local to the node. The DNS settings and System Time are not set for the cluster here.

Table A-1 (Cont.) Oracle Key Vault Multi-Master Cluster Operations on Cluster Nodes

Management Console Tab and Operation	Read-Only Node	Read-Write Node in Read-Only Restricted Mode	Read-Write Node in normal (Read-Write) Mode
System tab Cluster System Settings • Edit System Time • Edit DNS • Edit Max Disable Node Duration • Enable RESTful Services • Configure Syslog	Updated only with replication from a read-write node	Updated only with replication from a read-write node	Updated using client tools on this node Also updated by replication from other read-write nodes
System tab Audit Settings, Scope 'Node' • Enable Auditing • Replicate Audit Records • Send Audit to Syslog	Node is used to update these settings. The updates are local to the node.	Node is used to update these settings. The updates are local to the node.	Node is used to update these settings. The updates are local to the node.
System tab Audit Settings, Scope 'Cluster' • Enable Auditing • Replicate Audit Records • Send Audit to Syslog	Updated only with replication from a read-write node	Updated only with replication from a read-write node	Updated using client tools on this node. Also updated by replication from other read-write nodes
System tab Email Settings • Edit	Node is used to update these settings. The updates are local to the node.	Node is used to update these settings. The updates are local to the node.	Node is used to update these settings. The updates are local to the node.
System tab Monitoring Settings, Scope 'Node' • Enable Monitoring • Limit Access • Edit	Node is used to update these settings. The updates are local to the node.	Node is used to update these settings. The updates are local to the node.	Node is used to update these settings. The updates are local to the node.
System tab Monitoring Settings, Scope 'Cluster' • Enable Monitoring • Limit Access • Edit	Updated only with replication from a read-write node	Updated only with replication from a read-write node	Updated using client tools on this node Also updated by replication from other read-write nodes

Table A-1 (Cont.) Oracle Key Vault Multi-Master Cluster Operations on Cluster Nodes

Management Console Tab and Operation	Read-Only Node	Read-Write Node in Read-Only Restricted Mode	Read-Write Node in normal (Read-Write) Mode
System tab System Backup • Configure • Perform Backup • Perform Restore	Node is used to update these settings. The updates are local to the node. A backup can only be restored to a standalone Oracle Key Vault server. Restoring a backup implies that the entire cluster has failed and needs to be rebuilt.	Node is used to update these settings. The updates are local to the node. A backup can only be restored to a standalone Oracle Key Vault server. Restoring a backup implies that the entire cluster has failed and needs to be rebuilt.	Node is used to update these settings. The updates are local to the node. A backup can only be restored to a standalone Oracle Key Vault server. Restoring a backup implies that the entire cluster has failed and needs to be rebuilt.
System tab Console Certificate • Generate, Upload	Node is used to update these settings. The updates are local to the node.	Node is used to update these settings. The updates are local to the node.	Node is used to update these settings. The updates are local to the node.
System tab SSH Tunnel Settings • Add, Delete • Edit	Node is used to update these settings. The updates are local to the node.	Node is used to update these settings. The updates are local to the node.	Node is used to update these settings. The updates are local to the node.
System tab HSM • All operations	Node is used to update these settings. The updates are local to the node.	Node is used to update these settings. The updates are local to the node.	Node is used to update these settings. The updates are local to the node.
Cluster tab Management section • Add Node • Delete Node • Force Delete Node • Disable Node • Enable Node	A node in the `ACTIVE` state may be used to add, delete, force delete, or disable a node. When adding a node, selecting **Add Node as a Read-Write Peer** creates a read-write pair. Only a disabled node may enable itself. Delete and force delete have special considerations as noted.	A node in the `ACTIVE` state may be used to add, delete, force delete, or disable a node. When adding a node, this node cannot be added as a read-write peer to the new node, as it is already in a read-write pair. Only a disabled node may enable itself. Delete and force delete have special considerations as noted.	A node in the `ACTIVE` state may be used to add, delete, force delete, or disable a node. When adding a node, this node cannot be added as a read-write peer to the new node, as it is already in a read-write pair. Only a disabled node may enable itself. Delete and force delete have special considerations as noted.
Cluster tab Monitoring • View information • Enable, Disable link state	Node can access and update these settings. The updates are local to the node.	Node can access and update these settings. The updates are local to the node.	Node can access and update these settings. The updates are local to the node.

Table A-1 (Cont.) Oracle Key Vault Multi-Master Cluster Operations on Cluster Nodes

Management Console Tab and Operation	Read-Only Node	Read-Write Node in Read-Only Restricted Mode	Read-Write Node in normal (Read-Write) Mode
Cluster tab Conflict Resolution • Edit • Accept	Node can access but not resolve conflicts. Updates are received only from active read-write nodes in the cluster through replication.	Node can access but not resolve conflicts. Updates are received only from active read-write nodes in the cluster through replication.	Node can access and resolve conflicts. Updates are propagated to all other nodes in the cluster.
Join read-write pair	Only through induction from a read-only node. Requires **Add Node as Read-Write Peer** set to **Yes**.	Not applicable. Since this node is already a member of a read-write pair, when replication is once again available from this node to its read-write peer, it will return to its read-write state.	Not applicable

B

Oracle Key Vault okvutil Endpoint Utility Reference

The `okvutil` utility enables you to perform tasks uploading and downloading security objects.

- About the okvutil Utility
  The `okvutil` utility is a command-line utility that you can use to manage security objects.
- okvutil Command Syntax
  The `okvutil` utility syntax provides short and long options for specifying commands.
- okvutil changepwd Command
  The `okvutil changepwd` command changes the password associated with the credentials used to connect to Oracle Key Vault.
- okvutil diagnostics Command
  The `okvutil diagnostics` command collects diagnostic and environmental information on an endpoint to troubleshoot deployment issues.
- okvutil download Command
  The `okvutil download` command downloads security objects from Oracle Key Vault to the endpoint
- okvutil list Command
  The `okvutil list` command lists the available security objects that are uploaded.
- okvutil upload Command
  The `okvutil upload` command uploads security objects to Oracle Key Vault.

B.1 About the okvutil Utility

The `okvutil` utility is a command-line utility that you can use to manage security objects.

The `okvutil` command-line utility enables you to locate, upload, and download security objects to and from Oracle Key Vault. You can also use `okvutil` to change the wallet password and collect system diagnostics.

The `okvutil` utility uses the Transport Layer Security (TLS) credentials provisioned for the endpoint to authenticate to Oracle Key Vault.

B.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 B-1 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: • `-v 0` disables verbose mode. • `-v 1` includes debug messages. • `-v 2` includes more detailed debug messages.
`-h, --help`	Use option to get help with any `okvutil` command. For example: `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.

B.3 okvutil changepwd Command

The `okvutil changepwd` command changes 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 does not need to 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 B-2 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 B-1 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 B-1 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
```

B.4 okvutil diagnostics Command

The `okvutil diagnostics` command collects diagnostic and environmental information on an endpoint to troubleshoot deployment issues.

The information is placed in a `diagnostics.zip` file, which can be given to Oracle support for further analysis and debugging.

The information gathered includes information on the following:

- The shell environment variables: `OKV_HOME`, `ORACLE_HOME`, `ORACLE_BASE`, `ORACLE_SID`, `PATH`, `CLASSPATH`
- Configuration and IP address of the Oracle Key Vault server from `okvclient.ora`
- Directory listing of `OKV_HOME` and its sub-directories
- Oracle Key Vault log files from the endpoint
- Listing of symbolic links created by the Oracle Key Vault endpoint installer
- Network settings and ping results

The okvutil diagnostics command does not collect sensitive information such as user credentials or security objects.

Syntax

Short format:

```
okvutil diagnostics [-v verbosity_level]
```

Long format:

```
okvutil diagnostics  [--verbose verbosity_level]
```

Parameters

Table B-3 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 collects system diagnostics in a zip file.

Example B-2 shows how to execute the command. After you execute the command, when the `Diagnostics complete` message appears, then the `diagnostics.zip` file will be available in the current directory.

Example B-2 Collecting System Diagnostics

```
$ okvutil diagnostics
Diagnostics collection complete.
ls
diagnostics.zip
```

B.5 okvutil download Command

The `okvutil download` command downloads security objects from Oracle Key Vault to the endpoint

These security objects include 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.

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 B-4 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 from Oracle Key Vault. It must be a value from the following list: • `WALLET` for an Oracle wallet • `JKS` for a Java keystore • `JCEKS` for a Java Cryptography Extension keystore (JCEKS) • `SSH` for an SSH key file, to be downloaded as an opaque object. • `KERBEROS` for a Kerberos keytab, to be downloaded as an opaque object. • `OTHER` for opaque objects, which are other files that store secrets. 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. This setting is not case-sensitive. This setting is mandatory.
`-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 this password for the future. Remember that the group name is case-sensitive. 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. In a multi-master cluster, only the default wallet assigned to the endpoint can be specified when the name status is `PENDING`.
`-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).

Table B-4 (Cont.) okvutil download Command Options

Parameter	Description
`-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. 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 downloads 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 B-3 downloads the Oracle Key Vault virtual wallet `FinanceWallet` to a Java keystore.

Example B-3 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
```

Related Topics

- okvutil list Command
  The `okvutil list` command lists the available security objects that are uploaded.
- Downloading Oracle Wallets
  The `okvutil download` command downloads an Oracle wallet from the Oracle Key Vault server to an endpoint.

B.6 okvutil list Command

The `okvutil list` command lists 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 B-5 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: • `WALLET` for an Oracle wallet • `JKS` for the Java keystore • `JCEKS` for the Java Cryptography Extension keystore (JCEKS) • `OKV_PERSISTENT_CACHE` for the persistent cache of Oracle Key Vault (Note that this setting becomes unavailable if `EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN` has been set for a given endpoint database.) 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. 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. Only the default wallet assigned to the endpoint can be specified when the name status is `PENDING`.
`-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 B-4 lists 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 B-5 shows the contents of an Oracle wallet file.

Example B-4 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 B-5 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/
tv0ZwOlDeIp4CAwAAAAAAAAAAAAAAAAAAAAAAAAA
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
```

B.7 okvutil upload Command

The `okvutil upload` command uploads security objects to Oracle Key Vault.

These security objects can be 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 Oracle 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 B-6 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).
`-t, --type`	Specifies the data type of the object being uploaded to Oracle Key Vault. It must be a value from the following list: • `WALLET` for an Oracle wallet • `JKS` for a Java keystore • `JCEKS` for a Java Cryptography Extension keystore (JCEKS) • `SSH` for an SSH key file, to be uploaded as an opaque object. The maximum size is 120 KB. • `KERBEROS` for a Kerberos keytab, to be uploaded as an opaque object. The maximum size is 120 KB. • `TDE_KEY_BYTES` for a user-defined key to be used as a TDE master encryption key. • `OTHER` for opaque objects, which are other files that store secrets. The maximum size is 120 KB. 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. This setting is not case-sensitive.

Table B-6 (Cont.) okvutil upload Command Options

Parameter	Description
`-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. Only the default wallet assigned to the endpoint can be specified when the name status is `PENDING`.
`-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. 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 -v 2 Option

The `okvutil upload` command enables you to upload a Java keystore.

The following example shows 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 18.3.0.0.0
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.

Uploading a Password-Protected Wallet File

The `okvutil upload` command uploads a password-protected wallet file.

The following example shows 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.

Uploading a User-Defined Key to Use as a TDE Master Encryption Key

The `okvutil upload` command can upload a user-defined key to use as a TDE master encryption key.

The following example shows 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 April 2019\"
```

Related Topics

- Step 1: Upload the User-Defined Key
  Use the `okvutil upload` command to upload user-defined master encryption keys to Oracle Key Vault.
- Adding Security Objects to a Virtual Wallet
  You can add new security objects to a virtual wallet at any time as needed.
- Uploading Oracle Wallets
  The `okvutil upload` command uploads wallets to Oracle Key Vault.
- About the orapki Utility

C

Troubleshooting Oracle Key Vault

Oracle provides checklists and tips for commonly encountered errors that will help you install and deploy Oracle Key Vault.

- Oracle Key Vault Pre-Installation Checklist
  The pre-installation checklist covers all the requirements to successfully install Key Vault.
- Integrating Oracle Key Vault with Oracle Audit Vault and Database Firewall
  You can consolidate audits between Oracle Audit Vault and Database Firewall (AVDF) with Oracle Key Vault.
- RESTful Services Troubleshooting Help
  The Oracle Key Vault log files capture all the error messages sent by the server.
- 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.
- 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.
- WARNING: Could Not Store Private Key Errors
  If you upload two keystores with the same file name but different contents, a `WARNING: Could not store private key` error appears.
- Errors After Upgrading Oracle Key Vault
  Some errors that appear after the upgrade can be ignored.
- Error: Failed to Open Wallet
  An `Failed to Open Wallet` error can appear if you attempt to use an online master key.
- Transaction Check Error: Diagnostics Generation Utility
  If you are trying to perform an upgrade of Oracle Key Vault, a transaction check error may appear.
- Fast-Start Failover (FSFO) Suspended (ORA-16818)
  An `ORA-16818: Fast-Start Failover suspended` error can appear as a result of a fast start failover operation failing.
- SSH Tunnel Add Failure
  While you are configuring the SSH tunnel, a `Failed to establish SSH tunnel. Refer to Oracle documentation error` may appear.
- 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 points to an incorrect Java directory.
- TDE Endpoint Integration Issues
  Several issues related to Transparent Data Encryption (TDE) endpoint integration problems can arise.

- Failover Situations in Primary-Standby Mode
  Failover situations can occur with or without read-only restricted mode or during a planned shutdown operation for both primary and standby servers.
- Performing a Planned Shutdown
  A user who has the System Administrator role can perform planned shutdowns during an upgrade or a maintenance window.

Related Topics

- Guidelines for Uploading and Downloading Oracle Wallets
  Oracle provides guidelines for uploading and downloading wallets to and from Oracle Key Vault.
- Guidelines for Uploading and Downloading JKS and JCEKS Keystores
  Oracle provides recommendations for when you upload and download JKS and JCEKS keystores.
- Guidelines for Uploading and Downloading Credential Files
  Oracle provides recommendations for when you upload and download credential files.

C.1 Oracle Key Vault Pre-Installation Checklist

The pre-installation checklist covers all the requirements to successfully install Key Vault.

Table C-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 in System Requirements.
2. [x]	Open all the required network ports in your firewall	For details on network ports, see Network Port Requirements.
3. [x]	Supported endpoint platforms	See Supported Endpoint Platforms.
4. [x]	Set the `COMPATIBLE` initialization parameter for online master key (previously TDE direct connect).	Guidance for setting this parameter for Oracle Database 11.2.0.0 and later is in Supported Endpoint Platforms.
5. [x]	Get a fixed IP address, network mask, and gateway address from your network administrator.	You will need this information for Step 9 in Installing the Oracle Key Vault Appliance Software

C.2 Integrating Oracle Key Vault with Oracle Audit Vault and Database Firewall

You can consolidate audits between Oracle Audit Vault and Database Firewall (AVDF) with Oracle Key Vault.

To do this, you must integrate Audit Vault and Database Firewall with Oracle Key Vault.

- Step 1: Check the Environment
  Before you begin the integration, you should ensure that the required components are all in place.
- Step 2: Register Oracle Key Vault as a Secured Target with AVDF
  You must register the Oracle Key Vault server as a secured target on the Oracle Audit Vault and Database Firewall server.
- Step 3: Register Oracle Key Vault as a Host with AVDF
  Next, you must register the Oracle Key Vault server as a host on the Audit Vault and Database Firewall server.
- Step 4: Download the AVDF Agent and Upload it to Oracle Key Vault
  You must next download the Oracle Audit Vault and Database Firewall agent and then upload it to the Oracle Key Vault server.
- Step 5: Install the AVDF agent.jar File on the Oracle Key Vault Server
  At this stage, you are ready to install the `agent.jar` file on the Oracle Key Vault server.
- Step 6: Add the Oracle Key Vault Audit Trail to AVDF
  Next, you can add the audit trail to Oracle Audit Vault and Database Firewall.
- Step 7: View Oracle Key Vault Audit Data Collected by AVDF
  After you have completed the integration and are collecting data, you can view data collected by Oracle Audit Vault and Database Firewall.

C.2.1 Step 1: Check the Environment

Before you begin the integration, you should ensure that the required components are all in place.

1. Ensure that Oracle Audit Vault and Database Firewall is properly installed and configured, and that it has the Audit Vault administrator and auditor user accounts.
2. In Oracle Key Vault, enable SSH access.

   Log in to the Oracle Key Vault management console as a user who has the System Administrator role. Select the **System Settings** tab, then under Network Services, select **SSH Access**. Select **IP address(es)** and then enter only the IP addresses that you need. Click **Save**.

C.2.2 Step 2: Register Oracle Key Vault as a Secured Target with AVDF

You must register the Oracle Key Vault server as a secured target on the Oracle Audit Vault and Database Firewall server.

1. Log in to the Oracle Audit Vault Server console as an administrator (`avadmin`).
2. Click the **Secured Targets** tab, and then select **Register**.
3. In the **New Secured Target Name** field, enter the name of the Oracle Key Vault server.
4. For **Secured Target Type**, enter `Oracle Key Vault`.
5. In the Add Secured Target Location section, click **Advanced**, and then add the following for the connect string:

   ```
   jdbc:oracle:thin:@//127.0.0.1:1521/dbfwdb
   ```

6. For the user name and password, enter the following: `avcollector/`*`integration_password`*
7. Add the following collection attribute:

   ```
   av.collector.securedTargetVersion 12.2.0.0.0
   av.collector.TimeZoneOffset +5:30
   ```

   You can use any `TimeZoneOffset` setting that applies to your situation.

C.2.3 Step 3: Register Oracle Key Vault as a Host with AVDF

Next, you must register the Oracle Key Vault server as a host on the Audit Vault and Database Firewall server.

1. Log in to the Oracle Audit Vault Server console as an administrator (`avadmin`).
2. Click the **Hosts** tab, then select **Register**.
3. In the **Host Name** field, enter the name of the Oracle Key Vault server.
4. In the **Host IP** field, enter the IP address of the Oracle Key Vault server.
5. Click **Save**.
6. When the new entry appears with an agent activation key, copy this value and store it in a safe place.

   You will need this agent activation key value later on in these steps.

C.2.4 Step 4: Download the AVDF Agent and Upload it to Oracle Key Vault

You must next download the Oracle Audit Vault and Database Firewall agent and then upload it to the Oracle Key Vault server.

1. Log in to the Oracle Audit Vault Server console as an administrator (`avadmin`).
2. Select the **Agent** tab, and then select **Agent Release**.
3. Select the agent, which should be the first item in the list of agents.
4. Save this item as `agent.jar`.
5. Use `scp` to upload `agent.jar` to the Oracle Key Vault server.

C.2.5 Step 5: Install the AVDF agent.jar File on the Oracle Key Vault Server

At this stage, you are ready to install the `agent.jar` file on the Oracle Key Vault server.

1. Log on to the Oracle Key Vault command line as user `support` and then `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/*
   ```

2. `su` to `oracle` and extract the agent:

   ```
   su - oracle
   cd /usr/local/okv/avdf
   java -jar agent.jar -d /usr/local/okv/avdf
   ```

3. As `oracle`, start the agent and enter the Oracle Audit Vault and Database Firewall agent activation key that you generated earlier when you registered the Oracle Key Vault server as a host on the Audit Vault and Database Firewall server.

   ```
   cd /usr/local/okv/avdf/bin
   ./agentctl start -k
   ```

4. In the Oracle Key Vault management console, enable the database user.

   a. Select **System**, then **System Settings**.

   b. In the Oracle Audit Vault Integration section, check **Enable**.

   c. In the **Password** and **Reenter Password** fields, enter the integration password.

      This is the password of the user in the database that Audit Vault and Database Firewall will use to extract audit records.

   d. Click **Save**.

5. Because you no longer need to copy files from one server to another, disable SSH access.

   Log in to the Oracle Key Vault management console as a user who has the System Administrator role. Select the **System Settings** tab, then under Network Services, select **Disabled**. Click **Save**. Restart the Oracle Key Vault server by clicking **Reboot** on the top right.

C.2.6 Step 6: Add the Oracle Key Vault Audit Trail to AVDF

Next, you can add the audit trail to Oracle Audit Vault and Database Firewall.

1. Log in to the Oracle Audit Vault Server console as an administrator (`avadmin`).
2. Select **Secured Targets**, and then under **Monitoring**, select **Audit Trails**.
3. To add the new audit trail, click **Add**.
4. In the **Collection Host** field, specify the host computer where you installed the `agent.jar` agent file. (You can use the **Search** icon to find this host computer.)

5. In the **Secured Target Name** field, enter the name of the secured target.
6. From the **Audit Trail Type** drop-down list, select **TABLE**. Then choose the secured target and host that you created.
7. In the **Trail Location** field, enter `keyvault.audit_trail`.
8. Click **Save**.
9. Select this audit trail and then start the audit trail collection.

   In the Audit Trails page, select the audit trail and then click **Start**.

C.2.7 Step 7: View Oracle Key Vault Audit Data Collected by AVDF

After you have completed the integration and are collecting data, you can view data collected by Oracle Audit Vault and Database Firewall.

1. Log in to the Oracle Audit Vault Server console as an auditor (`avauditor`).
2. Select the **Reports** tab.
3. Select **All Activity**.
4. Select **All Activity Report**.

C.3 RESTful Services Troubleshooting Help

The Oracle 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.

1. Log in to the Oracle Key Vault server.
2. As the root user, check for log file errors as follows:

```
root# vi /var/log/messages
```

C.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.

To remedy this problem, 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.

C.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.

The KMIP error can occur for other scenarios, but this scenario is the most common.

The steps that result in this error are 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. To avoid the `Invalid Field`, create an endpoint group so that you can share the wallet with multiple endpoints.

Related Topics

- Creating an Endpoint Group
  Endpoints that must share a common set of security objects stored in wallets can be grouped into an endpoint group.

C.6 WARNING: Could Not Store Private Key Errors

If you upload two keystores with the same file name but different contents, a `WARNING: Could not store private key` error appears.

This error 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.

- To remedy this problem, download the second keystore using a unique alias.

Related Topics

- Downloading JKS or JCEKS Keystores
  The `okvutil download` command can download an uploaded JKS or JCEKS keystore.

C.7 Errors After Upgrading Oracle Key Vault

Some errors that appear after the upgrade can be ignored.

After you perform an upgrade of Oracle Key Vault on an standalone server, `ORA-1109: database does not open`, `ORA-00313: open failed for members`, and `ORA-00312: online log 3 thread 1` 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.

C.8 Error: Failed to Open Wallet

An `Failed to Open Wallet` error can appear if you attempt to use an online master key.

If you are attempting to use an online master key (previously called TDE direct connect) and encounter this error, then first check your environment variable `ORACLE_BASE`. In an Oracle Real Application Clusters environment, you must perform this step on all database instances.

You must also set the `ORACLE_SID`, `ORACLE_HOME`, and `OKV_HOME` environment variables needed by the PKCS #11 library as follows:

1. Log in to the server where the PKCS #11 library resides and then set the `ORACLE_SID`, `ORACLE_HOME`, and `OKV_HOME` environment variables.
2. Log in to the database instance using SQL*Plus as a user with the `SYSDBA` administrative privilege.

   ```
   sqlplus sys / as sysdba
   Enter password: password
   ```

3. Shut down the database.

   For example:

   ```
   SHUTDOWN IMMEDIATE
   ```

4. From the command line, restart the database service.

   ```
   su - oracle
   lsnrctl start
   ```

5. In SQL*Plus, restart the database.

   ```
   STARTUP
   ```

Related Topics

- How To Create a TDE Auto_Login Wallet For A Database With Oracle Key Vault OKV TDE Direct Connection / Online Master Key (Doc ID 2120160.1)

C.9 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-18.3.0.0.0-52_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.

C.10 Fast-Start Failover (FSFO) Suspended (ORA-16818)

An `ORA-16818: Fast-Start Failover suspended` error can appear as a result of a fast start failover operation failing.

If the primary server was shut down gracefully in a controlled way, such as by clicking a **Power Off** button instead of manually turning off the computer, then a fast start failover cannot be performed and the `ORA-16818: Fast-Start Failover suspended` error appears. In a graceful shutdown operation, the primary server's failover status goes into a suspended state with the standby waiting indefinitely for the primary server to be available. This is the expected behavior for a fast-start failover (FSFO) operation, as defined by Oracle Data Guard avoid a split brain scenario. By design, a fast-start failover operation error occurs only when the primary server shuts down unexpectedly. If you perform a `SHUTDOWN IMMEDIATE` or `SHUTDOWN NORMAL` command in SQL*Plus, then the FSFO does not occur because the database shuts down gracefully.

C.11 SSH Tunnel Add Failure

While you are configuring the SSH tunnel, a `Failed to establish SSH tunnel. Refer to Oracle documentation error` may appear.

You might get the following error message while trying to set up the SSH tunnel:

The failure may be due to one or more of the following problems:

- The following settings may be invalid:
  - Invalid IP address
  - Invalid port
  - Invalid user name
- The public SSH Oracle Key Vault key was 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.

To remedy this problem, check your input values and connection and retry.

C.12 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 points to an incorrect Java directory.

To remedy this problem, ensure that the Java version is 1.7.21 or later. You can create a soft link to the Java home directory as follows:

```
ln -s Java_home_directory/bin/java /usr/bin/java
```

C.13 TDE Endpoint Integration Issues

Several issues related to Transparent Data Encryption (TDE) endpoint integration problems can arise.

Common Transparent Data Encryption (TDE) endpoint integration problems caused by installation errors, `svrctl` misuse, and the mismanagement of security objects in a primary-standby environment can arise.

- **Installing the Oracle Key Vault library:** You must run `root.sh` to install the Oracle Key Vault library only once on a computer that has multiple Oracle databases. During an upgrade you must upgrade the library only after you have shut down all the associated endpoints. Oracle Key Vault servers are backward compatible with endpoint libraries.
- **Using `svrctl` to manage the database:** If you use the `svrctl` utility to manage the database, remember that `svrctl` can set the `ORACLE_BASE` environment variable to `NULL`. Oracle recommends that you set `ORACLE_BASE` to `ORACLE_HOME` if `ORACLE_BASE` is not used in your environment.
- **Managing security objects the same way in a primary-standby environment:** In a primary-standby configuration, ensure that both the primary and standby servers use the same mechanism to manage security objects. These servers should either both use a wallet or both use Oracle Key Vault.

C.14 Failover Situations in Primary-Standby Mode

Failover situations can occur with or without read-only restricted mode or during a planned shutdown operation for both primary and standby servers.

- About Failover Situations in Primary-Standby Mode
  Failover situations in primary-standby node can occur with read-only restricted mode disabled, and with read-only restricted mode enabled.
- Failover Situations Without Read-Only Restricted Mode
  If read-only restricted mode is not used, then various failover situations may occur in Oracle Key Vault.
- Failover Situations with Read-Only Restricted Mode
  The use read-only restricted mode affects failover operations in Oracle Key Vault.

C.14.1 About Failover Situations in Primary-Standby Mode

Failover situations in primary-standby node can occur with read-only restricted mode disabled, and with read-only restricted mode enabled.

The types of failover situations are as follows:

- **Planned shutdown of the primary server:** A system administrator shuts down the primary server during an upgrade or maintenance window.
- **Planned shutdown of the standby server:** A system administrator shuts down the standby server 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.

C.14.2 Failover Situations Without Read-Only Restricted Mode

If read-only restricted mode is not used, then various failover situations may occur in Oracle Key Vault.

- Primary Server: Planned Shutdown During an Upgrade
  In a failover, the use of read-only restricted mode affects a planned shutdown of a primary server during an upgrade.
- Primary Server: Planned Shutdown During Maintenance
  In a failover, not using read-only restricted mode affects the planned shutdown of a primary server during maintenance.
- Standby Server: Planned Shutdown
  In a failover, not using read-only restricted mode affects a planned shutdown in a standby server.
- Primary Server: Unplanned Shutdown
  In a failover, not using read-only restricted mode affects an unplanned shutdown in a primary server.
- Standby Server: Unplanned Shutdown
  In a failover, not using read-only restricted mode affects a standby server during an unplanned shutdown.

C.14.2.1 Primary Server: Planned Shutdown During an Upgrade

In a failover, the use of read-only restricted mode affects a planned shutdown of a primary server during an upgrade.

If read-only restricted mode is not used, when the primary server goes offline during an upgrade, then the standby server waits in read-only mode for the primary server to return online. During the upgrade, you cannot access the Oracle Key Vault management console.

- **Recovery process:** When the primary server is back online after the 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 primary-standby mode.

- **Primary server state:** Down
- **Standby server state:** Up
- **Does failover occur?** No

Related Topics

- Performing a Primary Server Planned Shutdown During an Upgrade
  You must upgrade both servers in a pair when you perform a primary server shutdown.

C.14.2.2 Primary Server: Planned Shutdown During Maintenance

In a failover, not using read-only restricted mode affects the planned shutdown of a primary server during maintenance.

If read-only restricted mode is not used, when the primary server is powered off or restarted during maintenance, then the standby server takes over from the primary server.

- **Recovery process:** The standby server is now the new primary server. When the old primary server is back online after maintenance, it will automatically synchronize with the new primary server and take over the role of standby server. Both servers will continue to operate in primary-standby mode. Be aware that 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.
- **Primary server state:** Down
- **Standby server state:** Up
- **Does failover occur?** Yes

Related Topics

- Performing a Primary Server Planned Shutdown During Maintenance
  To perform a primary server planned shutdown during maintenance, power off or restart Oracle Key Vault.

C.14.2.3 Standby Server: Planned Shutdown

In a failover, not using read-only restricted mode affects a planned shutdown in a standby server.

If read-only restricted mode is not used, when the standby server is powered off during upgrade or maintenance, the primary server continues operating as the primary server. Read and write operations are allowed.

- **Recovery process:** 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 primary-standby mode. Be aware that when the standby server is offline, data replication is disabled. If the primary server goes offline before synchronizing with the standby server, then it may cause a loss of critical data.
- **Primary server state:** Up
- **Standby server state:** Down

- **Does failover occur?** No

Related Topics

- Performing a Standby Server Planned Shutdown During an Upgrade
  You must upgrade both servers in a pair when you perform a standby server shutdown.
- Performing a Standby Server Planned Shutdown During Maintenance
  You can perform a standby server planned shutdown during maintenance from the standby server using SSH.

C.14.2.4 Primary Server: Unplanned Shutdown

In a failover, not using read-only restricted mode affects an unplanned shutdown in a primary server.

If read-only restricted mode is not used, 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 in the Oracle Key Vault management console. If the primary server cannot be reached after the specified duration has elapsed, then the standby server takes over from the primary server.

- **Recovery process:** The standby server is now the new primary server. Rectify the failure that affected the primary server by restarting the server or restoring network connectivity. When the primary server is back online, it will automatically synchronize with the new primary server and take over the role of standby server.
- **Primary server state:** Down
- **Standby server state:** Up
- **Does failover occur?** Yes

C.14.2.5 Standby Server: Unplanned Shutdown

In a failover, not using read-only restricted mode affects a standby server during an unplanned shutdown.

If read-only restricted mode is not used, when the standby server goes offline because of power loss, network failure, or hardware failure, then the primary server becomes unavailable. All operations are disabled.

- **Recovery process:** Rectify the failure that affected the standby server by restarting the server or restoring network connectivity. When the standby server is back online, it will automatically synchronize with the primary server. You cannot re-establish synchronization or network connectivity between the primary and standby servers, contact Oracle Support.
- **Primary server state:** Up
- **Standby server state:** Down
- **Does failover occur?** No

C.14.3 Failover Situations with Read-Only Restricted Mode

The use read-only restricted mode affects failover operations in Oracle Key Vault.

- Primary Server: Planned Shutdown During an Upgrade
  In a failover, not using read-only restricted mode affects a planned shutdown of a primary server during an upgrade
- Primary Server: Planned Shutdown During Maintenance
  In a failover, using read-only restricted mode affects the planned shutdown of a primary server during maintenance.
- Standby Server: Planned Shutdown
  In a failover, using read-only restricted mode affects a planned shutdown in a standby server.
- Primary Server: Unplanned Shutdown
  In a failover, using read-only restricted mode affects an unplanned shutdown in a primary server.
- Standby Server: Unplanned Shutdown
  In a failover, using read-only restricted mode affects a standby server during an unplanned shutdown.

C.14.3.1 Primary Server: Planned Shutdown During an Upgrade

In a failover, not using read-only restricted mode affects a planned shutdown of a primary server during an upgrade

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.

- **Recovery process:** 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 primary-standby mode.
- **Primary server state:** Down
- **Standby server state:** Up
- **Does failover occur?** No

Related Topics

- Performing a Primary Server Planned Shutdown During an Upgrade
  You must upgrade both servers in a pair when you perform a primary server shutdown.

C.14.3.2 Primary Server: Planned Shutdown During Maintenance

In a failover, using read-only restricted mode affects the planned shutdown of a primary server during maintenance.

When the primary server is powered off or restarted 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.

- **Recovery process:** The standby server is now the new primary server. When the old primary server is back online after maintenance, it will automatically synchronize with the new primary server and take over the role of standby server. Both servers will continue to operate in primary-standby mode.
- **Primary server state:** Down

- **Standby server state:** Up
- **Does failover occur?** Yes

Related Topics

- Performing a Primary Server Planned Shutdown During Maintenance
  To perform a primary server planned shutdown during maintenance, power off or restart Oracle Key Vault.

C.14.3.3 Standby Server: Planned Shutdown

In a failover, using read-only restricted mode affects a planned shutdown in a standby server.

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.

- **Recovery process:** When the standby server is back online post-upgrade or after 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 primary-standby mode.
- **Primary server state:** Up
- **Standby server state:** Down
- **Does failover occur?** No

Related Topics

- Performing a Standby Server Planned Shutdown During an Upgrade
  You must upgrade both servers in a pair when you perform a standby server shutdown.
- Performing a Standby Server Planned Shutdown During Maintenance
  You can perform a standby server planned shutdown during maintenance from the standby server using SSH.

C.14.3.4 Primary Server: Unplanned Shutdown

In a failover, using read-only restricted mode affects an unplanned shutdown in a primary server.

When the primary server goes offline because of power loss, network failure, or hardware failure, then the standby server waits for the duration specified in the **Fast Start Failover Threshold** field on the Configure Primary-Standby page in the Oracle Key Vault management console. If the primary server is not reachable after the specified duration has elapsed, then the standby server enters read-only restricted mode, and takes over from the primary server.

- **Recovery process:** The standby server is now the new primary server. Rectify the failure that affected the primary server by restarting the server or restoring network connectivity. When the primary server is back online, it will automatically synchronize with the new primary server and take over the role of standby server.
- **Primary server state:** Down
- **Standby server state:** Up
- **Does failover occur?** Yes

C.14.3.5 Standby Server: Unplanned Shutdown

In a failover, using read-only restricted mode affects a standby server during an unplanned shutdown.

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 Primary-Standby page of the Oracle Key Vault management console. 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.

- **Recovery process:** Rectify the failure that affected the standby server by restarting the server or restoring network connectivity. When the standby server is back online, it will automatically synchronize with the primary server. If you cannot re-establish synchronization or network connectivity between the primary and standby servers, then contact Oracle Support.
- **Primary server state:** Up
- **Standby server state:** Down
- **Does failover occur?** No

C.15 Performing a Planned Shutdown

A user who has the System Administrator role can perform planned shutdowns during an upgrade or a maintenance window.

- Primary Server Planned Shutdown
  A user who has the System Administrator role can plan a shutdown of the primary server during an upgrade or a maintenance window.
- Standby Server Planned Shutdown
  A user who has the System Administrator role can plan a shutdown of the primary server during an upgrade or a maintenance window.

C.15.1 Primary Server Planned Shutdown

A user who has the System Administrator role can plan a shutdown of the primary server during an upgrade or a maintenance window.

- Performing a Primary Server Planned Shutdown During an Upgrade
  You must upgrade both servers in a pair when you perform a primary server shutdown.
- Performing a Primary Server Planned Shutdown During Maintenance
  To perform a primary server planned shutdown during maintenance, power off or restart Oracle Key Vault.

C.15.1.1 Performing a Primary Server Planned Shutdown During an Upgrade

You must upgrade both servers in a pair when you perform a primary server shutdown.

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. After an upgrade, the primary and standby server retain their old roles.

- To perform a primary upgrade, upgrade both of the Oracle Key Vault server pair used in the primary-standby configuration.

Related Topics

- Upgrading a Pair of Oracle Key Vault Servers in a Primary-Standby Deployment
  You should allocate several hours to upgrade the primary server after upgrading the standby.

C.15.1.2 Performing a Primary Server Planned Shutdown During Maintenance

To perform a primary server planned shutdown during maintenance, power off or restart Oracle Key Vault.

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 Settings**.
3. Do one of the following:
   - Click the **Power Off** button.
   - Click the **Reboot** button.

   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 Primary-Standby 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.

After a maintenance window, the primary and standby server switch roles. When the old primary server is back online after maintenance, it will take over the role of standby server.

C.15.2 Standby Server Planned Shutdown

A user who has the System Administrator role can plan a shutdown of the primary server during an upgrade or a maintenance window.

During upgrade, the standby server shuts down automatically, and no manual steps are necessary.

- Performing a Standby Server Planned Shutdown During an Upgrade
  You must upgrade both servers in a pair when you perform a standby server shutdown.
- Performing a Standby Server Planned Shutdown During Maintenance
  You can perform a standby server planned shutdown during maintenance from the standby server using SSH.

C.15.2.1 Performing a Standby Server Planned Shutdown During an Upgrade

You must upgrade both servers in a pair when you perform a standby server shutdown.

When you restart the standby server during the upgrade, the upgrade script initiates an automatic shutdown. There are no manual steps to be performed after the standby server is restarted.

- To perform a standby upgrade, upgrade both of the Oracle Key Vault server pair used in the primary-standby configuration.

Related Topics

- Upgrading a Pair of Oracle Key Vault Servers in a Primary-Standby Deployment
  You should allocate several hours to upgrade the primary server after upgrading the standby.

C.15.2.2 Performing a Standby Server Planned Shutdown During Maintenance

You can perform a standby server planned shutdown during maintenance from the standby server using SSH.

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. Log in to the standby database instance as a user who as the `ALTER DATABASE` system privilege.

   For example:

   ```
   sqlplus sec_admin
   Enter password: password
   ```

4. Execute the `ALTER DATABASE` statement as follows:

   ```
   ALTER DATABASE RECOVER MANAGED STANDBY DATABASE CANCEL;
   ```

5. Shut down the database.

   ```
   SHUTDOWN IMMEDIATE
   ```

6. Power off the standby server.
   a. Log in to the Oracle Key Vault management console as a user who has the System Administrator role.
   b. Select the **System** tab, and then select **System Settings**.
   c. Click the **Power Off** button.

D

Security Technical Implementation Guides Compliance Standards

Oracle Key Vault follows the Security Technical Implementation Guides (STIG)-based compliance standards.

- About Security Technical Implementation Guides
  A Security Technical Implementation Guide (STIG) is a methodology followed by the U.S. Department of Defense (DOD).
- Enabling and Disabling STIG Rules on Oracle Key Vault
  You can enable STIG rules on Oracle Key Vault by enabling Strict mode.
- Current Implementation of STIG Rules on Oracle Key Vault
  You should be aware of the vulnerability categories that STIG recommendations addresses.
- Current Implementation of Database STIG Rules
  The current implementation of the database STIG rules encompass a wide range of rules.
- Current Implementation of Operating System STIG Rules
  The current implementation of the operating system STIG rules encompass a wide range of rules.

D.1 About Security Technical Implementation Guides

A Security Technical Implementation Guide (STIG) is a methodology followed by the U.S. Department of Defense (DOD).

STIG is designed 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

D.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
  You enable STIG rules (strict mode) from the command line.
- Disabling STIG Rules on Oracle Key Vault
  You disable STIG rules (strict mode) from the command line.

D.2.1 Enabling STIG Rules on Oracle Key Vault

You enable STIG rules (strict mode) from the command line.

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
```

D.2.2 Disabling STIG Rules on Oracle Key Vault

You disable STIG rules (strict mode) from the command line.

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
```

D.3 Current Implementation of STIG Rules on Oracle Key Vault

You should be aware of the vulnerability categories that STIG recommendations addresses.

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 D-1 lists the three vulnerability categories that STIG recommendations address.

Table D-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.

D.4 Current Implementation of Database STIG Rules

The current implementation of the database STIG rules encompass a wide range of rules.

Table D-2 shows the current implementation of Database STIG rules on Oracle Key Vault.

Table D-2 Current Implementation of Database STIG Rules

STIG ID	Title	Severity	Addressed by Script	Addressed by Documentation	Action required	Implemented	Notes
DG0004-ORACLE11	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-ORACLE11	DBMS application object ownership	No	No	Yes	No	No	The object owner accounts in the Oracle Key Vault server are as follows: • `KEYVAULT` • `APEX_040200` • `AVSYS` • `MANAGEMENT`
DG0014-ORACLE11	DBMS demonstration and sample databases	CAT II	No	No	None	No	All default demonstration and sample database objects have been removed.
DG0071-ORACLE11	DBMS password change variance	CAT II	No	No	No	No	Currently not supported

Table D-2 (Cont.) Current Implementation of Database STIG Rules

STIG ID	Title	Severity	Addressed by Script	Addressed by Documentation	Action required	Implemented	Notes
DG0073-ORACLE11	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-ORACLE11	DBMS links to external databases	CAT II	No	Yes	No	No	No
DG0077-ORACLE11	Production data protection on a shared system	CAT II	No	No	None	No	No
DG0116-ORACLE11	DBMS privileged role assignments	CAT II	Yes	Yes	No	No	No
DG0117-ORACLE11	DBMS administrative privilege assignment	CAT II	No	No	No	No	Currently not supported
DG0121-ORACLE11	DBMS application user privilege assignment	CAT II	No	No	No	No	Currently not supported
DG0123-ORACLE11	DBMS Administrative data access	CAT II	No	No	No	No	Currently not supported
DG0125-ORACLE11	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-ORACLE11	DBMS account password reuse	CAT II	No	No	None	No	No.
DG0128-ORACLE11	DBMS default passwords	CAT I	Yes	No	No	No	Account CTXSYS is assigned a random password in the script.
DG0133-ORACLE11	DBMS Account lock time	CAT II	Yes	No	No	No	No
DG0141-ORACLE11	DBMS access control bypass	CAT II	Yes	No	No	No	Users can use a script to audit the following events: DROP ANY SYNONYM DROP ANY INDEXTYPE

Table D-2 (Cont.) Current Implementation of Database STIG Rules

STIG ID	Title	Severity	Addressed by Script	Addressed by Documentation	Action required	Implemented	Notes
DG0142-ORACLE11	DBMS Privileged action audit	CAT II	No	No	None	No	No
DG0192-ORACLE11	DBMS fully-qualified name for remote access	CAT II	Yes	No	No	No	Currently not supported
DO0231-ORACLE11	Oracle application object owner tablespaces	CAT II	No	No	No	No	Currently not supported
DO0250-ORACLE11	Oracle database link usage	CAT II	No	Yes	No	No	No
DO0270-ORACLE11	Oracle redo log file availability	CAT II	No	No	No	No	Currently not supported
DO0350-ORACLE11	Oracle system privilege assignment	CAT II	No	No	No	No	Currently not supported
DO3475-ORACLE11	Oracle `PUBLIC` access to restricted packages	CAT II	No	No	No	No	Currently not supported
DO3536-ORACLE11	Oracle `IDLE_TIME` profile parameter	CAT II	Yes	No	No	No	No
DO3540-ORACLE11	Oracle `SQL92_SECURITY` parameter	CAT II	No	No	None	No	Parameter `SQL92_SECURITY` is already set to `TRUE`.
DO3609-ORACLE11	System privileges granted WITH ADMIN OPTION	CAT II	No	No	No	No	Currently not supported
DO3610-ORACLE11	Oracle minimum object auditing	CAT II	No	No	No	No	Currently not supported

Table D-2 (Cont.) Current Implementation of Database STIG Rules

STIG ID	Title	Severity	Addressed by Script	Addressed by Documentation	Action required	Implemented	Notes
DO3689-ORACLE11	Oracle object permission assignment to PUBLIC	CAT II	No	No	No	No	Currently not supported
DO3696-ORACLE11	Oracle RESOURCE_LIMIT parameter	CAT II	No	No	No	No	Currently not supported

D.5 Current Implementation of Operating System STIG Rules

The current implementation of the operating system STIG rules encompass a wide range of rules.

Table D-3 shows the current implementation of Operating System STIG Rules on Oracle Key Vault.

Table D-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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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
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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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
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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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
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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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
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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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
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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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
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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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
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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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
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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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
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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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 D-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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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
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

Table D-3 (Cont.) Current Implementation of Operating System STIG Rules

STIG ID	Title	Severity	Key Vault Server - Default	Key Vault Server - STIG
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 D-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

Glossary

appliance

The format in which Oracle Key Vault is made available. The Oracle Key Vault software appliance includes the operating system, the software that implements the Oracle Key Vault functionality, the database, the replication software, and other related components. Oracle Key Vault is delivered as a software image that is installed on a standalone computer, or machine, supplied by the user. Oracle provides all updates for the software on the appliance, including the operating system. Do not load additional software on the Oracle Key Vault appliance.
You can deploy an Oracle Key Vault appliance as a standalone server, a member of a primary-standby configuration, or a node in a multi-master cluster.

Audit Manager

An Oracle Key Vault administrative role that enables a user to manage audit lifecycle and policies and to separate the role of auditing from the role of managing the Oracle Key Vault server.

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.

candidate node

During node induction, an Oracle Key Vault server to be added to a multi-master cluster. A candidate node must be a freshly installed Oracle Key Vault appliance, except when it is the initial node, in which case it provides the entirety of the cluster's initial data. A candidate node must be at the same release and patch level as the multi-master cluster to which it is being added.

After the server has been inducted into a cluster, it is a called a node. After a successful node induction, you can configure the server to use the cluster-wide configuration settings. The cluster data set is then replicated to the node.

cluster data set

The set of all security objects managed by the cluster. When creating the cluster, the initial node provides all of the security objects that will be part of the initial cluster data set.

cluster link

A link that represents the outbound network connection (to the node) and the inbound replication process (from the node). You can enable or disable the link to manage node data replication.

cluster subgroup

A group of one or more nodes that is a subgroup of a cluster. Each node in a cluster can belong to only one subgroup. The node is assigned to a subgroup when the node is added to the multi-master cluster and this assignment remains unchanged for the lifetime of the node. The assignment is for each node, and members of a read-write pair can be in different subgroups.

The subgroup implements a notion of endpoint affinity. It is used when you set the endpoint's node search order in the endpoint node scan list. Nodes in the same subgroup as the node where the endpoint was added are considered local to the endpoint. The local subgroup is scanned first before communicating with nodes that are not in the local subgroup.

The cluster topology can change when you add or remove new nodes to and from the cluster. The endpoints get this information with the response messages for the operations the endpoint initiated. Oracle Key Vault periodically sends the updated endpoint node scan list back to the endpoint even if there is no change to cluster topology. This is to account for any lost messages.

controller node

A node that controls or manages a cluster reconfiguration change, such as adding, enabling, disabling, or removing nodes. A node is only a controller node while the change is being made. During node induction, the controller node provides the server certificate and the data that is used to initialize the candidate node.

Each concurrent operation will have its own controller node. One controller node can only control one cluster configuration transaction at a time.

credential file

A file that contains sensitive information such as user IDs, passwords, and keys. The file, such as a Kerberos keytab file, is stored as an opaque object, which means that its individual contents are not interpreted by Oracle Key Vault. The entire file is uploaded and downloaded as an object.

See also security object.

default wallet

A special virtual wallet that is associated with an endpoint, into which all the endpoint's security objects can be automatically uploaded.

deleted node

A node that has been disassociated from the cluster, either by using the **Delete** or **Force Delete** buttons on the Oracle Key Vault management console. If it has been disabled for longer than the Maximum Disable Node duration, then you must delete the node.

Once a node has been deleted, you cannot re-associate it with the cluster. If it is to be inducted into the cluster, then you must re-image it and then convert into a freshly installed server.

You can use the **Delete** option under normal operating circumstances. Only use the **Force Delete** option if the node is unreachable when the **Delete** option does not work.

endpoint

A computer system such as a database server, an application server, and other information systems, where keys are used to access encrypted data and credentials are used to authenticate to other systems.

endpoint administrator

Owner of an endpoint. Endpoint administrators can be 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 collection of endpoints that are created to share a set of security objects.

endpoint node scan list

A list of nodes to which an endpoint can connect.

heartbeat lag

A monitored metric that determines the health of the multi-master cluster. This is an indication of the node and network health. It is the time since the current node received a heartbeat message from a given node. A heartbeat is sent out from each node every two minutes. Every heartbeat should be received on each other node shortly thereafter.

A higher heartbeat lag indicates that the user operations that require conflict resolution like creating a wallet will take longer. Heartbeat lags between any two nodes affect the operations cluster wide. If the heartbeat lag is high, ensure that the cluster services are active and that replication is active. Disable and then re-enable the links between the two nodes between which the heartbeat lag is significant.

initial node

The first, or initial, node of an Oracle Key Vault Multi-Master Cluster. You create a multi-master cluster by converting a single Oracle Key Vault server to become the initial node. The Oracle Key Vault server can be a clean installed Oracle Key Vault server, or it can already be in service with active data. A standalone server or a member of a primary-standby configuration can be converted to be the initial node of a cluster. If you want to use a member of a primary-standby configuration, then you must first break the primary-standby relationship splitting the pair.

If the initial node has been active and therefore has data, then Oracle Key Vault uses this data as the cluster data set to initialize the cluster.

Initialization can occur only once in the life of the cluster.

installation passphrase

A password that is specified during the Oracle Key Vault installation. The installation passphrase is used to log in to Oracle Key Vault and complete the post-installation tasks. The installation passphrase can only be changed on the Oracle Key Vault management console after installation but before post-installation. After you complete the post-installation process, this option no longer appears on the management console.

JAVA_HOME

The environment variable that points to the location of Java files (JDK/JRE) in the system. This allows Java applications to look up the `JAVA_HOME` variable in order to operate.

Java keystore file

A file that can hold multiple security objects such as keys and certificates. It uses the Java Keystore File (JKS) format.

Key Administrator

An Oracle Key Vault administrator role that enables a user to manage the key lifecycle and control access to all security objects within Oracle Key Vault. This is a highly sensitive role and should be granted with care.

keystore

A generalized term for a container that stores encryption keys including but not limited to TDE master encryption keys.

Management Information Base (MIB)

See MIB.

master encryption key

See TDE master encryption key.

maximum disable node duration

The time, in hours, that a node may remain in the disabled state. If the node has been disabled for a longer duration, it can no longer be enabled.

The default maximum disable node duration is 24 hours.

MIB

Management information base; a text file that, if Oracle Key Vault is monitored through SNMP, 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 that is monitored.

name resolution time

A monitored metric used to determine the health of the multi-master cluster. It is the average time taken to ascertain that there is no name conflict in the cluster or to resolve the name conflict after an attempt to use conflicting names took place.

node

A Oracle Key Vault server that has been converted to be member of a Oracle Key Vault multi-master cluster. It is known as an Oracle Key Vault cluster node or simply a node.

node induction

The process of converting an Oracle Key Vault server to be a node in the multi-master cluster.

The initial node in a cluster provides the initial cluster data set. Subsequently, only new Oracle Key Vault servers can be inducted to the multi-master cluster, and the current data in the multi-master cluster is loaded into the new nodes.

OKV_HOME

The environment variable that points to the location in which the Oracle Key Vault endpoint software will reside. It contains sub-directories for endpoint software such as the configuration files, log files, libraries, binaries, and other files that the endpoint software utility needs.

online master key

A TDE-generated master encryption key that is stored in Oracle Key Vault. The online master key enables Oracle Key Vault administrators to have full control over the TDE master encryption keys that Key Vault protects. When a key rotation is performed on the online master key, the change is reflected in all other nodes in a cluster. In previous releases, the term for online master key was TDE direct connection.

Opaque Object

A security object that Oracle Key Vault cannot interpret.

Oracle Key Vault appliance

See appliance.

Oracle Key Vault multi-master cluster

A distributed set of Oracle Key Vault nodes that are grouped together so that they all communicate with one another. Some pairs of nodes are configured as read-write pairs. In a read-write pair, an update to one node is replicated to the other node, and the update must be verified on the other node before the update is considered successful.

All nodes in the multi-master cluster connect to all other nodes. Data updated in a read-write pair is replicated to all nodes.

Oracle Key Vault node

See node.

Oracle Key Vault server

An Oracle Key Vault server that is a standalone installation of the Oracle Key Vault appliance. It provides all the core functionality related to endpoints and wallets.

Oracle wallet file

A container that can hold multiple security objects such as keys and certificates. It uses the PKCS#12 cryptographic standard.

You can manage Oracle wallets in Oracle Key Vault just like other security objects. Optionally, you can encrypt them and protect them with a password. An Oracle wallet that can be accessed without a password is called an auto-login wallet.

See also password-protected wallet.

ORACLE_BASE

The environment variable that points to 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 environment variable that points to the directory path to install Oracle components (for example, `/u01/app/oracle/product/18.3.0/db_n`). You are prompted to enter an Oracle home in the **Path** field of the Specify File Locations window.

`ORACLE_HOME` corresponds to the environment in which Oracle Database products run. If you install an OFA-compliant database, using Oracle Universal Installer defaults, then the 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 executable files and network files.

ORACLE_SID

The environment variable that represents the Oracle System ID (SID), which uniquely identifies a particular database on a system. For this reason, you cannot have more than one database with the same SID on a computer system.

When using Oracle Real Application Clusters, you must ensure that all instances that belong to the same database have a unique SID.

oraenv

Along with `coraenv`, a Unix/ Linux command line utility that sets 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, then commands such as `sqlplus`, `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 encryption keys.

PKCS#12 file

In cryptography, PKCS#12 defines an archive file format for storing many cryptographic objects as a single file. Wallet files are stored in PKCS#12 format.

read-only node

A node that is not part of a replication pair. Most data cannot be directly updated using the Oracle Key Vault management console, or with Oracle Key Vault client software. Critical data such as keys, wallets, and certificates in a read-only node is only updated through replication from read-write nodes.

read-only restricted mode

A node enters read-only restricted mode when it has no read-write pair, or if its read-write peer is unavailable. The Oracle Key Vault console displays a warning that the node is operating in read-only restricted mode. In read-only restricted mode, updates using the Oracle Key Vault management console, or Oracle Key Vault client software are restricted. However, you can still perform system configuration on the node.

When the node is a member of a read-write pair, this indicates the other node has been disabled but not deleted from the cluster, or the heartbeat is not detected for other reasons.

read-write mode

A node is in read-write mode when it is available for endpoint and wallet data updates using the Oracle Key Vault management console, or Oracle Key Vault client software. The node must be a member of a read-write pair, and the read-write peer must be online and active.

When both nodes in the pair are available, both nodes can accept updates, and all updates to one node are synchronously replicated to the peer. If one of the nodes in the pair becomes unavailable, then the remaining node enters read-only restricted mode and will not accept any data updates until the peer is restored.

The node state is displayed on the Monitoring page of the **Cluster** tab of the node management console. The **Cluster** tab of the node management console displays the type and status of all nodes in the cluster.

read-write node

An active, connected, member of a read-write pair of nodes.

read-write pair

A pair of nodes that operates with bidirectional synchronous replication. You create the read-write pair by pairing a new node with a read-only node. You can update data, including the endpoint and wallet data, in either node by using the Oracle Key Vault management console, or Oracle Key Vault client software. The updates are replicated immediately to the other node in the pair. Updates are replicated asynchronously to all other nodes.

A node can be a member of at most one bidirectional synchronous pair.

A multi-master cluster requires at least one read-write pair to be fully operational. It can have a maximum of 8 read-write pairs.

read-write peer

The specific member of one, and only one, read-write pair in the cluster. Each read-write pair consists of only two nodes. You configure nodes as peers by setting **Add Candidate Node as Read-Write Peer** to **Yes** on the controller node during induction of the candidate node. Peers are identified on the Cluster Management Configuration page.

If one member of the pair is deleted, then the peer automatically becomes a read-only node.

recovery passphrase

A secret token that is created during the installation of an Oracle Key Vault appliance. The recovery passphrase created for the initial node is subsequently used by the cluster and propagated to all other nodes in the cluster.

You enter the existing recovery passphrase on both the controller page and the candidate page during induction of any nodes into the cluster. Because there is only one recovery passphrase, you must use that same recovery passphrase when the recovery passphrase is required.

replication

The process of replicating data changes that were made to a read-write node to all other nodes. The read-write peer is updated immediately. Replication is used to distribute the data to all other nodes in the cluster.

replication lag

A monitored metric that determines the health of the multi-master cluster. It is the time taken for an object to be replicated to another node.

A higher replication lag indicates that the Oracle Key Vault operations like changing the access permissions for an endpoint on the wallet will take longer to replicate. Depending on the operation, a replication lag may or may not have a cluster-wide impact. If the replication lag is significant between two notes, then you should disable and re-enable the cluster links.

security object

An object that contains critical data provided by the user. A security object can be of the following types:

- private encryption key
- Oracle wallet
- Java keystore
- Java Cryptography Extension keystore
- certificate
- credential file

software appliance

A self-contained preconfigured product that can be installed on supported hardware dedicated for a specific purpose.

sqlnet.ora

An Oracle Database configuration file for the client or server. By default, the `sqlnet.ora` file resides in `$ORACLE_HOME/network/admin` directory. It specifies the following connection information:

- 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 Oracle Key Vault 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. This is a highly sensitive role and should be granted with care.

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, then the attributes in the template are automatically assigned to the new object.

user

A staff member who uses Oracle Key Vault. Users can be administrators, auditors, or ordinary users with no administrative roles.

user group

A named collection of Oracle Key Vault users. A user group can collectively be granted privileges or roles.

virtual wallet

A container for security objects such as 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

A

about managing, *7-19*
access control, *2-3*
  access grants for virtual wallets, *2-4*
  how configuration works, *2-3*
access control options, *2-4*
access grants, *2-4*
Actions menu, *4-34*
activate Command, *13-82*
add_attr command, *13-75*
add_custom_attr command, *13-75*
add_epg_member command, *13-35*
add_member command, *13-85*
add_wallet_access_ep command, *13-43*
add_wallet_access_epg command, *13-45*
adding user to user group, *7-22*
administration users
  multi-masters clusters effect on, *7-4*
administrative roles
  about, *2-5*
  about managing, *7-8*
  Audit Manager
    about, *2-7*
  granting or changing, *7-8*
  Key Administrator
    about, *2-7*
  revoking, *7-10*
  separation of duty, *2-6*
  System Administrator
    about, *2-7*
alerts, *1-10*
  about, *17-17*
  configuring, *17-19*
  types of, *17-17*
  viewing open alerts, *17-20*
all_attr command, *13-76*
appliance automation
  commands
    enrollment token management, *13-17*
    virtual wallet management, *13-42*
architecture, *3-3*
archiving
  credential files, *18-25*
Audit Manager
  about, *2-7*
Audit Manager role
  multi-master cluster effect on, *7-3*
Audit Vault and Database Firewall
  consolidating audit records, *17-25*
auditing
  about, *17-22*
  Audit Vault and Database Firewall,
      consolidating audit records, *17-25*
  deleting audit records, *17-25*
  exporting audit records to file, *17-25*
  multi-master clusters, *17-24*
  syslog file, *17-23*
  viewing audit records, *17-24*
Automatic Storage Management
  uploading keystores
    about, *19-9*
    copying keystore to, *19-11*
    procedure, *19-10*

B

backing up data
  about, *14-1*
  best practices, *14-16*
  chanding schedule, *14-11*
  deleting schedule, *14-11*
  primary-standby deployments, *14-11*
  recovery passphrase, *14-12*
  scheduling backup, *14-10*
backup destinations
  about, *14-2*
  changing settings, *14-6*
  creating remote, *14-4*
  deleting remote, *14-7*
backup scheduling, *14-9*
  about, *14-8*
  types, *14-8*
backups
  console certificates, *16-8*
  protecting with recovery passphrase, *14-12*
  reports, *17-29*
  types, *14-8*

benefits
centralizing key lifecyle management, 1-2
centralizing key storage, 1-2
fighting security threats, 1-2

C

candidate nodes, 3-10
centralized storage
Java keystores, 1-4
Oracle wallet files, 1-4
centralized storage and management of security objects, 1-9
certifcates
rotating, about, 16-1
rotating, procedure, 16-3
rotation, checking status, 16-5
certificate
get_cert command, 13-70
reg_cert command, 13-72
certificates, 16-6
rotating, advice, 16-2
rotating, factors that may affect process, 16-2
See also console certificates
changepwd command (okvutil), B-3
changing a user group description, 7-23
check_object_status command, 13-46
cluster node types
about, 3-6
cluster nodes
about, 3-4
limitation, 3-4
cluster size and availability guidance, 3-14
cluster subgroup
about, 3-5
clusters
creating first node, 5-2
deleting a node, 5-9
disabling a node, 5-8
disabling node replication, 5-11
enabling a node, 5-9
enabling node replication, 5-11
force deleting a node, 5-10
management information, 5-11
monitoring information, 5-13
read-only, creating, 5-5
read-write pair of nodes, creating, 5-3
read-write pairs of nodes, creating, 5-7
restarting cluster services, 5-11
terminating node pairing, 5-7
Commercial National Security Algorithm (CNSA)
about, 15-27
backup and restore operations, 15-29
running scripts, 15-28
Commercial National Security Algorithm (CNSA) *(continued)*
upgrading primary-standby Oracle Key Vault servers, 15-31
upgrading standalone Oracle Key Vault server, 15-30
configuration files
endpoint configuration file, 10-11
configuration parameters
endpoints, 9-25
configuring a primary-standby deployment, 6-2
conflicts in names of objects, 5-15
console certificates, 16-6
about managing, 16-6
backup data restored, 16-8
downloading CA request, 16-6
having signed, 16-7
primary-standby environments, 16-8
RESTful services, 16-8
uploading, 16-7
controller nodes
about, 3-9
create_endpoint command, 13-18
create_endpoint_group command, 13-36
create_key command, 13-68
create_unique_endpoint command, 13-19
create_unique_endpoint_group command, 13-37
create_unique_wallet command, 13-47
create_wallet command, 13-49
creating a user group, 7-20
creating user accounts, 7-4
credential files
about archiving and downloading, 18-25
change to content guidance, 18-28
downloading
guidance, 18-28
procedure, 18-27
overwriting danger of, 18-28
sharing with multiple endpoints guidance, 18-28
uploading
guidance, 18-28
procedure, 18-26
critical data, 3-5

D

dashboard
status panes, 15-4
viewing, 15-2
data
backing up, about, 14-1
restoring, about, 14-1
Database as a Service
about configuring for Key Vault, 12-2
configuring instance, 12-2

Database as a Service *(continued)*
creating low privileged user, *12-3*
deleting SSH tunnel, *12-11*
disabling SSH tunnel, *12-9*
enrolling instance as endpoint
about, *12-12*
installing Oracle Key Vault software onto, *12-15*
post-installation tasks, *12-17*
preparing environment, *12-15*
registering, *12-13*
resuming access to Oracle Key Vault, *12-20*
reverse SSH tunnel in multi-master cluster, *12-8*
reverse SSH tunnel in primary-standby configuration, *12-8*
SSH tunnel between Oracle Key Vault and DBaas instance, *12-5*
SSH tunnel not active, *12-11*
suspending access to Oracle Key Vault
about, *12-18*
procedure, *12-19*
users
low privileged user for DBaaS, *12-3*
viewing SSH tunnel details, *12-9*
del_attr command, *13-77*
del_custom_attr command, *13-78*
del_member command, *13-86*
delete_endpoint command, *13-21*
delete_endpoint_group command, *13-38*
delete_wallet command, *13-50*, *13-61*
deleting user accounts, *7-7*
deleting user groups, *7-24*
deployment
architecture, *2-2*
overview, *1-16*
deployment scenarios
cluster size and availability, *3-14*
mid-size cluster, *3-16*
two data centers, *3-16*
two nodes, *3-14*
deployments
credential files, archiving and downloading, *18-25*
Java keystores, uploading and downloading, *18-19*
JKS and JCEKS keystores, archiving and downloading, *18-23*
migrating standalone Key Vault server to multi-master cluster, *3-13*
online master keys for TDE wallets, *1-5*
Oracle wallets, uploading and downloading, *18-19*
primary-standby to multi-master cluster, *3-13*
recommendations for, *5-16*
destroy command, *13-82*
diagnostic reports, *17-14*
about, *17-14*
generating file, *17-16*
installing generation utility, *17-15*
removing diagnostic generation utility, *17-16*
removing temp files, *17-16*
diagnostics
accessing with okvutil diagnostics, *B-3*
diagnostics generation utility
transaction check error, *C-8*
disk space, adding, *4-25*
DNS
configuring NTP servers for non-multi-master clusters, *15-9*
nodes, *15-15*
DNS settings
multi-master clusters, *15-18*
download command, *13-22*
download command (okvutil), *B-4*
downloading
credential files, *18-27*
JKS and JCEKS keystores, *18-23*, *18-24*
wallets, *18-21*
downtime, minimizing, *15-32*
drop_epg_member command, *13-39*
drop_wallet_access_ep command, *13-51*
DSA keys, removing after upgrade, *4-28*

E

email
modify_endpoint_email command, *13-29*
email addresses
changing, *7-18*
disabling email notifications, *7-18*
email notification
about, *17-9*
configuring, *17-10*
disabling, *17-13*
testing, *17-12*
emergency system recovery, *2-8*
endpoint administrators
about, *2-9*
endpoint database requirements, *4-4*
endpoint groups, *9-18*
access grant to virtual wallet, *9-21*
add_epg_member command, *13-35*
add_wallet_access_epg command, *13-45*
adding endpoint too, *9-21*
create_endpoint_group command, *13-36*
create_unique_endpoint_group command, *13-37*
creating, *9-18*
delete_endpoint_group command, *13-38*

endpoint groups *(continued)*
deleting, *9-23*
drop_epg_member command, *13-39*
modfify_endpoint_group_desc command, *13-40*
modify_wallet_access_epg command, *13-52*, *13-58*
modify_wallet_name command, *13-60*
modifying details, *9-20*
modifying virtual wallets from Keys & Wallets tab, *8-7*
multi-master clusters, effect on, *9-17*
removing access to virtual wallets from Keys & Wallets tab, *8-6*
removing endpoint, *9-22*
endpoint node scan lists
about, *3-20*
endpoint platforms, supported, *4-3*
endpoint self-enrollment, about, *9-3*
endpoints, *9-18*
about, *9-10*
about managing, *9-1*
add_epg_member command, *13-35*
adding access to virtual wallet, *9-15*
adding to an endpoint group, *9-21*
adding using administrator-initiated enrollment, *9-5*
adding using self-enrollment, *9-8*
adding using self-enrollment, about, *9-8*
adding using self-enrollment, procedure for, *9-9*
administrators for, *9-1*
alternative for individual, *9-11*
associating default wallet with, *9-13*
configuration file, *10-11*
configuration parameters, about, *9-25*
configuration parameters, setting, *9-26*
create_endpoint command, *13-18*
create_unique_endpoint command, *13-19*
DBaaS
enrolling, *12-12*
registering, *12-12*
default wallet, setting for, *9-13*
delete_endpoint command, *13-21*
deleting, *9-10*, *9-11*
details
about, *9-23*
configuration parameters, about, *9-25*
configuration parameters, setting, *9-26*
modifying, *9-24*
diagnostics, *B-3*
download command, *13-22*
downloading software, *10-3*
drop_epg_member commandd, *13-39*
drop_wallet_access_ep command, *13-51*

endpoints *(continued)*
endpoint node scan lists, *3-20*
enrolling and provisioning, *10-3*
enrollment
about, *10-1*
administrator initiated, about, *9-3*
types of enrollment, *9-3*
enrollment in multi-master cluster, *9-4*
enrollment process
about, *10-1*
enrollment types, *9-3*
get_enrollment_token command, *13-23*
guidance on enrolling across deployments, *3-16*
installing software for new enrollment, *10-5*
Java home, how determined, *10-8*
limitiations of TDE endpoint integration, *18-2*
modify_endpoint_desc command, *13-30*
modify_endpoint_email command, *13-29*
modify_endpoint_group_name command, *13-41*
modify_endpoint_name command, *13-31*
modify_endpoint_platform command, *13-32*
modify_endpoint_type command, *13-33*
modifying virtual wallets from Keys & Wallets tab, *8-7*
multi-master clusters, effect on, *9-2*
nodes available for connection, *3-20*
not using Oracle Key Vault client software, *10-10*
okvclient.ora file, *10-11*
okvutil utility for provisioning, *B-1*
one or more endpoints, *9-10*
Oracle Cloud Infrastructure database instance
about, *12-1*
password, changing, *B-3*
post-installation for new enrollment, *10-7*
preparing environment for new enrollment, *10-5*
provision command, *13-24*
provisioning
about, *10-1*
re_enroll command, *13-27*
re_enroll_all command, *13-28*
reenrolling, *9-12*
removing access to virtual wallets from Keys & Wallets tab, *8-6*
removing from an endpoint group, *9-22*
reports, *17-27*
revoking access to virtual wallet, *9-16*
suspending, *9-11*
TDE endpoint management, *10-10*
upgrading for enrolled, *9-32*

endpoints *(continued)*
upgrading for unenrolled
downloading Oracle Key Vault okvclient.jar software, *9-28*
installing Oracle Key Vault okvclient.jar software, *9-29*
post-installation tasks, *9-30*
preparing environment, *9-27*
upgrading from unenrolled endpoint, *9-27*
wallet items, viewing, *9-16*
See also endpoint groups
enrolling endpoints
about, *10-1*
administrator initiated
about, *9-3*
script to automatically enroll using RESTful commands, *13-10*
self-initiated
about, *9-3*
environment variables
JAVA_HOME, how determined during client installation, *10-8*
OKV_HOME
non-database utilities, *10-9*
set during installation, *10-9*
okvclient.ora location of, *10-9*
persistent master encryption key cache, *18-11*
sqlnet.ora file, *10-10*
Error
Object is Unstorable in Container error, *B-4*
errors
about, *13-62*
at command line, *13-63*
error reporting
while running commands from script, *13-63*
EXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN parameter, *18-16*

F

failover situations
read-only restricted mode, *C-11*
failovers
restoring primary-standby after, *6-12*
FIPS 140–2, *2-9*
FIPS mode, *2-9*
disabling, *15-15*
disabling for non-multi-master clusters, *15-9*
enabling, *15-15*
enabling for non-multi-master clusters, *15-9*
FIPS-Inside
See FIPS mode

G

get_attr command, *13-78*
get_cert command, *13-70*
get_default_wallet command, *13-53*
get_enrollment_token command, *13-23*
get_key command, *13-70*
get_object_name command, *13-54*
get_opaque command, *13-71*
get_secret command, *13-71*
get_wallets command, *13-55*
granting access to objects or users, *2-4*

I

initial node
creating, *5-2*
creation of, *3-7*
installation and configuration
about, *4-1*
downloading appliance software, *4-5*
endpoint database requirements, *4-4*
endpoint platform, supported, *4-3*
installing appliance software procedure, *4-6*
network port requirements, *4-3*
post-installation tasks, *4-10*
system requirements, *4-2*
installing appliance software, *4-6*
interfaces, *1-15*
management console, *1-15*
okvutil endpoint utility, *1-16*
RESTful services, *1-16*

J

Java keystores
downloading, *B-4*
uploading, *B-8*
JAVA_HOME environment variable
how determined during client installation, *10-8*
location determined during installation, *10-8*
JKS and JCEKS keystores
downloading
JKS and JCEKS keystores, *18-23*
procedure, *18-24*
uploading
procedure, *18-23*
JKS and JCKS keystores
change to content guidance, *18-25*
downloading
guidance, *18-25*
overwriting danger of, *18-25*
sharing with multiple endpoints guidance, *18-25*

JKS and JCKS keystores *(continued)*
uploading
guidance, *18-25*

K

Kerberos keytabs
downloading, *B-4*
kernels, removing after upgrade, *4-23*
key
get_key command, *13-70*
reg_key command, *13-69*
Key Administrator role
about, *2-7*
multi-master cluster effect on, *7-3*
key attributes
add_attr command, *13-75*
all_attr command, *13-76*
custom, add_custom_attr command, *13-75*
custom, mod_custom_attr command, *13-81*
del_attr command, *13-77*
get_attr command, *13-78*
list_attr command, *13-79*
mod_attr command, *13-80*
key lifecycle management, *1-10*
key rotation, *1-5*
keyattributes
custom, del_custom_attr command, *13-78*
keys
activate Command, *13-82*
activating, *8-12*
create_key command, *13-68*
deactivating, *8-12*
deleting, *8-13*
destroy command, *13-82*
finding for Key Vault, *B-6*
get_object_name command, *13-54*
locate command, *13-83*
multi-master clusters, effect on, *8-11*
query command, *13-84*
reports, *17-28*
revoke command, *13-84*
revoking, *8-12*
state of, managing, *8-11*
keystores
Automatic Storage Management
about uploading from, *19-9*
copying keystore to, *19-11*
procedure for uploading from, *19-10*
KMIP Protocol, *1-13*

L

list command (okvutil), *B-6*
list_attr command, *13-79*
list_wallet command, *13-86*
local backup destinations
about, *14-2*
locate command, *13-83*
log file locations, *C-6*
logging in to management console, *4-15*

M

management console
about, *1-15*
logging in to, *4-15*
overview of, *4-34*
Management Information Base (MIB) variables, *17-6*
master encryption keys
persistent master encryption key cache, *1-12*, *18-11*
TDE,
See persistent master encryption key cache
user-defined key as, *18-28*
maximum disable node duration
multi-master clusters, *15-18*
mod_attr command, *13-80*
mod_custom_attr command, *13-81*
modfify_endpoint_group_desc command, *13-40*
modify_endpoint_desc command, *13-30*
modify_endpoint_email command, *13-29*
modify_endpoint_group_name command, *13-41*
modify_endpoint_name command, *13-31*
modify_endpoint_platform command, *13-32*
modify_endpoint_type command, *13-33*
modify_wallet_access_ep command, *13-56*
modify_wallet_access_epg command, *13-52*, *13-58*
modify_wallet_desc command, *13-59*
modify_wallet_name command, *13-60*
monitoring
diagnostic reports, *17-14*
email notification, *17-9*
Oracle Audit Vault, *15-16*, *17-17*
remote monitoring, *17-2*
reports, *17-26*
SNMP, *17-2*
syslog configuration, *17-13*
monitoring information for clusters, *5-13*
multi-master clusters, *3-3*
about managing, *5-2*
addition of new server to cluster, *3-10*
addition of nodes, *3-10*
administration users, effect on, *7-4*
Audit Manager role, affect on, *7-3*
auditing, *17-24*
backup and restore operations, *14-1*
benefits, *3-2*

multi-master clusters *(continued)*
building and managing, about, *3-7*
candidate node, *3-10*
changing recovery passphrase, *15-22*
checking upgrade success, *4-32*
cluster node, *3-4*
cluster subgroup, *3-5*
controller node, *3-9*
critical data, *3-5*
difference from primary-standby configuration, *6-4*
DNS for individual nodes configuring, *15-15*
DNS settings, *15-18*
downtime, minimizing, *15-32*
effect on role management, *7-8*
endpoint enrollment, *9-4*
endpoint groups, effect on, *9-17*
endpoints, *9-4*
endpoints, effect on, *9-2*
expansion of
about, *3-9*
addition of more nodes, *3-10*
controller node, *3-9*
FIPS mode for individual nodes, setting, *15-15*
host name network setting for individual nodes, *15-13*
inconsistency resolution, *3-19*
initial node, *3-7*
Key Administrator role, effect on, *7-3*
keys, effect on, *8-11*
maximum disable node duration, *15-18*
mid-size cluster, *3-16*
migrating standalone Key Vault server to, *3-13*
mode types, *3-6*
multi-master clusters expansion of candidate nodes, *3-10*
name conflict resolution, *3-19*
network services for individual nodes, *15-14*
node limitations, *3-4*
operations permitted on modes, *3-7*
Oracle Audit Vault, *15-16*, *17-17*
Oracle Key Vault management console, setting timeout, *15-20*
overview, *3-1*
pre-upgrade, *4-31*
preparation for pre-upgrade, *4-30*
primary-standby to multi-master cluster, *3-13*
read-only mode, *3-6*
read-only node, *3-6*
read-only restricted mode, *3-6*
read-write mode, *3-6*
multi-master clusters *(continued)*
read-write node, *3-5*
reconfiguration changes, *3-9*
RESTful services enablement, *15-19*
restore operations, *14-14*
reverse SSH tunnels, *12-8*
rolling back pre-upgrade, *4-33*
security objects, effect on, *8-11*
size and availability, *3-14*
SNMP settings, *15-19*
syslog destination
clearing, *17-14*
setting, *17-13*
syslog settings, *15-19*
syslog settings, node, *15-16*
System Administrator role, effect on, *7-3*
system settings
about, *15-17*
system settings for individual nodes, *15-13*
system time for individual nodes
setting, *15-14*
system users, effect on, *7-4*
time settings, *15-18*
two data centers, *3-16*
two nodes, *3-14*
upgrade, about, *4-29*
upgrading each node, *4-31*
user accounts, effect on, *7-2*
user groups, changing description, *7-23*
user groups, creating in, *7-20*
user groups, deleting, *7-24*
user groups, effect on, *7-19*
user groups, removing users from, *7-23*
user groups, renaming, *7-23*
users, effect on, *7-4*
virtual wallet user access to, *7-9*
MySQL integration with Oracle Key Vault, *19-12*

N

naming conflicts
about, *5-15*
accepting suggested name, *5-16*
changing suggested name, *5-16*
finding, *5-15*
network details
configuring for non-multi-master clusters, *15-5*
network port requirements, *4-3*
network services
configuring for non-multi-master clusters, *15-6*
nodes
creating first node, *5-2*
deleting, *5-9*

nodes *(continued)*
disabling, *5-8*
disabling replication, *5-11*
enabling, *5-9*
enabling replication, *5-11*
force deleting, *5-10*
restarting cluster services for, *5-11*
terminating pairing of, *5-7*
NTP servers
configuring DNS for non-multi-master clusters, *15-9*

O

OASIS Key Management Interoperability Protocol (KMIP)
Oracle Key Vault implementation of, *1-13*
OKV_HOME environment variable
non-database utilities, *10-9*
okvclient.jar
downloading for installation on endpoint, *10-3*
okvclient.ora file
about, *10-11*
okvutil utility, *1-15*, *1-16*
about, *1-15*, *1-16*
changepwd command, *B-3*
diagnostics command, *B-3*
download command, *B-4*
list command, *B-6*
syntax, *B-1*
upload command, *B-8*
used to manage endpoints, *B-1*
online master keys, *1-5*
about using with Oracle Key Vault, *1-5*
centralized management of TDE keys, *1-5*
Oracle Data Guard connection, *19-6*
Oracle GoldenGate, *19-3*
opaque
get_opaque command, *13-71*
reg_opaque command, *13-73*
operations, restrictions and conditions of, *A-1*
options for access control, *2-4*
Oracle Active Data Guard
support for data moves, *19-12*
Oracle Audit Vault
enabling Oracle Key Vault to send data to, for non-multi-master clusters, *15-11*
integration for node, *15-16*, *17-17*
Oracle Cloud Infrastructure database instance endpoints
about, *12-1*
Oracle Data Guard
migrating Oracle wallets, *19-6*
online master keys connection, *19-6*
reverse migrating wallets, *19-7*
Oracle Data Guard *(continued)*
uploading wallets to Oracle Key Vault, *19-5*
Oracle Data Pump support for data moves, *19-12*
Oracle GoldenGate
online master keys with
about, *19-3*
TDE wallet migration
about, *19-3*
wallets used with, *19-2*
Oracle Key Vault
administering cluster environments, *15-12*
benefits, *1-2*
deployment architecture, *2-2*
deployment overview, *1-16*
key management, about, *1-1*
RESTful services, *13-1*
standards and protocols, *1-13*
who should use, *1-8*
Oracle Key Vault client software
endpoints not using, *10-10*
Oracle Key Vault compute instance
about, *11-1*
about provisioning, *11-3*
benefits, *11-2*
finding image, *11-5*
launching process, *11-5*
launching, about, *11-4*
migrating data out of compute instance, *11-10*
migrations, about, *11-9*
post-installation tasks, *11-6*
post-launch tasks, *11-6*
prerequisites, *11-4*
restarting, *11-7*
starting, *11-7*
stopping, *11-7*
system settings, *11-8*
terminating, *11-8*
transitioning data into a compute instance, *11-10*
Oracle Key Vault compute instances
backup operations, *11-8*
restore operations, *11-8*
Oracle Key Vault concepts, *2-1*
Oracle Key Vault endpoint utility, *1-15*, *1-16*
about, *1-15*, *1-16*
See also okvutil utility
Oracle Key Vault features
ASM cluster file system encryption key management, *1-14*
audit and monitoring services, external support for, *1-14*
backup and restore support for security objects, *1-12*

Oracle Key Vault features *(continued)*
centralized storage and management of security objects, *1-9*, *1-13*
database release and platform support, *1-14*
DBaaS endpoint support, *1-15*
endpoint enrollment, automatic using RESTful services, *1-13*
HSM integration, *1-15*
key lifecycle management, *1-10*
MySQL integration, *1-14*
persistent master encryption key cache, *1-12*
primary-standby environment support, *1-11*, *15-26*
reporting and alerts, *1-10*
RESTful service support for key management, *1-13*
separation of duties, *1-11*
Oracle Key Vault general system administration
about, *15-2*
Oracle Key Vault interfaces, *1-15*, *1-16*
management console, *1-15*
RESTful services, *1-16*
Oracle Key Vault keys
finding, *B-6*
Oracle Key Vault maintenance
dashboard, *15-2*
system settings, *15-13*
Oracle Key Vault management console
about, *1-15*
overview of, *4-34*
timeout for multi-master cluster nodes, *15-20*
timeout for Web sessions for non-multi-master clusters, *15-11*
Oracle Key Vault Multi-Master Cluster, *A-1*
Oracle Key Vault status
viewing, *15-2*
Oracle Key Vault use cases, *1-4*
Oracle Real Application Clusters
RESTful services, *13-1*
support for data moves, *19-12*
wallets, *19-1*
Oracle Recovery Manager (RMAN) support for data moves, *19-12*
Oracle wallets
downloading
about, *18-19*
uploading
about, *18-19*

P

passphrases, *15-21*
changing in clusters environment, *15-22*
changing in non-clusters environment, *15-21*
recovering credentials, *15-21*
passphrases *(continued)*
recovering system, *15-21*
See also passwords
passwords, *15-21*
about changing, *7-11*, *7-12*
changing endpoint password, *B-3*
changing operating system passwords, *7-14*
changing password automatically, *7-13*
changing password manually, *7-13*
changing your own, *7-12*
controlling manual password reset operations, about, *7-16*
controlling manual password reset operations, configuration, *7-17*
See also passphrases
persistent master encryption key cache, *18-10*
about, *18-10*
architecture, *18-10*
caching master encryption keys in-memory, *18-11*
contents of, listing, *18-17*
environment variables, importance of setting, *18-11*
modes of operation
first mode, *18-12*
Oracle Key Vault first mode, *18-12*
Oracle Database deployments, *18-18*
PEXPIRE PKCS11 PERSISTENT CACHE ON DATABASE SHUTDOWN parameter, *18-16*
PKCS11_CACHE_TIMEOUT parameter, *18-14*
PKCS11_CONFIG_PARAM_REFRESH_INTERVAL parameter, *18-15*
PKCS11_PERSISTENT_CACHE_FIRST parameter, *18-15*
PKCS11_PERSISTENT_CACHE_REFRESH_WINDOW parameter, *18-16*
PKCS11_PERSISTENT_CACHE_TIMEOUT parameter, *18-14*
refresh window, *18-13*
storage location, *18-11*
PKCS11_CACHE_TIMEOUT parameter, *18-14*
PKCS11_CONFIG_PARAM_REFRESH_INTERV AL parameter, *18-15*
PKCS11_PERSISTENT_CACHE_FIRST parameter, *18-15*
PKCS11_PERSISTENT_CACHE_REFRESH_WI NDOW parameter, *18-16*
PKCS11_PERSISTENT_CACHE_TIMEOUT parameter, *18-14*
post-installation tasks, *4-10*
powering off Oracle Key Vault for non-multi-master clusters, *15-12*
powering off Oracle Key Vault nodes, *15-16*
primary servers
role in primary-standby configuration, *6-5*
primary-standby
restore operations, *14-15*

primary-standby configuration
about, 6-2
benefits, 6-4
best practices, 6-22
changing SNMP settings on standby server, 17-4
checking TDE wallet migration for logical standby, 19-8
configuring primary server, 6-5
configuring standby server, 6-7
difference from multi-master clusters, 6-4
disabling, 6-13
downtime, minimizing, 15-32
enabling primary-standby on primary, 6-9
migrating TDE wallets to Oracle Key Vault for standby, 19-8
persistent master encryption key cache
downtime, minimizing, 15-32
primary server
configuring, 6-5
enabling for primary-standby, 6-9
primary server role, 6-5
read-only restricted mode
downtime, minimizing, 15-32
read-only restricted mode disabled, 6-16
read-only restricted mode enabled, 6-16
read-only restricted mode impact, 6-15
read-only restricted mode state during network failure, 6-19
read-only restricted mode state during primary server failure, 6-19
read-only restricted mode state during standby server failure, 6-19
read-only restricted mode states, 6-17
read-only restricted mode, disabling, 6-21
read-only restricted mode, enabling, 6-20
read-only restricted mode, recovering from, 6-21
restoring primary-standby after, 6-12
reverse SSH tunnels, 12-8
standby server
configuring, 6-7
standby server role, 6-5
switching servers, 6-11
unpairing, 6-13
primary-standby environments
console certificates, 16-8
primary-standby server
moving to multi-master cluster, 3-13
privileges, 2-3
access control options, 2-4
access grants for virtual wallets, 2-4
RESTful services, 13-2
See also access control
provision command, 13-24

Q

query command, 13-84

R

re_enroll command, 13-27
re_enroll_all command, 13-28
read-only mode
about, 3-6
read-only nodes
about, 3-6
creating, 5-5
read-only restricted mode
about, 3-6
disabling, 6-21
enabling, 6-20
failover, planned shutdown in standby server, C-12, C-15
failover, planned shutdown of primary server during upgrade, C-11, C-14
failover, planned shutodwn on primary server during maintenance, C-12, C-14
failover, unplanned shutdown in primary server, C-13, C-15
failover, unplanned shutdown in standby server, C-13, C-16
notifications, 6-22
primary-standby configuration with read-only restricted mode enabled, 6-16
primary-standby configuration without read-only restricted mode enabled, 6-16
primary-standby configuration, impact on, 6-15
recovering primary-standby, 6-21
read-only restricted mode states
network failure in primary-standby configuration, 6-19
primary server failure, 6-19
primary-standby configuration, 6-17
standby server failure, 6-19
read-write mode
about, 3-6
read-write nodes
about, 3-5
read-write pair of nodes
creating, 5-3
read-write pairs of nodes
creating additional, 5-7
rebooting Oracle Key Vault for non-multi-master clusters, 15-12
rebooting Oracle Key Vault nodes, 15-16
recovery passphrase
about recovering, 15-21
changing in clusters environment, 15-22

recovery passphrase *(continued)*
changing in non-clusters environment, *15-21*
protecting the backup, *14-12*
recovering credentials, *15-21*
reg_cert command, *13-72*
reg_key command, *13-69*
reg_opaque command, *13-73*
reg_secret command, *13-73*
rekey operation, *1-5*
remote backup destination
about, *14-2*
remote backup destinations
changing settings, *14-6*
creating, *14-4*
deleting, *14-7*
remote monitoring
about, *17-2*
changing settings on standby server, *17-4*
changing user name and password, *17-4*
granting SNMP access to users, *17-3*
remotely monitoring using SNMP, *17-5*
removing user from a user group, *7-23*
renaming a user group, *7-23*
reporting, *1-10*
reports
about, *17-26*
endpoint reports, *17-27*
keys reports, *17-28*
notification report, *17-29*
system reports, *17-29*
user reports, *17-28*
wallets reports, *17-28*
restarting Oracle Key Vault for non-multi-master clusters, *15-12*
restarting Oracle Key Vault nodes, *15-16*
RESTful administrative commands
add_epg_member, *13-35*
add_wallet_access_ep, *13-43*
add_wallet_access_epg, *13-45*
check_object_status, *13-46*
check_unique_wallet, *13-47*
create_endpoint, *13-18*
create_endpoint_group, *13-36*, *13-37*
create_unique_endpoint, *13-19*
create_wallet, *13-49*
delete_endpoint_group, *13-38*
delete_unique_endpoint, *13-21*
delete_wallet, *13-50*
download, *13-22*
drop_epg_member, *13-39*
drop_wallet_access_ep, *13-51*
drop_wallet_access_epg, *13-52*
error reporting
about, *13-62*
RESTful administrative commands *(continued)*
error reporting *(continued)*
while running commands from script, *13-63*
error reporting at command line, *13-63*
get_default_wallet, *13-53*
get_enrollment_token, *13-23*
get_object_name, *13-54*
get_wallets, *13-55*
help information, *13-64*
modify_endpoint_desc, *13-30*
modify_endpoint_email, *13-29*
modify_endpoint_group_desc, *13-40*
modify_endpoint_group_name, *13-41*
modify_endpoint_name, *13-31*
modify_endpoint_platform, *13-32*
modify_endpoint_type, *13-33*
modify_wallet_access_ep, *13-56*
modify_wallet_access_epg, *13-58*
modify_wallet_desc, *13-59*
modify_wallet_name, *13-60*
mset_default_wallet, *13-61*
provision, *13-24*
RESTful APIs
reports, *17-29*
RESTful key management commands
about, *13-65*, *13-84*
activate, *13-82*
add_attr, *13-75*
add_custom_attr, *13-75*
add_member, *13-85*
all_attr, *13-76*
commands listed, *13-67*
create_key, *13-68*
del_attr, *13-77*
del_cust_attr, *13-78*
del_member, *13-86*
destroy, *13-82*
get_attr, *13-78*
get_cert, *13-70*
get_key, *13-70*
get_opaque, *13-71*
get_secret, *13-71*
list_attr, *13-79*
list_wallet, *13-86*
locate, *13-83*
mod_attr, *13-80*
mod_custom_attr, *13-81*
privileges required, *13-65*
reg_cert, *13-72*
reg_key, *13-69*
reg_opaque, *13-73*
reg_secret, *13-73*
revoke, *13-84*
usage, *13-65*

RESTful services
about, *1-16*, *13-1*
command syntax, *13-16*
administrative commands, *13-15*
configuration file
about, *13-5*
creating, *13-6*
creation guidelines, *13-6*
example of creating automatic endpoint enrollment, *13-10*
examples, *13-8*
executing multiple commands in script, *13-9*
executing single command, *13-8*
console certificates, *16-8*
disabling, *13-14*
disabling for non-multi-master clusters, *15-11*
downloading software utility, *13-4*
enabling for non-multi-master clusters, *15-11*
enabling network services, *13-4*
enabling RESTful services, *13-4*
multi-master clusters, enablement, *15-19*
multitenant environments, *13-1*
Oracle Real Application Clusters, *13-1*
privileges, *13-2*
system requirements, *13-3*
RESTful Services
commands
endpoint management, *13-34*
enabling, *13-3*
restoring data
about, *14-13*
best practices, *14-16*
multi-master clusters, *14-14*
primary-standby deployment, *14-15*
procedure, *14-13*
system state after, *14-15*
third-party certificates, *14-15*
revoke command, *13-84*
roles
about, *2-5*
root user
about, *2-8*
Roots of Trust (RoT), *1-15*

S

search bars, *4-35*
searches
how to perform searches in Oracle Key Vault management console, *4-34*
secret
get_secret command, *13-71*
reg_secret command, *13-73*
secure user management, *7-16*
security objects
activating, *8-12*
adding to virtual wallets, *8-4*
deactivating, *8-12*
deleting, *8-13*
details of, about, *8-13*
downloading to different types, *B-4*
modifying details of, *8-18*
multi-master clusters, effect on, *8-11*
removing from virtual wallets, *8-5*
revoking, *8-12*
searching for object items, *8-14*
state of, managing, *8-11*
viewing details of, *8-15*
virtual wallets, creating for, *8-2*
self-enrollment, for endpoints, *9-8*
separation of duties, *1-11*
how Oracle Key Vault manages, *2-6*
users, *7-1*
SNMP
about, *17-2*
changing settings on standby server, *17-4*
changing user name and password, *17-4*
example of simplified remote monitoring, *17-7*
granting access to user, *17-3*
Management Information Base (MIB) variables, *17-6*
remotely monitoring Oracle Key Vault, *17-5*
SNMP settings
multi-master clusters, *15-19*
split-brain scenarios, *6-2*
sqlnet.ora file
environment variables and, *10-10*
SSH key files
downloading from Key Vault to a wallet, *B-4*
SSH tunnels
creating between Oracle Key Vault and DBaas instance, *12-5*
deleting, *12-11*
disabling, *12-9*
multi-master clusters, *12-9*, *12-11*
not active, *12-11*
reverse SSH tunnel in multi-master cluster, *12-8*
reverse SSH tunnel in primary-standby configuration, *12-8*
viewing details, *12-9*
standby servers
role in primary-standby configuration, *6-5*
support user
about, *2-8*
swap space, extending, *4-25*

syslog
configuring for non-multi-master clusters, 15-10
syslog configuration
audit records, 17-23
destination
clearing, 17-14
setting, 17-13
syslog settings
multi-master cluster node, 15-16
multi-master clusters, 15-19
System Administrator role
about, 2-7
multi-master cluster effect on, 7-3
system diagnostics
See diagnostic reports
system recovery, 2-8, 15-21
system requirements, 4-2
system time
setting for non-multi-master clusters, 15-7
system users
multi-master cluster effect on, 7-4

T

TDE direct connect
See online master keys
TDE master encryption keys
centralized management, 1-5
TDE wallets
Oracle GoldenGate, 19-3
TDE-enabled databases, 18-2
about configuring Key Vault for, 18-2
configuring environment for, 18-3
integrating TDE with Key Vault, 18-4
limitations of TDE endpoint integration, 18-2
third-party certificates
restoring data, 14-15
time
setting for non-multi-master clusters, 15-7
time settings
multi-master clusters, 15-18
time, setting for node, 15-14
Transparent Data Encryption, 18-2
downtime, minimizing for TDE heartbeat, 15-32
endpoint management, 10-10
See also TDE-enabled databases
transportable tablespaces support for data moves, 19-12
troubleshooting
finding log files, C-6
upgrade errors, C-8
uploading Java keystores, C-6
troubleshooting (continued)
uploading keystores with same file name but different contents, C-7
uploading the same Oracle wallet multiple times, C-7
types of backups, 14-8

U

upgrades
error handling, C-8
upgrading
backing up
Oracle Key Vault server, 4-28
backing up upgraded Oracle Key Vault server, 4-28
endpoint software, 4-22
Oracle Key Vault pair
about, 4-18
procedure, 4-21
Oracle Key Vault standalone
about, 4-18
procedure, 4-19
removing DSA keys, 4-28
removing old kernels, 4-23
upgrading endpoint software
unenrolled endpoint, 9-27
upgrading multi-master cluster nodes
about, 4-29
checking upgrade success, 4-32
pre-upgrading, 4-31
pre-upgrading preparation, 4-30
rolling back pre-upgrade script, 4-33
upgrading each node, 4-31
upgrading standalone or primary-server
about, 4-16
upload command (okvutil), B-8
uploading
credential files, 18-26
JKS and JCEKS keystores, 18-23
wallet to Oracle RAC, 19-1
wallets, 18-20
use cases, 1-4
centralized storage, 1-4
key rotation, 1-5
online management of keys and secret data, 1-7
storage of credential files, 1-7
user accounts
multi-master clusters, effect on, 7-2
user groups, 7-19
adding a user, 7-22
changing description, 7-23
creating, 7-20
deleting, 7-24

user groups *(continued)*
granting access to virtual wallet, *7-22*
modifying virtual wallets from Keys & Wallets tab, *8-7*
multi-master clusters, effect on, *7-19*
removing access to virtual wallets from Keys & Wallets tab, *8-6*
removing access to virtual wallets from User's tab, *8-9*
removing user from, *7-23*
renaming, *7-23*
revoking access to virtual wallets, *8-10*
user-defined keys
about, *18-28*
activating, *18-30*
uploading to Oracle Key Vault, *18-29*
users, *7-19*
about changing password, *7-12*
about user accounts, *7-1*
administrative roles, about, *7-8*
administrative roles, granting or changing, *7-8*
administrative roles, revoking, *7-10*
administrator roles, *2-5*
changing operating system passwords, *7-14*
changing own password, *7-12*
changing password automatically, *7-13*
changing password manually, *7-13*
changing passwords, about, *7-11*
changing user email address, *7-18*
controlling manual password reset operations, about, *7-16*
controlling manual password reset operations, configuration, *7-17*
creating accounts, *7-4*
deleting accounts, *7-7*
disabling email notifications, *7-18*
endpoint administrators
about, *2-9*
granting access to virtual wallet, *7-9*
modifying virtual wallets from Keys & Wallets tab, *8-7*
multi-master cluster effect on, *7-4*
removing access to virtual wallets from Keys & Wallets tab, *8-6*
removing access to virtual wallets from User's tab, *8-9*
reports, *17-28*
root user
about, *2-8*
support user
about, *2-8*
view account details, *7-6*
See also user groups

V

viewing user account details, *7-6*
virtual wallets
about, *8-1*
access management from Keys and Wallets tab, *8-6*
adding endpoint access to, *9-15*
adding security objects to, *8-4*
creating, *8-2*
deleting, *8-5*
endpoint group access grant, *9-21*
granting access to from Keys & Wallets tab, *8-6*
granting access to from Users tab, *8-8*
granting user access to, *7-9*, *7-22*
granting user group access to from User's tab, *8-9*
modifying from Keys & Wallets tab, *8-7*
removing security objects from, *8-5*
removing user access to from Users tab, *8-9*
revoking endpoint access, *9-16*
revoking user group access from, *8-10*

W

wallets
add_member command, *13-85*
add_wallet_access_ep command, *13-43*
add_wallet_access_epg command, *13-45*
check_object_status command, *13-46*
checking TDE wallet migration for logical standby
about, *19-8*
create_unique_wallet command, *13-47*
create_wallet command, *13-49*
del_member command, *13-86*
delete_wallet command, *13-50*, *13-61*
downloading
guidance, *18-22*
procedure, *18-21*
downloading from Key Vault to a wallet, *B-4*
drop_wallet_access_ep command, *13-51*
endpoint group access grant, *9-21*
endpoints, associating, *9-13*
endpoints, viewing wallet items for, *9-16*
get_default_wallet command, *13-53*
get_wallets command, *13-55*
key rotation guidance, *18-22*
list_wallet command, *13-86*
migrating existing TDE wallet to Key Vault
about, *18-6*
procedure, *18-6*

wallets *(continued)*
migrating TDE to Key Vault for logical standby database
about, *19-8*
migrating to Oracle Data Guard, *19-6*
modify_wallet_access_ep command, *13-56*
modify_wallet_access_epg command, *13-52*, *13-58*
modify_wallet_desc command, *13-59*
modify_wallet_name command, *13-60*
Oracle GoldenGate use with, *19-2*
Oracle Real Application Clusters environment, *19-1*
overwriting danger of, *18-22*
reports, *17-28*
restoring database contents previously encrypted by TDE
about, *18-8*
wallets *(continued)*
reverse migrating in Oracle Data Guard
about, *19-7*
setting default for endpoint, *9-13*
sharing with multiple endpoints guidance, *18-22*
uploading
guidance, *18-22*
procedure, *18-20*
uploading contents to Key Vault server, *B-8*
uploading in Oracle Data Guard
about, *19-5*
procedure, *19-5*