Oracle® Fusion Middleware
Administrator's Guide for Oracle Privileged Account Manager
11g Release 2 (11.1.2)
E27152-08

May 2014
Documentation for administrators and end users that describes how to use Oracle Privileged Account Manager to administer, audit, and provide better security for privileged accounts and passwords in your organization.
## Contents

### Preface

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audience</td>
<td>xv</td>
</tr>
<tr>
<td>Documentation Accessibility</td>
<td>xv</td>
</tr>
<tr>
<td>Related Documents</td>
<td>xv</td>
</tr>
<tr>
<td>Conventions</td>
<td>xvi</td>
</tr>
</tbody>
</table>

### What's New In This Guide

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>New and Changed Features for 11g Release 2 (11.1.2)</td>
<td>xvii</td>
</tr>
<tr>
<td>Other Significant Changes in this Book</td>
<td>xix</td>
</tr>
</tbody>
</table>

### Part I  Introduction to Oracle Privileged Account Manager

#### 1  Introduction to Oracle Privileged Account Manager

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is Oracle Privileged Account Manager?</td>
<td>1-1</td>
</tr>
<tr>
<td>Why Use Oracle Privileged Account Manager?</td>
<td>1-2</td>
</tr>
<tr>
<td>Features</td>
<td>1-3</td>
</tr>
<tr>
<td>Functionality</td>
<td>1-5</td>
</tr>
<tr>
<td>Architecture and Topology</td>
<td>1-6</td>
</tr>
<tr>
<td>How Oracle Privileged Account Manager is Deployed in Oracle Fusion Middleware</td>
<td>1-9</td>
</tr>
<tr>
<td>Understanding the Relationship between Oracle Privileged Account Manager Entities</td>
<td>1-10</td>
</tr>
</tbody>
</table>

#### 2  Understanding Oracle Privileged Account Manager Security

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview</td>
<td>2-1</td>
</tr>
<tr>
<td>Understanding Oracle Privileged Account Manager Authentication</td>
<td>2-2</td>
</tr>
<tr>
<td>Authentication for the Oracle Privileged Account Manager Console</td>
<td>2-3</td>
</tr>
<tr>
<td>Authentication for the Oracle Privileged Account Manager Server</td>
<td>2-4</td>
</tr>
<tr>
<td>Understanding Oracle Privileged Account Manager Authorization</td>
<td>2-4</td>
</tr>
<tr>
<td>Administration Role Types</td>
<td>2-4</td>
</tr>
<tr>
<td>End Users</td>
<td>2-6</td>
</tr>
<tr>
<td>Securing Oracle Privileged Account Manager</td>
<td>2-6</td>
</tr>
<tr>
<td>Securing the Network Channel</td>
<td>2-6</td>
</tr>
<tr>
<td>Connecting to Target Systems</td>
<td>2-7</td>
</tr>
<tr>
<td>Securing the End User Interface</td>
<td>2-7</td>
</tr>
<tr>
<td>Securing Shared Accounts</td>
<td>2-8</td>
</tr>
</tbody>
</table>
What is a Shared Account? ................................................................. 2-8
Security Limitations .......................................................................... 2-8
How to Secure the Account ................................................................. 2-9
Enabling Password Resets ................................................................... 2-9
Avoiding Assignments through Multiple Paths ..................................... 2-9
Defining Richer Password Policies ...................................................... 2-10
Hardening the Back-End Oracle Privileged Account Manager Database ...................................................... 2-10
Understanding Session Management Security .................................. 2-12
Understanding Plug-In Security ......................................................... 2-12

Part II  Basic Administration

3  Getting Started with Managing Oracle Privileged Account Manager

Before You Begin ............................................................................... 3-1
Understanding ICF Connectors in Oracle Privileged Account Manager ........ 3-3
  About the ICF Connectors .................................................................. 3-3
  Locating the Oracle Privileged Account Manager Connector Bundles .......... 3-4
  Consuming ICF Connectors ................................................................. 3-4
Starting Oracle Privileged Account Manager ........................................ 3-5
  Starting WebLogic ............................................................................ 3-6
  Configuring an External Identity Store for Oracle Privileged Account Manager 3-7
  Preparing the Identity Store ............................................................... 3-9
    Using the idmConfigTool to Prepare the Directory ............................ 3-10
    Manually Preparing the Directory .................................................. 3-14
  Assigning the Application Configurator Role to a User ......................... 3-14
Administering Oracle Privileged Account Manager ............................... 3-15
  Working with Oracle Privileged Account Manager Self-Service ............ 3-16

4  Starting and Using the Oracle Privileged Account Manager Console

Before You Begin ............................................................................... 4-1
Invoking Oracle Privileged Account Manager’s Web-Based Console .......... 4-1
Navigating Oracle Privileged Account Manager’s Console ...................... 4-2
  Working with the Home Accordion ................................................ 4-3
  Working with the Administration Accordion .................................... 4-4
  Working with the Reports Accordion .............................................. 4-4
  Working with the Configuration Accordion ..................................... 4-5
  Working with the Search Portlet ..................................................... 4-5
  Working with a Search Results Table ............................................. 4-7

5  Configuring and Managing the Servers

Understanding the Servers .................................................................. 5-1
  Oracle Privileged Account Manager Server ..................................... 5-1
  Oracle Privileged Session Manager Server ..................................... 5-2
Managing an Oracle Privileged Account Manager Server ...................... 5-3
  Before You Begin ............................................................................ 5-3
  Configuring a Connection to the Oracle Privileged Account Manager Server 5-4
### 6 Working with Targets

**What Are Targets?** .......................................................... 6-1  
**Adding Targets to Oracle Privileged Account Manager** .......................................................... 6-2  
  - database Target Type Parameters .................................................. 6-3  
  - ldap Target Type Parameters ...................................................... 6-5  
  - lockbox Target Type Parameters .................................................. 6-7  
  - unix Target Type Parameters ...................................................... 6-7  
**Managing a Target's Service Account Password** .......................................................... 6-8  
**Opening a Target** .................................................................................. 6-9  
**Managing a Target's Service Account Password** .......................................................... 6-9  
**Removing Targets from Oracle Privileged Account Manager** .......................................................... 6-10

### 7 Working with Service Accounts

**Understanding Service Accounts** .......................................................... 7-1  
**Creating Service Accounts** .......................................................... 7-1  
**Managing Service Account Passwords** .......................................................... 7-2  
  - Showing Service Account Passwords .................................................. 7-3  
  - Viewing the Password History ...................................................... 7-3  
  - Resetting Service Account Passwords .................................................. 7-4  
  - Understanding Service Account Password Rollover .................................................. 7-4

### 8 Working with Privileged Accounts

**What is a Privileged Account?** .......................................................... 8-1  
**Managing System Accounts** .......................................................... 8-2  
**Managing Application Accounts** .......................................................... 8-3  
**Understanding Sharing Accounts** .......................................................... 8-4  
**Adding Privileged Accounts into Oracle Privileged Account Manager** .......................................................... 8-4  
  - Adding the Account ........................................................................ 8-5  
  - Adding Grantees ........................................................................ 8-7  
  - Adding CSF Mappings ........................................................................ 8-8  
**Searching for Privileged Accounts** .......................................................... 8-8  
**Opening Privileged Accounts** .......................................................... 8-9  
**Checking Out Privileged Accounts** .......................................................... 8-10  
  - Checking Out Passwords ........................................................................ 8-10  
  - Checking Out Privileged Account Sessions .................................................. 8-11  
**Checking In Privileged Accounts** .......................................................... 8-12  
**Viewing a Session Recording** .......................................................... 8-13  
**Managing Privileged Account Passwords** .......................................................... 8-16
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>Working with Policies</td>
<td></td>
</tr>
<tr>
<td></td>
<td>What Are Oracle Privileged Account Manager Policies?</td>
<td>9-1</td>
</tr>
<tr>
<td></td>
<td>Working with Password Policies</td>
<td>9-2</td>
</tr>
<tr>
<td></td>
<td>Searching for Password Policies</td>
<td>9-3</td>
</tr>
<tr>
<td></td>
<td>Viewing Password Policies</td>
<td>9-3</td>
</tr>
<tr>
<td></td>
<td>Modifying the Default Password Policy</td>
<td>9-4</td>
</tr>
<tr>
<td></td>
<td>Creating a Password Policy</td>
<td>9-6</td>
</tr>
<tr>
<td></td>
<td>Assigning Password Policies</td>
<td>9-7</td>
</tr>
<tr>
<td></td>
<td>Deleting Password Policies</td>
<td>9-8</td>
</tr>
<tr>
<td></td>
<td>Working with Usage Policies</td>
<td>9-9</td>
</tr>
<tr>
<td></td>
<td>Searching for Usage Policies</td>
<td>9-9</td>
</tr>
<tr>
<td></td>
<td>Viewing Usage Policies</td>
<td>9-9</td>
</tr>
<tr>
<td></td>
<td>Modifying the Default Usage Policy</td>
<td>9-10</td>
</tr>
<tr>
<td></td>
<td>Creating a Usage Policy</td>
<td>9-11</td>
</tr>
<tr>
<td></td>
<td>Assigning Usage Policies</td>
<td>9-12</td>
</tr>
<tr>
<td></td>
<td>Deleting Usage Policies</td>
<td>9-12</td>
</tr>
<tr>
<td>10</td>
<td>Working with Grantees</td>
<td></td>
</tr>
<tr>
<td></td>
<td>What Are Grantees?</td>
<td>10-1</td>
</tr>
<tr>
<td></td>
<td>Granting Accounts to Users</td>
<td>10-2</td>
</tr>
<tr>
<td></td>
<td>Granting Accounts to Groups</td>
<td>10-3</td>
</tr>
<tr>
<td></td>
<td>Searching for Grantees</td>
<td>10-3</td>
</tr>
<tr>
<td></td>
<td>Opening a Grantee</td>
<td>10-4</td>
</tr>
<tr>
<td></td>
<td>Removing Grantees from an Account</td>
<td>10-4</td>
</tr>
<tr>
<td>11</td>
<td>Working with Plug-Ins</td>
<td></td>
</tr>
<tr>
<td></td>
<td>What is a Plug-In?</td>
<td>11-1</td>
</tr>
<tr>
<td></td>
<td>Developing Plug-Ins for Oracle Privileged Account Manager</td>
<td>11-2</td>
</tr>
<tr>
<td></td>
<td>Overview</td>
<td>11-2</td>
</tr>
<tr>
<td></td>
<td>Supported Languages</td>
<td>11-3</td>
</tr>
<tr>
<td></td>
<td>Prerequisites</td>
<td>11-3</td>
</tr>
<tr>
<td></td>
<td>Oracle Privileged Account Manager Plug-In Benefits</td>
<td>11-4</td>
</tr>
<tr>
<td></td>
<td>Design Guidelines</td>
<td>11-4</td>
</tr>
<tr>
<td></td>
<td>Framework Description</td>
<td>11-4</td>
</tr>
<tr>
<td></td>
<td>Supported Operations and Timings</td>
<td>11-5</td>
</tr>
<tr>
<td></td>
<td>Pre-Operation Plug-Ins</td>
<td>11-5</td>
</tr>
<tr>
<td></td>
<td>Post-Operation Plug-Ins</td>
<td>11-5</td>
</tr>
<tr>
<td></td>
<td>Filtering Rules</td>
<td>11-6</td>
</tr>
<tr>
<td></td>
<td>Creating a Plug-In Configuration</td>
<td>11-7</td>
</tr>
<tr>
<td></td>
<td>Searching for Plug-In Configurations</td>
<td>11-10</td>
</tr>
<tr>
<td></td>
<td>Opening a Plug-In</td>
<td>11-10</td>
</tr>
</tbody>
</table>
# Deleting a Plug-In

Deleting a Plug-In ......................................................................................................................... 11-11

## 12 Working with Self-Service

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction to Using Self Service</td>
<td>12-1</td>
</tr>
<tr>
<td>Viewing Your Accounts</td>
<td>12-2</td>
</tr>
<tr>
<td>Searching for Accounts</td>
<td>12-2</td>
</tr>
<tr>
<td>Opening Accounts</td>
<td>12-2</td>
</tr>
<tr>
<td>Checking Accounts Out and In</td>
<td>12-3</td>
</tr>
<tr>
<td>Viewing Your Checked-Out Accounts</td>
<td>12-3</td>
</tr>
<tr>
<td>Checking Out Privileged Account Sessions</td>
<td>12-3</td>
</tr>
<tr>
<td>Showing a Password</td>
<td>12-3</td>
</tr>
</tbody>
</table>

## Part III Monitoring Oracle Privileged Account Manager

## 13 Working with Reports

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Viewing a Report</td>
<td>13-1</td>
</tr>
<tr>
<td>Working with Deployment Reports</td>
<td>13-2</td>
</tr>
<tr>
<td>Working with Usage Reports</td>
<td>13-3</td>
</tr>
<tr>
<td>Working with Failure Reports</td>
<td>13-3</td>
</tr>
<tr>
<td>Working with Checkout History Reports</td>
<td>13-4</td>
</tr>
</tbody>
</table>

## 14 Managing Oracle Privileged Account Manager Auditing and Logging

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Understanding Oracle Privileged Account Manager Auditing</td>
<td>14-1</td>
</tr>
<tr>
<td>Configuring Auditing in Oracle Privileged Account Manager</td>
<td>14-2</td>
</tr>
<tr>
<td>Configuring File-Based Auditing in Oracle Privileged Account Manager</td>
<td>14-3</td>
</tr>
<tr>
<td>Configuring Database-Based Auditing in Oracle Privileged Account Manager</td>
<td>14-4</td>
</tr>
<tr>
<td>Deploying Oracle Privileged Account Manager Audit Reports in BI Publisher</td>
<td>14-7</td>
</tr>
<tr>
<td>Setting the Audit Logging Levels</td>
<td>14-9</td>
</tr>
<tr>
<td>Understanding Oracle Privileged Account Manager Audit Reports</td>
<td>14-10</td>
</tr>
<tr>
<td>Auditing Application Consumption of Credentials from CSF</td>
<td>14-11</td>
</tr>
<tr>
<td>Understanding Oracle Privileged Account Manager Logging</td>
<td>14-12</td>
</tr>
<tr>
<td>Configuring Basic Logging</td>
<td>14-13</td>
</tr>
<tr>
<td>Example Logging Data</td>
<td>14-14</td>
</tr>
</tbody>
</table>

## Part IV Advanced Administration

## 15 Performing Advanced Configuration Tasks for Oracle Privileged Account Manager

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuring Oracle Privileged Account Manager to Communicate With Target Systems Over SSL</td>
<td>15-1</td>
</tr>
<tr>
<td>Securing Data On Disk</td>
<td>15-3</td>
</tr>
<tr>
<td>Enabling TDE Mode</td>
<td>15-3</td>
</tr>
<tr>
<td>Enable TDE in the Database</td>
<td>15-3</td>
</tr>
<tr>
<td>Enable Encryption in the Oracle Privileged Account Manager Schema</td>
<td>15-4</td>
</tr>
<tr>
<td>Enable TDE Mode in the Oracle Privileged Account Manager Server Configuration</td>
<td>15-4</td>
</tr>
</tbody>
</table>
16 Developing Plug-Ins for Oracle Privileged Account Manager

Overview ........................................................................................................................................ 16-1
Oracle Privileged Account Manager Framework Packages .................................................... 16-1
Special Considerations for Using Oracle Privileged Account Manager Plug-Ins ................. 16-2
Setting Up a Plug-In .................................................................................................................. 16-2
Understanding the Plug-In API ................................................................................................ 16-3
Communication between the Server and Plug-In .................................................................. 16-3
Plug-In Structure ....................................................................................................................... 16-4
Plug-In Interfaces and Classes ................................................................................................ 16-4
  PluginContext ......................................................................................................................... 16-4
  PluginResult .......................................................................................................................... 16-6
  PrePlugin .............................................................................................................................. 16-7
  PostPlugin ............................................................................................................................ 16-7
Debugging and Logging for Plug-Ins ....................................................................................... 16-8
Example Plug-ins ..................................................................................................................... 16-8
  Pre Plug-In Example ............................................................................................................ 16-8
  Post Plug-In Example ........................................................................................................... 16-11
Managing Plug-Ins ................................................................................................................... 16-15

17 Configuring Oracle Privileged Account Manager for Integrated Solutions

Integrating with Oracle Identity Manager ................................................................................ 17-1
Overview .................................................................................................................................... 17-1
Before You Begin ....................................................................................................................... 17-3
Installing Oracle Identity Manager ......................................................................................... 17-3
Configuring an Oracle Identity Manager Administrator ....................................................... 17-3
Configuring the External Identity Stores ............................................................................... 17-4
Creating LDAP Groups .......................................................................................................... 17-4
Adding the Oracle Privileged Account Manager CA Certificate ......................................... 17-4
Setting Up Oracle Identity Manager for the Integration ......................................................... 17-6
Installing and Configuring the Generic LDAP Connector ..................................................... 17-6
Creating an Application Instance ............................................................................................ 17-6
Part V Appendixes and Glossary

A Working with the Command Line Tool

Using the Command Line Tool .................................................................................................................. A-2
 Launching the Command Line Tool ................................................................................................................ A-2
   Launching the Command Line Tool from IAM_HOME .................................................................................. A-2
   Launching the Command Line Tool from Oracle Privileged Account Manager Client Archive ................ A-3

Issuing Commands ........................................................................................................................................... A-3

Working with the Server ............................................................................................................................... A-4
 getconfig Command ....................................................................................................................................... A-4
 getserverstatus Command ............................................................................................................................... A-5
 modifyconfig Command ................................................................................................................................. A-5

Working with Policies ..................................................................................................................................... A-6
 addpasswordpolicy Command ......................................................................................................................... A-6
 addusagepolicy Command ............................................................................................................................... A-8
 modifypasswordpolicy Command .................................................................................................................. A-9
 modifyusagepolicy Command ......................................................................................................................... A-9
 removepasswordpolicy Command .................................................................................................................. A-10
 removeusagepolicy Command ......................................................................................................................... A-10
 retrievepasswordpolicy Command .................................................................................................................. A-11
 retrieveusagepolicy Command ......................................................................................................................... A-11

Working with Targets ...................................................................................................................................... A-11
 addtarget Command ...................................................................................................................................... A-12
   ldap Target Type Parameters ......................................................................................................................... A-13
   database Target Type Parameters .................................................................................................................. A-14
   unix Target Type Parameters .......................................................................................................................... A-15
   lockbox Target Type Parameters ................................................................................................................... A-17
   displayalltargets Command .......................................................................................................................... A-17
   modifytarget Command ................................................................................................................................. A-17
   removetarget Command ................................................................................................................................. A-18

Running the opamSetup Script ......................................................................................................................... 17-6
 Creating the OPAM_TAGS UDF ......................................................................................................................... 17-8
 Tagging Catalog Entries with Oracle Privileged Account Manager Metadata ............................................. 17-8

Integrating with Oracle Access Management Access Manager .................................................................. 17-9
 Before You Begin ............................................................................................................................................. 17-9
 Enabling Single Sign-On ................................................................................................................................ 17-10
   Configure a New Resource for the Agent ......................................................................................................... 17-11
   Configure Oracle HTTP Server for the Access Manager Domain ............................................................... 17-11
   Add New Identity Providers ............................................................................................................................ 17-12
   Configure Access to Multiple Applications .................................................................................................... 17-13

Integrating with the Credential Store Framework ......................................................................................... 17-13
 Understanding Oracle Privileged Account Manager-Managed CSF Credentials ..................................... 17-13
 Provisioning ................................................................................................................................................... 17-14
 Lifecycle Management .................................................................................................................................... 17-15
 Application Consumption ............................................................................................................................... 17-15

Application Consumption ................................................................................................................................ A-17

Part V Appendixes and Glossary

A Working with the Command Line Tool

Using the Command Line Tool ......................................................................................................................... A-2
 Launching the Command Line Tool ................................................................................................................... A-2
   Launching the Command Line Tool from IAM_HOME .................................................................................. A-2
   Launching the Command Line Tool from Oracle Privileged Account Manager Client Archive ................ A-3

Issuing Commands ........................................................................................................................................... A-3

Working with the Server ..................................................................................................................................... A-4
 getconfig Command ....................................................................................................................................... A-4
 getserverstatus Command ............................................................................................................................... A-5
 modifyconfig Command ................................................................................................................................. A-5

Working with Policies ..................................................................................................................................... A-6
 addpasswordpolicy Command ......................................................................................................................... A-6
 addusagepolicy Command ............................................................................................................................... A-8
 modifypasswordpolicy Command .................................................................................................................. A-9
 modifyusagepolicy Command ......................................................................................................................... A-9
 removepasswordpolicy Command .................................................................................................................. A-10
 removeusagepolicy Command ......................................................................................................................... A-10
 retrievepasswordpolicy Command .................................................................................................................. A-11
 retrieveusagepolicy Command ......................................................................................................................... A-11

Working with Targets ...................................................................................................................................... A-11
 addtarget Command ...................................................................................................................................... A-12
   ldap Target Type Parameters ......................................................................................................................... A-13
   database Target Type Parameters .................................................................................................................. A-14
   unix Target Type Parameters .......................................................................................................................... A-15
   lockbox Target Type Parameters ................................................................................................................... A-17
   displayalltargets Command .......................................................................................................................... A-17
   modifytarget Command ................................................................................................................................. A-17
   removetarget Command ................................................................................................................................. A-18
resettargetpassword Command .............................................................. A-18
retrievetarget Command ........................................................................ A-19
searchtarget Command ........................................................................... A-20
showtargetpassword Command .............................................................. A-20
showtargetpasswordhistory Command .................................................. A-21
Working with Accounts........................................................................... A-21
addaccount Command ............................................................................ A-22
displayallaccounts Command ................................................................ A-23
checkin Command .................................................................................. A-23
checkout Command ................................................................................ A-24
displaycheckedoutaccounts Command ................................................. A-24
modifyaccount Command ...................................................................... A-25
removeaccount Command ...................................................................... A-25
resetpassword Command ....................................................................... A-26
retrieveaccount Command .................................................................... A-26
searchaccount Command ....................................................................... A-27
searchcheckouthistory Command ......................................................... A-28
showpassword Command ....................................................................... A-28
showpasswordhistory Command ............................................................ A-29
Working with Grantees........................................................................... A-29
displayallgroups Command .................................................................. A-30
displayallusers Command .................................................................... A-30
grantgroupaccess Command ................................................................ A-30
grantuseraccess Command ................................................................... A-31
removegroupaccess Command ................................................................ A-31
removeuseraccess Command ................................................................ A-32
retrievegrantees Command ................................................................... A-32
retrievegroup Command ........................................................................ A-33
retrieveuser Command .......................................................................... A-33
searchgroup Command ......................................................................... A-33
searchuser Command ............................................................................ A-34
Working with Plug-Ins............................................................................ A-34
addplugin Command ............................................................................. A-35
addplugincustomattr Command ............................................................. A-36
removeplugincustomattr Command ....................................................... A-37
retrieveplugin Command ...................................................................... A-37
searchplugin Command ......................................................................... A-38
modifyplugin Command ......................................................................... A-38
removeplugin Command ....................................................................... A-39
Exporting and Importing Data................................................................. A-40
export Command ................................................................................... A-40
filedecryption Command ...................................................................... A-42
import Command .................................................................................... A-43

B Working with Oracle Privileged Account Manager's RESTful Interface

Overview.................................................................................................. B-1
Server State Resource ............................................................................ B-2
Get Server State ..................................................................................................................... B-2

**Configuration Resource** .................................................................................................... B-3
Global Configuration Resource .......................................................................................... B-3
Get Configuration Resource ............................................................................................ B-3
Update Configuration Resource ....................................................................................... B-4
Oracle Privileged Session Manager Configuration Resource ....................................... B-4
Get Configuration Resource ............................................................................................ B-5
Update Configuration Resource ....................................................................................... B-6

**Policy Resource** .................................................................................................................. B-7
Search for Policies ................................................................................................................ B-7
Get Default Policies .......................................................................................................... B-8
Password Policy Resource ................................................................................................ B-9
Retrieve a Password Policy ............................................................................................. B-9
Update a Password Policy ................................................................................................. B-11
Create a Password Policy ................................................................................................ B-12
Get Accounts for Password Policy .................................................................................. B-13
Delete a Password Policy .................................................................................................. B-14

Usage Policy Resource ...................................................................................................... B-14
Retrieve a Usage Policy .................................................................................................... B-14
Update a Usage Policy ...................................................................................................... B-17
Create a Usage Policy ....................................................................................................... B-18
Get Grants for Usage Policy ............................................................................................. B-20
Delete a Usage Policy ........................................................................................................ B-21

**Target Resource** .................................................................................................................. B-21
Get Target Attributes ......................................................................................................... B-21
Add a Target ......................................................................................................................... B-25
Verify a Target ..................................................................................................................... B-27
Retrieve a Target ................................................................................................................ B-29
Update a Target .................................................................................................................... B-30
Remove a Target .................................................................................................................. B-31
Search for Targets ................................................................................................................. B-32
Get Available Accounts ..................................................................................................... B-33
Retrieve Accounts Registered on a Target ......................................................................... B-34
Get Target Types .................................................................................................................. B-35
Reset Password .................................................................................................................. B-35
Show Service Account Password ...................................................................................... B-36
Show Service Account Password (Deprecated) ................................................................. B-37
Show Service Account Password History ......................................................................... B-37

**Account Resource** ............................................................................................................... B-39
Add an Account to a Target ................................................................................................. B-39
Get Applicable Usage Policy for the Account ................................................................. B-40
Grant a User/Role Access to an Account ............................................................................... B-41
Add or Remove a CSF Map-Key for an Account ................................................................. B-42
Search Accounts ................................................................................................................ B-43
Search Assigned Accounts ................................................................................................. B-44
Retrieve an Account ............................................................................................................. B-45
Retrieve Grantees on an Account ......................................................................................... B-46
Troubleshooting Oracle Privileged Account Manager

C Troubleshooting Oracle Privileged Account Manager

Introduction to Troubleshooting Oracle Privileged Account Manager C-1
Getting Started with Troubleshooting and Logging Basics for Oracle Privileged Account Manager C-2
Increasing the Log Level C-2
Examining Exceptions in the Logs C-2
Resolving Common Problems and Solutions C-3
Console Cannot Connect to Oracle Privileged Account Manager Server ........................................C-3
Console Changes Are Not Reflected in Other, Open Pages ...............................................................C-4
Cannot Access Targets or Accounts ..................................................................................................C-4
Cannot Add Database Targets ........................................................................................................C-5
  Cannot Connect to Oracle Database with sysdba Role .........................................................C-5
  Cannot Find Special Options for Adding a Database Target ................................................C-5
Cannot Add an Active Directory LDAP Target ................................................................................C-6
Grantee Cannot Perform a Checkout .................................................................................................C-6
Cannot View Users or Roles from the Configured Remote Identity Store ........................................C-7
Group Membership Changes Are Not Immediately Reflected in Oracle Privileged Account Manager ............................................................C-7
Cannot Use Larger Key Sizes for Export/Import .............................................................................C-8
Oracle Privileged Account Manager End Users Gain Privileges They Were Not Explicitly Granted ........................................................................................................C-8
Cannot Access MSSQL Server Targets and Accounts .....................................................................C-9
Troubleshooting Issues with Using Oracle Database TDE ............................................................C-9
  TDE Wallet Errors ....................................................................................................................C-9
  The TDE Wallet is Open, but Columns Are Not Encrypted .....................................................C-10
Cannot Open Session Recordings .....................................................................................................C-10
  Cannot Access Recordings In Internet Explorer Browser ......................................................C-10
  Cannot Access Recordings in Any Browser .............................................................................C-11
Session Checkout Does Not Work, Even After Granting the Account ..............................................C-12
OPAM Console Login Does Not Work in Internet Explorer 11 Browser .....................................C-12
  Using My Oracle Support for Additional Troubleshooting Information ....................................C-12

Glossary

Index
Preface

Welcome to Administrator’s Guide for Oracle Privileged Account Manager. This guide describes how to use and administer Oracle Privileged Account Manager in an enterprise infrastructure.

Audience

The Administrator’s Guide for Oracle Privileged Account Manager is intended for Oracle Privileged Account Manager administrators who can configure connections to target systems and client applications, access passwords for target systems, and who can create roles and assign users to those roles.

Administrators must be familiar with either the UNIX operating system or the Microsoft Windows operating system to understand the command-line syntax and examples in this document. You also must be familiar with the Lightweight Directory Access Protocol (LDAP).

This book is also intended for Oracle Privileged Account Manager end-users who do not have administrative privileges, but who are authorized to check privileged accounts in and out.

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 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, refer to the following documents in the Oracle Identity Manager 11g Release 2 (11.1.2) documentation set:

- Oracle Database Advanced Security Administrator’s Guide
- Oracle Database Vault Administrator’s Guide
Conventions

The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong></td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td>italic</td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td>monospace</td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>
What's New In This Guide

This section summarizes the new features and significant product changes for Oracle Privileged Account Manager in the Oracle Fusion Middleware 11g Release 2 (11.1.2) release.

The topics in this chapter include

- New and Changed Features for 11g Release 2 (11.1.2)
- Other Significant Changes in this Book

Follow the pointers in this guide to get more information about the features and how to use them.

New and Changed Features for 11g Release 2 (11.1.2)

Oracle Privileged Account Manager 11g Release 2 (11.1.2) includes the following new and changed administrative, development, and security features:

- Updated Example 16-3, "Email Notification Post Plug-In" by changing
  result.setPlgErrorMsg(e.getMessage()); to
  result.setErrorMsg(e.getMessage());
- Updated all of the examples in Section B.10, "Plug-In Resource," as follows:
  - Changed the pluginFlexSecFields parameters to pluginCustomAttrs
  - Changed the pluginFlexSecField parameters to pluginCustomAttr
- Added a plug-in framework that enables you to extend and customize Oracle Privileged Account Manager functionality to better suit your specific requirements. This framework enables you to
  - Validate and manipulate data before Oracle Privileged Account Manager performs operations
  - Perform specific actions after Oracle Privileged Account Manager completes its operations
  - Register and manage plug-ins through the Oracle Privileged Account Manager Console, command line, or RESTful interface
  - Integrate Oracle Privileged Account Manager with third-party systems such as wallets, ticket management systems, and audit systems

In addition, a new Plug-in Configuration page and several new plug-in related options have been added to the Oracle Privileged Account Manager Console, command line tool, and RESTful interface. For more information about plug-ins and using the new interface features to configure plug-ins, refer to the following:
Added the Oracle Privileged Session Manager to manage the privileged sessions to the target system. By creating a single access point to the target resources, Oracle Privileged Session Manager (Session Manager) helps administrators easily control and monitor all of the activities within the privileged session.

In addition, a new Session Management page and several new session management-related updates have been made to the Console, command line tool, and RESTful interface. For more information about session management and configuring sessions, refer to the following:

- Chapter 1, "Introduction to Oracle Privileged Account Manager" for information about how the Session Manager functionality works within Oracle Privileged Account Manager.
- Section 2.5, "Understanding Session Management Security" for information about Session Manager security.
- Section 5.3, "Managing the Oracle Privileged Session Manager Server" for information about configuring the Session Manager server.
- Chapter 8, "Working with Privileged Accounts" for information about administering managed sessions from the Console and about privileged sessions.
- Section 12.7, "Checking Out Privileged Account Sessions" for information about how to check out sessions and passwords.
- Section 13.5, "Working with Checkout History Reports" for information about working with Session History reports.
- Appendix B, "Working with Oracle Privileged Account Manager's RESTful Interface" for information about administering managed sessions by using Oracle Privileged Account Manager's RESTful interface.
Added a new `opsmconfig` configuration object that represents the configuration information for Session Manager servers.

Added a new My Checkouts page where users can access a list of the accounts they currently have checked out and a Checkout History page where administrators can access information about account checkouts.

- For information about the My Checkouts page, refer to Section 4.3, "Navigating Oracle Privileged Account Manager's Console" and Section 8.5, "Checking Out Privileged Accounts."
- For information about the Checkout History page, refer to Section 13.5, "Working with Checkout History Reports."
- Added new Checkout History Report that enables administrators to view information about any account checkouts performed over a specified period of time. Refer to Section 13.5, "Working with Checkout History Reports" for information.

Various other changes have been made to the Console, command line, and RESTful interfaces. Information about these new or updated interface changes is provided throughout this guide.

**Other Significant Changes in this Book**

For 11g Release 2 (11.1.2), this guide has been reorganized and updated as follows:

- Added and updated various parameter labels, procedure descriptions, and screenshots throughout book based on changes to the user interface, command line tool commands, and RESTful APIs.

- Reorganized Chapter 5, "Configuring and Managing Oracle Privileged Account Manager," into smaller, separate chapters. Refer to the Contents for more information.

- Added the following new chapters and appendixes:
  - Chapter 4, "Starting and Using the Oracle Privileged Account Manager Console," describes how to invoke and work with Oracle Privileged Account Manager’s web-based graphical user interface, or Console.
  - Section 7 describes how to configure and manage OPAM Service Accounts.
  - Chapter 11, "Working with Plug-Ins," describes how to configure and deploy an Oracle Privileged Account Manager plug-in.
  - Chapter 16, "Developing Plug-Ins for Oracle Privileged Account Manager," describes how to write your own, custom plug-ins.

- Reorganized Chapter A, "Working with the Command Line Tool," by combining related commands into sections. For example, all of the server related commands are now located in Section A.2, "Working with the Server." Refer to the Contents for more information.
This part contains introductory and conceptual information about Oracle Privileged Account Manager, and it includes the following chapters:

- Introduction to Oracle Privileged Account Manager
- Understanding Oracle Privileged Account Manager Security
Introduction to Oracle Privileged Account Manager

This chapter introduces you to Oracle Privileged Account Manager by describing key concepts, features, and functionality.

This chapter includes the following sections:

- Section 1.1, "What is Oracle Privileged Account Manager?"
- Section 1.2, "Why Use Oracle Privileged Account Manager?"
- Section 1.3, "How Oracle Privileged Account Manager is Deployed in Oracle Fusion Middleware"
- Section 1.4, "Understanding the Relationship between Oracle Privileged Account Manager Entities"

1.1 What is Oracle Privileged Account Manager?

Oracle Privileged Account Manager manages privileged accounts that are not being managed by any other Oracle Identity Management components.

Accounts are considered "privileged," if they can access sensitive data, can grant access to sensitive data, or can both access and grant access to that data. Privileged accounts are your company's most powerful accounts and they are frequently shared.

Accounts become candidates for management via Oracle Privileged Account Manager if they are associated with elevated privileges, are used by multiple end-users on a task-by-task basis, and must be controlled and audited.

For example, these accounts require security and may fall under compliance regulations:

- UNIX root, Windows administrator, and Oracle Database SYSDBA system accounts
- Application accounts, such as the database user accounts used by an application server when it connects to a Human Resources application
- Traditional shared and elevated privilege user accounts, such as system administrators and database administrators

Administrators determine which accounts are privileged within a particular deployment, and they must configure Oracle Privileged Account Manager to manage those accounts.

While Oracle Privileged Account Manager most commonly manages shared and elevated privileged accounts, administrators can also use it to manage passwords for
any type of account. For example, if an employee is on extended leave and you have a business reason for allowing another employee to access the system using that person's email account, Oracle Privileged Account Manager can manage that privilege.

1.2 Why Use Oracle Privileged Account Manager?

Oracle Privileged Account Manager enables you to administer and provide better security for privileged accounts and passwords that are traditionally difficult to manage for several reasons.

First, privileged accounts generally have more access rights than a regular user's account. Because these accounts are not typically associated with one specific employee, they are often difficult to audit with existing tools and processes. Consequently, when employees leave the company, they might retain privileged account passwords that are still in use, which is a very serious compliance and security issue.

Also, changing privileged account passwords on a regular basis is difficult. If many people depend on the account, changing the password and notifying everyone requires a coordinated effort.

Finally, you typically do not want to store passwords in a central or well-known location, such as an external repository (like LDAP) or in application configuration files, because you cannot control access to those passwords.

Oracle Privileged Account Manager delivers a complete solution for securely managing privileged accounts and passwords because it provides

- Centralized password management for privileged and shared accounts, including UNIX and Linux root accounts, Oracle Database SYSDBA, application accounts, and LDAP admin accounts

- Interactive, policy-based account and session checkout and check-in

Oracle Privileged Account Manager requires all authorized users to check out an account before using it, and then to check that account back in when they are finished with it. Oracle Privileged Account Manager audits account check outs and check ins by tracking the real identity (the person’s name) of every shared administrator user at any given moment in time. By using this information, Oracle Privileged Account Manager can provide a complete audit trail that shows who accessed what, when, and where.

In addition, Oracle Privileged Session Manager (Session Manager) enables administrators to monitor and control which activities users can perform during a session. Users are never allowed direct access to resources or to privileged credentials.

- Automatic password changes using the Identity Connector Framework (ICF)

Oracle Privileged Account Manager modifies passwords when they are checked out and checked in (when configured to do so). Consequently, when a user checks out a password and then subsequently checks it back in, that user can no longer use the previously checked out password.

In addition, Oracle Privileged Account Manager can change application privileged account passwords at specified intervals, such as every 90 days, with no changes to those applications and Oracle Privileged Account Manager synchronizes those passwords on the target systems. For example, Oracle Privileged Account Manager can update service and scheduled task credentials.
Oracle Privileged Account Manager's key features include:

- Multiple access points, including
  - Oracle Privileged Account Manager's web-based user interface (called the Console)

    Two interfaces are associated with the Console:
    * **Administrator**: Oracle Privileged Account Manager administrators use this interface to create and manage policies, targets, accounts, grants, and reports.
    * **Self-Service**: Oracle Privileged Account Manager end users use this interface to search for, view, check out, and check in accounts.

    Refer to Chapter 4, "Starting and Using the Oracle Privileged Account Manager Console" for more information.

  - Oracle Privileged Account Manager's command line tool (CLI)

    You can use the CLI to perform many of the same tasks you perform from the Console. For example, you can use the CLI to check out and check in accounts or to create and manage policies, targets, accounts, and grants.

    Refer to Appendix A, "Working with the Command Line Tool" for more information.

  - RESTful APIs

    Oracle Privileged Account Manager uses RESTful APIs to expose internal functionality to applications and scripts. These APIs also provide the integration point to be leveraged by third parties that want to integrate with Oracle Privileged Account Manager functionality.

    **Note:** These APIs are considered to be RESTful because they conform to Representative State Transfer (REST) standards.

    Refer to Appendix B, "Working with Oracle Privileged Account Manager's RESTful Interface" for more information.

- Integration with Oracle technologies, including
  - **Oracle Platform Security Services (OPSS) Policy Store** for authorization
  - **Oracle Platform Security Services (OPSS) Trust Service** to authenticate and propagate identities from the Oracle Privileged Account Manager user interface to the Oracle Privileged Account Manager server
Why Use Oracle Privileged Account Manager?

- **Identity Connector Framework** (ICF) to connect to target systems and to discover, update, or discover and update the passwords for privileged accounts on those systems.

  In addition, because ICF is an open standard, you can write your own connectors against other types of targets for which Oracle has not yet created an ICF connector.

  For more information about ICF and about developing your own connector, refer to "Understanding the Identity Connector Framework" and "Developing Identity Connectors Using Java" or "Developing Identity Connectors Using .Net" in the *Oracle Fusion Middleware Developer’s Guide for Oracle Identity Manager*.

- Ability to manage and audit privileged sessions to the target system
  - Session Manager creates a single access point to target resources, which enables administrators to easily control and monitor all the activities within the privileged session.
  - Session Manager also maintains historical records (transcripts) to support forensic analysis and audit data.

- Support for multiple target types; including
  - UNIX and Linux operating systems
  - Oracle, MSSQL, MySQL and Sybase databases
  - LDAP v3-compliant directories

- Advanced reporting capabilities
  - Oracle Privileged Account Manager's out-of-the box audit reports are integrated with Oracle Business Intelligence Publisher 11g (BI Publisher) so you know who is using your privileged accounts. BI Publisher also enables you to create and manage formatted reports from different data sources.
  - The Oracle Fusion Middleware Audit Framework logs audit events in a centralized database. Oracle Privileged Account Manager uses these events to generate audit reports.
  - Events related to privileged account access roll up into Oracle Identity Manager and Oracle Identity Analytics for audit and attestation.

- Policy-driven access to privileged accounts

  In Oracle Privileged Account Manager, there are two types of policies for granting access to privileged accounts:

  - **Password Policy**: This policy type captures the password construction rules enforced by a specific target on an associated privileged account. For example, you can specify the minimum and maximum number of numeric characters for a password for an account. In addition, you use a password policy to create a password value that Oracle Privileged Account Manager uses to reset a password for a privileged account.

  - **Usage Policy**: This policy type defines when and how often a user or group can access a privileged account.

---

**Note**: If you do not specify a time interval by using a Usage Policy, the user or group can access the privileged account at any time (24x7).
Ability to manage attended and unattended accounts

- An attended account is an account assigned to a particular group or user.
- An unattended account is an account that is never used by an end user.

For example, Oracle Privileged Account Manager uses an unattended account, called the OPAM service account, to connect to and manage target systems. This account performs all Oracle Privileged Account Manager-related operations (such as discovering accounts, resetting passwords, and so forth) on the target system, which is why the OPAM service account (service account) must have some special privileges and properties.

Oracle Privileged Account Manager can also manage other kinds of unmanaged accounts, such as an application account or a service account with CSF mappings that enable applications to pick up a password at run-time by using CSF.

**Note:** You must never use the same account as a service account and a privileged account to be managed by Oracle Privileged Account Manager.

For more information about working with service accounts in Oracle Privileged Account Manager, refer to Section 7, "Working with Service Accounts."

### 1.2.2 Functionality

In addition to the functionality described in Section 1.2, "Why Use Oracle Privileged Account Manager?," Oracle Privileged Account Manager

- Associates privileged accounts with targets
- Grants users and roles access to privileged accounts, and removes that access
- Provides an extensible plug-in framework that enables you to use Oracle or third-party plug-ins to perform operations such as custom notifications, extended usage policies, and custom logic to synchronize passwords with external repositories
- Provides role-based access to accounts maintained in the Oracle Privileged Account Manager accounts request system
- Provides password check out and check in, as well as session checkout to control access to accounts
- Provides “over-the-shoulder” session management by enabling administrators to
  - Control session initiation
  - Control sessions through policy-based and administrator-initiated session termination and lockout
  - Monitor and audit sessions
- Eliminates the potential of having unmanaged privileged accounts when your unattended applications use client-certificate authentication

Client-certificate authentication is using an SSL certificate to perform authentication (in lieu of a password) against an Oracle Privileged Account Manager server.

- Resets passwords to a random value on check in and check out by default
You can configure Oracle Privileged Account Manager to automatically check in privileged accounts after a specified time to protect against users who check out that privileged account and do not bother to explicitly check in the account.

You can also constrain how long users can check out a privileged account.

- Manages password resets on supported targets
- Makes authorization decisions to determine
  - Which targets, privileged accounts, and policies are exposed to an end user or administrator
  - Which operations (such as add, modify, check-in, and checkout) end users and administrators can perform
- Associates policies with privileged accounts
- Performs and supports Create, Read, Update, Delete, and Search (CRUDS) operations on targets, privileged accounts, and policies

This core functionality is exposed through Oracle Privileged Account Manager’s RESTful APIs. Check-ins, checkouts, and so forth are also supported through the RESTful interface.

- Uses Oracle’s common auditing, logging, and reporting to monitor and report access

With Oracle Privileged Account Manager, you can use the auditing, logging, and reporting capabilities of Oracle Fusion Middleware Control and Oracle BI Publisher to monitor and report access that users and groups have to privileged accounts.

- Offers multiple high availability capabilities

### 1.2.3 Architecture and Topology

The following diagram illustrates Oracle Privileged Account Manager’s architecture and topology:
Why Use Oracle Privileged Account Manager?

Introduction to Oracle Privileged Account Manager

Figure 1–1 Oracle Privileged Account Manager Architecture and Topology

Figure illustrating OPAM’s architecture and topology.

As you examine this figure, it is important to note the following points:

- All of Oracle Privileged Account Manager's core logic resides on the Oracle Privileged Account Manager server. This functionality is exposed through a Representational State Transfer (REST or RESTful) service, where the data is encoded as JavaScript Object Notation (JSON).

Note: Oracle Privileged Account Manager provides a web-based user interface (known as the Console) and an Oracle Privileged Account Manager command line tool (CLI). Both interfaces are essentially clients of the Oracle Privileged Account Manager server. However, third parties can write their own clients, such as custom applications, by leveraging the open RESTful service. For more information, refer to Appendix B, "Working with Oracle Privileged Account Manager’s RESTful Interface."

- Session Manager is an Oracle Privileged Account Manager subcomponent that empowers Oracle Privileged Account Manager’s session management capabilities. Session Manager is a J2EE application that interacts with the Oracle Privileged
Account Manager Server through the Oracle Privileged Account Manager RESTful interfaces and shares the same database that is used by the Oracle Privileged Account Manager Server. In addition, the Session Manager listens and responds to SSH traffic to establish privileged sessions against SSH-capable Oracle Privileged Account Manager targets.


For more information about Oracle Privileged Account Manager authentication, refer to Section 2.2, "Understanding Oracle Privileged Account Manager Authentication."

- All communication with, and between, Oracle Privileged Account Manager-related components (including Oracle Privileged Account Manager’s Console, command-line interface, and server) occurs over SSL. In addition, Oracle Privileged Account Manager’s RESTful interfaces are exposed over SSL.

- Oracle Privileged Account Manager relies on and transparently uses the identity store, Policy Store, and credential store configured for the WebLogic domain in which Oracle Privileged Account Manager is deployed. (Because the Policy Store and credential store are implicitly part of the WebLogic domain, they are not depicted in this diagram.)

The identity store is the centralized repository for Oracle Privileged Account Manager users and groups.

Refer to Section 1.3, "How Oracle Privileged Account Manager is Deployed in Oracle Fusion Middleware" for more information.

- The Oracle Privileged Account Manager Console leverages, and is rendered by, Oracle Application Development Framework (ADF).

For more information about ADF, refer to the following website:

- Oracle Privileged Account Manager connects to targets by using Identity Connector Framework (ICF) connectors. As shown in Figure 1–1, Oracle Privileged Account Manager uses the following connectors, which are constructed by using the ICF:

  - **Generic Database User Management connector**: Connects to Oracle, MSSQL, Sybase, MySQL databases.
  - **Generic Unix connector**: Connects to any UNIX system.
  - **Generic LDAP connector**: Connects to LDAP targets (such as Oracle Internet Directory, Oracle Universal Directory, and Active Directory).
  - **Custom connector**: Connects to a target that does not have a predefined connector associated with it.

For additional information, refer to "Understanding the Identity Connector Framework" in the Oracle Fusion Middleware Developer’s Guide for Oracle Identity Manager.
1.3 How Oracle Privileged Account Manager is Deployed in Oracle Fusion Middleware

The following figure illustrates how Oracle Privileged Account Manager is deployed within Oracle Fusion Middleware.

**Figure 1–2  Oracle Privileged Account Manager Deployed Within Oracle Fusion Middleware**

As you examine this figure, note the following points:

- All components are deployed within a single WebLogic domain.
- Oracle Privileged Account Manager stores its application data in the Oracle Privileged Account Manager database. In addition, the Oracle Privileged Account Manager schema is created in this database via the Oracle Repository Creation Utility.
- Oracle Privileged Session Manager relies on the Oracle Privileged Account Manager Database for persistence and communicates with Oracle Privileged Account Manager through its RESTful interfaces.
- Oracle Privileged Account Manager’s web-based user interface (the Console) is deployed in the Oracle WebLogic Server Managed Server, along with the Oracle Privileged Account Manager Server and the Session Manager. The Console communicates with the Oracle Privileged Account Manager Server. This server is created as a server that is managed by the Oracle WebLogic Server Managed Server (or Managed Server).
- The OPSS identity store and the OPSS security store (which includes the Policy Store and credential store) are WebLogic domain-wide constructs, so there is one of each per domain. (Because the OPSS security store is implicitly part of the WebLogic domain, it is not depicted in this diagram.)

Oracle Privileged Account Manager simply works with what is configured for that domain. You are not required to use an Oracle Privileged Account.
Manager-specific configuration to use these constructs and services. In addition, Oracle Privileged Account Manager abstracts out the use of these constructs and services so that you do not have to understand what goes on “under the covers” in great detail.

- The OPSS identity store can point to the LDAP embedded in WebLogic (out of the box) or to an external LDAP server.

  Refer to “Configuring the Identity Store Service” in the Oracle Fusion Middleware Application Security Guide for configuration instructions.

- For information about managing the Policy Store and the credential store, refer to “Managing the Policy Store” and “Managing the Credential Store” in the Oracle Fusion Middleware Application Security Guide.

### 1.4 Understanding the Relationship between Oracle Privileged Account Manager Entities

Before you start working with the different Oracle Privileged Account Manager entities, you should understand how those entities relate to each other. Figure 1–3 illustrates this relationship.

**Figure 1–3  Oracle Privileged Account Manager Entity Relationships**

An Oracle Privileged Account Manager Password Policy can apply on both a target or a privileged account. When applied on a privileged account, that account's password construction (its complexity) and lifecycle (how often it changes) is governed by the effective Oracle Privileged Account Manager Password Policy. Similarly, when applied on a target, the target's service account is governed by the Oracle Privileged Account Manager Password Policy.

Targets are software systems that contain one or more privileged accounts.

A Usage Policy applies on a grant and it controls when and how grantees can use a privileged account. For example, you can configure a Usage Policy to control when a user's access to an account will expire.

Users and groups (roles) are maintained in the Oracle Privileged Account Manager identity store. These users and groups can only access a privileged account through a grant. If a user or group member tries to access a privileged account, and Oracle Privileged Account Manager finds a grant, then the grantee is allowed to access the account based on that grant and its associated Usage Policy.
This chapter describes how Oracle Privileged Account Manager authenticates and authorizes different types of users by using the authentication and authorization framework provided in the Oracle Privileged Account Manager server. In addition, this chapter explains various methods that you can use to further secure Oracle Privileged Account Manager in your deployment environment.

This chapter includes the following sections:

- **Section 2.1, "Overview"**
- **Section 2.2, "Understanding Oracle Privileged Account Manager Authentication"**
- **Section 2.3, "Understanding Oracle Privileged Account Manager Authorization"**
- **Section 2.4, "Securing Oracle Privileged Account Manager"**
- **Section 2.5, "Understanding Session Management Security"**
- **Section 2.6, "Understanding Plug-In Security"**

### 2.1 Overview

The authentication and authorization framework provided in the Oracle Privileged Account Manager server provides the following features and functionality:

- Supports OPSS-Trust tokens and HTTP-Basic Authentication
  
  You can also configure the Oracle Privileged Account Manager Console to work alongside Oracle Access Management. (Refer to **Section 17.2, "Integrating with Oracle Access Management Access Manager"** for more information.)

- Leverages the Java Authentication & Authorization Service (JAAS) for authentication

  **Note:** Oracle Privileged Account Manager authentication relies on JAAS support in the J2EE container on which its deployed. Refer to "WebLogic Security Service Architecture" in *Oracle Fusion Middleware Understanding Security for Oracle WebLogic Server* for more information.

- Defines different Oracle Privileged Account Manager-specific Admin Roles and their Oracle Privileged Account Manager-specific responsibilities

- Enforces authorization decisions that determine
Understanding Oracle Privileged Account Manager Authentication

- Which targets and privileged accounts are exposed to an administrator or to an end-user
- Which operations (such as add, modify, check-in, and checkout) an end-user or an administrator can perform on targets, privileged accounts, and policies
- Supports Usage Policies and Password Policies for privileged accounts

2.2 Understanding Oracle Privileged Account Manager Authentication

Oracle Privileged Account Manager uses Security Assertions Markup Language-based (SAML-based) tokens as its authentication mechanism. SAML-based token authentication is provided by using the OPSS Trust Service in the J2EE container on which its deployed.

The following figure illustrates Oracle Privileged Account Manager authentication.

Figure 2–1  Trust-Based Authentication in Oracle Privileged Account Manager

Trust Service instances are typically configured to securely propagate user identities from the client application to the Oracle Privileged Account Manager server as part of the Oracle Privileged Account Manager installation and configuration process.

Oracle Privileged Account Manager requires authentication when

- Users interact with Oracle Privileged Account Manager’s web-based user interface (the Console) and the command line tool (CLI)
- Users and clients interact directly with the Oracle Privileged Account Manager server through its RESTful interfaces

In both cases, Oracle Privileged Account Manager supports the following authentication modes, over SSL, out of the box:

- **HTTP Basic-Authentication:** The user sends the user name and password as unencrypted base64 encoded text
OPSS-Trust Service Assertions: The OPSS Trust Service allows the propagation of identities across HTTP-enabled applications by providing and validating tokens. This service uses an asserter that is available through Oracle WebLogic Server.

In addition, Oracle Privileged Account Manager and Oracle Identity Navigator can support ADF-based authentication for UI-based interactions, which is done transparently against the domain-specific identity store.

2.2.1 Authentication for the Oracle Privileged Account Manager Console

The Oracle Privileged Account Manager web-based user interface, or Console, supports the same authentication mechanisms as Oracle Identity Navigator and you can configure the interface with Oracle Access Management.

When a user interacts with the Oracle Privileged Account Manager Console and Oracle Identity Navigator, the following occurs:

1. The user authenticates against the Oracle Privileged Account Manager Console and Oracle Identity Navigator by using ADF authentication.

2. The Oracle Privileged Account Manager Console and Oracle Identity Navigator call the OPSS-Trust Service to request a token that asserts the identity of the user logged into the Oracle Privileged Account Manager Console.

3. Now, whenever the Oracle Privileged Account Manager Console and Oracle Identity Navigator make RESTful calls to the Oracle Privileged Account Manager server to execute Oracle Privileged Account Manager functionality, the Oracle Privileged Account Manager Console and Oracle Identity Navigator present the generated token to the Oracle Privileged Account Manager server.

4. Because the OPSS Trust Service Asserter is configured by default, the Asserter examines the token presented in the previous step, validates the token, and then asserts that the identity performing the RESTful call against the Oracle Privileged Account Manager server is the one contained in the token.

This process is called identity propagation. An end-user only authenticates against the Oracle Privileged Account Manager Console and Oracle Identity Navigator, but the Oracle Privileged Account Manager Console and Oracle Identity Navigator can securely convey to the Oracle Privileged Account Manager server the identity for which they are making a request.

The important point to note about identity propagation is that it removes the need for end users to authenticate themselves against the Oracle Privileged Account Manager Console, Oracle Identity Navigator, and the Oracle Privileged Account Manager server.

Note: If you deploy your own client applications against the Oracle Privileged Account Manager server, then you must have identity propagation. In such a context, it is recommended that you use OPSS-Trust Service based Identity Assertions. For more information, refer to the Oracle Fusion Middleware Application Security Guide.
2.2.2 Authentication for the Oracle Privileged Account Manager Server

The Oracle Privileged Account Manager server only exposes RESTful interfaces and supports HTTP-Basic Authorization or OPSS-Trust. In addition, the Oracle Privileged Account Manager server requires that all communication with that server occurs over an SSL-secured channel.

The Oracle Privileged Account Manager command line tool client uses HTTP Basic-Authentication over SSL to connect to, and authenticate against, the Oracle Privileged Account Manager server.

2.3 Understanding Oracle Privileged Account Manager Authorization

This section describes Oracle Privileged Account Manager authorization.

The topics include:

- Administration Role Types
- End Users

**Note:** If you are using Oracle Privileged Account Manager on IBM WebSphere, refer to "Differences in Oracle Privileged Account Manager Authorization" in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management for information about this topic.

2.3.1 Administration Role Types

Common Admin Roles are a set of predefined, standardized application roles for securing administrative access to Oracle Identity Management applications. These roles encapsulate the common administrative tasks across the Oracle Identity Management suite.

**Note:** For more information about Common Admin Roles, including the responsibilities of each role and the skills and expertise required to perform that role, refer to "Common Admin Roles" in the Oracle Fusion Middleware Administrator’s Guide for Oracle Identity Navigator.

Oracle Privileged Account Manager uses Admin Roles to manage access to targets and privileged accounts and to control which operations administrators can perform. Specifically, the Oracle Privileged Account Manager server renders different user interface components based on the Admin Role assigned to the user logging in.

Only administrators who are assigned the Oracle Privileged Account Manager-specific Common Admin Roles can administer Oracle Privileged Account Manager.

**Note:** Authorized administrators must configure and assign roles from the Administration tab in the Oracle Identity Navigator Console. Refer to "Configuring Enterprise Roles" in the Oracle Fusion Middleware Administrator’s Guide for Oracle Identity Navigator for detailed information.

The following table describes the Common Admin Roles that are specific to Oracle Privileged Account Manager.
### Table 2–1  Supported Admin Roles

<table>
<thead>
<tr>
<th>Admin Role</th>
<th>Access Rights</th>
</tr>
</thead>
</table>
| Application Configurator (OPAM_APPLICATION_CONFIGURATOR) | ■ Configure Oracle Privileged Account Manager Console and servers.  
■ Manage plug-in configurations (create plug-in configurations, modify configuration attributes the plug-in needs to work, and delete the configurations).  
  
  **Note:** When a new plug-in configuration is created, the status is disabled and the plug-in cannot be executed. This role cannot enable a plug-in configuration. Only the Security Administrator (OPAM_SECURITY_ADMIN) role can enable plug-in configurations and determine under which conditions those plug-ins can be executed.  
■ Manage properties on the Session Manager Configuration page. |
| Security Administrator (OPAM_SECURITY_ADMIN)      | ■ Manage targets (add, edit, and remove targets) and view the Password History for a target.  
■ Manage accounts (add, edit, and remove accounts) and view the Password History for an account.  
  
  **Note:** This role cannot assign grantees to privileged accounts.  
■ Manage Password and Usage Policies (create, edit, and delete policies).  
■ Assign Password Policies to accounts.  
  
  **Note:** This role cannot assign Usage Policies because they are always associated with a grant. Only the User Manager (OPAM_USER_MANAGER) role can assign grants and Usage Policies.  
■ Edit plug-in configurations, enable or disable plug-ins, and configure Filter Rules (such as enabling users or groups) to decide under which conditions a plug-in can be executed.  
  
  **Note:** This role cannot create or delete a plug-in configuration. Only the Application Configurator (OPAM_APPLICATION_CONFIGURATOR) role can create or delete a plug-in configuration.  
■ View the Session Manager Configuration page. |
| Security Auditor (OPAM_SECURITY_AUDITOR)          | ■ Open and review Oracle Privileged Account Manager reports.  
■ View Oracle Privileged Account Manager Audit reports in the Oracle Identity Navigator Reports portlet. |
| User Manager (OPAM_USER_MANAGER)                  | ■ Assign end users with grants to privileged accounts.  
■ Manage Usage Policies (create, edit, and delete Usage Policies).  
■ Assign Usage Policies to grants.  
  
  **Note:** The relationship between an account and a grantee (end user) of that account is called a grant. The User Manager can assign different Usage Policies to different grantees of the same account.  
This role cannot assign Password Policies to accounts.  
■ View the Plug-In Configuration page and search for plug-ins.  
■ Terminate all Oracle Privileged Session Manager sessions for a selected account. |

**Note:** The following information about the weblogic user is specific to working in a WebLogic environment. If you are using IBM WebSphere, refer to "Differences in Oracle Privileged Account Manager Authorization" in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management for information.
After installation, the default administrator is the weblogic user (also known as the bootstrap user) who is a member of the Administrators group. You must use the weblogic user to create and assign users to the Oracle Privileged Account Manager Admin Roles described in Table 2-1. Those users can then perform the administration tasks described in this table.

**Note:** Although it is possible for the default administrator to assign all those roles to himself or herself, this is not typical.

After installation, use the weblogic user as the bootstrap user who can map users from the domain identity store to the Oracle Privileged Account Manager Common Admin Roles described in Table 2-1. Users mapped to the Security Administrator role can assign the Common Admin Roles to other users, and can later replace the weblogic user in your environment. After you complete the initial user mapping, replace the default administrator user by mapping the Security Administrator role to at least one administrator user that is defined in your domain identity store. Refer to Section 3.3.3, "Preparing the Identity Store" for more information.

### 2.3.2 End Users

Oracle Privileged Account Manager End Users or Enterprise Users are not assigned any roles, so they have limited access to Oracle Privileged Account Manager user interface components. These users are only entitled to perform certain tasks; which includes viewing, searching, checking out, and checking in privileged accounts for which they have been granted access.

**Note:** Refer to Chapter 12, "Working with Self-Service" for more information.

### 2.4 Securing Oracle Privileged Account Manager

You can implement the recommendations described in this section to further secure Oracle Privileged Account Manager in your deployment environment.

The topics include:

- **Securing the Network Channel**
- **Securing Shared Accounts**
- **Enabling Password Resets**
- **Avoiding Assignments through Multiple Paths**
- **Defining Richer Password Policies**
- **Hardening the Back-End Oracle Privileged Account Manager Database**

#### 2.4.1 Securing the Network Channel

As part of its normal functionality, Oracle Privileged Account Manager performs remote password resets on target systems. Because these passwords allow access to those systems as privileged identities (Oracle Privileged Account Manager manages privileged accounts and identities) you must ensure that these remote password resets occur over a secured network channel.
After being reset, Oracle Privileged Account Manager propagates these passwords to end users who are requesting access to the target system as a privileged account. Again, you must ensure that these newly reset passwords are propagated to the end users over a secured channel.

Considering these points, there are two aspects of an Oracle Privileged Account Manager deployment that must be closely examined and secured:

- **Connecting to Target Systems**
- **Securing the End User Interface**

### 2.4.1.1 Connecting to Target Systems

Oracle Privileged Account Manager leverages ICF connectors to communicate with target systems. These connectors are highly flexible and they can be configured in several ways. To allow flexibility in testing (and even production), Oracle Privileged Account Manager does not mandate that this connectivity always occurs over a secure channel.

Except for the Generic UNIX targets, which mandates SSH, the Generic LDAP and Generic DB targets allow connections through both secured (encrypted) and clear channels. Therefore, it is important for an Oracle Privileged Account Manager administrator to consider all relevant factors when deciding what type of channel to use when connecting to target systems.

Oracle recommends that you always use secured channels to mitigate the risk of password compromise due to packet sniffing. If the target system (either LDAP or DB) supports SSL and is listening on an SSL port, then Oracle Privileged Account Manager can communicate with that target over SSL.

Consult your target systems' product documentation for information about configuring your targets so that they are listening on an SSL port. To configure Oracle Privileged Account Manager to communicate through SSL, refer to Section 15.1, "Configuring Oracle Privileged Account Manager to Communicate With Target Systems Over SSL." Securing these connections through SSL ensures that the password reset operations performed by Oracle Privileged Account Manager occur in a secure manner.

### 2.4.1.2 Securing the End User Interface

There are two primary interfaces open to an Oracle Privileged Account Manager end user:

- **Console** (refer to Chapter 4, "Starting and Using the Oracle Privileged Account Manager Console" for more information)
- **Command line tool** (refer to Appendix A, "Working with the Command Line Tool" for more information)

Oracle Privileged Account Manager’s Console is hosted in Oracle Identity Navigator. However, Oracle Identity Navigator is also used for other purposes, so it can be deployed with SSL enabled or disabled.

If you deploy Oracle Identity Navigator with SSL disabled, even if Oracle Identity Navigator communicates with the Oracle Privileged Account Manager server over an SSL secured channel, then the connectivity between Oracle Identity Navigator (for example, the Oracle Privileged Account Manager Console) and the end user browser is not secured, which can cause security concerns.
Oracle recommends that if you use Oracle Identity Navigator to serve the Oracle Privileged Account Manager Console, you must deploy Oracle Identity Navigator in an SSL (and only SSL) -enabled mode.

**Note:** For more information about configuring SSL for Oracle Identity Navigator, refer to “Configuring Secure Socket Layer” in the Oracle Fusion Middleware Administrator’s Guide for Oracle Identity Navigator.

Because the Oracle Privileged Account Manager server mandates SSL connectivity, the Oracle Privileged Account Manager command line tool always uses SSL and communicates over a secure channel. Consequently, when the Oracle Privileged Account Manager server propagates a password to an end user through the command line tool, it always uses a secured channel and prevents compromises from packet sniffing.

### 2.4.2 Securing Shared Accounts

Oracle Privileged Account Manager enables you to specify whether a privileged account is *shared* or *not shared*. This section defines shared accounts, explains some security considerations, and describes how to improve security for a shared account.

#### 2.4.2.1 What is a Shared Account?

By default, Oracle Privileged Account Manager allows only one user to check out an account at a time. If a second user tries to check out an already checked-out account, an error message displays stating the account is already checked out.

Oracle Privileged Account Manager also enables you to configure a *shared* account, which enables multiple users to check out the account at the same time.

When multiple users check out a shared account, Oracle Privileged Account Manager shares the password generated by the first user instead of generating a new password for each user. (Setting a new password would affect the existing check out.) Oracle Privileged Account Manager does not reset that password until all users have checked in the account and the last person has checked in the password.

Oracle recommends that you designate an account as shared only if there are compelling business reasons to do so. For example, sharing a database account might be advantageous if that account is being administered by multiple people.

#### 2.4.2.2 Security Limitations

When you configure a shared account, keep in mind the following security limitations:

- Users can still use the password after checking in an account because Oracle Privileged Account Manager does not reset the password until the last user checks it in.
- Sharing accounts presents a problem with achieving a fine-grained audit. Oracle Privileged Account Manager can provide an audit trail that shows when the account was checked out and which users had access to that account at any given time. However, if multiple end users have the same privileged account checked out at the same time, then Oracle Privileged Account Manager cannot isolate the actions taken by an individual end user.
2.4.2.3 How to Secure the Account

If you do have a compelling reason for sharing an account, it’s useful to take the following steps to secure that account:

1. Configure the Usage Policy to automatically check in the privileged account after a specified period of time. Automatic check-ins ensure that shared privileged accounts get checked in and that passwords get cycled in a timely manner.

2. Limit the number of users to whom you assign the privileged account and try to further segregate these users by specifying when they can access the account. You can configure the Usage Policy to specify which days of the week and what times of the day a user can access an account. These limitations can minimize overlapping checkouts, which improves Oracle Privileged Account Manager’s ability to audit.

---

**Note:** For more information about configuring a Usage Policy, refer to Section 9.3.3, "Modifying the Default Usage Policy" or Section 9.3.4, "Creating a Usage Policy."

---

2.4.3 Enabling Password Resets

Oracle Privileged Account Manager allows you to configure the Password Policy for a privileged account so that Oracle Privileged Account Manager automatically resets the privileged account's password when the account is checked-out, checked-in, in both cases, or in neither case.

At a minimum, Oracle recommends that you configure and apply a Password Policy to reset the privileged account's password on check in. Resetting the password on check in prevents end users from using that account after checking it in because the password they used is no longer associated with that privileged account. This feature is one of the fundamental innovations in Oracle Privileged Account Manager and should be used.

---

**Note:** For more information about configuring and working with Password Policies, refer to Chapter 9, "Working with Policies."

---

2.4.4 Avoiding Assignments through Multiple Paths

In addition to directly assigning privileged accounts to end users, Oracle Privileged Account Manager allows you to assign privileged accounts to groups. For example, you might want to create a "Data Center Product UNIX Administrators" group and give that group access to certain privileged accounts.

When designing your deployment, it is important to ensure that a given end user is granted access to a privileged account through only one path (either directly or through a single group). When Oracle Privileged Account Manager discovers multiple grant paths, it picks the first path retrieved from its back-end, which leads to non-deterministic behavior. This behavior can cause the effective Usage Policy to be different from the intended Usage Policy.

On a related note, you must avoid creating groups with multiple naming attribute values or you might enable users to access groups for which they were not explicitly granted access. Refer to Section C.3.10, "Oracle Privileged Account Manager End Users Gain Privileges They Were Not Explicitly Granted" for more for more information.
2.4.5 Defining Richer Password Policies

The primary purpose of an Oracle Privileged Account Manager’s Password Policy is to ensure the success of an Oracle Privileged Account Manager-initiated password reset that occurs against a target system.

At a minimum, Oracle Privileged Account Manager requires the effective Password Policy on a privileged account to describe the Password Policy being enforced on the target system. However, Oracle Privileged Account Manager administrators are not restricted to this requirement. You can define a much richer Password Policy in Oracle Privileged Account Manager that generates more complex and secure passwords during Oracle Privileged Account Manager reset operations.

Note: For more information about configuring and working with Password Policies, refer to Chapter 9, "Working with Policies."

2.4.6 Hardening the Back-End Oracle Privileged Account Manager Database

Beginning with the 11gR2 11.1.2.1.0 release, Oracle Privileged Account Manager moved to using the Oracle RDBMS as its data store. As such, it is important to ensure that the back-end Oracle Privileged Account Manager database is locked down for production deployments.

Oracle recommends that administrators perform the following three tasks to effectively lock down a back-end Oracle Privileged Account Manager database:

- Enable TDE
- Enable SSL
- Use Oracle Database Vault

Enable TDE

Oracle recommends enabling Transparent Data Encryption (TDE) mode on your back-end Oracle Privileged Account Manager database for all production deployments of Oracle Privileged Account Manager. Enabling TDE ensures that all sensitive information stored by Oracle Privileged Account Manager (such as account passwords) is encrypted on disk.

For security purposes, enabling TDE for Oracle Privileged Account Manager is a two-step process. You must

1. Enable TDE in the back-end database
   
   Refer to "Enabling TDE in the Database" in the Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management.

2. Configure Oracle Privileged Account Manager to expect a back-end database that is configured to use TDE.
   
   To configure Oracle Privileged Account Manager from the Console, refer to Section 5.2.3, "Managing Oracle Privileged Account Manager Server Properties."
To configure Oracle Privileged Account Manager from the command line, refer to Section 15.2.1, "Enabling TDE Mode."

---

**Note:** for more information about using Transparent Data Encryption, visit these websites:

---

**Enable SSL**

In addition to the on-disk encryption provided by TDE, Oracle Privileged Account Manager obfuscates all sensitive information that it stores. This obfuscation ensures that anyone gaining access to the Oracle Privileged Account Manager schema still does not have direct access to sensitive information. Furthermore, the obfuscation prevents any malicious elements monitoring network traffic from easily viewing sensitive information. However, Oracle recommends that you perform both of the following actions for production deployments of Oracle Privileged Account Manager:

- Enable SSL on the back-end RDBMS used by Oracle Privileged Account Manager.
- Update the OPAMDS data source to use the SSL endpoints on your back-end RDBMS.

---

**Note:** Refer to "Enabling SSL in Other Authentication Modes" and to "SSL-Enable a Data Source" in the Oracle Fusion Middleware Administrator’s Guide for instructions.

---

These actions encrypt network traffic between Oracle Privileged Account Manager and the back-end Oracle Privileged Account Manager database. They also ensure that the database is not vulnerable to exploitation by malicious elements.

**Use Oracle Database Vault**

Using Oracle Database Vault for production deployments of Oracle Privileged Account Manager secures the Oracle Privileged Account Manager schema and ensures that only the Oracle Privileged Account Manager application has access to the schema.

---

**Note:** Refer to "Configuring Secure Application Roles for Oracle Database Vault" in the Oracle Database Vault Administrator’s Guide for instructions.

---

Thus, you prevent inadvertent or malicious access to the Oracle Privileged Account Manager application schema (and associated data) by database administrators and other implicitly authorized users. You also ensure that only the Oracle Privileged Account Manager application can access the sensitive information it maintains in the back-end Oracle Privileged Account Manager database.
2.5 Understanding Session Management Security

Currently, Oracle Privileged Account Manager only provides Session Management on SSH-enabled targets. Accessing a target through the Oracle Privileged Session Manager (Session Manager) prevents the end-user from being exposed to the password associated with the privileged account in use. Therefore, for all use-cases where the end-user does not require knowledge of the password associated with a privileged account, granting access to just the session, as opposed to the password or both the session and password, is recommended. Refer to step 3 in Section 9.3.4, "Creating a Usage Policy."

SSH Session Management support requires that the Session Manager perform the tasks associated with an SSH server. For server authentication and data privacy (such as encryption), Oracle Privileged Account Manager generates a new DSA 1024 bit (such as FIPS 186-2 compliant) SSH server key for every Oracle Privileged Account Manager instance. This server key is configurable and it can be changed by using Session Manager Configuration for key rollover.

2.6 Understanding Plug-In Security

The Oracle Privileged Account Manager plug-in framework provides for significant extensibility and customizability. However, this framework also allows custom code to interface closely with Oracle Privileged Account Manager functionality. Therefore, protecting Oracle Privileged Account Manager against malicious custom code is very important.

Because custom plug-in code can interface closely with internal Oracle Privileged Account Manager functionality, it is important that you closely scrutinize all custom plug-in code that is deployed on Oracle Privileged Account Manager. To prevent a single malicious entity from deploying malicious custom code, Oracle Privileged Account Manager enforces a requirement that two different Oracle Privileged Account Manager administrators (with different roles) are involved during the process of deploying custom plug-in code.

First, an administrator with the OPAM_APPLICATION_CONFIGURATOR Admin Role can add a new custom plug-in and configure it. However, the action of enabling the plug-in and determining the conditions under which it can run must be done by a second administrator with the OPAM_SECURITY_ADMIN Admin Role. For more information, refer to Section 11.3, "Creating a Plug-In Configuration." This design ensures that a single malicious entity cannot deploy a malicious plug-in by himself or herself.

Furthermore, because the custom plug-in code runs within the context of the Oracle Privileged Account Manager server, internal Oracle Privileged Account Manager APIs could potentially be accessible to the custom plug-in code. To pro-actively prevent this access, the Oracle Privileged Account Manager plug-in framework uses a custom class loader that explicitly prevents access to all internal Oracle Privileged Account Manager server APIs. Therefore, the custom plug-in code must interface with internal Oracle Privileged Account Manager logic through the well-defined APIs described in Section 16.3, "Understanding the Plug-In API."
Part II
Basic Administration

This part provides information about performing basic administration tasks for Oracle Privileged Account Manager from the Console, and it contains the following chapters:

- Getting Started with Managing Oracle Privileged Account Manager
- Starting and Using the Oracle Privileged Account Manager Console
- Configuring and Managing the Servers
- Working with Targets
- Working with Service Accounts
- Working with Privileged Accounts
- Working with Policies
- Working with Grantees
- Working with Plug-Ins
- Working with Self-Service
Getting Started with Managing Oracle Privileged Account Manager

This chapter describes how to finish configuring Oracle Privileged Account Manager after installation.

Note: You can manage Oracle Privileged Account Manager from the Console, from the command line, and by using Oracle Privileged Account Manager’s RESTful interface.

- For information for starting and using the Oracle Privileged Account Manager Command Line Tool (CLI), refer to Appendix A, "Working with the Command Line Tool."
- For information for starting and using the Oracle Privileged Account Manager RESTful interface, refer to Appendix B, "Working with Oracle Privileged Account Manager’s RESTful Interface.”

This chapter includes the following sections:

- Section 3.1, "Before You Begin"
- Section 3.2, "Understanding ICF Connectors in Oracle Privileged Account Manager”
- Section 3.3, "Starting Oracle Privileged Account Manager”
- Section 3.4, "Administering Oracle Privileged Account Manager”
- Section 3.5, "Working with Oracle Privileged Account Manager Self-Service”

Note: If you are using Oracle Privileged Account Manager on IBM WebSphere, refer to “Differences in Getting Started with Administering Oracle Privileged Account Manager” in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management for information about this topic.

3.1 Before You Begin

This chapter assumes that you have installed and configured Oracle Privileged Account Manager 11g Release 2 (11.1.2) as described in Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management.
Before starting the final configuration steps needed to start Oracle Privileged Account Manager, Oracle recommends the following:

- Read the "Configuring Oracle Privileged Account Manager" chapter in the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management.*
- Review Table 3–1 to understand the default application URLs for various interfaces that you use to manage Oracle Privileged Account Manager in this release:

<table>
<thead>
<tr>
<th>Interface</th>
<th>Default URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Identity Navigator</td>
<td>http://managedserver_host:managedserver_port/oinav/</td>
</tr>
<tr>
<td>Oracle WebLogic Server Administrative Console</td>
<td>http://adminserver_host:adminserver_port/console/</td>
</tr>
<tr>
<td>Oracle Privileged Account Manager Console</td>
<td>http://managedserver_host:managedserver_port/oinav/opam</td>
</tr>
<tr>
<td>Oracle Privileged Account Manager Server</td>
<td>http://managedserver_host:managedserver_sslport/opam</td>
</tr>
</tbody>
</table>

- Review Table 3–2 to understand the various default ports for Oracle Privileged Account Manager in this release:

<table>
<thead>
<tr>
<th>Port Type</th>
<th>Default Port</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Privileged Account Manager Server</td>
<td>18102</td>
<td>The default SSL-enabled port for the WebLogic Managed Server on which the Oracle Privileged Account Manager server is deployed.</td>
</tr>
<tr>
<td>Oracle Privileged Account Manager Console</td>
<td>18101 (non-SSL)</td>
<td>The WebLogic Managed Server port on which the Oracle Privileged Account Manager Console is available by default.</td>
</tr>
<tr>
<td></td>
<td>18102 (SSL)</td>
<td></td>
</tr>
<tr>
<td>Oracle Privileged Session Manager (SSH)</td>
<td>1222</td>
<td>The default WebLogic Managed Server port on which Oracle Privileged Session Manager listens for SSH traffic.</td>
</tr>
<tr>
<td>WebLogic Admin Console</td>
<td>7001 (non-SSL)</td>
<td>The default WebLogic Admin Server ports on which the WebLogic Admin Console is available.</td>
</tr>
<tr>
<td></td>
<td>7002 (SSL)</td>
<td></td>
</tr>
</tbody>
</table>

- Review Table 3–3 to become familiar with the common directory variables that are used throughout this book:

**Note:** For additional information about these directories, and other common directories used in most Oracle Identity and Access Management installations and configurations, refer to "Identifying Installation Directories" in the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management* and "Understanding Oracle Fusion Middleware Concepts and Directory Structure" in the *Oracle Fusion Middleware Installation Planning Guide for Oracle Identity and Access Management.*
3.2 Understanding ICF Connectors in Oracle Privileged Account Manager

Oracle Privileged Account Manager enables you to secure, share, audit, and manage administrator-identified account credentials. To provide these capabilities, Oracle Privileged Account Manager must be able to access and manage privileged accounts on a target system.

Connectors enable Oracle Privileged Account Manager to interact with target systems, such as LDAP or Oracle Database, and to perform Oracle Privileged Account Manager-relevant administrative operations on those systems.

Oracle Privileged Account Manager leverages connectors that are compliant with the Identity Connector Framework (ICF) standard. By using this standard, you separate Oracle Privileged Account Manager from the mechanism it uses for connecting to targets. Therefore, in addition to connectors provided by vendors such as Oracle, you are free to build, test, and deploy your own ICF connectors into Oracle Privileged Account Manager.

This section describes how Oracle Privileged Account Manager consumes these ICF connectors. The topics include:

- About the ICF Connectors
- Locating the Oracle Privileged Account Manager Connector Bundles
- Consuming ICF Connectors

**Note:** For more information about the Identity Connector Framework, refer to “Understanding the Identity Connector Framework” in the Oracle Fusion Middleware Developer’s Guide for Oracle Identity Manager.

### 3.2.1 About the ICF Connectors

Oracle Privileged Account Manager ships with the following ICF-compliant connectors that were developed by Oracle:

- Database User Management (DBUM) Connector
Understanding ICF Connectors in Oracle Privileged Account Manager

- Generic LDAP Connector
- Oracle Identity Manager Connector for UNIX

These connectors enable Oracle Privileged Account Manager to manage privileged accounts on a range of target systems belonging to the preceding types.

Oracle Privileged Account Manager can also use customer-created, ICF-compliant connectors, which empowers you to manage your proprietary systems by using Oracle Privileged Account Manager.

---

**Note:** If you are only interested in using the connectors that ship with Oracle Privileged Account Manager, then *no further action is required* because these connectors come pre-configured out-of-the-box.

If you want to use other Oracle connectors or a custom connector, then refer to [Section 15.3, "Adding New Connectors to an Existing Oracle Privileged Account Manager Installation"](http://example.com) for more information.

For additional information about developing ICF-compliant connectors, refer to "Developing Identity Connectors" in the [Oracle Fusion Middleware Developer’s Guide for Oracle Identity Manager](http://example.com).

---

### 3.2.2 Locating the Oracle Privileged Account Manager Connector Bundles

Because ICF connectors are generic, and useful in numerous contexts, a given Oracle installation puts all connector bundles into a single location on the file system. All components (such as Oracle Privileged Account Manager) that rely on these connector bundles can access them from this location:

```
ORACLE_HOME/connectors
```

The connectors that are pushed into `ORACLE_HOME/connectors` are actually shipped with Oracle Identity Manager. Of all the connectors in this directory, only the following three connectors are certified with Oracle Privileged Account Manager for this release:

- `org.identityconnectors.dbum-1.0.1116.jar`
- `org.identityconnectors.genericunix-1.0.0.jar`
- `org.identityconnectors.ldap-1.0.6380.jar`

---

**Note:** If you obtain any new ICF connectors from Oracle, you must place them in the location specified in the instructions provided.

Storing custom third-party connectors is at your discretion; however, you must ensure they can be read by Oracle Privileged Account Manager at run time.

---

### 3.2.3 Consuming ICF Connectors

Oracle Privileged Account Manager consumes ICF connectors by using the `opam-config.xml` file. The contents of this file provide the following information to Oracle Privileged Account Manager:

1. Where to pick up the ICF connector bundle (on the file system)
2. Which configuration attributes are relevant for the Oracle Privileged Account Manager use-cases
3. How to render the Oracle Privileged Account Manager Console when configuring connectivity to a target system using a particular connector

You will find the `opam-config.xml` file in the `ORACLE_HOME/opam/config` directory. The out-of-the-box image is configured to pick up and use the connector bundles that ship with the Oracle Identity Management Suite.

The `opam-config.xsd` file (also located in the `ORACLE_HOME/opam/config` directory) describes the schema for `opam-config.xml`. If you make any changes to `ORACLE_HOME/opam/config/opam-config.xml` file, verify them with the `opam-config.xsd` file.

**Caution:** Be sure to back-up the original `opam-config.xml` file before attempting to edit that file.

### 3.3 Starting Oracle Privileged Account Manager

This section provides some high-level information about starting and working with Oracle Privileged Account Manager. The topics include:

- **Starting WebLogic**
- **Configuring an External Identity Store for Oracle Privileged Account Manager**
- **Preparing the Identity Store**
- **Assigning the Application Configurator Role to a User**

The procedures described in this section reference information and instructions contained in the following Oracle publications. If necessary, review the referenced concepts, terminology, and procedures before starting these procedures.

<table>
<thead>
<tr>
<th>Reference Publications</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Table 3–4</strong> For Information About</td>
</tr>
<tr>
<td>Admin Roles</td>
</tr>
<tr>
<td>Supported identity and Policy Store configurations for Oracle Privileged Account Manager and Oracle Identity Navigator</td>
</tr>
<tr>
<td>Oracle WebLogic Server concepts and terminology</td>
</tr>
<tr>
<td>Creating a default authenticator in Oracle WebLogic Server</td>
</tr>
<tr>
<td>Configuring an identity store in your environment</td>
</tr>
<tr>
<td>Configuring Oracle Virtual Directory with the LDAP-based server</td>
</tr>
<tr>
<td>Configuring the OVD authenticator in Oracle WebLogic Server</td>
</tr>
<tr>
<td>Connecting the Node Manager to WLST</td>
</tr>
</tbody>
</table>
3.3.1 Starting WebLogic

Before you can start Oracle Privileged Account Manager, you must start the WebLogic servers and console.

Note:

- If you are using Oracle Privileged Account Manager on IBM WebSphere, you must start IBM WebSphere and perform some configuration steps before assigning the Application Configurator and invoking the Oracle Privileged Account Manager Console.

For more information about these tasks, refer to "Starting Oracle Privileged Account Manager on IBM WebSphere" in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management.

- Oracle Privileged Account Manager administrators and users will probably never have to use the Oracle Identity Navigator interface except during the initial set-up of Oracle Privileged Account Manager.

Note:

- For detailed information about starting WebLogic and Managed Servers, refer to "Starting or Stopping the Oracle Stack" in the Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management.

- You must have the appropriate Administration Role and credentials to start the server. Refer to Section 2.3.1, "Administration Role Types" for more information.

1. Connect the Node Manager to WLST by running the nmConnect command.

Refer to "Node Manager Commands" in the Oracle Fusion Middleware WebLogic Scripting Tool Command Reference for instructions.

2. Start the WebLogic Admin Server. For example,

On UNIX, type

```bash
MW_HOME/user_projects/domains/DOMAIN_NAME/bin/startWebLogic.sh
```

On Windows, type

```bat
MW_HOME\user_projects\domains\DOMAIN_NAME\bin\startWebLogic.bat
```
3. Start the Oracle Privileged Account Manager Managed Server.
4. Open a browser and start the WebLogic Console from the following location:
   http://adminserver_host:adminserver_port/console

### 3.3.2 Configuring an External Identity Store for Oracle Privileged Account Manager

This section describes how to configure a new, external identity store for Oracle Privileged Account Manager.

---

**Note:** If you are using IBM WebSphere, you must configure a registry rather than an external identity store. Refer to "Configuring a Registry" in the *Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management* for instructions.

---

You must configure a domain identity store before you can view users when searching from the Oracle Identity Navigator Access Privileges pane. To configure the identity store as the main authentication source, you must configure the Oracle WebLogic Server domain where Oracle Identity Navigator is installed.

You can configure the domain identity store using Oracle Internet Directory or Oracle Virtual Directory with a supported LDAP-based directory server. You configure the identity store in the WebLogic Server Administration Console.

---

**Note:**

- Oracle Privileged Account Manager can use any LDAP directory that is supported by Oracle WebLogic Server, as its identity store.
  
  For more information about configuring an identity store, refer to "Configuring the Identity Store Service" in the *Oracle Fusion Middleware Application Security Guide*.

- For information about other supported identity stores, refer to "System Requirements and Certification" in the *Oracle Fusion Middleware Administrator’s Guide for Oracle Identity Navigator*.

---

To configure the Oracle Internet Directory authenticator in Oracle WebLogic Server:

1. Log in to Oracle WebLogic Server Administration Console, and click **Lock & Edit** in the Change Center.
2. In Oracle WebLogic Server Administration Console, select **Security Realms** from the left pane and click the realm you are configuring. For example, the default realm is `myrealm`.
3. Select the Providers tab, then select the Authentication subtab.
4. Click **New** to launch the Create a New Authentication Provider page and complete the fields as follows:
   - **Name:** Enter a name for the Authentication provider. For example, `MyOIDDirectory`.
   - **Type:** Select `OracleInternetDirectoryAuthenticator` from the list.
Click OK to update the Authentication providers table.

5. In the Authentication providers table, click the newly added authenticator.

6. In Settings, select the Configuration tab, then select the Common tab.

7. On the Common tab, set the Control Flag to SUFFICIENT.

Setting the Control Flag attribute for the authenticator provider determines the ordered execution of the Authentication providers. The possible values for the Control Flag attribute are:

- **REQUIRED** - This LoginModule must succeed. Even if it fails, authentication proceeds down the list of LoginModules for the configured Authentication providers. This setting is the default.
- **REQUISITE** - This LoginModule must succeed. If other Authentication providers are configured and this LoginModule succeeds, authentication proceeds down the list of LoginModules. Otherwise, control is returned to the application.
- **SUFFICIENT** - This LoginModule need not succeed. If it does succeed, return control to the application. If it fails and other Authentication providers are configured, authentication proceeds down the LoginModule list.
- **OPTIONAL** - This LoginModule can succeed or fail. However, if all Authentication providers configured in a security realm have the JAAS Control Flag set to **OPTIONAL**, the user must pass the authentication test of one of the configured providers.

8. Click Save.

9. Select the Provider Specific tab and enter the following required settings using values for your environment:

- **Host**: The host name of the Oracle Internet Directory server.
- **Port**: The port number on which the Oracle Internet Directory server is listening.
- **Principal**: The distinguished name (DN) of the Oracle Internet Directory user to be used to connect to the Oracle Internet Directory server. For example: cn=OIDUser, cn=users, dc=us, dc=mycompany, dc=com.
- **Credential**: Password for the Oracle Internet Directory user entered as the Principal.
- **Group Base DN**: The base distinguished name (DN) of the Oracle Internet Directory server tree that contains groups.
- **User Base DN**: The base distinguished name (DN) of the Oracle Internet Directory server tree that contains users.
- **All Users Filter**: LDAP search filter. Click More Info for details.
- **User From Name Filter**: LDAP search filter. Click More Info for details.
- **User Name Attribute**: The attribute that you want to use to authenticate (for example, cn, uid, or mail). For example, to authenticate using a user's email address you set this value to mail.

10. Click Save.
11. From the Settings for myrealm page, select the Providers tab, then select the Authentication tab.

12. Click Reorder.

13. Select the new authenticator and use the arrow buttons to move it into the first position in the list.

14. Click OK.

15. Click DefaultAuthenticator in the Authentication providers table to display the Settings for DefaultAuthenticator page.

16. Select the Configuration tab, then the Common tab, and select SUFFICIENT from the Control Flag list.

17. In the Change Center, click Activate Changes.


19. Verify your configuration and set-up by confirming that the users present in the LDAP directory (Oracle Internet Directory or Oracle Virtual Directory) can log in to Oracle Privileged Account Manager with no issues.

To use Oracle Virtual Directory as the domain identity store, you must do the following:

- Configure the OVD authenticator in Oracle WebLogic Server as described in Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help.
- You must enable the Use Retrieved User Name As Principal option when configuring authenticators in Oracle WebLogic Server, as described in the preceding step 9.

**Note:** If you are using an SSL-enabled identity store, follow the steps described in “SSL for the Identity Store Service” in the Oracle Fusion Middleware Application Security Guide.

### 3.3.3 Preparing the Identity Store

If you want to use an external LDAP server to serve as an identity store, you must seed the identity store with the necessary Oracle Privileged Account Manager users and groups.

- If you are using an Oracle Internet Directory directory or a third-party directory that is fronted by Oracle Virtual Directory as the identity store, then you can use the idmConfigTool to prepare the directory. When using the idmConfigTool, you must also extend the schema. Refer to Section 3.3.3.1, “Using the idmConfigTool to Prepare the Directory” for more information.
- If you’re using a different directory, then you must manually prepare that directory. Refer to Section 3.3.3.2, “Manually Preparing the Directory” for more information.
3.3.3.1 Using the idmConfigTool to Prepare the Directory

If you are preparing an Oracle Internet Directory directory or a third-party directory fronted by OVD as an identity store, then you can use the idmConfigTool to perform the following tasks:

- Extend the Directory Schema for Oracle Privileged Account Manager
- Create Users and Groups for Oracle Privileged Account Manager

3.3.3.1.1 Extend the Directory Schema for Oracle Privileged Account Manager

Pre-configuring the identity store extends the schema in Oracle Internet Directory.

To pre-configure the identity store, you must perform the following tasks on IDMHOST1:

1. Set the environment variables: MY_HOME, JAVA_HOME, and ORACLE_HOME.
   - Set ORACLE_HOME to IAM_HOME

2. Create a properties file, called extend.props, with the following contents:

   ```
   IDSTORE_HOST: idstore.mycompany.com
   IDSTORE_PORT: 389
   IDSTORE_BINDDN: cn=orcladmin
   IDSTORE_USERNAMEATTRIBUTE: cn
   IDSTORE_LOGINATTRIBUTE: uid
   IDSTORE_USERSEARCHBASE: cn=Users,dc=mycompany,dc=com
   IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=mycompany,dc=com
   IDSTORE_SEARCHBASE: dc=mycompany,dc=com
   IDSTORE_SYSTEMIDBASE: cn=systemids,dc=mycompany,dc=com
   ```

   Where:

   - **IDSTORE_HOST** and **IDSTORE_PORT** are, respectively, the host and port of your identity store directory.
     - If you are using a non-OID directory, then specify the Oracle Virtual Directory host (which should be **IDSTORE.mycompany.com**).
     - If your identity store is in Oracle Internet Directory, then **IDSTORE_HOST** should point to Oracle Internet Directory, even if you are fronting Oracle Internet Directory with Oracle Virtual Directory.
   - **IDSTORE_BINDDN** is an administrative user in the identity store directory.
   - **IDSTORE_USERNAMEATTRIBUTE** is the LDAP attribute that contains the username. This attribute is usually **CN**.
3. Configure the identity store by using the `idmConfigTool` command, which is located at:

```
IAM_HOME/idmtools/bin
```

**Note:** When you run the `idmConfigTool` command, it creates or appends to the `idmDomainConfig.param` file. This file is generated in the same directory where you run the `idmConfigTool` command.

To ensure that you append to the same file each time you run the tool, always run `idmConfigTool` from the following directory:

```
IAM_HOME/idmtools/bin
```

- **On Linux**, the command syntax is:
  
  ```
  idmConfigTool.sh -preConfigIDStore input_file=configfile
  ```

- **On Windows**, the command syntax is:
  
  ```
  idmConfigTool.bat -preConfigIDStore input_file=configfile
  ```

For example:

```
idmConfigTool.sh -preConfigIDStore input_file=extend.props
```

When the command runs, you are prompted to enter the password of the account that you are using to connect to the identity store.

Sample command output, when running the command against Oracle Virtual Directory:

```
Enter ID Store Bind DN password:
```

```
May 25, 2011 2:37:18 PM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/app/oracle/product/fmw/iam/idmtools/templates/oid/idm_idstore_groups_template.ldif
May 25, 2011 2:37:18 PM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/app/oracle/product/fmw/iam/idmtools/templates/oid/idm_idstore_groups_acl_template.ldif
May 25, 2011 2:37:18 PM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/app/oracle/product/fmw/iam/idmtools/templates/oid/systemid_pwdpolicy.ldif
May 25, 2011 2:37:18 PM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/app/oracle/product/fmw/iam/idmtools/templates/oid/idstore_tuning.ldif
May 25, 2011 2:37:18 PM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/app/oracle/product/fmw/iam/idmtools/templates/oid/oid_schema_extn.ldif
May 25, 2011 2:37:18 PM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/app/oracle/product/fmw/iam/oblix/server/oim-intg/schema/0ID_oblix_pwd_schema_add.ldif
May 25, 2011 2:37:18 PM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/app/oracle/product/fmw/iam/oblix/server/oim-intg/schema/0ID_oblix_schema_add.ldif
```

```
``
4. A file named automation.log is created in the directory from where you ran the tool. Check this log file for any errors or warnings and correct them.

Note: In addition to creating users, the idmConfigTool creates these groups:
- OrclPolicyAndCredentialWritePrivilegeGroup
- OrclPolicyAndCredentialReadPrivilegeGroup

See Also:
Oracle Fusion Middleware Integration Overview for Oracle Identity Management Suite for more information about the idmConfigTool command.

3.3.3.1.2 Create Users and Groups for Oracle Privileged Account Manager If you plan to implement Oracle Privileged Account Manager in your topology, you must seed the identity store with the users and groups that are required by Oracle Privileged Account Manager.

Note: The use of apm and APM in the following procedure is appropriate for setting up the users and groups required by Oracle Privileged Account Manager.

To create the necessary users and groups, perform the following tasks on IDMHOST1:

1. Set the environment variables: MW_HOME, JAVA_HOME, and ORACLE_HOME.
   Set ORACLE_HOME to IAM_HOME.

2. Create a properties file, called apm.props with the following contents:

   IDSTORE_HOST: idstore.mycompany.com
   IDSTORE_PORT: 389
   IDSTORE BINDDN: cn=orcladmin
   IDSTORE_USERNAMEATTRIBUTE: cn
   IDSTORE LOGINATTRIBUTE: uid
   IDSTORE_USERSEARCHBASE: cn=Users,dc=mycompany,dc=com
   IDSTORE GROUPSEARCHBASE: cn=Groups,dc=mycompany,dc=com
   IDSTORE SEARCHBASE: dc=mycompany,dc=com
   POLICYSTORE_SHARES_IDSTORE: true
   IDSTORE APAMUSER: opamadmin

Where
- IDSTORE_HOST and IDSTORE_PORT are, respectively, the host and port of your identity store directory.
  - If you are using a non-OID directory, then specify the Oracle Virtual Directory host (which should be IDSTORE.mycompany.com).
Starting Oracle Privileged Account Manager

If your identity store is in Oracle Internet Directory, then 
**IDSTORE_HOST** should point to Oracle Internet Directory, even if you are 
fronting Oracle Internet Directory with Oracle Virtual Directory.

- **IDSTORE_BINDDN** is an administrative user in the identity store Directory.
- **IDSTORE_USERNAMEATTRIBUTE** is the LDAP attribute that contains the 
username. This attribute is usually **CN**.
- **IDSTORELOGINATTRIBUTE** is the LDAP attribute that contains the user’s Login 
name.
- **IDSTORE_USERSEARCHBASE** is the location in the directory where users are 
stored.
- **IDSTORE_GROUPSEARCHBASE** is the location in the directory where groups are 
stored.
- **IDSTORE_SEARCHBASE** is the location in the directory where users and groups 
are stored.
- **POLICYSTORE_SHARES_IDSTORE**
  - If your Policy and identity stores are in the same directory, set to **true**.
  - If your Policy and identity stores are **not** in the same directory, set to **false**.
- **IDSTORE_APMUSER** is the name of the user you want to create as your Oracle 
Privileged Account Manager administrator.

In addition to creating the users, this command assigns the users to the groups 
created in Section 3.1, "Before You Begin."

3. Configure the identity store by using the *idmConfigTool* command, which is 
located at:

IAM_HOME/idmtools/bin

---

**Note:** When you run the *idmConfigTool* command, it creates or 
appends to the *idmDomainConfig.param* file. This file is generated in 
the same directory where you run the *idmConfigTool* command.

To ensure that you append to the same file each time you run the tool, 
always run *idmConfigTool* from the following directory:

IAM_HOME/idmtools/bin

---

- **On Linux**, the command syntax is:

  idmConfigTool.sh -prepareIDStore mode=APM input_file=configfile

- **On Windows**, the command syntax is:

  idmConfigTool.bat -prepareIDStore mode=APM input_file=configfile

For example:

idmConfigTool.sh -prepareIDStore mode=APM input_file=apm.props

When the command runs, you are prompted to enter the password of the account 
that you are using to connect to the identity store.
Starting Oracle Privileged Account Manager

Sample command output:

Enter ID Store Bind DN password:
Feb 18, 2013 10:10:35 PM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/app/oracle/product/fmw/iam/common/templates/oinav_template_oid.ldif
*** Creation of APM User ***
Feb 18, 2013 10:10:35 PM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/app/oracle/product/fmw/iam/idmtools/templates/oid/oam_user_template.ldif
Enter User Password for opamadmin:
Confirm User Password for opamadmin:
The tool has completed its operation. Details have been logged to automation.log

4. A file named automation.log is created in the directory from where you ran the tool. Check this log file for any errors or warnings and correct them.

See Also:
Oracle Fusion Middleware Integration Overview for Oracle Identity Management Suite for more information about the idmConfigTool command.

3.3.3.2 Manually Preparing the Directory

Use the following steps to manually prepare your directory:

1. Identify the group container that you will be using for the Oracle Privileged Account Manager Admin Roles in your LDAP Directory. (The group container is the "Group Base DN" on your Weblogic Authenticator Configuration.)

   Refer to "Configuring Authentication Providers" in the Oracle Fusion Middleware Securing Oracle WebLogic Server for more information.

2. Create the Oracle Privileged Account Manager Admin Roles in the LDAP group container that you identified in Step 1.

   If necessary, refer to Section 2.3.1, "Administration Role Types" for a list of the roles that must be created and their purpose.

3. Assign the appropriate users to these roles.

   For example, Section 3.3.4, "Assigning the Application Configurator Role to a User" describes how to assign a role by using Oracle Identity Navigator. However, you can also assign roles by using other mechanisms (such as an LDAP Browser, a provisioning system such as Oracle Identity Manager, etc.) that can update the LDAP Directory.

3.3.4 Assigning the Application Configurator Role to a User

After installation, you do not have any users present with administrator roles. You must select a user and grant that person the Application Configurator role by using Oracle Identity Navigator.
When the Application Configurator user logs in by using the following URL, that user will see an empty screen with a Configure OPAM link.

http://managedserver_host:managedserver_port/oinav/opam

The Application Configurator user can use this link to let the Oracle Privileged Account Manager Console know where Oracle Privileged Account Manager server is running by providing the Oracle Privileged Account Manager server's host and port. When the Oracle Privileged Account Manager Console can successfully communicate with the Oracle Privileged Account Manager server, the Oracle Privileged Account Manager Console will be populated with content.

### Note:
Oracle Privileged Account Manager administrators and users will probably never have to use the Oracle Identity Navigator interface except during the initial set-up of Oracle Privileged Account Manager.

You are now ready to start using Oracle Privileged Account Manager.

For information about invoking and working with the Oracle Privileged Account Manager Console, refer to Chapter 4, "Starting and Using the Oracle Privileged Account Manager Console."

If you prefer using the Oracle Privileged Account Manager Command Line Tool (CLI), refer to Appendix A, "Working with the Command Line Tool."

If you prefer using the Oracle Privileged Account Manager RESTful interface, refer to Appendix B, "Working with Oracle Privileged Account Manager's RESTful Interface."

### 3.4 Administering Oracle Privileged Account Manager

The following table describes the basic workflows that are performed by Oracle Privileged Account Manager administrator users based on their different Admin Roles.

### Note:
An administrator with the Application Configurator Admin Role should have already configured a connection to the Oracle Privileged Account Manager servers. Refer to Section 5.2.2, "Configuring a Connection to the Oracle Privileged Account Manager Server" for more information.
The following steps describe the basic workflow of a Self-Service user with no administrator privileges:

1. View accounts
2. Search for an account
3. Check out accounts
4. View checked out accounts
5. Check in accounts
6. Check out a session
7. View checked out sessions
8. Check in a session
9. View an account password

**Note:** Refer to Chapter 12, "Working with Self-Service" for detailed information about how to perform these tasks.
This chapter describes how to start and work with Oracle Privileged Account Manager’s web user interface, known as the Console.

This chapter includes the following sections:

- Section 4.1, "Before You Begin"
- Section 4.2, "Invoking Oracle Privileged Account Manager's Web-Based Console"
- Section 4.3, "Navigating Oracle Privileged Account Manager's Console"

**Note:** You can also manage Oracle Privileged Account Manager from the command line or by using Oracle Privileged Account Manager's RESTful interface.

- Refer to Appendix A, "Working with the Command Line Tool" for information about using the command line tool.
- Refer to Appendix B, "Working with Oracle Privileged Account Manager's RESTful Interface" for information about using the RESTful interface.

### 4.1 Before You Begin

This chapter assumes that you have finished configuring Oracle Privileged Account Manager as described in Chapter 3, "Getting Started with Managing Oracle Privileged Account Manager."

### 4.2 Invoking Oracle Privileged Account Manager's Web-Based Console

You can access Oracle Privileged Account Manager's Console by opening a browser window and entering the following URL:

http://managedserver_host:managedserver_port/oinav/opam

When the Oracle Privileged Account Manager page displays with the Sign In screen, log in with the appropriate administrator or end user credentials.
4.3 Navigating Oracle Privileged Account Manager's Console

When you log in to Oracle Privileged Account Manager, the Console displays.

Access to certain features in the Console is based on your administration role (Admin Role) and credentials. For example, Figure 4–1 shows all of the features available in Oracle Privileged Account Manager. However, the Administration, Reports, and Configuration accordions, described later in this section, are not available to end users or to users with the Security Administrator role.

![Oracle Privileged Account Manager Console (Full Privileges View)](image)

Figure 4–2 shows the Console when you log in as a Self-Service user with no administrator privileges.
This section provides a high-level overview of the Oracle Privileged Account Manager Console. The topics in this section include:

- Working with the Home Accordion
- Working with the Administration Accordion
- Working with the Reports Accordion
- Working with the Configuration Accordion
- Working with the Search Portlet
- Working with a Search Results Table

**Tip:** Hover your mouse over elements in the Oracle Privileged Account Manager interface (such as parameter fields or information icons) to see helpful prompts.

### 4.3.1 Working with the Home Accordion

The Home accordion contains the following nodes:

- **My Accounts**: Select this node to access the My Accounts page where you can search, view, open, and check out accounts where you are a grantee.

- **My Checkouts**: Select this node to access the My Checkouts page where you can view your checked out accounts, view the password for those accounts, and check in your checked out accounts.

You must check out a privileged account to use it. Oracle Privileged Account Manager enables you to check out an account as a password or as a session. Refer to Section 8.5, “Checking Out Privileged Accounts” for more information.

Clicking either node opens a new page on the right side of the Console. Use these pages to manage your accounts.

---

**Note:** Refer to Section 2.3, “Understanding Oracle Privileged Account Manager Authorization” for more information about Admin Roles.
4.3.2 Working with the Administration Accordion

Based on your Admin Role and credentials, the Administration accordion contains some or all of the following nodes:

- **Accounts**: Select to open the Accounts page, where you can search, open, add, and remove accounts.
- **Targets**: Select to open the Targets page, where you can search, open, add, and remove targets.
- **Password Policies**: Select to open the Password Policies page, where you can search, open, create, and delete Password Policies.
- **Usage Policies**: Select to open the Usage Policies page, where you can search, open, create, and delete Usage Policies.
- **User Grantees**: Select to open the User Grantees page, where you can search, open, and view information about individual user grantees.
- **Group Grantees**: Select to open the Group Grantees page, where you can search, open, and view information about a group of grantees.

Clicking any of these nodes opens a new page on the right side of the Console. Use these pages to configure and manage Oracle Privileged Account Manager.

**Note:**
- For detailed information about configuring and managing Oracle Privileged Account Manager, refer to Chapter 3, "Getting Started with Managing Oracle Privileged Account Manager."
- For detailed information about configuring and managing an Oracle Privileged Account Manager server, refer to Section 5.2, "Managing an Oracle Privileged Account Manager Server."

4.3.3 Working with the Reports Accordion

Based on your Admin Role and credentials, the Reports accordion contains some or all of the following nodes:

- **Deployment Reports**: Select to open the Deployment Reports page, where you can view information about how targets and privileged accounts are currently deployed.
- **Usage Reports**: Select to open the Usage Reports page, where you can view information about how privileged accounts are being used in your deployment.
- **Failure Reports**: Select to open the Failure Reports page, where you can view information about the current state of target and account failures.
4.3.4 Working with the Configuration Accordion

Based on your Admin Role and credentials, the Configuration accordion contains some or all of the following nodes, which represent the common global configuration properties that apply to all Oracle Privileged Account Manager servers in a cluster:

- **Server Connection**: Select to configure a connection to the Oracle Privileged Account Manager server.
  
  **Note**: Refer to Section 5.2.2, "Configuring a Connection to the Oracle Privileged Account Manager Server" for more information.

- **Server Configuration**: Select to manage the following server properties:
  - Usage Policy scheduler interval
  - Password Policy scheduler interval
  - Target connection timeout in seconds
  - Oracle Database TDE Mode (Transparent Data Encryption)

  **Note**: Refer to Section 5.2.3, "Managing Oracle Privileged Account Manager Server Properties" for more information.

- **Plug-in Configuration**: Select to create, edit, and manage plug-in configurations for Oracle Privileged Account Manager.

  **Note**: Refer to Chapter 11, "Working with Plug-Ins" for more information.

- **Session Manager Configuration**: Select to configure the Session Manager properties, configure Oracle Privileged Account Manager server URLs, and SSH configuration.

  **Note**: Refer to Section 5.3.3, "Managing the Oracle Privileged Session Manager Properties" for more information.

4.3.5 Working with the Search Portlet

Use Oracle Privileged Account Manager’s Search portlet to search for accounts, targets, policies, users, groups, and plug-ins.

You configure searches by using one or more of the parameters displayed in the portlet. The availability of different search parameters depends on the type of search you are going to perform. For example, Figure 4–3 shows the Search Accounts portlet that you use to search for privileged accounts.
The following table describes the different search parameters and for which search types they are available:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
<th>Search Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Name</td>
<td>Enter one or more letters of the account name.</td>
<td>Accounts, My Accounts, Checkout History</td>
</tr>
<tr>
<td>Target Name</td>
<td>Enter one or more letters of the target name.</td>
<td>Accounts, My Accounts, Targets</td>
</tr>
<tr>
<td>Target Type</td>
<td>Select All to search all target types or limit the search to only ldap, unix, database, or lockbox target types.</td>
<td>Accounts, My Accounts, Targets</td>
</tr>
<tr>
<td>Domain</td>
<td>Enter one or more letters of the domain name.</td>
<td>Accounts, My Accounts, Targets</td>
</tr>
<tr>
<td>Description</td>
<td>Enter one or more letters of the account, target, or plug-in description.</td>
<td>Accounts, My Accounts, Plug-in Configuration</td>
</tr>
<tr>
<td>Host Name</td>
<td>Enter one or more letters of the host name on which to search.</td>
<td>Targets</td>
</tr>
<tr>
<td>Policy Name</td>
<td>Enter one or more letters of the policy name.</td>
<td>Password Policies, Usage Policies</td>
</tr>
<tr>
<td>Policy Status</td>
<td>Select All to search all policies or limit the search to only Active or only Disabled policies.</td>
<td>Password Policies, Usage Policies</td>
</tr>
<tr>
<td>User Name</td>
<td>Enter one or more letters of the user name.</td>
<td>User Grantees, Checkout History</td>
</tr>
<tr>
<td>Group Name</td>
<td>Enter one or more letters of the group name.</td>
<td>Group Grantees</td>
</tr>
<tr>
<td>Start Date and End Date</td>
<td>Use the Calendar/Time icon 🕒 to specify a date range and time in which to search.</td>
<td>Checkout History</td>
</tr>
<tr>
<td>Pattern</td>
<td>Enter one or more characters of a string in the recording of a checkout event. For example, sync:x:5:0:sync:/sbin:/bin/sync</td>
<td>Checkout History</td>
</tr>
<tr>
<td>Query Size</td>
<td>Use the counter to limit how many query results are returned.</td>
<td>Checkout History</td>
</tr>
<tr>
<td>Name</td>
<td>Enter one or more letters of plug-in name.</td>
<td>Plug-in Configuration</td>
</tr>
<tr>
<td>Resource Type</td>
<td>Select All to search all resource types or limit the search to only account, only server, or only target resource types.</td>
<td>Plug-in Configuration</td>
</tr>
<tr>
<td>Status</td>
<td>Select All to search all plug-in statuses or limit the search to only Active or only Disabled plug-ins.</td>
<td>Plug-in Configuration</td>
</tr>
<tr>
<td>Timing</td>
<td>Select All to search all plug-in timings or limit the search to only pre timing plug-ins or only post timing plug-ins.</td>
<td>Plug-in Configuration</td>
</tr>
<tr>
<td>Operation</td>
<td>Select All to search all plug-in operations or limit the search to only add, autotickin, checkin, checkout, passwordcycle, remove, resetpassword, retrieve, sessioncheckout, showpassword, showpassordhistory, test, or update operations.</td>
<td>Plug-in Configuration</td>
</tr>
</tbody>
</table>
The Search Portlet also supports the use of wildcards, as follows:

- Use the percentage symbol (%) to search for character strings of any length. You can also use multiple wildcards in the same search string. For example,
  - If you enter person%, then the results might include person1, person_2, and person1234.
  - If you enter %person%, then the results might include dperson, hrperson1, and hrperson2.

- Use an underscore symbol (_) to search for a single character. You can also use multiple wildcards in the same search string. For example,
  - If you enter person_, then the results might include person1, person2, and persons.
  - If you enter o_m_, then the results might include oam1, oem1, oem2, oem3, and oim1.

The general steps for performing a search are as follows:

1. Select the appropriate node in the Home, Administration, Reports, or Configuration accordion.
   For example, to search for an account, select Accounts.

2. When the Search portlet displays, configure a search as follows:
   - To search for all available results, such as all accounts, do not specify any search parameters in the portlet.
   - To refine your search, use one or more of the search parameters described in Table 4–1.
     For example, to see a list of the privileged accounts on a particular LDAP target, enter one or more letters of the target’s name in the Target Name field and select ldap from Target Type menu.

3. Click Search.
   The results are displayed in the Search Results table.

   **Note:** You can use the View menu, located above the Search Results table, to manage how the search results are displayed in the table. Refer to Table 4–2 in Section 4.3.6, "Working with a Search Results Table" for more information.

4. To perform another search, click Reset.

**4.3.6 Working with a Search Results Table**

You can use the drop-down menus and icons located along the top of the different Search Results tables to perform various tasks.
The following table describes these menus and icons:

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Search Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actions</td>
<td>All</td>
<td>Click this menu and select an action to perform. Note: The options on this menu duplicate the task icons displayed above the table.</td>
</tr>
<tr>
<td>View</td>
<td>All</td>
<td>Click this menu and select one of the following options to control how columns are displayed in the Search Results table:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>Columns &gt; Show All</strong>: Displays all columns in the table.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>Columns &gt; Column Name</strong>: Click a column name to display or hide that column in the table. The columns are displayed (checked) by default.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>Columns &gt; Manage Columns</strong>: Provides a dialog that enables you to display or hide columns.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>Reorder Columns</strong>: Select this option and the Reorder Columns dialog displays. Use this dialog to select the columns and shift their order in the table.</td>
</tr>
<tr>
<td>Open</td>
<td>All</td>
<td>Click to open the selected account, target, policy, user grantee, group grantee, or plug-in configuration.</td>
</tr>
<tr>
<td>Password Check Out</td>
<td>My Accounts</td>
<td>Select a row in the Search Results table and click this option to check out the account’s password.</td>
</tr>
<tr>
<td>Session Check Out</td>
<td>My Accounts</td>
<td>Select a row in the Search Results table and click this option to check out a session.</td>
</tr>
</tbody>
</table>
**Table 4–2 (Cont.) Search Results Table Features**

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Search Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Refresh</td>
<td>My Accounts, My Checkouts, Accounts, Checkout History, Plug-in Configuration</td>
<td>Click to re-display (refresh) the Search Results.</td>
</tr>
<tr>
<td>Check In</td>
<td>My Checkouts only</td>
<td>Click to check in the selected checked-out account. Refer to Section 8.6, &quot;Checking In Privileged Accounts&quot; for more information.</td>
</tr>
<tr>
<td>Show Password</td>
<td>My Checkouts, Accounts, Targets</td>
<td>Click to open the Show Current Password dialog where you can view the current password information about a selected account or target service target.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ For Accounts, this dialog lists the current Account Name and Password.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ For Targets, this dialog lists the current Target Name, Service Account Name, Current Password, and Password Change Time.</td>
</tr>
<tr>
<td>Password History</td>
<td>Accounts, Targets</td>
<td>Click to open the Show Password History dialog where you can view the password history for an account or a target.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ For Accounts, this dialog lists the current Account Name, Password, and Modification Time (date and time).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ For Targets, this dialog lists the Target Name, Passwords, and Modification Time (date and time).</td>
</tr>
<tr>
<td>Status</td>
<td>Accounts only</td>
<td>Click this menu and select one of the following options to limit which account results are displayed in the table:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ All: Lists all accounts on the target.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Checked-in Accounts: Lists only those accounts that are currently checked-in.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ Checked-out Accounts: Lists only those accounts that are currently checked-out.</td>
</tr>
<tr>
<td>Add</td>
<td>Accounts, Targets</td>
<td>Click to add a new account or a new target to the Oracle Privileged Account Manager repository.</td>
</tr>
<tr>
<td>Remove</td>
<td>Accounts, Targets</td>
<td>Click to remove the selected account or target from the Oracle Privileged Account Manager repository.</td>
</tr>
<tr>
<td>Reset Password</td>
<td>Accounts, Targets</td>
<td>Click to open the Reset Password dialog where you can manually reset the password for a selected account or target service account.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ For Accounts, this dialog lists the current Account Name and Target Name. Type a password in the New Password field to create a new password for the account.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>■ For Targets, this dialog lists the current Target Name and Service Account Name. You can either type a password in the New Password field or enable the Generate password automatically checkbox to automatically generate a new password.</td>
</tr>
<tr>
<td>Force Check In</td>
<td>Accounts only</td>
<td>Click to check in privileged accounts that have been checked-out by other users.</td>
</tr>
<tr>
<td>Create Password Policy</td>
<td>Password Policies only</td>
<td>Click to create a Password Policy. Refer to Section 9.2.4, &quot;Creating a Password Policy&quot; for more information.</td>
</tr>
<tr>
<td>Create Usage Policy</td>
<td>Usage Policies only</td>
<td>Click to create a Usage Policy. Refer to Section 9.3.4, &quot;Creating a Usage Policy&quot; for more information.</td>
</tr>
</tbody>
</table>
### Search Results Table Features

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Search Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete</td>
<td>Password Policies, Usage Policies, Plug-in Configuration</td>
<td>Click to delete a selected policy from the Oracle Privileged Account Manager repository.</td>
</tr>
<tr>
<td>Create</td>
<td>Plug-in Configuration</td>
<td>Click to create a plug-in configuration. Refer to Section 11.3, &quot;Creating a Plug-In Configuration&quot; for more information.</td>
</tr>
<tr>
<td>Recording</td>
<td>Checkout History</td>
<td>Click to view a recording, in transcript format, of the actions taken during an account checkout.</td>
</tr>
</tbody>
</table>
This chapter provides information that administrators must know to configure and manage an Oracle Privileged Account Manager server and an Oracle Privileged Session Manager (Session Manager) server.

This chapter includes the following sections:

- Section 5.1, "Understanding the Servers"
- Section 5.2, "Managing an Oracle Privileged Account Manager Server"
- Section 5.3, "Managing the Oracle Privileged Session Manager Server"

### 5.1 Understanding the Servers

This section provides a high-level overview of the following servers:

- Oracle Privileged Account Manager Server
- Oracle Privileged Session Manager Server

#### 5.1.1 Oracle Privileged Account Manager Server

The Oracle Privileged Account Manager server implements the core functionality of Oracle Privileged Account Manager and makes authorization decisions that determine:

- Which targets and privileged accounts are exposed to administrators and end-users
- Which operations administrators and end-users can perform on targets, privileged accounts, and policies

In addition, the Oracle Privileged Account Manager server

- Supports Usage and Password Policies for accounts
- Enforces its authorization decisions
- Supports authentication by using the SAML-based Oracle Security Token from OPSS Trust Services and HTTP-Basic Authentication

Note: If you are using Oracle Privileged Account Manager on IBM WebSphere, refer to "Differences in Configuring and Managing the Servers" in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management for information about this topic.
Supports different Admin Roles for the Oracle Privileged Account Manager server

---

**Note:** For security purposes, the Oracle Privileged Account Manager server only responds to SSL traffic.

When you add the Oracle Privileged Account Manager server target to the Oracle Privileged Account Manager user interface or to the Oracle Privileged Account Manager command line tool (CLI), you must provide the SSL endpoint as `https://hostname:sslport/opam`.

By default, WebLogic responds to SSL using port 7002 on the Admin Server and port 18102 on the Managed Server. You can use the WebLogic console to check the port for your particular instance.

The following figure illustrates the Oracle Privileged Account Manager server architecture.

*Figure 5–1  Server Architecture*

![Server Architecture Diagram]

---

### 5.1.2 Oracle Privileged Session Manager Server

The Oracle Privileged Session Manager creates a single access point to target resources and enables you to manage privileged sessions to the target system through

- **Session Initiation** by
  - Providing a single control point for privileged access
  - Never exposing privileged credentials
  - Supporting any compliant, third-party clients (such as Putty, OpenSSH, etc.)

- **Session Control** by providing control through policy-based and administrator-initiated session termination and lockout.

- **Session Monitoring and Auditing** by maintaining historical records (transcripts) to support forensic analysis and audit data.

The following figure illustrates how the Oracle Privileged Session Manager relates to the Oracle Privileged Account Manager server.
5.2 Managing an Oracle Privileged Account Manager Server

This section provides information administrators need to manage an Oracle Privileged Account Manager server, which includes the following topics:

- **Before You Begin**
- Configuring a Connection to the Oracle Privileged Account Manager Server
- Managing Oracle Privileged Account Manager Server Properties

5.2.1 Before You Begin

- You must be an Oracle Privileged Account Manager administrator with the Application Configurator Admin Role to add and manage an Oracle Privileged Account Manager server.

**Note:** For more information about this Admin Role, refer to Section 2.3.1, "Administration Role Types" and Section 3.3.4, "Assigning the Application Configurator Role to a User."

- The procedures described in this chapter reference information and instructions contained in the following Oracle publications. If necessary, review the referenced concepts, terminology, and procedures before you begin configuring the Oracle Privileged Account Manager server.

<table>
<thead>
<tr>
<th>Table 5–1 Reference Publications</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>For Information About</strong></td>
</tr>
<tr>
<td>Admin Roles</td>
</tr>
<tr>
<td>Oracle WebLogic Server concepts and terminology</td>
</tr>
</tbody>
</table>

Figure 5–2  How Session Manager Relates to the Oracle Privileged Account Manager Server

[Diagram of session manager relationships]
5.2.2 Configuring a Connection to the Oracle Privileged Account Manager Server

When you log into Oracle Privileged Account Manager, the Oracle Privileged Account Manager Server URL is automatically detected by default.

Use the following steps to configure a new connection to the Oracle Privileged Account Manager server from the Oracle Privileged Account Manager Console:

1. Open Oracle Privileged Account Manager by logging in to:
   http://managedserver_host:managedserver_port/oinav/opam

   **Note:** You must log in as a user with the Application Configurator Admin Role, or the Server Configuration page will not be accessible.

   For more information about this, and other, Admin Roles refer to Section 2.3.1, "Administration Role Types" and Section 3.3.4, "Assigning the Application Configurator Role to a User."

2. When the Oracle Privileged Account Manager Console displays, select Server Connection from the Configuration accordion.

3. When the Server Connection page displays, notice that the Oracle Privileged Account Manager Server URL is displayed as the Auto-Detect URL.

   To add a different server, enter that server's Host name and SSL Port number.

   **Note:** You must provide a fully qualified host name for the Host value. Using localhost can cause problems, such as described in Section C.3.13, "Cannot Open Session Recordings."

4. Click the Test button to test the connection settings.

   If the server configuration tested successfully, you should see a "Test Succeeded" message.

5. Click the Apply button to save this connection information.

5.2.3 Managing Oracle Privileged Account Manager Server Properties

You can use the Console or properties in the OPAM Global Config configuration entry to define server-level behavior for activities such as scheduler intervals, timeouts, etc. The available server properties are explained in detail in Section 5.2.3.1.

You can manage server properties defined in the OPAM Global Config configuration entry from two locations:
5.2.3.1 From the Console

Use the following steps to manage the Oracle Privileged Account Manager server properties from the Oracle Privileged Account Manager Console:

1. Open Oracle Privileged Account Manager by logging in to:

   http://managedserver_host:managedserver_port/oinav/opam

   **Note:** You must log in as a user with the Application Configurator Admin Role, or the Server Configuration page will not be accessible.

   For more information about this, and other, Admin Roles refer to Section 2.3.1, "Administration Role Types" and Section 3.3.4, "Assigning the Application Configurator Role to a User."

2. When the Oracle Privileged Account Manager Console displays, select Server Configuration from the Configuration accordion.

3. When the Server Configuration page displays, you can modify any of the following server property options:

   - **Usage policy enforcement interval in seconds.** Specify an interval (in seconds) in which Oracle Privileged Account Manager checks accounts and then automatically checks-in the accounts that have exceeded the expiration time defined in the Usage Policy. (Default is 3600 seconds)

   - **Password policy enforcement interval in seconds.** Specify an interval (in seconds) in which Oracle Privileged Account Manager checks and then resets the password for any accounts that have exceeded the maximum password age defined in the Password Policy. (Default is 3600 seconds)

   - **Target connection timeout in seconds.** Specify an interval (in seconds) in which Oracle Privileged Account Manager allows an ICF connector to wait for a response from the target system to which it is connecting.

     The default value for this setting is 20 seconds, but in some deployments where network latency is high and target systems take longer to respond, you may need to increase this value.

   - **Require TDE enabled backend.** Check this box to enable Oracle Privileged Account Manager to use Transparent Data Encryption (TDE) mode. (Default is TDE mode enabled.)

     Enabling TDE ensures that all sensitive information stored by Oracle Privileged Account Manager (such as account passwords) is encrypted on disk.

     Unchecking the box disables TDE mode.

     **Note:** Oracle strongly recommends that you enable TDE mode for enhanced security.

     Refer to Section 2.4.6, "Hardening the Back-End Oracle Privileged Account Manager Database" for more information about using TDE mode.
4. When you are finished, click the **Apply** button to save these configuration settings.

### 5.2.3.2 From the Command Line

To access the OPAM Global Config configuration entry and modify these server properties, use the `getconfig` and the `modifyconfig` commands from the command line.

**Note:** Refer to Section A.2.1, "getconfig Command" and Section A.2.3, "modifyconfig Command" for detailed information about using these commands.

Refer to Section 15.2, "Securing Data On Disk" for more information about enabling or disabling TDE mode from the command line.

### 5.3 Managing the Oracle Privileged Session Manager Server

This section provides information administrators need to manage a Session Manager Server, which includes the following topics:

- **Before You Begin**
- **Configuring a Connection to the Oracle Privileged Session Manager Server**
- **Managing the Oracle Privileged Session Manager Properties**

### 5.3.1 Before You Begin

- You must be an administrator with the **Application Configurator Admin Role** or the **Security Administrator role** to view the Session Manager Configuration page.
- Only administrators with the **Application Configurator Admin Role** can modify any of the settings on the Session Manager Configuration page.

**Note:** For more information about these Admin Roles refer to Section 2.3.1, "Administration Role Types" and Section 3.3.4, "Assigning the Application Configurator Role to a User."

### 5.3.2 Configuring a Connection to the Oracle Privileged Session Manager Server

Use the following steps to configure the Oracle Privileged Session Manager server from the Oracle Privileged Account Manager Console:

1. Open Oracle Privileged Account Manager by logging in to:
   
   `http://managedserver_host:managedserver_port/oinav/opam`

2. When the Oracle Privileged Account Manager Console displays, select **Session Manager Configuration** from the Configuration accordion.

   Use the properties on the Session Manager Configuration page to configure the Session Manager. Refer to Section 5.3.3, "Managing the Oracle Privileged Session Manager Properties" for instructions.

**Note:** You cannot run two instances of Oracle Privileged Session Manager on the same machine.
5.3.3 Managing the Oracle Privileged Session Manager Properties

Use the following steps to manage the Session Manager properties from the Oracle Privileged Account Manager Console:

---

**Note:**

- You can also configure Session Manager properties by using the Oracle Privileged Account Manager RESTful interface. Refer to Appendix B, "Working with Oracle Privileged Account Manager’s RESTful Interface" for more information.
- You cannot use the Oracle Privileged Account Manager Command Line Tool (CLI) to configure Session Manager properties.

---

1. Open Oracle Privileged Account Manager and navigate to the Session Manager Configuration page as described in Section 5.3.2, "Configuring a Connection to the Oracle Privileged Session Manager Server."

2. When the Session Manager Configuration page displays, configure the following options:

   - **Session Monitoring Update Interval in seconds.** Specify an interval (in seconds) in which Session Manager checks all checked-out sessions and updates their transcripts. Session Manager automatically terminates any sessions that have exceeded the expiration time defined in the Usage Policy. (Default is 60 seconds.)

   - **Max recording size per Session.** Specify the maximum recording size that is allowed per session (in KB). This recording size limits how much data a user can use after checking out a session. When this quota is reached, the session is automatically terminated. (Default is 10240)

   - **Oracle Privileged Account Manager URLs.** Use this table to manage an array of Oracle Privileged Account Manager servers to which Session Manager can connect:

     **Note:** Notice that the Oracle Privileged Account Manager Server URL is displayed by default in the first row of the table, as the Auto-Detect URL. Clicking the Add button removes the Auto-Detect URL. After adding one or more rows to the table, you must click Remove and remove all rows to use the Auto-Detect URL instead. The Auto-Detect URL only displays when the table is empty.

     The Oracle Privileged Account Manager Server URL is multi-valued to allow for High Availability (HA).

     Session Manager maintains the server list and, when required, uses it on a round-robin basis for connections to Oracle Privileged Account Manager. Connection attempts are made against all configured servers until one succeeds or all configured URLs are exhausted.

     - To add one or more Oracle Privileged Account Manager Server URLs, click Add.
When the new row is displayed in the table, enter the URL of an Oracle Privileged Account Manager server into the blank field. For example, 
https://<opamserver_host>:<port>/opam

- To delete one or more Oracle Privileged Account Manager Server URLs from the table, select the row and click Remove.

■ SSH Configuration. Use the following options to configure the connection details to be displayed for session checkouts:

- **Listener Port**: Provide the reserved SSH port on which the Session Manager listener protocol is listening. The value must be greater than 1024 and it defaults to 1222.

- **Session Checkout Instructions**: Enter an instruction message to be displayed when users check out a session. This message should describe the information a user must provide to connect to the Session Manager server by using a regular SSH client.

  For example:

  ```
  ssh -p <port> <opamuser>:<targetname>:<accountname>@<sessionmgrhost>  
  Use opam password on password prompt
  ```

3. When you are finished, click the **Apply** button to save these configuration settings.

4. Restart Oracle Privileged Account Manager.

---

**Note:** Anytime you change the Session Manager configuration, you must restart Oracle Privileged Account Manager for the changes to take effect.

For the detailed instructions you need to check out and check in sessions, refer to *Section 12.7, "Checking Out Privileged Account Sessions."*
This chapter describes the different tasks you can perform when working with targets in Oracle Privileged Account Manager.

This chapter includes the following sections:

- **Section 6.1, "What Are Targets?"**
- **Section 6.2, "Adding Targets to Oracle Privileged Account Manager"**
- **Section 6.3, "Searching for Targets"**
- **Section 6.4, "Opening a Target"**
- **Section 6.5, "Managing a Target's Service Account Password"**
- **Section 6.6, "Removing Targets from Oracle Privileged Account Manager"**

---

**Note:** You can also use Oracle Privileged Account Manager’s command line tool or Oracle Privileged Account Manager’s RESTful interface to perform many of the tasks described in this chapter.

If you prefer using these interfaces instead of the Oracle Privileged Account Manager Console, refer to Appendix A, "Working with the Command Line Tool" or Appendix B, "Working with Oracle Privileged Account Manager’s RESTful Interface" for instructions.

---

**Note:** You must be an Oracle Privileged Account Manager administrator with the Security Administrator Admin Role to add, edit, or remove targets.

---

### 6.1 What Are Targets?

A target is a software system that contains, uses, and relies on user, system, or application accounts.

You cannot create targets in, or delete targets from, your environment by using Oracle Privileged Account Manager. Rather, Oracle Privileged Account Manager manages existing targets that were provisioned using other mechanisms.

When you “add” a target in Oracle Privileged Account Manager, you are creating a reference to that target. In effect, you are registering the target and asking Oracle Privileged Account Manager to manage it. When you “remove” a target from Oracle Privileged Account Manager, you are only removing that reference.
Oracle Privileged Account Manager supports database, LDAP, lockbox, and UNIX target types.

A lockbox target provides password vault-like functionality in Oracle Privileged Account Manager. That is, it provides a secure mechanism for storing the passwords (or any kind of sensitive information) associated with privileged accounts in your deployment. This target type is different from the other, conventional Oracle Privileged Account Manager target types in the following ways:

- Oracle Privileged Account Manager does not interact with lockbox target systems. There is no connectivity to, or operations performed against, these systems.
- Oracle Privileged Account Manager does not manage the password lifecycle or reset passwords associated with accounts on lockbox targets.
- Password modifications are handled out-of-band and updated into Oracle Privileged Account Manager as an administrative action. Therefore, Oracle Privileged Account Manager does not randomize the passwords; but rather, they stored as given by the administrator.

A lockbox target may be preferable when you want to centrally store and securely grant privileged account passwords without having Oracle Privileged Account Manager automatically manage those accounts on the target systems. For example, if you want to control how and when the passwords on the those target systems are modified, as opposed to allowing Oracle Privileged Account Manager do so.

Additionally, a lockbox target may be useful when an appropriate ICF connector is unavailable for a specific target type, but you still want to manage access to that system through Oracle Privileged Account Manager.

### 6.2 Adding Targets to Oracle Privileged Account Manager

**Note:** When adding a target of any Target Type (except lockbox), you must configure a service account (also called an unattended account) with privileges that enable that account to

- Search for accounts on the target system
- Modify the passwords of accounts on the target system

You must never use the same account as a service account and as a privileged account to be managed by Oracle Privileged Account Manager.

For additional information about service accounts, see the description for attended and unattended accounts in Section 1.2.1, "Features" and refer to Chapter 7, "Working with Service Accounts."

**Note:** If you are using Oracle Privileged Account Manager on IBM WebSphere, refer to "Differences When Adding Targets to Oracle Privileged Account Manager on IBM WebSphere" in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management for information about this topic.

Use the following steps to add a target for Oracle Privileged Account Manager to manage:

1. Log in to Oracle Privileged Account Manager.
2. Select **Targets** from the Administration accordion to open the Targets page.

3. Click **Add**, located in the Search Results table toolbar to open a new Target: *Untitled* page displays with two tabs:
   - **General**. Contains two areas with parameters used to specify Basic Configuration and Advanced Configuration information for the target.
   - **Privileged Accounts**. Lists the privileged accounts currently being managed on the target and enables you to add, open, and remove the accounts that are managed by that target.

4. On the General tab, use the **Target Type** menu to select a target type (*database*, *ldap*, *lockbox*, or *unix*), and then set the remaining configuration parameters.

   **Note:** When you set the target type, the Target: *Untitled* page refreshes and the configuration parameters change, based on your selection.

   The following sections describe the available parameters for each target type:
   - **Section 6.2.1, "database Target Type Parameters"**
   - **Section 6.2.2, "ldap Target Type Parameters"**
   - **Section 6.2.3, "lockbox Target Type Parameters"**
   - **Section 6.2.4, "unix Target Type Parameters"**

   You must specify all of the required attributes (indicated by an asterisk *) symbol.

5. After setting the target configuration parameters, click **Test** to check the target's configuration.

   If the configuration is valid, a "Test Succeeded" message displays.

6. Click **Save** to add your new target on the Oracle Privileged Account Manager server.

   Oracle Privileged Account Manager automatically assigns a Target GUID and you can view this read-only value at the bottom of the Basic Configuration parameters section.

   You can now associate this target with a privileged account. For instructions, proceed to **Section 8.2, "Adding Privileged Accounts into Oracle Privileged Account Manager."**

### 6.2.1 database Target Type Parameters

When you select the **database** target type, the basic and advanced configuration parameters display. These parameters are described in the following tables:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Name</td>
<td>Enter a name for the new target.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a description for this target.</td>
</tr>
<tr>
<td>Organization</td>
<td>Enter the name of an organization to associate with the target.</td>
</tr>
<tr>
<td>Domain</td>
<td>Enter the domain of the target server.</td>
</tr>
</tbody>
</table>
Adding Targets to Oracle Privileged Account Manager

Table 6–1  (Cont.) Basic Configuration Parameters for the database Target Type

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Password Policy</td>
<td>Select a Password Policy to apply to the target’s service account. Oracle Privileged Account Manager uses this policy to auto-generate passwords.</td>
</tr>
</tbody>
</table>
| Enable Password Rollover | Enable this box to allow Oracle Privileged Account Manager to automatically change (rollover) the service account password for this target to a randomized value according to the **Expire password after** setting that is specified in the assigned Password Policy.  

**Note**: Password rollover for target service accounts is similar to password expiration for privileged accounts. If a password has not been changed by the expiration date configured in the associated Password Policy, then Oracle Privileged Account Manager will automatically change the password to a randomized value. |
| Host                 | Enter the host name of the target server.                                                                                                     |
| Database Connection  | Enter the JDBC URL used to identify the target system location. For example, **Oracle:jdbc:oracle:thin:@<host>:<port>:<sid>**  

**Note**: Oracle Privileged Account Manager supports the Oracle, MSSQL, Sybase, and MySQL database types.  
Refer to the *Oracle Identity Manager Connector Guide for Database User Management* for information about which special options are supported.
The following Advanced Configuration parameter is optional:

### Table 6–2 Advanced Configuration Parameters for the database Target Type

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Connection Properties</strong></td>
<td>Enter connection properties to use while configuring a secured connection. These properties must be name-value pairs given in following format: prop1=val1#prop2=val2</td>
</tr>
</tbody>
</table>

#### 6.2.2 Idap Target Type Parameters

When you select the Idap target type, the basic and advanced configuration parameters display. These parameters are described in the following tables:
## Table 6–3  Basic Configuration Parameters for the ldap Target Type

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Name</td>
<td>Enter a name for the new target.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a description for this target.</td>
</tr>
<tr>
<td>Organization</td>
<td>Enter the name of an organization to associate with the target.</td>
</tr>
<tr>
<td>Domain</td>
<td>Enter the domain of the target server.</td>
</tr>
<tr>
<td>Password Policy</td>
<td>Select a Password Policy to apply to the target’s service account. Oracle Privileged Account Manager uses this policy to auto-generate passwords.</td>
</tr>
<tr>
<td>Host</td>
<td>Enter the host name of the target server.</td>
</tr>
<tr>
<td>TCP Port</td>
<td>Enter the TCP/IP port to use when communicating with the LDAP server. You can use the up/down arrow icons to increment this value.</td>
</tr>
<tr>
<td>SSL</td>
<td>Enable this box to use Secure Socket Layer (SSL) when connecting to the LDAP server. <strong>Note:</strong> For SSL connectivity, you must import an SSL certificate to the J2EE container hosting Oracle Privileged Account Manager. For more information, refer to Section 15.1, &quot;Configuring Oracle Privileged Account Manager to Communicate With Target Systems Over SSL.&quot;</td>
</tr>
<tr>
<td>Principal</td>
<td>Enter the distinguished name (DN) to use when authenticating to the LDAP server. For example, cn=admin</td>
</tr>
<tr>
<td>Password</td>
<td>Enter the user’s password.</td>
</tr>
<tr>
<td>Base Contexts</td>
<td>Enter one or more starting points in the LDAP tree to use when searching the tree for users on the LDAP server or when looking for groups where the user is a member. Use a pipe (</td>
</tr>
<tr>
<td>Account User Name Attribute</td>
<td>Enter the attribute to be used as the account’s user name. (Default is uid.)</td>
</tr>
</tbody>
</table>

These Advanced Configuration parameters are *optional*:

## Table 6–4  Advanced Configuration Parameters for the ldap Target Type

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uid Attribute</td>
<td>Enter the name of the LDAP attribute that is mapped to the Uid attribute.</td>
</tr>
<tr>
<td>LDAP Filter for Retrieving Accounts</td>
<td>Enter an LDAP filter to control which accounts are returned from the LDAP resource. If you do not specify a filter, Oracle Privileged Account Manager returns only those accounts that include all of the specified object classes.</td>
</tr>
<tr>
<td>Password Attribute</td>
<td>Enter the name of the LDAP attribute that holds the password. When changing a user's password, Oracle Privileged Account Manager sets the new password to this attribute.</td>
</tr>
<tr>
<td>Account Object Classes</td>
<td>Enter one or more object classes to use when creating new user objects in the LDAP tree. Type each object class on its own line. Do not use commas or semicolons to separate entries. Some object classes require that you specify them in their class hierarchy, using a pipe (</td>
</tr>
</tbody>
</table>
6.2.3 lockbox Target Type Parameters

When you select the lockbox target type, only the following basic configuration parameters display:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Name</td>
<td>Enter a name for the new target.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a description for this target.</td>
</tr>
<tr>
<td>Organization</td>
<td>Enter the name of an organization to associate with the target.</td>
</tr>
<tr>
<td>Domain</td>
<td>Enter the domain of the target server.</td>
</tr>
<tr>
<td>Host</td>
<td>Enter the host name of the target server.</td>
</tr>
</tbody>
</table>

Note: You can add configuration parameters to this list by editing the opam-config.xml file as described in Section 3.2.3, "Consuming ICF Connectors."

6.2.4 unix Target Type Parameters

When you select the unix target type, the basic and advanced configuration parameters display. These parameters are described in the following tables:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Name</td>
<td>Enter a name for the new target.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a description for this target.</td>
</tr>
<tr>
<td>Organization</td>
<td>Enter the name of an organization to associate with the target.</td>
</tr>
<tr>
<td>Domain</td>
<td>Enter the domain of the target server.</td>
</tr>
<tr>
<td>Password Policy</td>
<td>Select a Password Policy to apply to the target's service account. Oracle Privileged Account Manager uses this policy to auto-generate passwords.</td>
</tr>
<tr>
<td>Enable Password Rollover</td>
<td>Enable this box to allow Oracle Privileged Account Manager to automatically change (rollover) the service account password for this target to a randomized value according to the Expire password after setting that is specified in the assigned Password Policy. Note: Password rollover for target service accounts is similar to password expiration for privileged accounts. If a password has not been changed by the expiration date configured in the associated Password Policy, then Oracle Privileged Account Manager will automatically change the password to a randomized value.</td>
</tr>
<tr>
<td>Host</td>
<td>Enter the host name of the target server.</td>
</tr>
<tr>
<td>Port</td>
<td>Enter the port used to connect with the UNIX server. You can use the up/down arrow icons to increment this value. Note: Only the SSH protocol is supported. (Default port is 22)</td>
</tr>
<tr>
<td>Login User</td>
<td>Enter the user name to use when connecting to this target.</td>
</tr>
<tr>
<td>(Service Account)</td>
<td></td>
</tr>
</tbody>
</table>
The following Advanced Configuration parameters are \textit{optional}:

\begin{table}[h]
\centering
\begin{tabular}{|l|p{0.7\textwidth}|}
\hline
\textbf{Parameter Name} & \textbf{Description} \\
\hline
\textbf{Command timeout} & Specify how long (in milliseconds) to wait for the command to complete before terminating that command. \\
\hline
\textbf{Password Expect Expressions} & Specify the expressions displayed on the target when setting the user's password. For example, if the 	exttt{Enter password} and 	exttt{Re-enter password} expressions are displayed when you run the \texttt{passwd} command, then the value for this field can be \texttt{enter password, re-enter password}. \\
& \textbf{Note:} You can provide a regular expression here. Use a comma to separate the two expressions. \\
\hline
\textbf{Pre-password expectExpression} & When you run the \texttt{passwd} command on some targets, prompts can be displayed before the password prompts appear. Specify the prompt expression and the expected input value, using a comma to separate these values. \\
\hline
\textbf{sudo password expectExpression} & Specify the password prompt to be displayed when running a command in sudo mode. (Default value is \texttt{password}) \\
\hline
\end{tabular}
\end{table}

\textbf{6.3 Searching for Targets}

If you have administrator privileges, you can search for targets using the following criteria or a combination of these items:

\begin{itemize}
  \item Target Name
  \item Target Type (\texttt{All}, \texttt{database}, \texttt{ldap}, \texttt{lockbox}, or \texttt{unix})
  \item Host Name
  \item Domain
  \item Description
\end{itemize}

To search for a target,
1. Select Targets in the Administration accordion.

2. When the Targets tab displays, use the Search portlet parameters to configure your search. For example,
   - To search for all LDAP targets, select ldap from the Target Type menu.
   - To search for all available targets, do not specify any search parameters.

3. Click Search.
   Review your search results in the Search Results table.

6.4 Opening a Target

You can open a target to review and edit the target’s configuration parameters and its associated privileged account parameters.

Use one of the following methods to open a target:

- Click the Target Name (an active link) in the Search Results table.
- Select the target’s Row number and then click the Open icon.

The Target: TargetName page opens where you can access the target and privileged account information.

6.5 Managing a Target’s Service Account Password

Oracle Privileged Account Manager provides several options for managing a target’s service account passwords, including:

- Showing passwords
- Viewing password history
- Resetting passwords
- Enabling password rollover

Administrators with the Security Administrator Admin Role can perform these password management tasks by using the Oracle Privileged Account Manager Console, command line tool, or REST API.

---

**Note:**

- For information about managing passwords by using the Console, refer to Section 7.3, "Managing Service Account Passwords."
- For command line instructions, refer to Section A.4, "Working with Targets."
- For REST API instructions, refer to Section B.5, "Target Resource."

---

Oracle Privileged Account Manager audits password management actions to keep track of password access.

---

**Note:** The procedures for showing and resetting a privileged account password are different from the procedures described in this section. Refer to Section 8.8, "Managing Privileged Account Passwords" for information.
6.6 Removing Targets from Oracle Privileged Account Manager

To remove a target, select the target from the Search Results table and then click the **Remove** icon.

**WARNING:** When you remove a target, you also remove all information about the target that is stored in Oracle Privileged Account Manager (including privileged accounts).

Before removing a target, it is critical that you first capture all relevant information from that target. For example, save the target's service account password and any current passwords that are associated with the privileged accounts on the target.
This chapter provides background information about OPAM service accounts, including an example for creating those accounts.

The topics in this chapter include:

- Section 7.1, "Understanding Service Accounts"
- Section 7.2, "Creating Service Accounts"
- Section 7.3, "Managing Service Account Passwords"

### 7.1 Understanding Service Accounts

Before adding a target to Oracle Privileged Account Manager, you must configure an OPAM service account (also called an unattended account) for that target. OPAM service accounts (service accounts) enable Oracle Privileged Account Manager to connect to and manage target systems.

You use an OPAM service account to configure the credentials for a target system.

**Note:**
- Service accounts do not apply for lockbox-type targets.
- Never use the same account as a service account and a privileged account to be managed by Oracle Privileged Account Manager.

A service account must have sufficient privileges to perform all Oracle Privileged Account Manager-related operations on the target system, such as:

- Searching for and viewing details about the accounts in the target, which is used for all operations such as looking up and adding privileged accounts to the system, locating the account during checkout, etc.
- Changing account passwords in the target, which is used for operations involving password changes such as checkout, check-in, resetpassword, etc.
- Changing self password, which is used for resetting target service account passwords and changing the password of the service account itself.

### 7.2 Creating Service Accounts

This section provides information about creating a service account to use when connecting to a target system.
Managing Service Account Passwords

Note: Never use the same account as both a service account and a privileged account to be managed by Oracle Privileged Account Manager.

The methods for creating a service account and assigning privileges to that account depend on the target system. For example, the steps for creating accounts and assigning roles on an Oracle Database system are different from the steps for a UNIX operating system.

The following examples illustrate two methods for creating a service account:

Note: These examples are only provided as a reference. You can achieve the same result by using other means.

On an Oracle Database System:
1. Use SQLPLUS and connect as the sys user.
2. Run the following commands to create the opamsrvc account:
   
   ```
   connect sys/<password> as sysdba
   create user opamsrvc identified by <password>;
   grant connect, dba to opamsrvc
   ```

On a Linux System:
1. Use Linux and connect as root.
2. Run the following commands to create the opam_service account:
   
   ```
   $ useradd -d /home/opam_service -m -g root -G bin,daemon,sys,adm,disk,wheel -o -u 0 opam_service
   $ passwd opam_service
   ```

7.3 Managing Service Account Passwords

Oracle Privileged Account Manager provides the following options for managing a target’s service account passwords:

- Showing Service Account Passwords
- Viewing the Password History
- Resetting Service Account Passwords
- Understanding Service Account Password Rollover

Administrators with the Security Administrator Admin Role can perform these password management tasks by using the Oracle Privileged Account Manager Console, command line tool, or REST API.
Managing Service Account Passwords

7.3.1 Showing Service Account Passwords

If necessary, you can review the stored password for a target's service account by using the Show Password option, located above the Search Results table on the Targets page.

Use the following steps:

1. Select Targets in the Administration accordion.
2. When the Targets tab displays, use the Search portlet to locate the target.
3. Select the target row number and then click Show Password.

   The Show Current Password dialog displays and provides the following information about the target's service account password:

   - Target Name
   - Service Account Name
   - Current Password
   - Password Change Time

4. When you are finished, click Close.

Note:

- For command line instructions, refer to Section A.4, "Working with Targets."
- For REST API instructions, refer to Section B.5, "Target Resource."
- The procedures for showing and resetting a privileged account password are different from the procedures described in this section. Refer to Section 8.8, "Managing Privileged Account Passwords" for information.

Oracle Privileged Account Manager audits password management actions to keep track of password access.

7.3.2 Viewing the Password History

Use the Password History option to view the password history for a target's service account.
To view a target's password history,

1. Select **Targets** in the Administration accordion to open the Search Targets page, and then click **Search**.
2. Select the row number of the target.
3. When the **Password History** icon becomes active, click the icon.

   The Show Password History dialog displays with the Target Name, and the Password in clear text, and the Modification Time (date and time of the password reset).
4. When you are finished click **Close**.

### 7.3.3 Resetting Service Account Passwords

If necessary, you can manually reset the stored password for a target's service account by using the **Reset Password** option, located above the Search Results table.

**Note:** The Reset Password option is not applicable for the lockbox target type or the ldap target type and, if selected, it will return an "Operation not supported" error message.

Use the following steps:

1. Select **Targets** in the Administration accordion.
2. When the Targets tab displays, use the Search portlet to locate the target.
3. Select the target row number and then click **Reset Password**.

   The Reset Password dialog displays and provides the following information about the target's service account password:

   - Target Name
   - Service Account Name

   This dialog also contains two options for resetting the password:

   - **New Password**: Type a new password into the space provided.
   - **Generate password automatically**: Enable the checkbox to automatically generate a password, according to the account's Password Policy.

4. Type a new password or enable the checkbox, and then click **Reset**.

### 7.3.4 Understanding Service Account Password Rollover

When you create a service account for a target, the service account is governed by the target's Password Policy.

Password rollover for a target's service account is similar to password expiration for privileged accounts. If you enable password rollover for the service account, and the password has not been changed by the expiration date configured in the associated Password Policy, then Oracle Privileged Account Manager will automatically change the password to a randomized value.
Note: Refer to Section 6.2, "Adding Targets to Oracle Privileged Account Manager" for information about enabling password rollover for the different target types.
This chapter provides some background information about privileged accounts and describes how to work with those accounts using the Oracle Privileged Account Manager Console.

This chapter includes the following sections:

- Section 8.1, "What is a Privileged Account?"
- Section 8.2, "Adding Privileged Accounts into Oracle Privileged Account Manager"
- Section 8.3, "Searching for Privileged Accounts"
- Section 8.4, "Opening Privileged Accounts"
- Section 8.5, "Checking Out Privileged Accounts"
- Section 8.6, "Checking In Privileged Accounts"
- Section 8.7, "Viewing a Session Recording"
- Section 8.8, "Managing Privileged Account Passwords"
- Section 8.9, "Removing Privileged Accounts from Oracle Privileged Account Manager"

**Note:** You can also manage Oracle Privileged Account Manager accounts from the command line or by using Oracle Privileged Account Manager’s RESTful interface.

- For information about using the Oracle Privileged Account Manager Command Line Tool (CLI), refer to Section A.5, "Working with Accounts" in Appendix A, "Working with the Command Line Tool."
- For information about using the Oracle Privileged Account Manager RESTful interface, refer to Section B.6, "Account Resource" in Appendix B, "Working with Oracle Privileged Account Manager’s RESTful Interface."

### 8.1 What is a Privileged Account?

An account on a target is considered privileged in a deployment when that account

- Is associated with elevated privileges
- Is used by multiple end-users on a task-by-task basis
What is a Privileged Account?

- Requires its usage to be controlled and audited

You cannot create accounts in, or delete accounts from, your environment by using Oracle Privileged Account Manager. Oracle Privileged Account Manager only manages existing accounts that were provisioned using other mechanisms.

When you "add" an account in Oracle Privileged Account Manager, you are creating a reference to that account. In effect, you are registering the account and asking Oracle Privileged Account Manager to manage it. When you "remove" the account from Oracle Privileged Account Manager, you are only removing the reference to that account.

---

**Note:** Administrators determine which accounts are privileged within a particular deployment, and they must configure Oracle Privileged Account Manager to manage those accounts.

You must be an Oracle Privileged Account Manager administrator with the Security Administrator Admin Role to add and manage accounts.

---

Oracle Privileged Account Manager enables you to manage both system and application accounts.

This section contains the following topics:

- Managing System Accounts
- Managing Application Accounts
- Understanding Sharing Accounts

### 8.1.1 Managing System Accounts

Oracle Privileged Account Manager's primary purpose is to manage privileged system accounts on a supported target system. Oracle Privileged Account Manager does not mandate what constitutes a privileged system account — it can manage any account on a target system. Administrators are responsible for identifying which accounts are privileged. A privileged account is typically a system account that allows a user to perform administration tasks.

Privileged accounts are suitable for management through Oracle Privileged Account Manager if they are used and shared by multiple individuals in the organization and administrators are required to track the use of these accounts.

Administrators perform the following steps to register an account as a privileged account to be managed by Oracle Privileged Account Manager:

1. Add the target to Oracle Privileged Account Manager (if this has not already been done). Refer to Section 6.2, "Adding Targets to Oracle Privileged Account Manager" for instructions.

2. Add the identified privileged account to the target and assign a Password Policy. Refer to Section 8.2, "Adding Privileged Accounts into Oracle Privileged Account Manager" and Section 9.2.5, "Assigning Password Policies" for instructions.

3. Grant access to end users directly or by using LDAP roles/groups and assign a Usage Policy. Refer to Section 10.2, "Granting Accounts to Users" and Section 9.2.5, "Assigning Password Policies" for instructions.
8.1.2 Managing Application Accounts

Applications use application accounts to connect to target systems at run time. Traditionally, administrators set up these accounts once during installation and then they are forgotten. Consequently, application accounts can potentially cause hidden vulnerabilities in your deployment. For example, passwords might become less secure over time because they were created using outdated policies or commonly used deployment passwords might be compromised.

Oracle Privileged Account Manager enables you to better manage application accounts. In particular, for applications that store their application accounts in the Credential Store. These applications consume the account credentials at run time from the Credential Store through the Credential Store Framework.

For example, because an application account is essentially a special version of a system account, you can register an application account in Oracle Privileged Account Manager as described in Section 8.1.1, "Managing System Accounts." You can then add the corresponding CSF mappings for every application that depends on that account, which is how CSF uniquely identifies a credential stored within CSF, and how an application finds its credential in CSF. For more information about CSF mapping, refer to "Guidelines for the Map Name" in the Oracle Fusion Middleware Application Security Guide.

If you register an account’s CSF mappings with Oracle Privileged Account Manager, then every time the account’s password changes, Oracle Privileged Account Manager can update the CSF entries that correspond to the registered mappings to reflect the new password and the applications continue to work without service interruption.

---

**Note:** Oracle Privileged Account Manager updates, or synchronizes, CSF only when a password change occurs. Refer to Section 17.3, "Integrating with the Credential Store Framework" for information about integrating Oracle Privileged Account Manager with CSF.

You can also use the plug-in framework to synchronize passwords to non-CSF application wallets. You can write a plug-in on the passwordcycle and resetaccountpassword operations for the Server resource to capture all password update operations, and then add custom logic to synchronize the resource to your application wallet. Refer to Section 11.2.7, "Supported Operations and Timings" for more information.

---

Additionally, you can apply a Password Policy to these applications that periodically cycles the account password. Cycling the password ensures that the application accounts are always compliant with the latest corporate policies and they remain secure. Oracle Privileged Account Manager performs this task with no service interruption.

Finally, it’s useful to note that Oracle Privileged Account Manager can support an account as both a system account (shared and used by multiple end-users) and as an application account (only used by an application at run time) at the same time. In this configuration, a human end-user who’s been granted access can "check out" the application account to perform manual administrative operations as that application without disrupting application functionality.
8.1.3 Understanding Sharing Accounts

Oracle Privileged Account Manager enables you to specify whether an account is shared or not shared.

- **Shared accounts** enable multiple users to check out the account at the same time.
- **Unshared accounts** (Default) enable only one user to check out an account at a time.

Because unshared accounts are more secure, Oracle recommends that you designate an account as shared only if there are compelling business reasons to do so. If sharing is necessary, be sure to read Section 2.4.2, "Securing Shared Accounts."

---

**Note:** If you configure a shared account, be aware that a user can still use the password after checking in the account. Oracle Privileged Account Manager does not reset the account password until the last user checks in the account.

This is a security limitation for shared accounts.

---

8.2 Adding Privileged Accounts into Oracle Privileged Account Manager

**Note:** Accounts are always added to a target, so you must add a target object before you can add an account. Refer to Section 6.2, "Adding Targets to Oracle Privileged Account Manager" for more information.

Never use the same account as the service account and as a privileged account to be managed by Oracle Privileged Account Manager. Refer to Chapter 7, "Working with Service Accounts" for information about service accounts.

---

You can add a new privileged account from either of the following pages:

- **From the Accounts Page**
- **From the Targets Page**

**From the Accounts Page**

To add an account by using the Accounts page,

1. Select Accounts from the Administration accordion.
2. Click the Add icon located above the Search results table.

**From the Targets Page**

To add an account by using the Targets page,

1. Select Targets in the Administration accordion.
2. Use the Search Targets portlet to populate the Search Results table with a list of all available targets.
3. Locate the target where you want to add the account and open it by clicking the Target Name link.
4. When the Target: TargetName page displays, select the Privileged Accounts tab.
5. Click Add in the table toolbar.

In both cases, when you click Add, the Account: Untitled page displays with the following subtabs:

- General: Use to specify information needed to add the account.
- Grants: Use to associate users and groups (grantees) with the account.
- Credential Store Framework: Use to add or remove Credential Store Framework (CSF) mappings for the account.
- Checkout History: Use to search for, and view information about, any users who check out this account. (Refer to "From the Checkout History Tab" on page 8-15 for more information.)

Use these subtabs and the instructions provided in the following sections to finish adding the account:

- Adding the Account
- Adding Grantees
- Adding CSF Mappings

### 8.2.1 Adding the Account

To add an account you must complete the Step 1: Set Target and Step 2: Add Account sections on the General tab as follows:

**Set the Target**

1. Provide a Target Name and Target Type.
   - If a Target Name and a Target Type are already displayed, proceed to Step 1 in the Set the Account section.
   - If the either parameter is <undefined>, click the search icon.

2. When the Set Target dialog displays, enter a value in the Target Name field and click the Search button to locate the target where you want to add the account.
   
   For example, if you know the target name begins with "r," you can type an r into the Target Name field and click the Search button.

3. When the search results display in the Search Results table, select (check) the Row box next to a target name and then click Set.
   
   The selected Target Name and its Target Type are displayed on the General tab.

**Set the Account**

1. If the Account Name field is blank, click the search icon.

2. When the Set Account dialog displays, enter one or more letters in the Account Name field and click the Search button to locate the account you want to add.
Adding Privileged Accounts into Oracle Privileged Account Manager

For example, if you know the account name begins with "s," you can type an s into the Account Name field and click the Search button.

Note: When you add privileged accounts to a lockbox target, a Password field is also displayed in the Console.

Oracle Privileged Account Manager does not manage accounts on lockbox targets; therefore it cannot reset the passwords on those accounts. You must provide the password to be used when users check out those privileged accounts.

For more information about lockbox targets, refer to Section 6.1, "What Are Targets?"

3. When the search results display in the Search Results table, select (check) the Row box next to an account name and then click Set.

Note: You must not add the target's service account as a privileged account to be managed by Oracle Privileged Account Manager.

The selected account is displayed as the Account Name on the General tab.

4. Enable the Shared Account box to allow multiple users to check out this account at the same time.

Note: Refer to Section 8.1.3, "Understanding Sharing Accounts" and Section 8.5, "Checking Out Privileged Accounts" for more information.

5. Specify a Password Policy.

Note: Oracle Privileged Account Manager automatically assigns the Default Password Policy to new accounts. However, Oracle Privileged Account Manager administrators with the Security Administrator or the User Manager Admin Role can create new policies.

You can leave the default policy set or choose a different policy from the Password Policy drop-down menu.

For more information about policies, refer to Chapter 9, "Working with Policies."

6. Click Test to confirm that the account can be managed by Oracle Privileged Account Manager with these settings.

If the account configuration settings are valid, a "Test Succeeded" message displays.

7. Click Save.
A new Current Checkouts section is displayed at the bottom of the General tab page. The table in this section enables you to view the following:

- Which users currently have the account checked out
- Type of checkout (password or session)
- Checkout expiration date
- Recordings (or transcripts) related to the account checkout

In addition, if you are an administrator with the User Manager Admin Role, you can use the Force check-in option to check in accounts. Refer to Forcing a Check-In.

You can now add grantees and CSF mappings to the account. Continue to Section 8.2.2, "Adding Grantees" and Section 8.2.3, "Adding CSF Mappings" for more information.

### 8.2.2 Adding Grantees

This section provides instructions for adding grantees to a privileged account.

**Note:**

- You must be an Oracle Privileged Account Manager administrator with the User Manager Admin Role to add, edit, or delete grantees.
- Adding a new account does not automatically grant you access to that account. You must complete the process for adding yourself as a grantees.
- Before adding grantees to an account, be sure to read Section 2.4.4, "Avoiding Assignments through Multiple Paths."

To associate users and groups with a new account, select the Grants tab and then complete the following steps:

- To associate users, click Add from the Users table toolbar.
  1. In the Add Users dialog, enter one or more letters of a name into the User Name field and click the arrow icon to search for that user.
  2. When the search results display, select (check) each user you want to associate with this account.
  3. When you are finished adding users, click Add and then click Close.

    Oracle Privileged Account Manager adds those user names to the Users table on the Grants tab and automatically assigns the Default Usage Policy.

    To assign a different policy, select it from the Usage Policy menu.

- To associate groups, click Add from the Groups table toolbar.
  1. In the Add Group dialog, enter a name into the Group Name field and click the arrow icon to search for that group.
2. When the search results display, select (check) each group you want to associate with this account.

3. When you are finished adding groups, click Add and then click Close.

Oracle Privileged Account Manager adds those group names to the Groups table on the Grants tab and automatically assigns the Default Usage Policy.

To assign a different policy, select it from the Usage Policy menu.

---

### Note: Removing Grants

Removing a user or group grant from an account does not automatically cancel all existing checkouts.

When grantees check out an account, they are guaranteed access to that account until one of the following events occur:

- The grantee checks in the account
- Oracle Privileged Account Manager automatically checks in the account because the checkout duration has exceeded the expiration period specified by the account’s Usage Policy
- An administrator forces an account check-in

However, after the account is checked in, the grantee cannot check out that account again unless an administrator re-adds them as a grantee.

---

### 8.2.3 Adding CSF Mappings

Oracle Privileged Account Manager enables you to securely store and synchronize account credentials with the Oracle Credential Store Framework (CSF). This capability is useful for managing the lifecycle of application passwords stored in CSF.

When you configure CSF synchronization for an account, Oracle Privileged Account Manager changes the account password based on the assigned Usage Policy.

---

### Note: OraclePrivileged Account Manager updates, or synchronizes, CSF only when a password change occurs.

For more information about CSF and how Oracle Privileged Account Manager manages CSF credentials, refer to Section 17.3, “Integrating with the Credential Store Framework.”

---

To add CSF mappings to an account, complete the following steps:

1. Select the Account Name link in the Search Results table.

2. When the Account: AccountName page displays, select the Credential Store Framework tab.

3. Click Add.

   A new row displays in the table with empty fields in each column.

4. Enter the following information into the empty fields:

   - **Administration Server URL**. Enter the server URL in this format, `protocol://listen-address:listen-port`
For example, if you are using the https protocol and the SSL port is 7002, you would enter

```
https://localhost:7002
```

- **Username** and **Password**: Enter the login credentials of the Oracle WebLogic Server administrator.
- **Mapping**: Enter the Map name you created in CSF.
- **Key**: Enter the unique Key you created in CSF.

5. Click **Add** again to create another mapping. You can create as many CSF mappings as needed.

6. When you are finished adding information, click **Test** to validate the mapping. A dialog displays with either a success message or an error message.

### 8.3 Searching for Privileged Accounts

You can search for accounts by using one or more of the following parameters:

- **Account Name**
- **Target Type** (*All*, *ldap*, *unix*, *database*, or *lockbox*)
- **Target Name**
- **Domain**
- **Description**

To search for an account,

1. Select **Accounts** in the Administration accordion.
2. When the Accounts tab displays, use the Search portlet parameters to configure your search.
   - For example, to search for a list of all accounts on a particular target, enter one or more letters of the target name into the **Target Name** field.
   - To search for all available accounts, do not specify any search parameters.
3. Click **Search**.

Review your search results in the Search Results table.

---

**Note:** You can use the **View** menu, located above the Search Results table, to manage how the search results are displayed in the table. Refer to the table in Section 4.3.6, "Working with a Search Results Table" for more information.

---

4. To perform another search, click **Reset**.

### 8.4 Opening Privileged Accounts

Opening an account enables you to view or edit the configuration parameters for that account.

You can open privileged accounts from any Search Results table containing an Account Name link. For example,
1. Select **Accounts** in the Administration accordion and click **Search**.

2. When the results display in the Search Results table, locate the account you want to open and perform one of the following actions:
   - Click the Account Name (an active link) in the Search Results table.
   - Select the account Row and then click **Open**.

The Account: *AccountName* page opens. From this page, depending on your Admin Role, you can view and configure account settings related to the associated target, grants, Credential Store Framework, and checkouts.

### 8.5 Checking Out Privileged Accounts

Oracle Privileged Account Manager enables grantees to check out an account in two ways:

- **Password Checkouts**: Enables grantees to access and check out granted account by using encrypted passwords.
- **Session Checkouts** *(on UNIX systems only)*: Enables grantees to access and check out granted accounts without ever knowing the actual account credentials.

**Note:** You can also use the Oracle Privileged Account Manager command line tool or the RESTful interface to check out accounts.

- To use the command line tool, refer to Section A.5.4, "checkout Command."
- To use the RESTful interface, refer to Section B.6.10, "Check Out an Account."

### 8.5.1 Checking Out Passwords

Any administrator or end user can check out a privileged account password if they have been granted access to that account. (Refer to Chapter 10, "Working with Grantees" for more information.)

**Note:** You must be an administrator with the Security Administration Admin Role to modify or remove an account.

Privileged accounts are *not shared* by default, which means when one user checks out the account, it becomes unavailable to other users and prevents conflicting actions. However, administrators can configure *shared* accounts, which enables multiple users to check out the account at the same time. (Refer to Section 8.1.3, "Understanding Sharing Accounts" for more information.)

The steps for checking out a password are as follows:

1. Select **My Accounts** in the Home accordion.

2. On the My Accounts page, locate the account you want to check out in the Search Results table and select that row (Figure 8–1).
3. Click **Password Check Out**.

4. The Check-Out Account dialog displays with the Account Name, Target Name, and a blank **Justifications** field. Enter a comment in this field if you choose to, and then click **Checkout**.

   If the checkout is successful, a Check-Out Account - Success dialog displays. This dialog contains an encrypted Password. You can view this Password in clear text by clicking the **Show Password** box.

   If the checkout fails, the Check-Out Account dialog displays with a message stating you cannot check out the account, which may indicate someone else has already checked out that account.

   **Note:** To see if an account is already checked out, click the account name. When the Account: *AccountName* page opens, you can review the Current Checkouts table to see who checked out the account, what type of checkout it was (password or session), when the account was checked out, the checkout expiration date, and view a recording (if available).

5. Click **Close** to close the dialog and return to the Search Results table.

6. To verify that you checked out the account successfully:
   - Select **My Checkouts** from the Home accordion. When the My Checkouts page displays, locate the account name in the table.
   - If you have the **Security Administrator** Admin Role or the **User Manager** Admin Role, you can select **Accounts** from the Administration accordion and click **Search**. When the Search Results display, select the account name in the table to open the Account: *AccountName* page. The account should be listed in the Current Checkouts table.

### 8.5.2 Checking Out Privileged Account Sessions

**Privileged sessions** provide an extra level of security for privileged accounts on UNIX targets. Through privileged sessions, a grantee can access the granted account without ever knowing the actual account credentials.

**Note:** Session checkout is not available for other target types.

Any administrator or end user can check out a privileged account session if they have been granted access to that account and if the Usage Policy associated with the account allows session checkouts. (Refer to Chapter 10, "Working with Grantees" and Section 9.3.3, "Modifying the Default Usage Policy" for more information.)
1. Select My Accounts in the Home accordion and then click **Search**.

   The My Accounts page is refreshed and all of your accounts are displayed in the Search Results table.

---

**Note:** If you already know how to establish the session using an SSH client, and you know the Oracle Privileged Account Manager server host, port, UNIX target, and UNIX account name, proceed to step 3.

---

2. Select the account row, and then click **Session Check Out** to view the connection information you need to establish the session using an SSH client.

   For example:

   ```
   Account Name:  opamuser1  
   Target Name:  sample-unix  
   SSH Port:      1222  
   Instruction:   ssh -p <port> <opamuser>:<targetname>:<accountname>@<sessionmgrhost>  
   ```

   Use opam password on password prompt.

   Where:
   - **port** is the port where Oracle Privileged Session Manager is running.
   - **opamuser** is the Oracle Privileged Account Manager end user.
   - **targetname** is the name of the target to which you are connecting.
   - **accountname** is the account you will be using on that target.
   - **sessionmgrhost** is the host on which you are running Session Manager.

---

**Note:** The preceding example uses default Oracle Privileged Account Manager connection settings and instructions. Oracle Privileged Account Manager administrators can configure this information to whatever is appropriate for their own environments.

Refer to Section 5.3, “Managing the Oracle Privileged Session Manager Server” for information about configuring these settings.

---

3. Use your favorite SSH client to connect to a target or an account through the Oracle Privileged Session Manager server.

   For example, using the SSH client on a standard Linux machine, you would perform the following steps:

   a. Open a command window.

   b. At the prompt, enter the connection information as noted in the Session Checkout dialog.

      ```
      prompt> ssh -p 1222 opamuser1:target_system:opamuser1@sessionmgrhost
      ```

      A message displays stating that you are authenticated with partial success.

4. Enter the appropriate Oracle Privileged Account Manager password when you see the prompt to complete the connection to the Oracle Privileged Session Manager server.
To confirm the connection, type `id` at the prompt, and the account's `uid`, `gid`, and group information will be returned.

5. Return to the My Accounts page in the Console

6. To verify that you checked out the session successfully:
   - Select My Checkouts from the Home accordion. When the My Checkouts page displays, locate the account name in the table and review the Checkout Type column.
   - If you have the Security Administrator Admin Role or the User Manager Admin Role, you can select Accounts from the Administration accordion and click Search. When the Search Results display, select the account name in the table to open the Account: AccountName page. The session should be listed in the Current Checkouts table.

   **Note:** You do not have to perform any special steps to check in a checked out session. If you use the procedure described in Section 8.6, "Checking In Privileged Accounts," then the account is checked back in regardless of the checkout type (password or session).

### 8.6 Checking In Privileged Accounts

Any administrator or end user can check in their checked-out accounts by using the steps described in Regular Check-In.

Administrators with the User Manager Admin Role can force an account check-in (check in privileged accounts that have been checked out by other users) when necessary. Use the steps described in Forcing a Check-In.

   **Note:** In either case, you use the same steps to check in an account password or an account session.

#### Regular Check-In

To check in a checked out privileged account:

1. Select My Checkouts on the Home accordion.
   - The My Checkouts page displays with all of your checked-out accounts (passwords and sessions) listed in the Search Results table.

2. Select the account row or rows you want to check in.

3. When the Check-in icon located above the table becomes active, click the icon.

4. When the Check-in Accounts dialog displays, click the Check In button.
   - If the check-in is successful, Oracle Privileged Account Manager removes the account name(s) from the My Checkouts table and the account becomes available for check out again.

#### Forcing a Check-In

To force an account check in:

1. Select Accounts in the Administration accordion, and then search for the account as described in Section 8.3, "Searching for Privileged Accounts."

2. Select (check) the account you want to check in.
3. When the **Force Check In** option located above the table becomes active, click the icon.

   The Confirm Forced Check In dialog displays, asking you to confirm that you want to check in the account. Be aware that forcing the check in will log out all users that currently have the account checked out.

4. To proceed, click the **Check In** button.

   If the check-in is successful, the account becomes Available for check out again.

---

**Note:** You can also use the Oracle Privileged Account Manager command line tool or the RESTful interface to check-in accounts.

- To use the command line tool, refer to Section A.5.3, "checkin Command."
- To use the RESTful interface, refer to Section B.6.15, "Check In an Account."

---

## 8.7 Viewing a Session Recording

If necessary, administrators can view a recording, in transcript format, of the actions taken by the user during an account checkout (password or session).

---

**Note:** The **Session Monitoring Update Interval in seconds** setting on the Session Manager Configuration Page controls how often on-going session transcripts are updated. Refer to step 2 in Section 5.3.3, "Managing the Oracle Privileged Session Manager Properties" for more information.

---

The following table describes the different transcript types, where you can access these recordings, and which Admin Roles are required to view the transcripts:

<table>
<thead>
<tr>
<th>Recording Type</th>
<th>Viewing Location</th>
<th>Admin Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>On-going session transcripts</td>
<td>The account's Current Checkouts table</td>
<td>Security Admin or User Manager</td>
</tr>
<tr>
<td>Expired session transcripts</td>
<td>The account's Checkout History tab</td>
<td>Security Admin or User Manager</td>
</tr>
<tr>
<td>Expired session transcripts</td>
<td>Checkout History Report page</td>
<td>Security Auditor</td>
</tr>
</tbody>
</table>

The next three sections provide instructions for accessing these recordings.

**From a Current Checkouts Table**

1. Open the account as described in Section 8.4, "Opening Privileged Accounts."

2. When the Account: *AccountName* page displays, locate the correct user in the Current Checkouts table, and click the **Recording** icon ( ) in that row.

   A new tab opens in your browser and the recording displays in a transcript format. For example,
Figure 8–2  Example Session Recording

This example shows a representative section of a session transcript that includes some the commands entered by the user during that session.

From the Checkout History Tab
1. Open the account as described in Section 8.4, "Opening Privileged Accounts."
2. When the Account: AccountName page displays, select the Checkout History tab. A new tab opens in your browser and the recording displays in a transcript format.
3. Specify the period in which to search by providing the Start Date and End Date (required fields). Include any other, optional search criteria in the Search Checkout History section, and then click Search.
4. When the search results display in the table, locate the user whose transcript you want to review, and click the Recording icon ( ) in that row.

From the Checkout History Page
1. Select the Checkout History Report link from the Reports accordion to open the Checkout History page.
2. Use the Search Checkout History portlet to configure search parameters:
   - You must specify a Start Date and an End Date range in which to search for checkouts. Type the date and time into the blank fields or use the Calendar icons.
   - Enter information into one or more of the Account Name, User Name, Target Name, or Pattern fields.

Note: Use the Pattern field to search for a string in the recording of a checkout event.
3. Click **Search** and the results will display in the table.

4. Locate the correct account and user row in the table, and click the **Recording** icon in that row.

5. You are prompted to select a program, such as Wordpad, in which to open the transcript. Select a program and click **Open**.

   The recording opens in the selected program, and displays in a transcript format (refer to Figure 8–2).

### 8.8 Managing Privileged Account Passwords

Oracle Privileged Account Manager provides the following options for managing privileged account passwords:

- **Showing an Account Password**
- **Viewing an Account's Password History**
- **Resetting an Account Password**

**Note:** You can also perform these password management tasks by using the Oracle Privileged Account Manager command line tool or REST API.

**Note:** The procedures for showing and resetting a target's service account password are different from the procedures described in this section. Refer to Section 7.3, "Managing Service Account Passwords" for information.

#### 8.8.1 Showing an Account Password

If necessary, you can view a password in clear text for an account that you have checked out by using the **Show Password** option. For example, if you forget a password, you can use this feature to view the password again.

Any user can review passwords for accounts they have checked out. However, you cannot access passwords after the account is checked back in or view passwords for accounts that are checked out by other users. Attempts to do so will cause an error.

**Note:** Administrators with the **Security Administration** Admin Role, who can access all system and target service accounts, can use this feature to view current the password for both checked out and checked in privileged accounts.
From the My Checkouts Page  
You can access the Show Password option from the My Checkouts page as follows:

1. Select My Checkouts in the Home accordion to open the My Checkouts page.
2. Select the account’s row number.
3. When the Show Password icon becomes active, click the icon.
   The Current Password dialog displays with the Account Name and the Password in clear text.
4. When you are finished click Close.

From the Accounts Page  
Administrators with the Security Administration or User Manager Admin Role can access the Show Password option as follows:

1. Ensure that you have the privileged account checked out.

   Note: For most users, if they try to view the password for an account that has already been checked back in, an error will result.

   However, if you are an administrator with the Security Administrator or User Manager Admin Role, you can use this command to reset a password for both checked out and checked-in accounts.

2. Select Accounts in the Administration accordion.
3. When the Accounts page displays, use the Search portlet to locate the account.
4. Select the account row number and when the Show Password icon becomes active, click the icon.
   The Current Password dialog displays with the Account Name and the Password in clear text.
5. When you are finished click Close.

8.8.2 Viewing an Account’s Password History

Use the Password History option to view the password history for an account.

   Note: You must be an administrator with the Security Administration Admin Role to view the password history for a privileged account.

To view a privileged account’s password history,

1. Select Accounts in the Administration accordion to open the Search Accounts page, and then click Search.
2. Select the row number of the account.
3. When the Password History icon becomes active, click the icon.
   The Show Password History dialog displays with the Account Name, and the Password in clear text, and the Modification Time (Date and time of the password reset).
4. When you are finished click Close.
8.8.3 Resetting an Account Password

If necessary, you can manually reset the existing password for an account that you have checked out by using the Reset Password option.

If Security Administrators do not want to use randomized password generation, they can manually set a password of their choosing. For example, administrators might prefer to set a simple, easy-to-type password for one time use, such as during a system upgrade.

To reset an account password, use the following steps

1. Ensure that you have the privileged account checked out.

   **Note:** For most users, if they try resetting the password for an account that has already been checked back in, an error will result.

   However, if you are an administrator with the Security Administrator Admin Role, you can use this command to reset a password for both checked out and checked-in accounts.

2. Select Accounts in the Administration accordion.

3. When the Accounts tab displays, use the Search portlet to locate the account.

4. Select the account row number and then click Reset Password.

   The Reset Password dialog displays and provides the following information about the account password:
   - Account Name
   - Target Name

   This dialog also contains a New Password field.

5. Type a password into the space provided and click Save.

   You can use a password string of your choosing. The string does not have to comply with the Oracle Privileged Account Manager Password Policy because the Password Policy is used for randomized password generation.

   A message displays with the name of the selected account and the new password.

8.9 Removing Privileged Accounts from Oracle Privileged Account Manager

You can remove a privileged account from Oracle Privileged Account Manager by using the Search Accounts page or the Targets page.

**WARNING:** When you remove a privileged account, you remove all information about the account that is stored in Oracle Privileged Account Manager.

Before removing a privileged account, it is critical that you first capture all relevant information from that account. For example, save the current password associated with that privileged account.
From the Search Accounts Page
To remove an account from the Search Accounts page,

1. Locate the account to remove.
   a. Select Accounts in the Administration accordion.
   b. Click Search in the Search Accounts portlet to populate the Search Results table with a list of all available accounts.
      To narrow the results or to locate a particular account, enter search criteria in one or more the Search Accounts fields, and then click Search.

2. In the Search Results table, select the account to be removed, and then click Remove.

3. When you are finished, click the Apply button located at the top of the page.

From the Target Page
To remove an account from a target,

1. Locate the target from which you want to remove the account.
   a. Select Targets in the Administration accordion.
   b. Click Search in the Search Targets portlet to populate the Search Results table with a list of all available targets.
      To narrow the results or to locate a particular target, enter search criteria in one or more the Search Targets fields, and then click Search.

2. Click the target name in the Search Results table to open the target.

3. Select the Privileged Accounts tab.

4. In the Search Results table, select the account to be removed and then click Remove.

5. When you are finished, click the Apply button located at the top of the page.
This chapter provides information about working with Oracle Privileged Account Manager Password Policies and Usage Policies from the Console.

This chapter includes the following sections:

- Section 9.1, "What Are Oracle Privileged Account Manager Policies?"
- Section 9.2, "Working with Password Policies"
- Section 9.3, "Working with Usage Policies"

**Note:** You can also manage Oracle Privileged Account Manager policies from the command line or by using Oracle Privileged Account Manager's RESTful interface. For information, refer to

- Section A.3, "Working with Policies" in Appendix A, "Working with the Command Line Tool."
- Section B.4, "Policy Resource" in Appendix B, "Working with Oracle Privileged Account Manager's RESTful Interface."

### 9.1 What Are Oracle Privileged Account Manager Policies?

In Oracle Privileged Account Manager, there are two types of policies:

- **Password Policy.** This policy type captures the password construction rules enforced by a specific target on an associated privileged account. For example, minimum and maximum number of numeric characters. You use a Password Policy to create a password value that Oracle Privileged Account Manager uses to reset a password for a privileged account.

  A Password Policy also governs a password *lifecycle*, or how often a password must change.

- **Usage Policy.** This policy type defines when and how a grantees can use a privileged account. (Default is 24x7 access to password checkouts.)

Every privileged account that is managed by Oracle Privileged Account Manager must have an associated Password Policy. A Usage Policy only applies at the level of a grant. You can associate a single Password Policy with multiple privileged accounts and a single Usage Policy with multiple grants.
Working with Password Policies

Oracle Privileged Account Manager provides both a Default Password Policy and a Default Usage Policy. You can use these default policies, modify them, or create your own, specialized policies.

**Note:** For Usage Policies,

- User grants are given first priority.
  
  If a user has direct access to an account through a user grant, then Oracle Privileged Account Manager applies the Usage Policy that corresponds to that grant.
  
  If Oracle Privileged Account Manager cannot find a user grant for the user, then it looks for any group grants that grant the user access to that account.

  If the user is a member of multiple granted groups, then Oracle Privileged Account Manager sorts the group names into alphabetical order and uses the Usage Policy assigned to the first group.

  For example, assume you have Group A with corresponding policy `UsagePolicyB` and Group B with `UsagePolicyA`. When Oracle Privileged Account Manager sorts the group names, Group A comes first alphabetically, so Oracle Privileged Account Manager will apply `UsagePolicyB`.

**Note:** If you want to modify the default policies, Oracle recommends making a back-up copy of the policy before you modify it. Use the `export` command as described in Section A.8.1, "export Command."

Oracle Privileged Account Manager provides both a Default Password Policy and a Default Usage Policy. You can use these default policies, modify them, or create your own, specialized policies.

**Note:** If you want to modify the default policies, Oracle recommends making a back-up copy of the policy before you modify it. Use the `export` command as described in Section A.8.1, "export Command."

To review the parameter settings for a policy, refer to Section 9.2.2, "Viewing Password Policies" or Section 9.3.2, "Viewing Usage Policies."

**Note:** Only administrators with the Security Administrator Admin Role or the User Manager Admin Role can work with policies.

- Administrators with the Security Administrator Admin Role can modify the Default Password Policy and Default Usage Policy, create new policies, or delete policies. (You cannot delete the Default Password Policy or the Default Usage Policy.)
- Administrators with the Security Administrator Admin Role can assign Password Policies, but they cannot assign Usage Policies.
- Administrators with the User Manager Admin Role can only assign a Usage Policy to accounts at the grantee-account pair level. In other words, the User Manager can assign different Usage Policies to different grantees of the same account.
  
Administrators with the User Manager Admin Role cannot assign Password Policies.

### 9.2 Working with Password Policies

This section describes the different tasks an administrator performs when working with Password Policies.
The topics include:

- Section 9.2.1, "Searching for Password Policies"
- Section 9.2.2, "Viewing Password Policies"
- Section 9.2.3, "Modifying the Default Password Policy"
- Section 9.2.4, "Creating a Password Policy"
- Section 9.2.5, "Assigning Password Policies"
- Section 9.2.6, "Deleting Password Policies"

### 9.2.1 Searching for Password Policies

To search for a Password Policy,

1. Select **Password Policies** from the Administration accordion.
2. When the Search Policies portlet displays, enter your search criteria into one or more of the following fields.
   - **Policy Name**: Enter all or any part of a policy name.
   - **Policy Status**: Select **All** (default) from the menu to search for all policies (active and inactive). Select **Active** or **Disabled** to limit the search to just active or inactive policies.
3. Click **Search**.

Review your search results in the Search Results table.

### 9.2.2 Viewing Password Policies

To review the parameter settings for a Password Policy,

1. Select **Password Policies** from the Administration accordion.
2. When the Policies page displays, click **Search**.

The existing Password Policies will display in the Search Results table.

3. Use one of the following methods to open a policy:
   - Click the **Row** number next to the policy name and then click the **Open** icon located above the Search Results table.
   - Click the policy name (an active link) in the Search Results table.

For example, clicking the **Default Password Policy** link opens the Password Policy: Default Password Policy page.

A Password Policy page contains three tabs:

- **General**: Contains parameters used to specify general information about the policy and Password Lifecycle Rules for the policy. Password Lifecycle Rules govern when Oracle Privileged Account Manager must automatically reset an account password.

- **Password Complexity Rules**: Contains parameters that govern the complexity requirements for account passwords.

- **Privileged Accounts**: Provides information about the privileged accounts currently using that Password Policy.
9.2.3 Modifying the Default Password Policy

After evaluating the Default Password Policy, you may decide you want to modify the settings to better suit your environment.

**Note:** Oracle recommends making a back-up copy of the Default Password Policy before you modify it. You can use the export command as described in Section A.8.1, "export Command."

To modify the Default Password Policy,

1. Select Password Policies from the Administration accordion.
2. When the Password Policies page displays, click Search to populate the Search Results table.
3. Click the Default Password Policy link in the Search Results table to open the Password Policy: Default Password Policy page.
4. Select the General tab to modify the Description in the General Fields area or to modify any of the following Password Lifecycle Rules:

**Note:** You cannot edit the Policy Name or Policy Status values for this policy.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save password history for</td>
<td>Use the counter and drop menus to specify how many days to save the password history for an account. The password history includes when accounts are checked out, checked in, and when their passwords were reset.</td>
</tr>
<tr>
<td>Expire password after</td>
<td>Use the counter and drop menus to specify a duration period (number of days, hours, or minutes) after which Oracle Privileged Account Manager must automatically reset the account password. For example, if your enterprise wants a security policy where account passwords must be changed every month, you would set this value to 30 days. Every time the account is checked out and its password gets changed (if the policy is configured so that passwords must be changed on checkout/check-in) Oracle Privileged Account Manager tracks the password change time. If Oracle Privileged Account Manager detects the account is idle and no password changes have occurred over the specified number of days, then Oracle Privileged Account Manager automatically resets the password to a new, randomized value, which helps the enterprise to automatically enforce the security policy without human intervention. To disable this automatic reset option, set the numeric value to 0. <strong>Note:</strong> The Oracle Privileged Account Manager scheduler periodically checks for accounts where the password maximum age has expired and resets them as described in this section. By default, the scheduler makes this check every 60 minutes (based on the passwordcyclerinterval property in the OPAM Global Config configuration entry, whose default setting is 60 minutes). You can view and modify the current interval by using Oracle Privileged Account Manager’s getconfig and modifyconfig command line options. For more information, refer to Section A.2.1, &quot;getconfig Command&quot; and to Section A.2.3, &quot;modifyconfig Command.&quot;</td>
</tr>
</tbody>
</table>
5. Select the Password Complexity Rules tab to change one or more of the parameters that define the default password requirements.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Characters for Password</td>
<td>Specify the minimum and maximum number of characters required.</td>
</tr>
<tr>
<td>Alphabetic Characters</td>
<td>Specify the minimum number of alphabetic characters required.</td>
</tr>
<tr>
<td>Numeric Characters</td>
<td>Specify the minimum number of numeric characters required.</td>
</tr>
<tr>
<td>Alphanumeric Characters</td>
<td>Specify the minimum number of alphanumeric characters required.</td>
</tr>
<tr>
<td>Special Characters</td>
<td>Specify the minimum and maximum number of special characters (such as * or @) required.</td>
</tr>
<tr>
<td>Repeated Characters</td>
<td>Specify the minimum and maximum number of repeated characters allowed.</td>
</tr>
<tr>
<td>Unique Characters</td>
<td>Specify the minimum number of unique characters required.</td>
</tr>
<tr>
<td>Uppercase Characters</td>
<td>Specify the minimum number of uppercase characters required.</td>
</tr>
<tr>
<td>Lowercase Characters</td>
<td>Specify the minimum number of lowercase characters required.</td>
</tr>
<tr>
<td>Start with Character (not digit)</td>
<td>Specify the first character required to start a password.</td>
</tr>
<tr>
<td>Required Characters</td>
<td>Specify which characters are required in a password.</td>
</tr>
</tbody>
</table>

Note:

- An administrator with the Security Administrator Admin Role can also manually reset a password by using the Reset Password option (described in Section 8.8.3, "Resetting an Account Password") and Oracle Privileged Account Manager tracks this password change time as well.

- For higher security, the Reset password on check-in and Reset password on check-out options are both enabled by default, but they can be disabled if required. For example, some enterprises may only require that passwords be reset every 30 days.

- If your enterprise prefers that passwords not be automatically managed at all; that they are only changed through human intervention, disable all three Password Lifecycle Rules options. However, after disabling these three options, the only way to manually change passwords is by using the Reset Password option (described in Section 8.8.3, "Resetting an Account Password"). Oracle Privileged Account Manager is still useful in this case, as you can reset and centrally manage passwords for multiple systems from one place by using Oracle Privileged Account Manager.
Working with Password Policies

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allowed Characters</td>
<td>Specify which characters are permitted in a password.</td>
</tr>
<tr>
<td>Disallowed Characters</td>
<td>Specify which characters are not permitted in a password.</td>
</tr>
<tr>
<td>Disallowed as Password</td>
<td>Enable (check) the Account Name box to prohibit the use of an account name in the password.</td>
</tr>
</tbody>
</table>

6. Select the Privileged Accounts tab to review which accounts are currently using the Default Password Policy.

Note: To specify a different Password Policy for any account listed in the table, click the Account Name link. When the Account page displays, select a different policy name from the Password Policy menu.

7. When you are finished editing the policy, click Apply to save your changes.

9.2.4 Creating a Password Policy

To create a Password Policy,

1. Select Password Policies from the Administration accordion.

2. When the Password Policies page displays, click Create at the top of the Search Results table.

   A new, Password Policy: Untitled page displays with three tabs.

3. Provide the following information on the General tab:
   a. Policy Name: Enter a name for the new policy.
   b. Policy Status: Click the button to specify whether the policy is Active or Disabled.

      Making the policy Active puts that policy into effect for all of the associated accounts and grants.

      Disabling a policy applies the Default Password Policy to all accounts and grants associated with that disabled policy. If you simply assigned a different policy to those accounts and grants, you would lose all information about the old policy assignment.

   c. Description (optional): Enter a descriptive statement about the new policy.
   d. Password Lifecycle Rules: Configure these parameters to enable Oracle Privileged Account Manager to auto-generate and set a randomized account password under certain conditions, as described in step 4 on page 9-4.

4. Select the Password Complexity Rules tab to specify password complexity rules for this policy. Refer to the table provided in step 5 on page 9-5 for a description of these parameter settings.

5. Select the Privileged Accounts tab to assign the new policy to accounts or grantees. Refer to Section 9.2.5, "Assigning Password Policies" for detailed instructions.

   After assigning this Password Policy to privileged accounts, you can select the Privileged Accounts tab to review which accounts are currently using this policy.

6. Click Save.
### 9.2.5 Assigning Password Policies

When you add a new privileged account, Oracle Privileged Account Manager automatically assigns the Default Password Policy to that account. However, if you have created other Password Policies, as described in Section 9.2.4, "Creating a Password Policy," you can assign a different policy to the account.

---

**Note:** Only administrators with the Security Administrator Admin Role can assign Password Policies to accounts.

---

You can assign Password Policies to an account
- **From the Accounts Page**
- **From the Targets Page**
- **From the Password Policies Page**

**From the Accounts Page**

To assign a Password Policy from the Accounts page,

1. Locate the account where you want to assign the policy.
   - **a.** Select **Accounts** in the Administration accordion.
   - **b.** Click **Search** in the Search Accounts portlet to populate the Search Results table with a list of all available accounts.

   To narrow the results or to locate a particular account, enter search criteria in one or more the Search Accounts fields, and then click **Search**. For example, if you know the account is assigned to a UNIX target, select **unix** from the **Target Type** menu.

2. When the Search Results display, click the account’s Account Name link in the table to open the Account: **AccountName** page.

3. On the General tab, select a different policy name from the **Password Policy** menu.

4. After selecting the new policy, click **Test** to verify that the account can be managed by Oracle Privileged Account Manager.

   If the test is successful, you should see a "Test Succeeded" message.

5. **Click Apply** to finish assigning the policy to the selected account.

**From the Targets Page**

To assign a Password Policy from the Targets page,

1. Locate the target where the account is located.
   - **a.** Select **Targets** in the Administration accordion.
   - **b.** Click **Search** in the Search Targets portlet to populate the Search Results table with a list of all available targets.

   To narrow the results or to locate a particular target, enter search criteria in one or more the Search Targets fields, and then click **Search**.

2. Click the account’s Target Name link in the Search Results table to open the Target: **TargetName** page.

3. Click the Privileged Accounts tab to view a list of the accounts currently managed on the target.
Notice that the table lists the Password Policy that is currently assigned to each account.

4. Locate the account in the Privileged Accounts table, and then click the Account Name link.

5. When the General tab displays, select a different policy name from the Password Policy menu.

6. After selecting the new policy, click Test to verify that the account can be managed by Oracle Privileged Account Manager.

   If the test is successful, you should see a "Test Succeeded" message.

7. Click Apply to finish assigning the policy to the selected account.

From the Password Policies Page
To assign a Password Policy from the Policies page,

1. Locate the Password Policy that you want to assign to the account.
   a. Select Password Policies in the Administration accordion.
   b. Click Search in the Search Policies portlet to populate the Search Results table with a list of all available Password Policies.

      To narrow the results or to locate a particular policy, enter search criteria in one or more the Search Policies fields, and then click Search.

2. Locate the policy in the Search Results table, and then click the Policy Name link to open the Password Policy: PolicyName page.

3. Select the Privileged Accounts tab.

4. Locate the account and click the Account Name link to open the Account: AccountName page.

5. When the General tab displays, select a different policy name from the Password Policy menu.

6. After selecting the new policy, click Test to verify that the account can be managed by Oracle Privileged Account Manager.

   If the test is successful, you should see a "Test Succeeded" message.

7. Click Apply to finish assigning the policy to the selected account.

9.2.6 Deleting Password Policies

   Note: You cannot delete the Default Password Policy.

To delete a Password Policy,

1. Locate and select the policy to be deleted.

2. Click the Delete icon.

3. When the Confirm Remove dialog displays, click the Remove button.

   The policy is immediately deleted. If you had any accounts assigned to that policy, they will all revert to using the Default Password Policy.
9.3 Working with Usage Policies

This section describes the different tasks an administrator performs when working with Usage Policies.

The topics include:

- Section 9.3.1, "Searching for Usage Policies"
- Section 9.3.2, "Viewing Usage Policies"
- Section 9.3.3, "Modifying the Default Usage Policy"
- Section 9.3.4, "Creating a Usage Policy"
- Section 9.3.5, "Assigning Usage Policies"
- Section 9.3.6, "Deleting Usage Policies"

9.3.1 Searching for Usage Policies

To search for a Usage Policy,

1. Select Usage Policies from the Administration accordion.

2. When the Search Policies portlet displays, enter your search criteria into one or more of the following fields.
   - **Policy Name**: Enter all or any part of a policy name.
   - **Policy Status**: Select All (default) from the menu to search for all policies (active and inactive). Select **Active** or **Disabled** to limit the search to just active or inactive policies.

3. Click **Search**.

Review your search results in the Search Results table.

9.3.2 Viewing Usage Policies

To review the parameter settings for a Usage Policy,

1. Select Usage Policies from the Administration accordion.

2. When the Policies page displays, click **Search**.

   The existing policies will display in the Search Results table.

3. Use one of the following methods to open a policy:
   - Click the Row number next to the policy name and then click the **Open** icon located above the Search Results table.
   - Click the policy name (an active link) in the Search Results table.

   For example, clicking the Default Usage Policy link opens the Usage Policy: Default Usage Policy page.

The Usage Policy page contains three tabs:

- **General Fields**: Contains parameters used to specify general information about the policy.
- **Usage Rules**: Contains parameters that govern the time zone to be associated with checking out a privileged account, when the account can be checked out, and when the check out expires.
■ Grantees. Provides information about the grantees who are authorized to use that account.

9.3.3 Modifying the Default Usage Policy

After evaluating the Default Usage Policy, you may decide you want to modify the settings to better suit your environment.

---

**Note:** Oracle recommends making a back-up copy of the Default Usage Policy before you modify it. You can use the `export` command as described in Section A.8.1, "export Command."

---

To modify the Default Usage Policy,

1. Select **Usage Policies** from the Administration accordion.

2. When the Usage Policies page displays, click **Search** to populate the Search Results table.

3. Select the **Default Usage Policy** link in the Search Results table to open the Usage Policy: Default Usage Policy page.

4. Select the General Fields tab, where you can modify one or both of the following parameters:

   ---

   **Note:** You cannot edit the **Policy Name** or **Policy Status** values for this policy.

   ---

   - **Description**: Highlight and delete the existing text, and then enter your new description.

   - **Allow Checkout Type**: Use this menu to specify one of the following checkout options for this policy:
     - **All**: Allow users to check out passwords and sessions.
     - **password** *(default)*: Allow users to only check out passwords.
     - **session**: Allow users to only check out sessions.

   - **Enable Session Recording**: Select to enable session recording when this Usage Policy is applied to a session checkout.

     Refer to Section 8.7, "Viewing a Session Recording" for more information about session recordings.

5. Select the Usage Rules tab to change one of more of following parameter settings:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Timezone</strong></td>
<td>Select a time zone from the menu to indicate when the policy will be applied. For example, if you set the time zone to GMT, and the policy allows check-outs between 9am to 5pm, you can only check out between 9am-5pm GMT, and not PST.</td>
</tr>
<tr>
<td><strong>Permitted Usage Dates</strong></td>
<td>Use the <strong>Monday</strong> through <strong>Sunday</strong> checkboxes and the <strong>From</strong> and <strong>To</strong> drop menus to specify when grantees are allowed to use the account. Select one or more days of the week and the periods of time when grantees can access this account. (Default access is 24x7.)</td>
</tr>
</tbody>
</table>
6. Select the Grantees tab to view which grantees this policy is assigned.

   **Note:** To specify a different Usage Policy for any grantee listed in the table, click the **Account Name** link. When the Account page displays, select a different policy name from the **Usage Policy** menu.

   **Tip:** Clicking the active links in the Grantee Name or Account Name columns enable you to navigate to other screens for additional information.

7. When you are finished editing the policy, click **Apply** to save your changes.

### 9.3.4 Creating a Usage Policy

To create a Usage Policy,

1. Select **Usage Policies** from the Administration accordion.
2. When the Policies page displays, click **Create** at the top of the Search Results table. A new, Usage Policy: **Untitled** page displays with three tabs.

3. Provide the following information on the General tab:
   a. **Policy Name**: Enter a name for the new policy.
   b. **Policy Status**: Click the button to specify whether the policy status is **Active** or **Disabled**.
      
      Making the policy Active puts that policy into effect for the associated accounts and grants.
      
      Disabling a policy applies the Default Usage Policy to all accounts and grants associated with that disabled policy. If you simply assigned a different policy to those accounts and grants, you would lose all information about the old policy assignment.
   c. **Description** *(optional)*: Enter a descriptive statement about the new policy.
   d. **Allow Checkout Type**: Use this menu to specify one of the following checkout options for this policy:
      - **All**: Allow users to check out passwords and sessions.
      - **password** *(default)*: Allow users to only check out passwords.
      - **session**: Allow users to only check out sessions.
   e. **Enable Session Recording**: Select to enable session recording when this Usage Policy is applied to a session checkout.
      
      Refer to Section 8.7, "Viewing a Session Recording" for more information about session recordings.

4. Select the Usage Rules tab to define rules for using a privileged account. Refer to the table in step 5 on page 9-5 for a description of these parameter settings.

5. Select the Grantees tab to assign the new policy to accounts or grantees. Refer to Section 9.3.5, "Assigning Usage Policies" for detailed instructions.

   After assigning this policy, you can select the Grantees tab to review which users or groups are using this policy.

6. Click **Save**.

### 9.3.5 Assigning Usage Policies

When you create a new privileged account, Oracle Privileged Account Manager automatically assigns the Default Usage Policy to that account. However, if you have created additional Usage Policies, as described in Section 9.3.4, "Creating a Usage Policy," then you can assign a different policy to the account.

---

**Note:**

- Administrators with the **Security Administrator** Admin Role can assign Usage Policies to accounts. However, this role can only apply a Usage Policy at the account level.

- Administrators with the **User Manager** Admin Role can assign a Usage Policy to accounts at the *grantee-account pair* level. In other words, the User Manager can assign different Usage Policies to different grantees of the same account.

---
You can assign a different Usage Policy

- From the Accounts Page
- From the Targets Page
- From the Usage Policies Page

**Note:**
- When you add grantees to an account, as described in Section 10.2, “Granting Accounts to Users” or Section 10.3, “Granting Accounts to Groups,” Oracle Privileged Account Manager adds the user or group name to the Users or Groups table on the Grants tab and automatically assigns the Default Usage Policy.

- When you create a new Usage Policy for an account, the new policy is not automatically assigned to the existing grantees on that account. Oracle Privileged Account Manager allows you to assign customized policies to individual grantees, so you do not want the new policy to override those other policy assignments.

However, if you create a new policy for an account and then add new grantees, those (and future) grantees will automatically be associated with that policy because it has become the new Default Usage Policy for the account.

**From the Accounts Page**

To assign a Usage Policy from the Accounts page,

1. Locate the account where you want to assign the policy.
   - a. Select **Accounts** in the Administration accordion.
   - b. Click **Search** in the Search Accounts portlet to populate the Search Results table with a list of all available accounts.
      
      To narrow the results or to locate a particular account, enter search criteria in one or more the Search Accounts fields, and then click **Search**.

2. Locate the account’s **Account Name** link to open the Account: **AccountName** page.
3. Select the Grants tab.
4. Locate the grantee in the Users or Groups table, and use the **Usage Policy** menu in that row to select a different policy.
5. Click **Apply** to add your changes.

**From the Targets Page**

To assign a Usage Policy from the Targets page,

1. Locate the target where the account is located.
   - a. Select **Targets** in the Administration accordion.
   - b. Click **Search** in the Search Targets portlet to populate the Search Results table with a list of all available targets.
      
      To narrow the results or to locate a particular target, enter search criteria in one or more the Search Targets fields, and then click **Search**.

2. Click the account’s **Target Name** in the Search Results table to open that target.
3. When the Target: TargetName page displays, click the Grants tab to view a list of the grantees currently granted access to that account. Notice that the table lists the Usage Policy that is currently assigned to each grantees.

4. Locate the grantee in the Users or Groups table, and use the Usage Policy menu in that row to select a different policy.

5. Click Apply to finish assigning the policy to the selected account.

**From the Usage Policies Page**
To assign a Usage Policy from the Policies page,

1. Locate the Usage Policy that you want to assign to the account.
   a. Select Usage Policies in the Administration accordion.
   b. Click Search in the Search Policies portlet to populate the Search Results table with a list of all available Usage Policies.
      To narrow the results or to locate a particular policy, enter search criteria in one or more the Search Policies fields, and then click Search.

2. When the search results display, locate the policy you want to assign. Click the Policy Name link to open the Usage Policy: PolicyName page.

3. Select the Grantees tab.

4. Locate the user or group name in the Grantees table and then click that grantee's Account Name link to open the account.

5. When the Account: AccountName page displays, click the Grants tab.

6. Locate the grantee in the Users or Groups table, and use the Usage Policy menu in that row to select a different policy.

7. Click Apply to add your changes.

### 9.3.6 Deleting Usage Policies

**Note:** You cannot delete the Default Usage Policy.

To delete a Usage Policy,

1. Locate and select the policy to be deleted.

2. Click the Delete icon.

3. When the Confirm Remove dialog displays, click the Remove button.
   The policy is immediately deleted. If you had any accounts assigned to that policy, they will all revert to using the Default Usage Policy.
This chapter describes the different tasks you can perform when working with grantees in Oracle Privileged Account Manager.

**Note:** You must be an Oracle Privileged Account Manager administrator with the User Manager Admin Role to add, edit, or delete grantees.

This chapter includes the following sections:

- Section 10.1, "What Are Grantees?"
- Section 10.2, "Granting Accounts to Users"
- Section 10.3, "Granting Accounts to Groups"
- Section 10.4, "Searching for Grantees"
- Section 10.5, "Opening a Grantee"
- Section 10.6, "Removing Grantees from an Account"

**Note:** You can also use Oracle Privileged Account Manager’s command line tool or Oracle Privileged Account Manager’s RESTful interface to perform many of the tasks described in this chapter.

If you prefer using these interfaces instead of the Oracle Privileged Account Manager Console, refer to Appendix A, "Working with the Command Line Tool" or Appendix B, "Working with Oracle Privileged Account Manager’s RESTful Interface" for instructions.

### 10.1 What Are Grantees?

*Grantees* are users or groups in the identity store that have been granted access to a privileged account managed by an Oracle Privileged Account Manager administrator. Users cannot check out a privileged account unless they have been granted access to that account.

Oracle Privileged Account Manager evaluates grants in the following sequence:

1. When a user tries to access and check out an account, Oracle Privileged Account Manager looks for a user grant for that user. If Oracle Privileged Account Manager finds a user grant, then the user is permitted to check out the account based on that grant and its associated Usage Policy.
2. If Oracle Privileged Account Manager does not find a user grant, it looks for group grants. A user can be a member of many groups. If Oracle Privileged Account Manager finds a group grant for any one of the user's groups, then the user is permitted to check out the account based on that group grant and its associated Usage Policy.

3. If the user is member of multiple groups, and more than one of those groups is available in group grants - then Oracle Privileged Account Manager can pick any one of the matching group grants at runtime. It is indeterministic to say exactly which matching group grant of the multiple ones Oracle Privileged Account Manager will pick at runtime.

4. If Oracle Privileged Account Manager cannot find a user grant or a group grant, then the user is denied access.

Note: Before granting privileged accounts to users or groups, be sure to read, Section 2.4.4, "Avoiding Assignments through Multiple Paths."

10.2 Granting Accounts to Users

Use the following steps to grant access to a privileged account:

1. Locate the account where you want to grant access.
   a. Select Accounts in the Administration accordion.
   b. Click Search in the Search Accounts portlet to populate the Search Results table with a list of all available accounts.

To narrow the results or to locate a particular account, enter search criteria in one or more the Search Accounts fields, and then click Search.

2. Select that account name in the Search Results table.

The Account: Account Name page displays with the General, Grants, Credential Store Framework, and Checkout History tabs.

3. Select the Grants tab.

If any users are already associated with this account, their names are listed in the table in the Users area.

4. Click Add to open the Add Users dialog.

5. In the Add Users dialog, enter all or part of a user name and then click Search.

For example, if you want to add the jjones user, then you could type j, jj, or jon and the search results will include any user names containing those letters.

6. Select (check) one or more user names, and then click Add to make them grantees.

7. Click Close to close the dialog.

The new user's name displays in the Users table.

Note: At this point, the Default Usage Policy is automatically assigned to the user. However, you can use the Usage Policy menu to select a different policy for that user.
10.3 Granting Accounts to Groups

Use the following steps to grant access to a privileged account:

1. Locate the account where you want to grant access.
   a. Select Accounts in the Administration accordion.
   b. Click Search in the Search Accounts portlet to populate the Search Results table with a list of all available accounts.

      To narrow the results or to locate a particular account, enter search criteria in one or more the Search Accounts fields, and then click Search.

2. Select the account name in the Search Results table.

   The Account: Account Name page displays with the General, Grants, Credential Store Framework, and Checkout History tabs.

3. Select the Grants tab.

   If any groups are already associated with this account, their names are listed in the table in the Groups area.

4. Click Add to open the Add Groups dialog.

5. In the Add Groups dialog, enter all or part of a group name and then click Search.

   For example, if you want to add the hr_admin group, then you could type h, hr, or admin and the search results will include any group names containing those letters.

6. Select (check) one or more group names, and then click Add to make them grantees.

7. Click Close to close the dialog.

   The new group name displays in the Groups table.

---

**Note:** At this point, the Default Usage Policy is automatically assigned to the group. However, you can use the Usage Policy menu to select a different policy for that group.

---

10.4 Searching for Grantees

If you have administrator privileges, you can search for grantees by using the following steps

1. Select User Grantees or Group Grantees in the Administration accordion.

2. When the User Grantees or the Group Grantees page displays, use the Search portlet to configure your search.

   ■ To search for a particular grantee, enter one or more letters of the name into the User Name or Group Name field.

   ■ To search for all available grantees, do not specify any search parameters.

3. Click Search.

   Review your search results in the Search Results table.

4. To perform another search, click Reset.
10.5 Opening a Grantee

You can open a grantee to view information about that user or group grantee.

Use one of the following methods to open a grantee from the User Grantees or the Group Grantees page:

- Click the User Name or the Group Name (an active link) in the Search Results table.
- Select the user or group Row number and then click the Open icon.

The User: **UserName** or the Group: **GroupName** page opens where you can review the information about that grantee and the privileged accounts for which they are granted access.

10.6 Removing Grantees from an Account

---

**Note:** Removing a user or group grant from an account does *not* automatically cancel all existing check-outs.

When grantees check out an account, they are guaranteed access to that account until one of the following events occur:

- The user checks in the account
- Oracle Privileged Account Manager automatically checks in the account because the checkout duration has exceeded the expiration period specified by the account’s Usage Policy
- An administrator forces an account check-in

However, after the account is checked in, the grantee cannot check out that account again unless an administrator re-adds them as a grantee.

---

To remove one or more grantees from an account

1. Open the account and select the Grants tab.
2. Select the user or group Row number in the Search Results table.
3. Click the **Remove** icon.
4. When you are prompted to confirm the removal, click the **Remove** button to continue, (or **Cancel** to terminate the operation).

The prompt closes and the user or group is removed from the table.
This chapter provides some background information about Java plug-ins for Oracle Privileged Account Manager and explains how to configure and deploy plug-ins by using the Oracle Privileged Account Manager Console.

This chapter includes the following sections:

- Section 11.1, "What is a Plug-In?"
- Section 11.2, "Developing Plug-Ins for Oracle Privileged Account Manager"
- Section 11.3, "Creating a Plug-In Configuration"
- Section 11.4, "Searching for Plug-In Configurations"
- Section 11.5, "Opening a Plug-In"
- Section 11.6, "Deleting a Plug-In"

**Note:** You can also manage Oracle Privileged Account Manager plug-ins from the command line or by using Oracle Privileged Account Manager’s RESTful interface.

- For information about using the Oracle Privileged Account Manager Command Line Tool (CLI), refer to Section A.7, "Working with Plug-Ins" in Appendix A, "Working with the Command Line Tool."
- For information about using the Oracle Privileged Account Manager RESTful interface, refer to Section B.6, "Account Resource" in Appendix B, "Working with Oracle Privileged Account Manager’s RESTful Interface."

### 11.1 What is a Plug-In?

A plug-in is a customized program that enables you to extend Oracle Privileged Account Manager’s functionality to better meet your specific business and technical requirements. A plug-in enables you to provide custom logic as part of a transaction or by connecting to a custom data source.

An Oracle Privileged Account Manager plug-in can be a Java program that has a configuration entry in the Oracle Privileged Account Manager server. That configuration entry specifies the conditions for invoking the plug-in, which include:

- An operation, such as checkout or update
- A resource, such as an account, a target, or a server
Oracle Privileged Account Manager plug-ins can provide various types of added functionality, such as:

- Validating data before the Oracle Privileged Account Manager server performs an operation on it and performing specified actions after the server performs an operation
- Sending notifications based on Oracle Privileged Account Manager operations
- Performing step-up authentication and authorization
- Authenticating users through external identity stores

Upon start-up, the Oracle Privileged Account Manager server loads your plug-in configuration and library. When the server processes requests, it calls the plug-in functions whenever the specified event takes place.

### 11.2 Developing Plug-Ins for Oracle Privileged Account Manager

This section provides an overview of how you develop plug-ins for Oracle Privileged Account Manager. The topics include:

- Section 11.2.1, "Overview"
- Section 11.2.2, "Supported Languages"
- Section 11.2.3, "Prerequisites"
- Section 11.2.4, "Oracle Privileged Account Manager Plug-In Benefits"
- Section 11.2.5, "Design Guidelines"
- Section 11.2.6, "Framework Description"
- Section 11.2.7, "Supported Operations and Timings"
- Section 11.2.8, "Filtering Rules"

#### 11.2.1 Overview

You can develop plug-ins by using the Oracle Privileged Account Manager plug-in framework, which is shipped in the following jar file:

```
ORACLE_HOME/opam/jlib/opam-plugin-framework.jar
```

**Note:** Currently, Oracle Privileged Account Manager does not ship with any complete plug-ins. Refer to Chapter 16, "Developing Plug-Ins for Oracle Privileged Account Manager" for additional information about developing plug-ins for Oracle Privileged Account Manager.

Figure 10-1 illustrates the Oracle Privileged Account Manager plug-in framework.
You can configure plug-ins for Oracle Privileged Account Manager operations. The plug-ins are invoked whenever the operations are performed and the plug-in filter rules are met. Any type of Oracle Privileged Account Manager client, such as the command line tool or the Oracle Privileged Account Manager Console, can perform these operations. The Oracle Privileged Account Manager server and the plug-in program communicate through the plug-in JAVA interface. When a plug-in is invoked, the Oracle Privileged Account Manager server sends information about the operation and the entities involved to the plug-in. The plug-in then operates on that information and, after completing execution, sends the result back to the Oracle Privileged Account Manager server.

After developing a plug-in, you register it with Oracle Privileged Account Manager. Registration enables Oracle Privileged Account Manager to discover and configure the plug-in to be invoked for Oracle Privileged Account Manager events such as check-ins, checkouts, and so forth.

Oracle Privileged Account Manager’s Java-based plug-in framework enables you to create new plug-ins as well as customize existing ones.

### 11.2.2 Supported Languages

Currently, Oracle Privileged Account Manager only supports plug-ins written in Java.

### 11.2.3 Prerequisites

To develop Oracle Privileged Account Manager plug-ins, you should be familiar with the following topics:

- Oracle Privileged Account Manager
- Oracle Privileged Account Manager RESTful API

In addition, you should have some proficiency programming in Java.
11.2.4 Oracle Privileged Account Manager Plug-In Benefits

Some of the ways in which you can use plug-ins to extend Oracle Privileged Account Manager operations include:

- Validating data before the server performs operations on that data
- Performing actions that you define after the server successfully completes an operation
- Defining extended operations
- Authenticating users through external credential stores
- Replacing an existing server module with a module of your own

Upon start-up, the Oracle Privileged Account Manager server loads your plug-in configuration and library. The server calls your plug-in functions while processing various Oracle Privileged Account Manager requests.

11.2.5 Design Guidelines

Oracle recommends using these guidelines when designing plug-ins:

- Use plug-ins to guarantee that when Oracle Privileged Account Manager performs a specific operation, related actions are also performed.
- Use plug-ins only for centralized, global operations that should be invoked for the program body statement, regardless of which user or application issues the statement.
- Do not create recursive plug-ins. For example, creating a \texttt{pre\_checkout} plug-in that itself issues a \texttt{checkout} statement will cause the plug-in to execute recursively until it has run out of resources.
- Use plug-ins judiciously. Remember, they are executed every time the associated operation occurs.

11.2.6 Framework Description

The Oracle Privileged Account Manager plug-in framework is the environment in which you develop, configure, and apply plug-ins. Each individual plug-in instance is called a \textit{plug-in module}.

The Oracle Privileged Account Manager plug-in framework includes the following:

- Plug-in configuration tools
- Plug-in module interface
- Plug-in RESTful APIs, including Java package \texttt{oracle.xxx.xxx}

To use the Oracle Privileged Account Manager server plug-in framework,

1. Write a user-defined plug-in procedure in Java.
2. Compile the plug-in module.
3. Register the plug-in module through the configuration entry interface by using Oracle Privileged Account Manager's
   - Console, as described in \textit{Section 11.3, "Creating a Plug-In Configuration."
   - Command line tool, as described in \textit{Section A.7, "Working with Plug-Ins."}
11.2.7 Supported Operations and Timings

Oracle Privileged Account Manager supports plug-ins for operations on the following resources:

<table>
<thead>
<tr>
<th>Operations For Target Resources</th>
<th>Operations For Account Resources</th>
<th>Operations For Server Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>■ add</td>
<td>■ add</td>
<td>■ autocheckin (pre/post)</td>
</tr>
<tr>
<td>■ test</td>
<td>■ test</td>
<td>■ passwordcycle (pre/post)</td>
</tr>
<tr>
<td>■ retrieve</td>
<td>■ retrieve</td>
<td>■ resetaccountpassword (post)</td>
</tr>
<tr>
<td>■ update</td>
<td>■ update</td>
<td></td>
</tr>
<tr>
<td>■ remove</td>
<td>■ remove</td>
<td></td>
</tr>
<tr>
<td>■ resetpassword</td>
<td>■ resetpassword</td>
<td></td>
</tr>
<tr>
<td>■ showpassword</td>
<td>■ showpassword</td>
<td></td>
</tr>
<tr>
<td>■ showpasswordhistory</td>
<td>■ showpasswordhistory</td>
<td></td>
</tr>
<tr>
<td>■ checkout</td>
<td>■ checkout</td>
<td></td>
</tr>
<tr>
<td>■ checkin</td>
<td>■ checkin</td>
<td></td>
</tr>
<tr>
<td>■ sessioncheckout</td>
<td>■ sessioncheckout</td>
<td></td>
</tr>
</tbody>
</table>

When developing plug-ins, Oracle Privileged Account Manager enables you to specify when those plug-ins are executed. Oracle Privileged Account Manager supports pre and post operation timings for plug-ins, which are described in the following sections:

- **Pre-Operation Plug-Ins**
- **Post-Operation Plug-Ins**

11.2.7.1 Pre-Operation Plug-Ins

Oracle Privileged Account Manager adds pre-operation plug-ins to a queue and executes them in a specified order before performing the designated operation. As Oracle Privileged Account Manager executes each pre plug-in that is in the queue, the results are passed from the current plug-in to the next one in the queue. For example, if a target-add pre plug-in modifies the plug-in description, then the next plug-in sees the modified description.

Oracle Privileged Account Manager will not perform the operation until all pre plug-ins have successfully finished executing. If a pre plug-in fails or times out, then the operation also fails and Oracle Privileged Account Manager will not execute any of the other pre plug-ins queued for that operation.

Adding lots of pre plug-ins may increase the time taken for the operation because Oracle Privileged Account Manager must execute all pre plug-ins before performing the operation.

11.2.7.2 Post-Operation Plug-Ins

Oracle Privileged Account Manager adds post-operation plug-ins to a queue and executes them after performing the designated operation. Oracle Privileged Account Manager executes post plug-ins for both successful and failed operations. Post plug-ins see the results, including the success or failure status of the operation, but they cannot modify the results because the operation has already completed. Oracle Account Manager
Privileged Account Manager executes all queued post plug-ins and it does not matter if some of them fail. Results from one post plug-in cannot be passed to another.

Adding post plug-ins does not increase the time taken for the operation because they are not executed until after the operation is performed.

11.2.8 Filtering Rules

The Oracle Privileged Account Manager server executes plug-ins on specifically configured operations for specific end users. You can configure filtering rules that determine for which users or groups a plug-in will or will not be executed, and for which results codes a post plug-in will be executed.

---

**Note:** Only administrators with the Security Administrator Admin role can add, edit, or remove Filter Rules.

- To configure filtering rules from the Console, refer to Section 11.3, "Creating a Plug-In Configuration," step 6 for instructions.
- To configure filtering rules from the command line, refer to Section A.7.1, "addplugin Command," for instructions.

---

**For Users and Groups**

You can specify the users or groups for which a plug-in should be executed by adding them to an Enabled user or group list. Similarly, you can specify the users or groups for which a plug-in should not be executed by adding them to a Disabled user or group list.

Oracle Privileged Account Manager evaluates filtering rules in the following sequence to decide which rule takes precedence over another:

1. HTTP Result Code (for post plug-ins)
2. Disable user
3. Enable user
4. Disable group
5. Enable group

In general, rules that are defined at the user level override those that are defined at the group level, because the user level rules are more specific. User level rules target a specific user rather than a group of users.

In addition, the Disabled lists take precedence over Enabled lists.

---

**Note:** Result codes override all other filtering rules.

---

For example, assume person1 is a user in the Administrators group. If you put person1 in the Disabled user list and Administrators into the Enabled group list, then the Oracle Privileged Account Manager will not invoke the plug-in for person1 because the server checks the Disabled user list before checking the Enabled group list, and because the user-level rule overrides the group-level rule.

However, if you put person1 in the Enabled user list and put Administrators in the Disabled group list, then person1 can invoke the plug-in because the Enabled user check is performed before the Disabled group check.
If there are no values in these four fields, then all users and groups can invoke the plug-in. However, as long as there is one user or group in the Enabled user list or Enabled group list, then only that user or group can invoke the plug-in. No others can invoke that plug-in. If person1 is the only user in the Enabled user list, then all other users and groups are prevented from invoking the plug-in.

**Note:** The Filtering Rules evaluation sequence stops when it finds a match. For example, if the filter finds an Enabled user that matches the user who is performing the action, then the filtering stage stops. It does not matter if the user is present in any Enable or Disable group filters.

**For Results Codes**

After performing an Oracle Privileged Account Manager operation, the server returns an HTTP response containing an HTTP status integer (such as 200 for success, 201 for creation, 401 for insufficient privileges, and so on.)

You can configure filtering rules for post plug-ins that are based on one or more HTTP result code values. For example, if you specify a filtering rule for result code 200, then the server will only execute the post plug-in when the result status is 200.

**Note:** Result codes override all other filtering rules.

### 11.3 Creating a Plug-In Configuration

Creating a plug-in configuration means to register details about the plug-in with the plug-in resource (an account, a server, or a target).

**Note:** You must be an Oracle Privileged Account Manager administrator with the Application Configurator Admin Role to create plug-ins.

When you create a new plug-in configuration, the status is disabled and the plug-in cannot be executed. Only an administrator with the Security Administrator Admin Role can enable plug-in configurations and determine under which conditions those plug-ins can be executed.

To create a plug-in configuration from the Console,

1. Select **Plug-in Configuration** in the Configuration accordion to open the Search Plug-in Configuration page.
2. Click **Create**, located in the Search Results table toolbar.
3. When the Plug-In Configuration: **Untitled** page displays, provide the following information in the Configuration Settings section:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name for the new plug-in configuration.</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a description for this plug-in configuration.</td>
</tr>
</tbody>
</table>
4. To configure a Java classpath where the plug-in jar file is located, use the Class Paths section as follows:
   - To add a classpath, click Add.
     When a new row is displayed in the table, type the Java classpath into the blank field. For example,
     /u01/plugins/emailplugin.jar
     
     **Note:** The Oracle Privileged Account Manager server process must be able to access the specified class path files. You can specify any type of location, such as local file system, network file system, etc.

   - To delete a classpath, select that classpath row in the table, and then click Remove.

---

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>Enable an option to configure the plug-in's execution status at runtime by selecting one of these options: ■ <strong>Active</strong> (default): Allow the plug-in to execute. ■ <strong>Disabled</strong> (default): Do not allow the plug-in to execute.</td>
</tr>
<tr>
<td>Resource Type</td>
<td>Choose the type of resource on which the plug-in will perform: ■ account ■ server ■ target</td>
</tr>
<tr>
<td>Operation</td>
<td>Choose the operation that the plug-in will perform. <strong>Note:</strong> Refer to Section 11.2.7, &quot;Supported Operations and Timings&quot; for a complete list of supported operations.</td>
</tr>
<tr>
<td>Timing</td>
<td>Specify when you want Oracle Privileged Account Manager to execute the plug-in by choosing one of the following options: ■ <strong>pre</strong>: Executes the plug-in before performing the Oracle Privileged Account Manager operation. ■ <strong>post</strong>: Executes the plug-in after performing the Oracle Privileged Account Manager operation. <strong>Note:</strong> Refer to Section 11.2.7, &quot;Supported Operations and Timings&quot; for more information.</td>
</tr>
<tr>
<td>Order</td>
<td>Enter a value to specify the order in which a plug-in is queued for execution in relation to other plug-ins. For example, plug-in 1 is executed before plug-in 2. (Minimum is 1)</td>
</tr>
<tr>
<td>Timeout</td>
<td>Specify a value to indicate the maximum duration (in seconds) for which the plug-in can be executed. When the plug-in execution exceeds this time out period, Oracle Privileged Account Manager aborts the plug-in execution. (Default is 120 seconds, minimum is 10 seconds) <strong>Note:</strong> For additional information, refer to Section 11.2.7, &quot;Supported Operations and Timings.&quot;</td>
</tr>
<tr>
<td>Plug-in Class Name</td>
<td>Enter the name of the Java class that implements the plug-in's interface.</td>
</tr>
<tr>
<td>Plug-in Version</td>
<td>Enter the plug-in's Java version number. <strong>Note:</strong> Oracle Privileged Account Manager does not actually use the plug-in version. Instead, Oracle Privileged Account Manager uses the jar file listed in the plug-in's directory.</td>
</tr>
</tbody>
</table>
5. To configure custom attributes for the plug-in, expand and use the Custom Attributes section as follows:
   - To add an attribute, click Add.
     When a new row is displayed in the table, type the Attribute Name and Value into the blank fields. For example, for an email notification plug-in, you might create a notificationemail attribute with a value of abc@abc.com.
   - To delete a custom attribute, select that attribute’s row in the table, and then click Remove.

6. To configure filtering rules that determine when Oracle Privileged Account Manager executes the plug-in, expand and use the Filter Rules section as follows:

   **Note:** You must be an Oracle Privileged Account Manager administrator with the Security Administrator Admin Role to add, edit, or remove Filter Rules.

   For information about how Oracle Privileged Account Manager uses filtering rules, refer to Section 11.2.8, "Filtering Rules."

   - Expand the Users or Groups sections to specify for which users or groups the server can or cannot invoke the plug-in.
     a. Select the Enabled tab or Disabled tab and click Add.
     b. When the Add dialog displays, enter one or more letters of a name into the User Name field or Group Name field and click Search.
     c. When the search results display, select the row you want, and click Add.
        A success message displays above the search results list.
     d. Continue selecting and adding users or groups until you are finished, then click Close.
   - Expand the Result Codes section to configure filtering rules for a post plug-in that are based on one or more result codes.
     **Note:** You cannot configure result codes for pre plug-ins.

   a. Click Add.
   b. When a new HTTP Result Code row displays, enter an enabled HTTP response code into the blank field. For example, type 200 to execute a post plug-in when the response status is a successful request.

   - To delete a Filter Rule, select that rule row in the applicable table, and then click Remove.

7. After setting all of the necessary plug-in configuration parameters, click Test to verify that the configuration is valid.

   This test checks whether Oracle Privileged Account Manager can load the configured plug-in and whether it implements the required plug-in interface. Testing catches common issues, such as plug-in jars that are configured with the wrong file paths, plug-ins that implement the wrong interface, pre plug-ins that implement a post plug-in interface or vice versa, etc.
This test does not execute the plug-in or validate any plug-in custom attributes, which are only used by the custom plug-in logic itself.

If the configuration is valid, a "Test Succeeded" message displays.

8. Click Save to create the new configuration.

Oracle Privileged Account Manager automatically assigns a Plug-In GUID, which is displayed in the Configuration Settings section.

11.4 Searching for Plug-In Configurations

You can search for plug-in configurations by using one or more of the following parameters:

- Name
- Description
- Resource Type (All, account, server, or target)
- Status (All, Active, or Disabled)
- Timing (All, pre, or post)
- Operation (All, accountpasswordchange, add, autocheckin, checkin, checkout, passwordcycle, remove, resetpassword, retrieve, sessioncheckout, showpassword, showpasswordhistory, test, or update)

To search for a plug-in,

1. Select Plug-in Configuration in the Configuration accordion.

2. When the Plug-in Configuration page displays, use the Search portlet parameters to configure your search.
   - For example, to search for a list of all active plug-ins, select Active from the Status menu.
   - To search for all available plug-ins, do not specify any search parameters.

3. Click Search.

   Review your search results in the Search Results table, which contains a column for all of the search fields and a column for the Plug-In Order.

4. To perform another search, click Reset.

11.5 Opening a Plug-In

You can open a plug-in to view or edit the configuration parameters for that plug-in.

Note: You must be an Oracle Privileged Account Manager administrator with the Security Administrator Admin Role or the Application Configurator Admin Role to search for and view plug-ins.
To open a plug-in, open the Plug-in Configuration page and perform one of the following actions:

- Click the Plug-in Name (an active link) in the Search Results table.
- Select the plug-in Row and then click Open.

The Plug-in Configuration: Plug-in Name page opens where you can access the plug-in’s configuration settings, custom attributes, users or groups, and current status (active or disabled).

If you edited any of these settings, click Test to validate your changes. When you see the “Test Succeeded” message, click Save.

### 11.6 Deleting a Plug-In

To delete a plug-in configuration,

1. Locate the plug-in to remove.
   a. Select Plug-in Configuration in the Configuration accordion.
   b. Click Search in the Search portlet to populate the Search Results table with a list of all available plug-ins.
      
      To narrow the results or to locate a particular plug-in, enter search criteria in one or more the Search fields, and then click Search.

2. In the Search Results table, select the plug-in to be removed and then click Delete.
   
   When you are prompted to confirm the deletion, click Delete to continue or Cancel.
Deleting a Plug-In
This chapter provides instructions for self-service end users working with Oracle Privileged Account Manager.

This chapter includes the following sections:

- Section 12.1, "Introduction to Using Self Service"
- Section 12.2, "Viewing Your Accounts"
- Section 12.3, "Searching for Accounts"
- Section 12.4, "Opening Accounts"
- Section 12.5, "Checking Accounts Out and In"
- Section 12.6, "Viewing Your Checked-Out Accounts"
- Section 12.7, "Checking Out Privileged Account Sessions"
- Section 12.8, "Showing a Password"

12.1 Introduction to Using Self Service

Self-service users do not have any Oracle Privileged Account Manager administrator privileges or Admin Roles.

The basic workflow for a self-service user includes:

1. Viewing your accounts
2. Searching for accounts
3. Checking out accounts
4. Viewing checked-out accounts
5. Checking in accounts
6. Checking out a session
7. Viewing checked-out sessions
8. Checking in a session
9. Viewing an account password
12.2 Viewing Your Accounts

To view a list of all the accounts for which you are currently a grantee, select **My Accounts** on the Home accordion and then click **Search**.

The My Accounts page is refreshed and lists all of your accounts in the Search Results table. From this page you can

- Search your accounts.
- View the account name, the associated target name and target type, the domain, and a description.
- Open an account to review the associated target information and account information, which includes the Usage Policy associated with that account.
- Check out passwords and sessions.
- Control how information is displayed in the table by managing which columns are displayed and in which order.
- Refresh the list of displayed accounts after making changes.

12.3 Searching for Accounts

To search for an account, follow the instructions provided in Section 8.3, "Searching for Privileged Accounts."

12.4 Opening Accounts

To view information about an account for which you are a grantee:

You can open privileged accounts from the My Accounts or My Checkouts page.

**From the My Accounts Page**

1. Select **My Accounts** in the Home accordion.
2. Click Search to see all of your accounts displayed.
   
   Alternatively, you can narrow the results by configuring one or more of the Search Accounts parameters, as described in Section 8.3, "Searching for Privileged Accounts," and then click **Search**.
3. When the results display in the Search Results table, locate the account you want to open, and perform one of the following actions:
   - Click the Account Name (an active link) in the Search Results table.
   - Select the account Row and then click **Open**.

The Account: *AccountName* page opens with the target and account information.
From the My Checkouts Page
1. Select My Checkouts in the Home accordion.
2. When the My Checkouts page displays, locate the account you want to open in the search results table, and then click the Account Name (an active link).

The Account: AccountName page opens with the target and account information.

12.5 Checking Accounts Out and In
To check out a privileged account granted to you, follow the instructions provided in Section 8.5, "Checking Out Privileged Accounts."

To check an account back in again, follow the instructions provided in Section 8.6, "Checking In Privileged Accounts."

12.6 Viewing Your Checked-Out Accounts
To view a listing of all accounts you currently have checked-out, select My Checkouts on the Home accordion.

The My Checkouts page displays with all of your checked-out accounts listed in the Search Results table, as shown in

Figure 12–1 Example of Checked Out Accounts

12.7 Checking Out Privileged Account Sessions
To check out a privileged account session granted to you, follow the instructions provided in Section 8.5.2, "Checking Out Privileged Account Sessions."

**Note:** You do not have to perform any special steps to check in a checked out session. If you use the procedure described in Section 8.6, "Checking In Privileged Accounts," then the account is checked back in regardless of the checkout type (password or session).

12.8 Showing a Password
If necessary, you can view the current password in clear text for an account that you have checked out by using the Show Password option. For example, if you forget a password, you can use this feature to view the password again.

Any user can view a password for an account they have checked out. However, you cannot access passwords for accounts that are checked in or for accounts that are checked out by other users. Attempts to do so will cause an error.

To view a password, refer to Section 8.8.1, "Showing an Account Password."
Part III

Monitoring Oracle Privileged Account Manager

This part provides information about monitoring Oracle Privileged Account Manager, and it contains the following chapters:

■ Working with Reports
■ Managing Oracle Privileged Account Manager Auditing and Logging
This chapter provides information about how to work with the different Oracle Privileged Account Manager reports.

This chapter includes the following sections:

- Section 13.1, "Viewing a Report"
- Section 13.2, "Working with Deployment Reports"
- Section 13.3, "Working with Usage Reports"
- Section 13.4, "Working with Failure Reports"
- Section 13.5, "Working with Checkout History Reports"

**Note:** You can manage Oracle Privileged Account Manager reports from the command line or by using Oracle Privileged Account Manager’s RESTful interface.

- For information about using the Oracle Privileged Account Manager Command Line Tool (CLI), refer to Appendix A, "Working with the Command Line Tool."
- For information about using the Oracle Privileged Account Manager RESTful interface, refer to Appendix B, "Working with Oracle Privileged Account Manager’s RESTful Interface."

### 13.1 Viewing a Report

Oracle Privileged Account Manager reports are real-time reports that provide information about the current status of accounts and targets being managed by Oracle Privileged Account Manager.

**Note:** You must be an Oracle Privileged Account Manager administrator with the Security Auditor Admin Role to open and review Oracle Privileged Account Manager reports.

To view a report, expand the Reports accordion and click a Report link. The report information is displayed in the Reports page on the right.
13.2 Working with Deployment Reports

Select the Deployment Reports link to view information about how targets and privileged accounts are currently deployed. Figure 13–1 shows a sample report.

Figure 13–1  Sample Deployment Report

Information about the deployment is organized into the following portlets:

- **Target and Accounts Deployment table.** Provides a list of targets, including their target type and host names. Expand the arrow icon next to a target name to view the accounts associated with that target.

  **Tip:** You can click a link in the Target/Account column to open the configuration page for that target or account.

- **Target Distribution.** This portlet illustrates how targets are distributed within your deployment.

- **Account Distribution.** This portlet illustrates how accounts are distributed within your deployment, by Organization.

You can use the Show and Filter drop-down menus to control how the report content is displayed.

- Use the Show menu to view all targets or to just view a particular target.

- Use the Filter menu to view the report information as a bar graph, pie chart, or in tabular format.
13.3 Working with Usage Reports

Select the **Usage Report** link to view information about how privileged accounts are currently being used in your deployment.

**Figure 13–2 Sample Usage Report**

![Sample Usage Report](image)

This usage information is organized into the following portlets:

- **Account Usage.** This portlet provides a list of targets and accounts, including the target types, host names, and the last checked out date.
  
  - Expand the arrow icon next to a target name to view the accounts associated with that target.
  
  - Use the **Show** menus to constrain the view to just currently checked out accounts or accounts that were checked out in the last hour, day, or week. Also you can view all accounts or a subset of accounts.

- **Checked Out Accounts.** This portlet illustrates which accounts are checked out within your deployment. You can use the **Filter** menu to display the report information as a bar graph, pie chart, or in tabular format.

13.4 Working with Failure Reports

Select the **Failure Report** link to view information about the current state of target and account failures.

Information about target or account failures is organized into the following portlets:

- **Targets and Accounts Failures.** This portlet provides a list of targets, the target status, last error message, and the last failure date. Expand the arrow icon next to a target to view the accounts associated with that target.

- **Target Failures.** This portlet illustrates the target failures within your deployment.

- **Account Failures.** This portlet illustrates the account failures within your deployment.

You can use the **Show** and **Filter** drop-down menus to control how the report content is displayed.
- Use the **Show** menu to view the errors that occurred during the last 24 or 48 hours, the last week, or the last 30 days.
- Use the **Filter** menu to view the report information as a bar graph, pie chart, or in tabular format.

### 13.5 Working with Checkout History Reports

Use the Checkout History report to view information about the account checkouts performed over a specified period of time. This report consists of the following information:

- **Start Date** and **End Date**: Date that the account was checked out and checked back in, respectively.
- **Account Name**: Name of the checked out account.
- **User Name**: Name of the user who checked out the account.
- **Target Name**: Name of the target associated with the account.
- **Recording**: Indicates a recording (or transcript) of the user’s actions during the checkout is available for viewing.

---

**Note:** Only an administrator can view session recordings. Refer to Section 8.7, “Viewing a Session Recording” for more information.

---

To view a Checkout History report,

1. Select the **Checkout History Report** link from the Reports accordion to open the Checkout History page.
2. Use the Search Checkout History portlet to configure search parameters:
   - You must specify a **Start Date** and an **End Date** range in which to search for checkouts. Type the date and time into the blank fields or use the **Calendar** icons.
   - Enter information into one or more of the **Account Name**, **User Name**, **Target Name**, or **Pattern** fields.

---

**Note:** Use the **Pattern** field to search for a string in the recording of a checkout event.

---

- Enter a value into the **Query** field to limit the number of returned results.
3. Click **Search** and the results will display in the table. For example,
Figure 13–3  Example Checkout History Table

![Example Checkout History Table]

**Note:** Click the **Recording** icon ( ) to open a transcript of the user’s actions during the checkout.
Managing Oracle Privileged Account Manager Auditing and Logging

This chapter describes how to configure and use Oracle Privileged Account Manager’s auditing and logging functionality.

This chapter includes the following sections:

- Section 14.1, "Understanding Oracle Privileged Account Manager Auditing"
- Section 14.2, "Understanding Oracle Privileged Account Manager Logging"

Note: If you are using Oracle Privileged Account Manager on IBM WebSphere, refer to "Differences in Managing Oracle Privileged Account Manager Auditing and Logging" in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management for information about this topic.

14.1 Understanding Oracle Privileged Account Manager Auditing

Oracle Privileged Account Manager audits all security events that occur under its purview, which gives you better visibility into how privileged accounts are used within your organization and enables you to effectively manage sensitive information.

Specifically, the Oracle Privileged Account Manager audit logger logs any events that modify entity states; such as when you add, modify, or remove new accounts, targets, or policies.

The following table describes all of the event categories and event types for which an audit can be generated:

<table>
<thead>
<tr>
<th>Event Category</th>
<th>Event Types</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Management</td>
<td>Add Account</td>
<td>Events related to managing principal accounts</td>
</tr>
<tr>
<td></td>
<td>Change Password</td>
<td>Changes to user passwords</td>
</tr>
<tr>
<td></td>
<td>Disable Account</td>
<td>Disabling users, groups, or any other principal accounts</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Note: A principal can be an end-user or a pseudo-user (a service within the system).</td>
</tr>
</tbody>
</table>

Table 14–1 Audited Oracle Privileged Account Manager Events
Logging these audit events creates a processing history that allows reporting tools to gather statistics, as described in Section 14.1.2, "Understanding Oracle Privileged Account Manager Audit Reports."

### 14.1.1 Configuring Auditing in Oracle Privileged Account Manager

You can configure Oracle Privileged Account Manager to save audit events into a database or a file. When a database is not available, Oracle Privileged Account Manager saves its audit logs into this file,

```
DOMAIN_HOME/servers/<opamserv>/logs/auditlogs/OPAM
```

You can also configure Oracle Privileged Account Manager to deploy audit reports in BI Publisher (version 11.1.1.5.0 or higher), and use BI Publisher to view audit events in the database. Reports in BI Publisher are only possible if the audit events are being pushed into a database and not a file.

The following topics provide instructions for configuring auditing in Oracle Privileged Account Manager:

- Configuring File-Based Auditing in Oracle Privileged Account Manager
- Configuring Database-Based Auditing in Oracle Privileged Account Manager
- Deploying Oracle Privileged Account Manager Audit Reports in BI Publisher
- Setting the Audit Logging Levels
Note: To configure auditing for Oracle Privileged Account Manager on an IBM WebSphere server, refer to "Configuring Auditing for Oracle Privileged Account Manager" in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management before starting the procedures described in this section.

14.1.1.1 Configuring File-Based Auditing in Oracle Privileged Account Manager

This section describes how to configure file-based auditing in Oracle Privileged Account Manager.

Before You Begin

Before starting the following configuration steps, review these publications:

- "Using WLST Online or Offline" in the Oracle Fusion Middleware Oracle WebLogic Scripting Tool
- "OPSS Scripts for Auditing" in the Oracle Fusion Middleware Application Security Guide for detailed information about the getAuditPolicy, setAuditPolicy, getAuditRepository, and setAuditRepository WLST audit commands used in the configuration steps.

To configure Oracle Privileged Account Manager for file-based auditing:

1. Start the WebLogic Scripting Tool (WLST) and connect to the Oracle WebLogic Server:
   a. Open a command window and navigate to the following directory, which contains the WLST:
      
      MW_HOME/oracle_common/common/bin
   
   b. Start WLST by typing one of the following commands:
      
      On UNIX, type: sh wlst.sh
      On Windows, type: wlst.cmd
      
      You know that WLST has started when the command prompt changes to wls:/offline.
   
   c. Connect to the Oracle WebLogic Server by typing the following command:
      
      connect('WLS_Admin_Name','WLS_Admin_Password','WLS_Machine_Name:Port')
      
      For example,
      
      connect('weblogic','Welcome1','localhost:7004')
      
      WLST validates the administrator's username and password, the machine name, and the port that are associated with the WebLogic Admin Server. If all of these values are correct, WLST connects to the WebLogic Admin Server and the command prompt changes to
      
      wls:/base_domain/serverConfig

Note: Refer to "Securing Access from WLST Online" in the Oracle Fusion Middleware Oracle WebLogic Scripting Tool for additional information.
2. To set the audit logging level for Oracle Privileged Account Manager:
   
a. If the `filterPreset` parameter is set to `NONE`, use the `setAuditPolicy` command to change the value to `All`, `Medium`, or `Low`, based on how much logging you want Oracle Privileged Account Manager to provide:

```
setAuditPolicy(filterPreset='All')
```

A confirmation message displays to indicate the audit logging level was successfully updated.

---

**Note:** For a description of the different logging levels, refer to the table on page 14-10.

---

b. Verify the current logging level for Oracle Privileged Account Manager, by typing `getAuditPolicy()` at the prompt, and then checking the `filterPreset` parameter value.

3. To change the Repository Type to database (DB):
   
a. Type the `setAuditRepository` command as follows:

```
setAuditRepository(switchToDB='true')
```

A confirmation message displays to let you know that the audit repository was successfully updated.

b. You can use the WLST `getAuditRepository` command to verify that the audit repository is set to database-based auditing:

```
getAuditRepository()
```

The `setAuditRepository` parameter value (as indicated by the Repository Type field) should be `FILE`.

4. Restart both the Administration Server and the Oracle Privileged Account Manager Managed Server.

---

**Note:** For detailed information about starting a Managed Server, refer to “Starting or Stopping the Oracle Stack” in the Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management.

---

You must restart both servers for your changes to take effect. After the server restarts, audit logs will start appearing in this location:

```
DOMAIN_HOME/servers/<opamserv>/logs/auditlogs/OPAM
```

### 14.1.1.2 Configuring Database-Based Auditing in Oracle Privileged Account Manager

This section describes how to configure Oracle Privileged Account Manager to save audit events into the Oracle database that is associated with Oracle Privileged Account Manager.

**Prerequisites**

Before starting the following configuration steps,
Review these publications:

- "Using WLST Online or Offline" in the Oracle Fusion Middleware Oracle WebLogic Scripting Tool
- "WLST Commands for Auditing" in the Oracle Fusion Middleware Application Security Guide for detailed information about the getAuditPolicy, setAuditPolicy, getAuditRepository, and setAuditRepository WLST audit commands used in the configuration steps.

Install the following

- A database
- The Repository Creation Utility application, which is used to create a schema and load a repository into the database.


To configure database-based auditing:

1. Start the WebLogic Scripting Tool (WLST) and connect to the Oracle WebLogic Server:
   a. Open a command window and navigate to the following directory, which contains the WLST:
   
   \MW_HOME/oracle_common/common/bin

   b. Start WLST by typing one of the following commands:

   On UNIX, type: sh wlst.sh
   
   On Windows, type: wlst.cmd

   You know that WLST has started when the command prompt changes to wls:/offline.

   c. Connect to the Oracle WebLogic Server by typing the following command:

   connect('WLS_Admin_Name','WLS_Admin_Password','WLS_Machine_Name:Port')

   For example,

   connect('weblogic','Welcome1','localhost:7004')

   WLST validates the administrator's username and password, the machine name, and the port that are associated with the WebLogic Admin Server. If all of these values are correct, WLST connects to the WebLogic Admin Server and the command prompt changes to wls:/base_domain/serverConfig

Note: Refer to "Securing Access from WLST Online" in the Oracle Fusion Middleware Oracle WebLogic Scripting Tool for additional information.
2. To set the audit logging level for Oracle Privileged Account Manager:
   a. If the filterPreset parameter is set to NONE, use the setAuditPolicy command to change the value to All, Medium, or Low, based on how much logging you want Oracle Privileged Account Manager to provide:

      ```
      setAuditPolicy(filterPreset='All')
      ```

      A confirmation message displays to indicate the audit logging level was successfully updated.

      **Note:** For a description of the different logging levels, refer to the table on page 14-10.

   b. Verify the current logging level for Oracle Privileged Account Manager, by typing `getAuditPolicy()` at the prompt, and then checking the filterPreset parameter value.

3. To change the Repository Type to database (DB):
   a. Type the setAuditRepository command as follows:

      ```
      setAuditRepository(switchToDB='true')
      ```

      A confirmation message displays to let you know that the audit repository was successfully updated.

   b. You can use the WLST `getAuditRepository` command to verify that the audit repository is set to database-based auditing:

      ```
      getAuditRepository()
      ```

      The `setAuditRepository` parameter value (as indicated by the Repository Type field) should be DB.

4. Use the Repository Creation Utility to create and load the audit schema into the database, and then use the WebLogic Server Administrative Console to create a new JDBC data source.

   A data source contains credentials that BI Publisher needs to connect to the Oracle database associated with Oracle Privileged Account Manager. BI Publisher uses this connection to retrieve data from the Oracle Privileged Account Manager database. BI Publisher then uses this data to generate reports for targets, privileged accounts, grants, and policies.

   **Note:** Instructions for creating the audit schema and for creating a JDBC data source are provided in the "Configuring and Managing Auditing" section of the Oracle Fusion Middleware Application Security Guide.

5. Restart both the Administration Server and the Oracle Privileged Account Manager Managed Server.

   You must restart both servers for your changes to take effect. After restarting both servers, audit logs will start appearing in the installed database.
14.1.1.3 Deploying Oracle Privileged Account Manager Audit Reports in BI Publisher

This section describes how to deploy Oracle Privileged Account Manager audit reports in Oracle Business Intelligence Publisher (BI Publisher), a component used to manage and deliver reports.

Use the following steps:

1. Install and configure BI Publisher version 11.1.1.5.0 or higher if it is not already installed.

   Refer to "Configuring Oracle Business Intelligence Publisher" in the Oracle Fusion Middleware Administrator’s Guide for Oracle Identity Navigator for instructions.

2. After installing BI Publisher, locate the following directory in the WebLogic domain:

   \( \text{BI\_DOMAIN\_HOME/cfg/bupublisher/repository/Reports} \)

3. Locate the \text{opam\_product\_BIP11gReports\_11\_1\_2\_1\_0.zip} file in the following directory:

   \( \text{ORACLE\_HOME/opam/reports} \)

   Unzip this file into the Reports folder noted in step 2 and verify that the following directory was created:

   \( \text{ORACLE\_HOME/opam/reports/Oracle Privileged Account Manager} \)

4. To set up the catalog and configure data sources, open a browser window and enter the URL for BI Publisher.

   The format for this URL is

   \( \text{http://hostname:port/xmlpserver/} \)

   For example

   \( \text{http://localhost:7001/xmlpserver/} \)

5. When the BI Publisher login page displays, log in as a user with WebLogic privileges and click Sign In.

6. Set up the catalog as follows:

   a. Select Administration > System Maintenance > Server Configuration.

   b. When the System Maintenance page displays, go to the Path field in the Configuration Folder section and enter the path to your Configuration folder.

   For example,

   \( \text{BI\_DOMAIN\_HOME/cfg/bupublisher/repository} \)

   The files that contain your server configuration settings (such as the JDBC data source you created in step 4 of Section 14.1.1.2) are stored in a Configuration folder. The path to this folder is stored in the xmlp-server-config.xml configuration file. The xmlp-server-config.xml file is located in

   \( \text{BI\_DOMAIN\_HOME/cfg/bupublisher/repository/Admin/Configuration} \)
c. Locate the Catalog section on the System Maintenance page and specify the following information:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Parameter Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catalog Type</td>
<td>Select BI Publisher - File System from the menu.</td>
</tr>
<tr>
<td>Path</td>
<td>Enter the path to the BI Publisher Catalog folder. For example, <code>BI_DOMAIN_HOME/config/bipublisher/repository</code></td>
</tr>
</tbody>
</table>

**Caution:** The path to the BI Publisher Catalog includes the `reports` subdirectory where you unpacked the Oracle Privileged Account Manager reports. Do not include the `reports` subdirectory in the Path field or you will corrupt BI Publisher.

**Note:** Because the file system contains the reports repository, the platform where you are running BI Publisher determines the case-sensitivity of folder and report names. Repository object names are not case-sensitive in a Windows-based environment, but they are case-sensitive in a UNIX-based environment.

d. Click Apply.

A confirmation message is displayed.

e. Log in as an administrator.

f. Click Catalog to open the Shared Folder/Oracle Privileged Account Manager folder.

**Note:** If this folder does not display, restart the application from the WebLogic console.

7. One JDBC (Oracle Privileged Account Manager JDBC) connection is required for Oracle Privileged Account Manager reports. Use the following steps to define an Oracle Privileged Account Manager JDBC connection and define the data sources:

a. Click the Administration link found on the right side of the BI Publisher page.

The BI Publisher Administration page displays. (Note the Data Sources section on this page.)

b. Click the JDBC Connection link found in the Data Sources section.

c. When the Data Sources page displays, click Add Data Source in the JDBC section to create a JDBC connection to your database.

d. On the Add Data Source page, enter the following information:

<table>
<thead>
<tr>
<th>Data Source Name</th>
<th>OPAM JDBC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Driver Type</td>
<td>Select a driver type to suit your database (for example, Oracle 10g or Oracle 11g).</td>
</tr>
<tr>
<td>Database Driver Class</td>
<td>oracle.jdbc.driver.OracleDriver (Define a driver class to suit your database.)</td>
</tr>
<tr>
<td>Connection String</td>
<td>Provide the database connection details. For example, <code>hostname:port:sid</code>.</td>
</tr>
<tr>
<td>User name</td>
<td>Provide the Oracle Privileged Account Manager Audit DB user name.</td>
</tr>
</tbody>
</table>
If the connection to the database is established, a confirmation message is displayed indicating the success.

e. Click Apply.

You should see this newly defined connection (Oracle Privileged Account Manager JDBC) in the list of JDBC Data Sources.

f. Navigate to Oracle Privileged Account Manager Audit Reports.

The Catalog page is displayed as a tree structure on the left side of the page with details on the right.

g. Expand Shared Folders and select the Oracle Privileged Account Manager folder to view all of the objects in that folder.

8. Use Oracle Identity Navigator to configure a connection to the BI Publisher server.

Refer to "Creating a Connection to BI Publisher" in the Oracle Fusion Middleware Administrator's Guide for Oracle Identity Navigator for the necessary instructions.

When you configure the connection successfully, the My Reports section of the Oracle Identity Navigator Dashboard page will contain the link, **Click here to create reports**.

In addition, users with the Security Auditor role can now perform the following tasks:

- View Oracle Identity Management BI Publisher reports and audit reports
- Select and add reports to the My Reports list
- View and run any reports for which you have access privileges

You can now navigate in BI Publisher and use the Oracle Privileged Account Manager 11g BI reports.

### 14.1.1.4 Setting the Audit Logging Levels

To change the amount of audit logging provided by Oracle Privileged Account Manager, use the following steps:

1. Launch an application server shell (WLST) and establish a connection to the Oracle WebLogic Server as described in step 4 of Section 14.1.1.2, "Configuring Database-Based Auditing in Oracle Privileged Account Manager."

   **Note:** Refer to "Securing Access from WLST Online" in the Oracle Fusion Middleware Oracle WebLogic Scripting Tool for more information.

2. Use the `getAuditPolicy` command to get the current audit policy.
If the FilterPreset field is set to NONE, use the `setAuditPolicy` command to change the value. Choose one of the options noted the following table, depending on the type of events to be audited:

<table>
<thead>
<tr>
<th>Option</th>
<th>Logged Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>Logs all event types.</td>
</tr>
<tr>
<td>Medium</td>
<td>Logs the following event types:</td>
</tr>
<tr>
<td></td>
<td>- In the AccountManagement category: ChangePassword, CheckinAccount, CreateAccount, DeleteAccount, DisableAccount, EnableAccount, ModifyAccount, and QueryAccount</td>
</tr>
<tr>
<td></td>
<td>- In the PolicyManagement category: All</td>
</tr>
<tr>
<td></td>
<td>- In the TargetManagement category: All</td>
</tr>
<tr>
<td>Low</td>
<td>Logs the following event types:</td>
</tr>
<tr>
<td></td>
<td>- In the AccountManagement category: ChangePassword, CheckinAccount, CreateAccount, DeleteAccount, DisableAccount, EnableAccount, and ModifyAccount</td>
</tr>
<tr>
<td></td>
<td>- In the PolicyManagement category: CreatePolicy, DeletePolicy, and ModifyPolicy</td>
</tr>
<tr>
<td></td>
<td>- In the TargetManagement category: CreateTarget, DeleteTarget, and ModifyTarget</td>
</tr>
<tr>
<td>None</td>
<td>No logging is performed.</td>
</tr>
</tbody>
</table>

3. Restart the Oracle Privileged Account Manager server.

Note: For detailed information about starting a Managed Server, refer to "Starting or Stopping the Oracle Stack" in the Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management.

After the server restarts, audit logs will start appearing in this location:

`DOMAIN_HOME/servers/<opamservlet>/logs/auditlogs/OPAM`

### 14.1.2 Understanding Oracle Privileged Account Manager Audit Reports

Oracle Privileged Account Manager supplies a set of default audit reports that are integrated with BI Publisher 11g and the Oracle Fusion Middleware Audit Framework. Oracle Privileged Account Manager generates these reports based on the audit events logged in the audit store.

The default audit report types include:

- **Accounts Checkin Checkout Report**: Provides account checkout and check-in history.
- **All Events Report**: Includes all audit events that are logged in the audit store.
- **Error Events Report**: Provides information about any errors that occur in Oracle Privileged Account Manager, such as authentication and authorization failures.
- **General Report**: Provides information about events related to checking in, checking out, or modifying privileged accounts and events related to queries about privileged accounts and targets.

- **Target Management Report**: Provides information about events related to adding, modifying, querying, or removing targets.

Oracle Privileged Account Manager audit reports can show who checked out an account and on which system it was checked out, justifications, requests for a system that is already checked out, and requests for a system to which a user does not have privileges.

For example, the following figure shows a typical Oracle Privileged Account Manager audit report as viewed in BI Publisher.

---

**Note:** You can view Oracle Privileged Account Manager audit reports in BI Publisher.

---

**Figure 14–1  Example Oracle Privileged Account Manager Audit Report**

<table>
<thead>
<tr>
<th>Event</th>
<th>Status</th>
<th>User ID</th>
<th>Target</th>
<th>Resource ID</th>
<th>Message</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Checkout</td>
<td>1</td>
<td>Chris Paul</td>
<td>demo06.DEV_IAU_APPEND</td>
<td>70fa175040c7f2b1f8330e2b27</td>
<td>Checkout Account: demo06.DEV_IAU_APPEND/70fa175040c7f2b1f8330e2b27(NO_COMMENTS_PROVIDED)</td>
<td>11/16/19 12:14 PM Midway</td>
</tr>
<tr>
<td>Checkout</td>
<td>1</td>
<td>Les Paul</td>
<td>demo06.DEV_IAU_APPEND</td>
<td>70fa175040c7f2b1f8330e2b27</td>
<td>Checkout Account: demo06.DEV_IAU_APPEND/70fa175040c7f2b1f8330e2b27(NO_COMMENTS_PROVIDED)</td>
<td>11/16/19 12:15 PM Midway</td>
</tr>
<tr>
<td>Checkout</td>
<td>1</td>
<td>Tak Matsumoto</td>
<td>demo06.DEV_OPSS</td>
<td>7c03731d3e04b28ae9757f146104c</td>
<td>Checkin Account: demo06.DEV_OPSS/7c03731d3e04b28ae9757f146104c(NO_COMMENTS_PROVIDED)</td>
<td>11/16/19 12:16 PM Midway</td>
</tr>
<tr>
<td>Checkin</td>
<td>1</td>
<td>Tak Matsumoto</td>
<td>demo06.DEV_OPSS</td>
<td>7c03731d3e04b28ae9757f146104c</td>
<td>Checkin Account: demo06.DEV_OPSS/7c03731d3e04b28ae9757f146104c(NO_COMMENTS_PROVIDED)</td>
<td>11/16/19 12:17 PM Midway</td>
</tr>
<tr>
<td>Checkout</td>
<td>1</td>
<td>Chris Paul</td>
<td>demo06.DEV_IAU_APPEND</td>
<td>70fa175040c7f2b1f8330e2b27</td>
<td>Checkout Account: demo06.DEV_IAU_APPEND/70fa175040c7f2b1f8330e2b27(NO_COMMENTS_PROVIDED)</td>
<td>11/16/19 12:17 PM Midway</td>
</tr>
<tr>
<td>Checkout</td>
<td>1</td>
<td>Chris Paul</td>
<td>demo06.DEV_IAU_APPEND</td>
<td>70fa175040c7f2b1f8330e2b27</td>
<td>Checkout Account: demo06.DEV_IAU_APPEND/70fa175040c7f2b1f8330e2b27(NO_COMMENTS_PROVIDED)</td>
<td>11/16/19 12:17 PM Midway</td>
</tr>
<tr>
<td>Checkin</td>
<td>1</td>
<td>Les Paul</td>
<td>lockUser1account1</td>
<td>3e22c0f1c5460c1b7f04c5b0c</td>
<td>Checkin Account: lockUser1account1/3e22c0f1c5460c1b7f04c5b0c(NO_COMMENTS_PROVIDED)</td>
<td>11/16/19 12:17 PM Midway</td>
</tr>
</tbody>
</table>

Notice that this report provides the following information:

- **Event**: Type of event that occurred
- **Status**: Event results, where 1 is success and 0 is a failure
- **User ID**: User that initiated the event
- **Target**: Target on which the event occurred
- **Resource ID**: Resource identifier
- **Message**: Message returned from server
- **Time**: Date and time the event occurred

### 14.1.3 Auditing Application Consumption of Credentials from CSF

Oracle Privileged Account Manager can synchronize passwords to CSF, as described in Section 17.3, "Integrating with the Credential Store Framework." However, Oracle Privileged Account Manager cannot audit any CSF content because Oracle Privileged
Account Manager and CSF are two separate entities in the WebLogic domain. If you want to audit CSF access, then you must enable auditing in CSF itself.

**Note:** For information about enabling auditing in CSF, refer to the following sections in the Oracle Fusion Middleware Application Security Guide:

- For a list of the audit events that are supported by CSF, refer to "Oracle Platform Security Services Events and their Attributes."
- For information about the WLST commands used to enable auditing in CSF, refer to "WLST Commands for Auditing" or enter the following command from the command line:
  
  ```bash
  help('Audit WLST command')
  ```
- For information about using Enterprise Manager to manage this type of auditing, refer to "Managing Audit Policies."

For information about using WSAdmin commands to enable auditing in CSF, refer to "Executing Common Audit Framework wsadmin Commands" in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management.

### 14.2 Understanding Oracle Privileged Account Manager Logging

Oracle Privileged Account Manager is fully integrated with Oracle Fusion Middleware Logging and the Oracle Diagnostic Logging (ODL) framework.

The Oracle Privileged Account Manager generic logger (oracle.idm.opam) takes care of all logs not recorded by the audit logger, which includes debugging statements and exception messages. Processing tools can use these logs to diagnose problems that occur within the Oracle Privileged Account Manager server.

Table 14–2 describes the different Oracle Privileged Account Manager-related log files:

<table>
<thead>
<tr>
<th>File Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AdminServer.log</td>
<td>Generic log file where the WebLogic Admin Server writes messages from its subsystems and applications.</td>
</tr>
<tr>
<td>AdminServer-diagnostic.log</td>
<td>Diagnostic log file used to store messages generated by the WebLogic Admin Server.</td>
</tr>
<tr>
<td>base_domain.log</td>
<td>Generic log file where the WebLogic Admin Server writes messages about the overall status of the domain.</td>
</tr>
<tr>
<td>access.log</td>
<td>Generic log file used to store information about requests to access privileged accounts and targets.</td>
</tr>
<tr>
<td>opam_server1.log</td>
<td>Generic log file where the Oracle Privileged Account Manager Server writes messages from its subsystems and applications.</td>
</tr>
<tr>
<td>opam_server1-diagnostic.log</td>
<td>Diagnostic log file used to store messages generated by the Oracle Privileged Account Manager Server.</td>
</tr>
</tbody>
</table>

Oracle Privileged Account Manager log files are stored in the following locations:

- Server log files are stored in

  `DOMAIN_HOME/servers/OPAM managed server/logs`
Server application logging is spooled to

_OPAM managed server-diagnostic.log

- Console log files are stored in

_DOMAIN_HOME/servers/AdminServer/logs

---

**Note:** For more information about Oracle Fusion Middleware Logging and the Oracle Diagnostic Logging (ODL) framework, refer to "Managing Log Files and Diagnostic Data" in the *Oracle Fusion Middleware Administrator’s Guide*.

---

### 14.2.1 Configuring Basic Logging

You can configure Oracle Privileged Account Manager logging by using the standard WLST commands as described in "Logging Custom WLST Commands" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Following are some task-based invocations based on the preceding reference:

---

**Note:** The same commands apply if you are configuring logging on an IBM WebSphere server, however there are some differences to consider.

Before using these commands, refer to "Configuring Basic Logging for Oracle Privileged Account Manager" in the *Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management*.

---

- To list all of the available Oracle Privileged Account Manager loggers and their current configured levels, run the `listLoggers` command:

  ```
  listLoggers(target="<opamserver>",pattern="oracle.idm.opam.*")
  ```

  For example,

  ```
  listLoggers(target="opam_server1",pattern="oracle.idm.opam.*")
  ```

- To check Oracle Privileged Account Manager's current log level, run the `getLogLevel` command:

  ```
  getLogLevel(logger="oracle.idm.opam",target="<opamserver>")
  ```

  For example,

  ```
  getLogLevel(logger="oracle.idm.opam",target="opam_server1")
  ```

- To set the log level for a particular logger, run the `setLogLevel` command:

  ```
  setLogLevel(target="<opamserver>",logger="oracle.idm.opam",level="TRACE:32", persist=1)
  ```

  For example,

  ```
  setLogLevel(target="opam_server1",logger="oracle.idm.opam",level="TRACE:32", persist=1)
  ```
14.2.2 Example Logging Data

This figure shows some example logging data as viewed from the WebLogic console.

**Figure 14–2 Example Logging Report**

<table>
<thead>
<tr>
<th>Date/Time</th>
<th>Subsystem</th>
<th>Severity</th>
<th>Message ID</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oct 12, 2011 10:48:25 AM PDT</td>
<td>OPAM</td>
<td>Info</td>
<td>BSA-000000</td>
<td>UIResource/getAccount</td>
</tr>
<tr>
<td>Oct 12, 2011 10:48:28 AM PDT</td>
<td>OPAM</td>
<td>Info</td>
<td>BSA-000000</td>
<td>UIResource/getAccount</td>
</tr>
<tr>
<td>Oct 13, 2011 10:48:48 AM PDT</td>
<td>OPAM</td>
<td>Info</td>
<td>BSA-000000</td>
<td>ContentManager added session 7920936130191956 with result = true</td>
</tr>
<tr>
<td>Oct 13, 2011 10:58:26 AM PDT</td>
<td>OPAM</td>
<td>Info</td>
<td>BSA-000000</td>
<td>UIResource/getACheckedOutAccounts</td>
</tr>
<tr>
<td>Oct 13, 2011 10:58:35 AM PDT</td>
<td>OPAM</td>
<td>Info</td>
<td>BSA-000000</td>
<td>PrivilegedAccountResource/checkIn</td>
</tr>
<tr>
<td>Oct 13, 2011 10:58:39 AM PDT</td>
<td>OPAM</td>
<td>Info</td>
<td>BSA-000000</td>
<td>ContentManager removed session 7920936130191956 with result = true</td>
</tr>
</tbody>
</table>

Notice that this report provides the following information:

- Date and timestamp when the event occurred
- Subsystem on which the event occurred
- Message severity
- Message ID
- Message describing the operation that was performed
Advanced Administration

This part provides information about performing advanced administration tasks for Oracle Privileged Account Manager, and it contains the following chapters:

- Performing Advanced Configuration Tasks for Oracle Privileged Account Manager
- Developing Plug-Ins for Oracle Privileged Account Manager
- Configuring Oracle Privileged Account Manager for Integrated Solutions
15
Performing Advanced Configuration Tasks for Oracle Privileged Account Manager

This chapter provides information about performing some advanced configuration for Oracle Privileged Account Manager.

This chapter includes the following sections:

- Section 15.1, "Configuring Oracle Privileged Account Manager to Communicate With Target Systems Over SSL"
- Section 15.2, "Securing Data On Disk"
- Section 15.3, "Adding New Connectors to an Existing Oracle Privileged Account Manager Installation"
- Section 15.4, "Advanced Management of Session Manager Data"
- Section 15.5, "Moving from a Test Environment to a Production Environment"
- Section 15.6, "Rebranding Oracle Privileged Account Manager"

---

**Note:** If you are using Oracle Privileged Account Manager on IBM WebSphere, refer to "Differences in Performing Advanced Configuration Tasks for Oracle Privileged Account Manager on IBM WebSphere" in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management for information about this topic.

---

### 15.1 Configuring Oracle Privileged Account Manager to Communicate With Target Systems Over SSL

Oracle Privileged Account Manager can connect to target systems through Secure Socket Layer (SSL) or non-SSL options. The SSL option is more secure, but requires some additional configuration.

To communicate securely over SSL with a target system, the WebLogic instance running Oracle Privileged Account Manager must trust the SSL certificate used by the target system because Oracle Privileged Account Manager inherits its SSL configuration from the WebLogic container in which it runs. To have the WebLogic instance running Oracle Privileged Account Manager (and therefore Oracle Privileged Account Manager) trust the target system’s SSL certificate, you must import the certificate into the truststore used by that WebLogic instance.
Use the following steps to enable SSL communication between the target system and Oracle Privileged Account Manager:

1. Export the SSL certificate from the target system host computer.

   **Note:** The steps for exporting an SSL certificate are different for each target system type. Refer to the product documentation provided for your target system for detailed instructions.

2. Copy the certificate to the machine where you have the WebLogic instance running Oracle Privileged Account Manager.

   If you have the Oracle Privileged Account Manager/Oracle Identity Navigator Console and the Oracle Privileged Account Manager server running on different machines, you must copy the SSL certificate to the Oracle Privileged Account Manager server machine.

3. Run the following command to import the certificate into the JVM truststore of the WebLogic Server on which Oracle Privileged Account Manager is running:

   ```
   JAVA_HOME\bin\keytool -import -file FILE_LOCATION -keystore TRUSTSTORE_LOCATION -storepass TRUSTSTORE_PASSWORD -trustcacerts -alias ALIAS
   ```

   Where
   - `JAVA_HOME` is the location used by your WebLogic server. For example.
     - `MW_HOME/jrockit..`
     - `MW_HOME/jdk..`
     - The location where you installed the Java software
   - `FILE_LOCATION` is the full path and name of the certificate file.
   - `TRUSTSTORE_LOCATION` is one of the following truststore paths:

   **Table 15–1 Truststore Locations**
<table>
<thead>
<tr>
<th>If you are using:</th>
<th>Import the Certificate into the Keystore in This Directory:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle jrockit_R27.3.1-jdk</td>
<td>JROCKIT_HOME/jre/lib/security</td>
</tr>
<tr>
<td>The default Oracle WebLogic Server JDK</td>
<td>WEBLOGIC_HOME/java/jre/lib/security/cacerts</td>
</tr>
<tr>
<td>A JDK other than Oracle jrockit_R27.3.1-jdk or Oracle WebLogic Server JDK</td>
<td>JAVA_HOME/jre/lib/security/cacerts</td>
</tr>
</tbody>
</table>

   - `TRUSTSTORE_PASSWORD` is the password for the truststore.
   - `ALIAS` is an alias for the certificate.

**Note:** The steps for configuring SSL communication are different if you are using an IBM WebSphere instance.

Refer to “Differences When Configuring Oracle Privileged Account Manager to Communicate with Target Systems Over SSL” in the *Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management* for instructions.
15.2 Securing Data On Disk

Oracle Privileged Account Manager can operate with or without Oracle Database Transparent Data Encryption (TDE) mode.

Note: Oracle strongly recommends that you enable TDE mode for enhanced security.

You can enable or disable TDE mode at any point after installing and configuring Oracle Privileged Account Manager.

This section describes how to change the TDE mode for Oracle Privileged Account Manager. The topics include:

- **Enabling TDE Mode**
- **Disabling TDE Mode**

Note: The instructions for enabling or disabling TDE mode are essentially the same whether you are using a WebLogic server or an IBM WebSphere server.

Refer to "Differences When Securing Data On Disk" in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management for information about the minor differences if you are using Oracle Privileged Account Manager on IBM WebSphere.

15.2.1 Enabling TDE Mode

You can enable TDE mode by performing the following steps:

1. **Enable TDE in the Database**
2. **Enable Encryption in the Oracle Privileged Account Manager Schema**
3. **Enable TDE Mode in the Oracle Privileged Account Manager Server Configuration**

15.2.1.1 Enable TDE in the Database

To enable TDE in the database, refer to "Enabling Transparent Data Encryption" in the Oracle Database Advanced Security Administrator’s Guide.
15.2.1.2 Enable Encryption in the Oracle Privileged Account Manager Schema

You can enable encryption in the Oracle Privileged Account Manager schema by using sqlplus (or any other client) to run the following opamxencrypt.sql script with the Oracle Privileged Account Manager schema user:

IAM_HOME/opam/sql/opamxencrypt.sql

For example,
sqlplus DEV_OPAM/welcome1 @IAM_HOME/opam/sql/opamxencrypt.sql

15.2.1.3 Enable TDE Mode in the Oracle Privileged Account Manager Server Configuration

You can enable TDE mode in the Oracle Privileged Account Manager server configuration by using one of the following methods:

- From the Oracle Privileged Account Manager Console
- From the Oracle Privileged Account Manager Command Line Tool

From the Oracle Privileged Account Manager Console
To enable TDE mode by using the Console, refer to step 3 in Section 5.2.3.1, "From the Console."

From the Oracle Privileged Account Manager Command Line Tool
To enable TDE mode (if the tdemode flag is set to false) by using the command line tool, complete the following steps:

1. Set the environment variables, ORACLE_HOME and JAVA_HOME.

2. Run the following script:
   - On UNIX, type:
     ```bash
     ORACLE_HOME/bin/opam.sh -url OPAM_Server_Url -x modifyconfig -configtype global -propertyname tdemode -propertyvalue true -u OPAM_APPLICATION_CONFIGURATOR_USER -p Password
     ```
   - On Windows, type:
     ```bat
     ORACLE_HOME\bin\opam.bat -url OPAM_Server_Url -x modifyconfig -configtype global -propertyname tdemode -propertyvalue true -u OPAM_APPLICATION_CONFIGURATOR_USER -p Password
     ```


---

**Note:** For additional information, refer to "Securing Stored Data Using Transparent Data Encryption" in the Oracle Database Advanced Security Administrator’s Guide.
15.2.2 Disabling TDE Mode

You can switch to non-TDE mode by performing the following steps:

1. Disable TDE Mode in the Oracle Privileged Account Manager Server Configuration
2. Disable Encryption in the Oracle Privileged Account Manager Schema

15.2.2.1 Disable TDE Mode in the Oracle Privileged Account Manager Server Configuration

You can disable TDE mode in the Oracle Privileged Account Manager server by using one of the following methods:

- From the Oracle Privileged Account Manager Console
- From the Oracle Privileged Account Manager Command Line Tool

From the Oracle Privileged Account Manager Console

To disable TDE mode by using the Console, refer to step 3 in Section 5.2.3.1, "From the Console."

From the Oracle Privileged Account Manager Command Line Tool

To disable TDE mode by using the command line tool, complete the following steps:

1. Set the environment variables, ORACLE_HOME and JAVA_HOME.
2. Run the following script:

   On UNIX:
   
   ```
   ORACLE_HOME/opam/bin/opam.sh -url OPAM_Server_Url -x modifyconfig
   -configtype global -propertyname tdemode -propertyvalue false
   -u OPAM_APPLICATION_CONFIGURATOR_USER -p Password
   ```

   Where **OPAM_Server_Url** is of the form:
   
   `https://OPAM_Managed_Server_Hostname:OPAM_Managed_Server_SSL_port/opam`

   On Windows:
   
   ```
   ORACLE_HOME\opam\bin\opam.bat -url OPAM_Server_Url -x modifyconfig
   -configtype global -propertyname tdemode -propertyvalue false
   -u OPAM_APPLICATION_CONFIGURATOR_USER -p Password
   ```

   Where **OPAM_Server_Url** is of the form:
   
   `https://OPAM_Managed_Server_Hostname:OPAM_Managed_Server_SSL_port/opam`

15.2.2.2 Disable Encryption in the Oracle Privileged Account Manager Schema

You can disable encryption in the Oracle Privileged Account Manager schema by using sqlplus (or any other client) to run the following `opamxunencrypt.sql` script with the Oracle Privileged Account Manager schema user:

```
IAM_HOME/opam/sql/opamxunencrypt.sql
```
15.3 Adding New Connectors to an Existing Oracle Privileged Account Manager Installation

This section describes the processes for adding new connectors to your existing Oracle Privileged Account Manager installation. The topics include:

- Adding Connectors Supplied by Oracle
- Adding Custom Connectors

15.3.1 Adding Connectors Supplied by Oracle

If you are adding new ICF connectors that are supplied by Oracle, then they will be accompanied by installation instructions. These instructions describe where to store the connector bundle and how to modify the installation specific opam-config.xml file.

15.3.2 Adding Custom Connectors

Oracle Privileged Account Manager can use custom connectors that you created or that were created by a third party. However, these connectors must strictly adhere to the ICF standard. After verifying that the connector is ICF-compliant, perform the following steps to deploy the connector for Oracle Privileged Account Manager consumption:

1. Put the connector bundle in a location on the file system where the bundle can be read by the Oracle Privileged Account Manager at run time.

2. Perform the following steps to create a configuration block for the connector and include that block in the installation specific opam-config.xml file:
   a. Design and create a relevant configuration block.
      Both the opam-config.xml and opam-config.xsd files contain documentation and an example at the beginning of the file describing how to create a configuration block.
   b. Ensure that this connector configuration block includes the file system location you specified for the connector bundle in step 1.
   c. Add the new connector configuration block to the opam-config.xml file by containing it in a <connectorConfig> block.
   d. Validate the modified opam-config.xml file against the opam-config.xsd file to ensure that the Oracle Privileged Account Manager server can read the modified file. You can use your favorite XML schema validation tool for this purpose.

3. Restart the Oracle Privileged Account Manager server.

4. Connect to Oracle Privileged Account Manager, and then add and configure a new target system using the newly added connector type.
15.4 Advanced Management of Session Manager Data

This section describes how to manage your Oracle Privileged Session Manager (Session Manager) data. The topics include:

- Overview
- Partitioning
- Partition OPSM_SESSIONS Table
- Purging

15.4.1 Overview

The Session Manager stores all of its session recording data in the Oracle Privileged Account Manager database schema. Over time, as more information is recorded, the disk footprint for this database schema will grow. Therefore, having a strategy to effectively manage this data is important.

Compliance regulations may require that you store audit data (such as session recordings) for long periods. You need a good backup and recovery plan to protect the data.

A good backup plan accounts for these basic guidelines:

- **Growth rate of session recordings**: The growth rate depends on the number of sessions and the type of activity (which results in recordable data) that occurs on those sessions. The growth of the session recording data generated daily determines, in turn, how often you want to perform backups.

- **Compliance regulations**: Consult your organization’s compliance regulations to determine how frequently backups are required and for how many years session recording storage is mandatory.

- **Online or offline data management**: Consult your organization’s compliance regulations to determine how frequently backups are required and what portion of session recording data must be easily accessible.

Oracle Database uses Oracle Recovery Manager (RMAN) for backup and recovery. For details, refer to:


**Note:** The Oracle Privileged Account Manager schema is created using the Oracle Repository Creation Utility (RCU) and the session recording data in the Oracle Privileged Account Manager schema is stored in the OPSM_SESSIONS table.

---

15.4.2 Partitioning

The Oracle Privileged Account Manager schema is unpartitioned by default. However, session recording data is cumulative and older data is never removed. If you store a high volume of session recording data, then you should consider partitioning the OPSM_SESSIONS table, which allows for easier archiving.

Benefits of partitioning include:
Improved Performance: If a table is range-partitioned by Timestamps, for example, queries by Timestamps can be processed on the partitions within that time-frame only.

Better Manageability: You can create partitions on separate tablespaces (thus different disks), which enables you to move older data to slower and larger disks, while keeping newer data in faster and smaller disks.

In addition, partitioning makes archiving much easier. For example, you can compress a single partition rather than having to partition the entire table.

Increased Availability: If a single partition is unavailable, for example, and you know that your query can eliminate this partition from consideration, then the query can be successfully processed without needing to wait for the unavailable partition.

### 15.4.3 Partition OPSM_SESSIONS Table

In this example, the `OPSM_SESSIONS` table is partitioned on a quarterly basis. Depending on your needs, you can choose to implement a different partitioning scheme.

To minimize application down time, Oracle recommends that partitioning is done before using this schema for an Oracle Privileged Account Manager deployment. If you are partitioning on an active Oracle Privileged Account Manager deployment, then you must first shut down all Oracle Privileged Account Manager processes before proceeding with the following steps.

The partitioning steps are as follows:

1. Login to the database using SQLPlus as the Oracle Privileged Account Manager schema user.

2. Rename the existing unpartitioned table. For example:

   ```sql
   RENAME OPSM_SESSIONS TO OPSM_SESSIONS_NONPART;
   ```

3. Create a new partitioned table that follows the table structure of the unpartitioned table. This example uses the range-partitioning (by Timestamp) scheme:

   ```sql
   CREATE TABLE OPSM_SESSIONS
   PARTITION BY RANGE (STARTTIME)
   (    PARTITION OPSM_SESSIONS_DEFAULT VALUES LESS THAN (MAXVALUE)
   )
   AS SELECT * FROM OPSM_SESSIONS_NONPART;
   ```

4. Enable row movement to allow data to automatically move from partition to partition when new partitions are created. For example:

   ```sql
   ALTER TABLE OPSM_SESSIONS ENABLE ROW MOVEMENT;
   ```

5. You can now create partitions. In this example, partitions are created by calendar quarter:

   ```sql
   ALTER TABLE OPSM_SESSIONS
   Split Partition OPSM_SESSIONS_DEFAULT AT (TO_DATE('01/04/2013', 'DD/MM/YYYY'))
   INTO (PARTITION OPSM_SESSIONS_Q1_2013, PARTITION OPSM_SESSIONS_DEFAULT)
   UPDATE INDEXES;
   ```

   ```sql
   ALTER TABLE OPSM_SESSIONS
   Split Partition OPSM_SESSIONS_DEFAULT AT (TO_DATE('01/07/2013', 'DD/MM/YYYY'))
   INTO (PARTITION OPSM_SESSIONS_Q1_2013, PARTITION OPSM_SESSIONS_DEFAULT)
   UPDATE INDEXES;
   ```
Rebranding Oracle Privileged Account Manager

INTO (PARTITION OPSM_SESSIONS_Q2_2013, PARTITION OPSM_SESSIONS_DEFAULT)
UPDATE INDEXES;

ALTER TABLE OPSM_SESSIONS
SPLIT PARTITION OPSM_SESSIONS_DEFAULT AT (TO_DATE('01/10/2013', 'DD/MM/YYYY'))
INTO (PARTITION OPSM_SESSIONS_Q3_2013, PARTITION OPSM_SESSIONS_DEFAULT)
UPDATE INDEXES;

ALTER TABLE OPSM_SESSIONS
SPLIT PARTITION OPSM_SESSIONS_DEFAULT AT (TO_DATE('01/01/2014', 'DD/MM/YYYY'))
INTO (PARTITION OPSM_SESSIONS_Q4_2013, PARTITION OPSM_SESSIONS_DEFAULT)
UPDATE INDEXES;

**Note:** You should periodically create new partitions for new quarters.

### 15.4.4 Purging

Purging removes the Oracle Privileged Account Manager session recording data from the Oracle Privileged Account Manager schema. Therefore, if you foresee needing to revisit this data at a later point, then use Oracle Recovery Manager (RMAN) for backup and recovery.

Keep in mind that with a range-partitioned table it is much more efficient to drop a partition when you want to remove old data, rather than deleting individual rows.

ALTER TABLE OPSM_SESSIONS DROP PARTITION OPSM_SESSIONS_Q1_2013;

Once partitions are created, you can purge and back up a particular partition. Refer to the Oracle Database documentation for details.

### 15.5 Moving from a Test Environment to a Production Environment

For information about moving Oracle Fusion Middleware components from one environment to another, refer to "Moving from a Test to a Production Environment" in *Oracle Fusion Middleware Administrator’s Guide*.

For information about moving Identity Management components, including Oracle Privileged Account Manager, from a test environment to a production environment, refer to "Moving Identity Management Components to a Target Environment" in *Oracle Fusion Middleware Administrator’s Guide*.

### 15.6 Rebranding Oracle Privileged Account Manager

If necessary, you can rebrand the Login and Oracle Privileged Account Manager pages. The following topics contain instructions for changing the page title, branding text, and logo image on these pages:

- **Customizing the Login Page**
- **Customizing the Oracle Privileged Account Manager Page**

**Tip:** Create a back-up copy before you modify any files.
15.6.1 Customizing the Login Page

You configure branding changes for the Login page in the oinav.ear/oiNavApp-war.war/SignIn.jspx file.

**Login Page Title**
To change the Login page title, modify the title in \texttt{af:document \\
"\{signinBean.signInTitle\}".}

Refer to the following code sample:
<af:document id="d1" title="\{signinBean.signInTitle\}" theme="dark" \\
initialFocusId="pt1:_pt_it1"/>

**Login Page Branding Text**
To change the branding text on the Login page, modify the value of \texttt{af:outputText \\
"\{signinBean.title\}"}, which is defined in the branding facet.

Refer to the following code sample:
<f:facet name="branding"> \\
<af:outputText values="\{signinBean.title\}" id="ot1"/>
</f:facet>

**Login Page Logo Image**
To change the logo image on the Login page, perform these steps:

1. Copy the new image, for example newlogo.png, into the following directory:

   oinav.ear/oiNavApp-war.war/images

2. To skip the default logo, add the following line to the oinav.ear/oiNavApp-war.war/SignIn.jspx file:

   <f:attribute name="brandingLogoCls" value=""/>

3. If the new logo's image size is larger than the default size 30, add the following line to adjust the header size:

   <f:attribute name="globalBrandingSize" value="60"/>

4. Modify the branding facet by replacing \texttt{newlogo.png, newlogo mouse over text}, and \texttt{new branding text}.

Refer to the following code sample:
<f:facet name="branding"> \\
<af:panelGroupLayout layout="horizontal"> \\
<af:image source="/images/newlogo.png" shortDesc='newlogo mouse over text' \\
id="im1"/> \\
<af:spacer width="5"/> \\
<af:outputText value='new branding text' id='ot1'/> \\
</af:panelGroupLayout> \\
</f:facet>

15.6.2 Customizing the Oracle Privileged Account Manager Page

You configure branding changes for the Oracle Privileged Account Manager page in the oinav.ear/oiNavApp-war.war/opam.jspx file.
Oracle Privileged Account Manager Page Title
To change the page title on the Oracle Privileged Account Manager page, modify the title in \texttt{af:document \{resBundle.PRODUCT_OPAM\}}

Refer to the following code sample:

\begin{verbatim}
<af:document title="\{resBundle.PRODUCT_OPAM\}" id="d1" theme="contentBody"/>
\end{verbatim}

Oracle Privileged Account Manager Branding Text
To change the branding text on the Oracle Privileged Account Manager page, modify the value of \texttt{af:outputText \{resBundle.OPAM_PRODUCT_TITLE\}}, which is defined in the branding facet.

Refer to the following code sample:

\begin{verbatim}
<f:facet name="branding">
  <af:outputText value="\{resBundle.OPAM_PRODUCT_TITLE\}" id="ot1"/>
</f:facet>
\end{verbatim}

Oracle Privileged Account Manager Page Logo Image
To change the logo image on the Oracle Privileged Account Manager page, perform these steps:

1. Copy the new image, for example \texttt{newlogo.png}, into the following directory:
   \begin{verbatim}
oinav.ear/oiNavApp-war.war/images
\end{verbatim}

2. To skip the default logo, add the following line to the \texttt{oinav.ear/oiNavApp-war.war/opam.jspx} file:
   \begin{verbatim}
<f:attribute name="brandingLogoCls" value=""/>
\end{verbatim}

3. If the new logo's image size is larger than the default size 30, add the following line to adjust the header size:
   \begin{verbatim}
<f:attribute name="globalHeaderSize" value="30"/>
\end{verbatim}

4. Modify the branding facet by replacing \texttt{newlogo.png}, \texttt{newlogo mouse over text}, and \texttt{new branding text}.

Refer to the following code sample:

\begin{verbatim}
<f:facet name="branding">
  <af:panelGroupLayout layout="horizontal">
    <af:image source="/images/newlogo.png" shortDesc="newlogo mouse over text" id="im1"/>
    <af:spacer width="5"/>
    <af:outputText value="new branding text" id="ot1"/>
  </af:panelGroupLayout>
</f:facet>
\end{verbatim}
This chapter describes how to develop your own plug-ins for Oracle Privileged Account Manager.

**Note:** For basic information about managing Oracle Privileged Account Manager plug-ins, including how to configure and deploy plug-ins, refer to Section 11, "Working with Plug-Ins."

This chapter includes the following sections:

- Section 16.1, "Overview"
- Section 16.2, "Setting Up a Plug-In"
- Section 16.3, "Understanding the Plug-In API"
- Section 16.4, "Debugging and Logging for Plug-Ins"
- Section 16.5, "Example Plug-ins"
- Section 16.6, "Managing Plug-Ins"

### 16.1 Overview

You can use Oracle Privileged Account Manager’s Java-based plug-in framework to create plug-ins that extend Oracle Privileged Account Manager’s functionality to accommodate your specific business and technical requirements. Plug-ins enable you to provide custom logic within a transaction or to connect to a custom data source.

The topics in this section include:

- Oracle Privileged Account Manager Framework Packages
- Special Considerations for Using Oracle Privileged Account Manager Plug-Ins

#### 16.1.1 Oracle Privileged Account Manager Framework Packages

The Oracle Privileged Account Manager plug-in framework contains the plug-in interfaces and classes you need to develop a plug-in implementation. This framework is shipped in the following jar file:

```
ORACLE_HOME/opam/jlib/opam-plugin-framework.jar
```

You can use this jar file to develop, implement, and compile plug-ins.
16.1.2 Special Considerations for Using Oracle Privileged Account Manager Plug-Ins

Following are some special considerations and dependencies that you must consider when developing plug-ins:

- Although an Oracle Privileged Account Manager server runtime is not required when developing plug-ins, the server runtime is required for deployment and testing purposes.

16.2 Setting Up a Plug-In

To set-up an Oracle Privileged Account Manager Java plug-in,

1. Create a standalone Java program using the predefined interface and implement the required methods.

   You can execute the plug-in using pre or post timing for an operation. You must implement the corresponding pre plug-in or post plug-in interface.

   Note: Refer to Section 11.2, "Developing Plug-Ins for Oracle Privileged Account Manager" for additional information about developing plug-ins for Oracle Privileged Account Manager.

   Note: Refer to Section 11.2.7.1, "Pre-Operation Plug-Ins" and Section 11.2.7.2, "Post-Operation Plug-Ins" for a description of these timings.

2. Before compiling, place the following jar in your classpath:

   ORACLE_HOME/opam/jlib/opam-plugin-framework.jar

3. Compile the plug-in Java files and create the class or jar file. Ensure the compilation completes without errors.

4. Put the class or jar file in a file system location that is accessible to the Oracle Privileged Account Manager server. For high-availability cluster configurations, you may want to place the class or jar file in each individual node or in a shared location that is accessible to all nodes.

5. Register the plug-in by adding the plug-in configuration entry.

   Note:

   - For information about managing plug-in configurations from the Console, refer to Section 11.3, "Creating a Plug-In Configuration."
   - For information about managing plug-in configurations from the command line, refer to Section A.7, "Working with Plug-Ins."

You can choose any name and package for the class and jar files. However, you must be sure to use the same name and package when configuring the Plug-in Class Name and the Plug-in Class Path attributes when registering the plug-in.

For example, if you create a plug-in using a fully qualified name, such as

my.sample.OpamPlugin, that is compiled to the /u01/myplugin.jar file and that has a dependency on some classes in /u01/myutils.jar, then your plug-in configuration must use the following:
After the plug-in configuration is registered and enabled, the server invokes the plug-in whenever the invocation criteria are met.

### 16.3 Understanding the Plug-In API

This section presents a high-level overview of the plug-in API and explains the role of the main classes and interfaces.

**Note:** Do not use `System.exit()` in a plug-in implementation because it might cause failures in the server runtime.

The topics in this section include:

- Communication between the Server and Plug-In
- Plug-In Structure
- Plug-In Interfaces and Classes
- Pre Plug-In Example
- Post Plug-In Example

#### 16.3.1 Communication between the Server and Plug-In

Oracle Privileged Account Manager plug-ins use the `PrePlugin` or `PostPlugin` interface to communicate with the Oracle Privileged Account Manager server. When invoking a plug-in, the server constructs the `PluginContext` object and passes details about the operation (such as target, account, and so forth) to the plug-in invoked by that operation. The server also passes the operation request body and the plug-in configuration that invokes the plug-in to the plug-in.

The plug-in constructs a `PluginResult` object. After completing its task, the plug-in passes the `PluginResult` object back to the server. The `PluginResult` object can contain a success or failure code, custom error messages, and log messages. In some cases, the plug-in can change or add details to the request body and pass those details back to the server.

The following figure illustrates how Oracle Privileged Account Manager plug-ins communicate with the server.

*Figure 16–1  How Plug-Ins Communicate with the Server*
16.3.2 Plug-In Structure

The general structure for a Java plug-in is as follows:

For a Pre Plug-In

```java
public class OPAM_PLUGIN_CLASSNAME implements PrePlugin {
    public void runPrePlugin(PluginContext ctx, String reqBodyJSON, String pluginCfgJSON) {
        // Plugin Code
    }
}
```

For a Post Plug-In

```java
public class OPAM_PLUGIN_CLASSNAME implements PostPlugin {
    public void runPostPlugin(PluginContext ctx, String reqBodyJSON, String pluginCfgJSON) {
        // Plugin Code
    }
}
```

16.3.3 Plug-In Interfaces and Classes

This section describes the plug-in interfaces and classes to use for Oracle Privileged Account Manager.

The topics in this section include:

- PlugInContext
- PluginResult
- PrePlugin
- PostPlugin

16.3.3.1 PlugInContext

The Oracle Privileged Account Manager server creates the `PluginContext` object during plug-in invocation and sends it to the plug-in. This object contains the following information:

- **Account information in JSON format.** Obtained by using the `getAccountJSON()` method.
  - Account information is present in operations involving accounts, such as `checkin` and `checkout`.
  - Account information is not present in operations that do not involve accounts, such as adding or deleting a target. Also, account information is not present when you add a new account because the account is not yet created, and the details sent by the client can be obtained in the request body JSON.

---

Note: Refer to the Oracle Privileged Account Manager Plug-In Framework Java API Reference for more information.
Understanding the Plug-In API

- **Target information in JSON format.** Obtained by using the `getTargetJSON()` method.
  Similar to **Account information in JSON format**, target information is only present in operations involving targets.

  **Note:** Refer to Section B.5, "Target Resource" in Appendix B, "Working with Oracle Privileged Account Manager's RESTful Interface" for more information about the Target Resource and its JSON format.

- **Operation information in JSON format.** Obtained by using the `getOperationJSON()` method.
  This method contains information about on which resource this operation is being performed (such as target, account, etc.) and which operation is being performed (such as add, delete, etc.) For example:

  ```
  {
      "resourceType": "account",
      "operationName": "add"
  }
  ```

- **Authentication information in JSON format.** Obtained using the `getAuthContextJSON()` method.
  This method contains information about which user is performing the operation and in which groups that user is a member. For example

  ```
  {
      "requestor": "johndoe",
      "requestorGroups": ["ITADMINS", "MANAGERS"]
  }
  ```
  
  This information is not present for operations of resource type server because the Oracle Privileged Account Manager server performs the operation rather than an end user.

- **PluginResult Object.** Must be created by the plug-in to pass information back to the server. Set this object by using the `setPluginResult()` method and obtain the value that was set by using `getPluginResult()` method.

  **Note:** Refer to Section 16.3.3.2, "PluginResult" for additional information about the PluginResult object.

- **HTTP Response Status (for post plug-ins only).** Obtained using the `getHTTPResponseStatus()` method.
For post plug-ins, when the Oracle Privileged Account Manager server REST API-based operation has completed, the operation response code (such as 200 OK, 401 Unauthorized, etc.) is sent to the plug-in.

- **HTTP Response Entity** (for post plug-ins only). Obtained using the `getHTTPResponseEntity()` method.

  For post plug-ins, when the Oracle Privileged Account Manager Server REST API-based operation has completed, the operation results are sent to the plug-in. For example, operation results might include a list of accounts in JSON format from a search operation, an account token in JSON format from a checkout operation, and so forth.

- **HTTP Response Location > Created GUID** (for post plug-ins only). Obtained using the `getHTTPResponseLocation() -> getGuid()` method.

  For post plug-ins and add operations, when the Oracle Privileged Account Manager Server REST API-based add operation has completed, the newly created entity’s GUID is sent to the plug-in. This GUID is only present for creation operations. The GUID is not present for operations such as delete target, modify target, and so forth.

### 16.3.3.2 PluginResult

The plug-in creates `PluginResult` and uses it to send information back to the Oracle Privileged Account Manager server.

To create this object, the plug-in uses the `PluginResult(java.lang.String resultJSON, int plgErrorCode, java.lang.String plgErrorMsg)` method. After creation, `PluginResult` is passed back to the server by storing it in the `PluginContext` sent by the server. For example

```java
PluginResult result = new PluginResult(resultJSON, plgErrorCode, plgErrorMsg);
ctx.setPluginResult(result);
```

The `PluginResult` object contains the following information:

- **Plugin modified request body JSON.** Result information in JSON format.
  - In **pre** plug-in operations, the plug-in uses this information to change or add details to the request body JSON that is passed to it and then returns the updated content as result JSON. The server uses the result JSON to replace the request body JSON before performing the operation.

    If multiple pre plug-ins are executed for the same operation, then each plug-in gets the updated request body based on their order of execution. After the last pre plug-in is executed, the final request body is used to execute the operation in the server.

    If the plug-in does not want to modify the request body JSON, it can be passed as is into the plug-in result JSON. The plug-in can set this value when creating the `PluginResult` object and can access it by using the `getResultJSON()` and `setResultJSON()` methods.

  - In **post** plug-in operations, the plug-in can read this value but modifying the value has no impact because the operation has already been completed.

- **Error code.** Specifies success (PluginResult.CODE_SUCCESS) or failure (PluginResult.CODE_FAILURE).

  The plug-in can set this value during the `PluginResult` object creation and can access it by using the `getErrorCode()` and `setErrorCode()` methods.
- **Error message.** Describes the error in case of failures.

  The plug-in can set any string message and pass it back to the client. The plug-in can set the message during the PluginResult object creation and can access it by using the `getPlgErrorMsg()` and `setPlgErrorMsg()` methods.

  For example, assume you configured a pre plug-in for checkout operations to validate checkout dates and prevent account usage on blacklisted dates, such as regional holidays, weekends, etc. The plug-in can set an error message on checkout such as, "Checkout is not allowed on holidays." In this case, the checkout operation will fail and it will include the error message, "Plug-in execution failed with error message: Checkout is not allowed on holidays."

- **Debug logging.** Provides information about the plug-in execution.

  The plug-in can pass log messages back to the server that will be logged in the server log file. You can use these logs for debugging the plug-in execution. access the log message by using the `getLog()`, `appendLog(String)`, and `clearLog()` methods.

### 16.3.3 PrePlugin

You must implement the PrePlugin interface to create a pre plug-in for Oracle Privileged Account Manager operations. Use following method to implement the PrePlugin interface:

```java
void runPrePlugin(PluginContext ctx, java.lang.String reqBodyJSON, java.lang.String pluginCfgJSON)
```

The following information is passed from the server to the pre plug-in through `runPrePlugin` method:

- **PluginContext object.** Contains details about the operation.
  
  Refer to Section 16.3.3.1, "PlugInContext" for more information.

- **Request Body JSON.** Provides the request sent by the client to Oracle Privileged Account Manager for the REST API operation.
  
  The plug-in can modify this information for pre plug-ins and it can send the newly updated JSON back to the server in the PluginResult object. Refer to Section 16.3.3.2, "PluginResult" for more information.

- **Plug-in configuration.** Invokes this operation in JSON format.

  You can use the same Java plug-in implementation for many operations. For example, you can use the same plug-in that performs email notification for both checkout and checkin operations.

  When the plug-in configuration is passed to the plug-in, it provides details to the plug-in on which the configuration is causing this invocation, which is also useful for passing custom attributes that are present in the configuration to the plug-in. For example, in the email notification case, you can store mail server details as custom attributes in the plug-in configuration and they will be passed to the plug-in.

### 16.3.3.4 PostPlugin

You must implement the PostPlugin interface to create a post plug-in for Oracle Privileged Account Manager operations. Use following method to implement the PostPlugin interface:
void runPostPlugin(PluginContext ctx, java.lang.String reqBodyJSON, java.lang.String pluginCfgJSON)

The information that is passed from the server to the post plug-in through the runPostPlugin method is the same as the information described in the previous section. Review the list in Section 16.3.3.3, "PrePlugin."

16.4 Debugging and Logging for Plug-Ins

A plug-in can maintain its own log file and can log to that file in real time. In addition, a plug-in can log debug messages in the Oracle Privileged Account Manager server log file during execution by using PluginResult object debug logging methods, as described in Section 16.3.3.2, "PluginResult."

Messages logged using the PluginResult method will be present in the Oracle Privileged Account Manager Server log file. To view these messages, you must set the logging level to TRACE:32 (very detailed trace or debug information).

The following example shows the sample code used to implement a plug-in's logging functionality.

Example 16–1 Sample Code Used to Implement Plug-In Logging

```java
public void runPostPlugin(PluginContext ctx, String reqBodyJSON, String pluginCfgJSON) { // the parameters are the same for runPrePlugin()

    PluginResult result = ctx.getPluginResult(); // get the PluginResult object
    result.appendLog("Here is the log"); // append log

    // System.out.println(result.getLog()); // getLog() will return current log
    // result.clearLog(); // clearLog() will remove the log that has been recorded

}
```

16.5 Example Plug-ins

The examples in this section include

- Pre Plug-In Example
- Post Plug-In Example

16.5.1 Pre Plug-In Example

Example 16–3 (located at the end of this section) illustrates a pre plug-in that blocks an operation, based on the specified dates, before the operation is executed.

Your organization may have some blacklist dates, such as regional holidays and yearly closures, when access to privileged accounts should not be allowed. This pre plug-in performs the validation and extends Oracle Privileged Account Manager's Usage Policy functionality.

This plug-in uses the following custom attributes to specify the blacklist dates.
To configure the plug-in, use the attributes described in the following table:

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Attribute Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>date</td>
<td>Month/Day, for example 10/01.</td>
</tr>
<tr>
<td>Note:</td>
<td>You can specify more than one date attribute.</td>
</tr>
<tr>
<td></td>
<td>Do not ignore the zeros (0) used in the plug-in sample.</td>
</tr>
<tr>
<td></td>
<td>For example, to block June 9, you must specify 06/09 instead of 6/9.</td>
</tr>
</tbody>
</table>

To compile the plug-in

1. If necessary, download the following files:

<table>
<thead>
<tr>
<th>Download From</th>
<th>Files to Download</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORACLE_HOME</td>
<td>opam-plugin-framework.jar</td>
</tr>
<tr>
<td></td>
<td>jettison-1.3.jar</td>
</tr>
<tr>
<td></td>
<td>jersey-bundle-1.4.jar</td>
</tr>
<tr>
<td></td>
<td>jsr311-api.jar</td>
</tr>
</tbody>
</table>

2. Use the following sample compile command:

```java
BlackListDates.java
```

3. Create a jar file:

```bash
tar -cf BlackListDates.jar BlackListDates.class
```
After you configure this plug-in for Oracle Privileged Account Manager, the plug-in will block the user from executing the Oracle Privileged Account Manager operation based on the date attributes. For example, if today is 10/01, and one of the date attributes is 10/01, then the user will not be able to perform this operation.

Note: In the following example, only the month and day are necessary. The year does not matter.

Example 16–2  Blacklist Dates Pre Plug-In

```java
import org.codehaus.jettison.json.JSONException;
import org.codehaus.jettison.json.JSONObject;
import org.codehaus.jettison.json.JSONStringer;
import org.codehaus.jettison.json.JSONArray;
import com.oracle.idm.opam.plugin.interfaces.PrePlugin;
import com.oracle.idm.opam.plugin.context.PluginContext;
import com.oracle.idm.opam.plugin.context.PluginResult;
import java.util.Date;
import java.text.SimpleDateFormat;
import java.util.Calendar;

public class BlackListDates implements PrePlugin {

    public void runPrePlugin(PluginContext ctx, String reqBodyJSON, String pluginCfgJSON) {
        System.out.println("===== In BlackListDates.runPrePlugin =====");
        try {
            JSONArray blackDates = null;
            JSONObject plugin = new JSONObject(pluginCfgJSON);
            JSONObject config = plugin.getJSONObject("plugin");
            JSONArray customAttrsArr = new JSONArray();
            if(config.has("pluginCustomAttrs"))
                customAttrsArr = config.getJSONArray("pluginCustomAttrs");
            for(int i=0; i<customAttrsArr.length(); i++) {
                JSONObject singleJSON = customAttrsArr.getJSONObject(i);
                JSONObject singleAttr = singleJSON.getJSONObject("pluginCustomAttr");
                String attrName = singleAttr.getString("attrname");
                if(attrName.equalsIgnoreCase("date"))
                    blackDates = singleAttr.getJSONArray("attrvalue");
            }
            for(int i=0; i<blackDates.length(); i++) {
                String date = blackDates.getString(i);
                String[] count = date.split("/");
                if(count.length != 2)// wrong format, ignore
                    continue;
                else {
                    if(isToday(date)) {
                        setResult(ctx, reqBodyJSON, PluginResult.CODE_FAILURE,
                                    "You are not allowed to do the operation on this date : " + date);
                        return;
                    }
                }
            }
        } catch (Exception e) {
            System.out.println("Exception happened: ");
        }
    }
}
```

Note: In the following example, only the month and day are necessary. The year does not matter.
16.5.2 Post Plug-In Example

Example 16–3 (located at the end of this section) illustrates a post plug-in that sends an email about the operation after the operation has completed.

To configure the plug-in, use the attributes described in the following table:

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Attribute Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>pluginName</td>
<td>EmailNotification</td>
</tr>
<tr>
<td>pluginDescription</td>
<td>This is an Email Notification Plug-in.</td>
</tr>
<tr>
<td>pluginResource</td>
<td>account</td>
</tr>
<tr>
<td>pluginOperation</td>
<td>checkout</td>
</tr>
<tr>
<td>pluginTiming</td>
<td>post</td>
</tr>
<tr>
<td>pluginOrder</td>
<td>1</td>
</tr>
<tr>
<td>pluginClassName</td>
<td>EmailNotificationPlugin</td>
</tr>
<tr>
<td>Note:</td>
<td>Class Name should be consistent with the class name in your plug-in jar file</td>
</tr>
<tr>
<td>pluginClassPath</td>
<td>/myhome/plugins/EmailNotificationPlugin.jar</td>
</tr>
<tr>
<td>Note:</td>
<td>The pluginClassPath is a multi-value attribute, so the JSON should be sent in JSON array format, similar to the following: pluginClassPath: [&quot;/myhome/plugins/EmailNotificationPlugin.jar&quot;]</td>
</tr>
</tbody>
</table>
| pluginCustomAttrs    | Use the following custom attributes to send information, such as the SMTP server to use for sending the email, the address where the email is sent, and so forth:  
  ■ smtp_server: SMTP host name  
  ■ smtp_port: 25  
  ■ to_addr: to_address@somedomain  
  ■ from_addr: from_address@somedomain |
You must also configure the following custom attributes to send information, such as the SMTP server to use for sending the email, the address where the email is sent, and so forth.

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Attribute Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>smtp_server</td>
<td>SMTP host name</td>
</tr>
<tr>
<td>smtp_port</td>
<td>25</td>
</tr>
<tr>
<td>to_addr</td>
<td>&lt;EMAIL ADDRESS&gt;</td>
</tr>
<tr>
<td>from_addr</td>
<td>&lt;EMAIL ADDRESS&gt;</td>
</tr>
<tr>
<td>user</td>
<td>Optional. SMTP login user name</td>
</tr>
<tr>
<td>password</td>
<td>Optional. SMTP login user password</td>
</tr>
</tbody>
</table>

This plug-in uses the JAVA Mail API for sending email. For more information about the JAVA Mail library and to download the library, refer to the following location:

http://www.oracle.com/technetwork/java/javamail/index.html

To compile the plug-in

1. If necessary, download the following files:

<table>
<thead>
<tr>
<th>Download From</th>
<th>Files to Download</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORACLE_HOME</td>
<td>opam-plugin-framework.jar</td>
</tr>
<tr>
<td></td>
<td>jettison-1.3.jar</td>
</tr>
<tr>
<td></td>
<td>jersey-bundle-1.4.jar</td>
</tr>
<tr>
<td></td>
<td>jsr311-api.jar</td>
</tr>
</tbody>
</table>

   | JAVA Mail API | mail.jar           |

2. Use the following sample compile command:


3. Create a jar file:

   jar -cf EmailNotificationPlugin.jar EmailNotificationPlugin.class

After you configure this plug-in for Oracle Privileged Account Manager, the plug-in will send an email to the address that you set-up in custom attributes whenever the configured operation is performed. For example, if you configure this plug-in for the account resource type and checkout operation, then the plug-in will send an email notification whenever a checkout is completed.

**Example 16–3  Email Notification Post Plug-In**

```java
import org.codehaus.jettison.json.JSONException;
import org.codehaus.jettison.json.JSONObject;
import org.codehaus.jettison.json.JSONStringer;
```
import org.codehaus.jettison.json.JSONArray;
import com.oracle.idm.opam.plugin.interfaces.PostPlugin;
import com.oracle.idm.opam.plugin.context.PluginContext;
import com.oracle.idm.opam.plugin.context.PluginResult;
import java.lang.Thread;
import java.io.*
import java.util.*
import javax.mail.*
import javax.mail.internet.*
import javax.activation.*
/* Sample post plugin that sends email notification */
public class EmailNotificationPlugin implements PostPlugin {
  
  public void runPostPlugin(PluginContext ctx, String reqBodyJSON,
                           String pluginCfgJSON) {
    
    PluginResult result =
      new PluginResult(reqBodyJSON, PluginResult.CODE_SUCCESS, null);
    
    try {
      result.appendLog("Starting EmailNotificationPlugin");
      
      /* Get the resource type and operation name from the context */
      JSONObject opJSON = new JSONObject(ctx.getOperationJSON());
      String resourceType = opJSON.getString("resourceType");
      String operationName = opJSON.getString("operationName");
      
      /* Get the target name */
      JSONObject json = null;
      String targetName = null;
      if (ctx.getTargetJSON() != null) {
        JSONObject targetJSON = new JSONObject(ctx.getTargetJSON());
        json = targetJSON.getJSONObject("target");
        targetName = json.getString("targetName");
      }
      
      /* Get the account name */
      String accountName = null;
      if (ctx.getAccountJSON() != null) {
        JSONObject accountJSON = new JSONObject(ctx.getAccountJSON());
        json = accountJSON.getJSONObject("account");
        accountName = json.getString("accountName");
      }
      
      /* Get which user performed the operation */
      JSONObject pluginAuthJSON =
        new JSONObject(ctx.getAuthContextJSON());
      String requestor = pluginAuthJSON.getString("requestor");
      
      /* Get custom attributes defined in plugin configuration such as email server, 
      to address etc */
      JSONObject plugin = new JSONObject(pluginCfgJSON);
      JSONObject config = plugin.getJSONObject("plugin");
      JSONArray customAttrsArr = new JSONArray();
      if (config.has("pluginCustomAttrs"))
        customAttrsArr = config.getJSONArray("pluginCustomAttrs");
  
  } catch (Exception e) {
    result.appendLog("Error: "+e.getMessage());
  }
  
  return result;
}
```java
String smtpServer = null;
String smtpPort = null;
String fromAddr = null;
String user = null;
String password = null;
JSONArray emailList = null;

for (int i = 0; i < customAttrsArr.length(); i++) {
    JSONObject singleJSON = customAttrsArr.getJSONObject(i);
    JSONObject singleAttr =
        singleJSON.getJSONObject("pluginCustomAttr");

    String attrName = singleAttr.getString("attrname");

    if (attrName.equalsIgnoreCase("smtp_server"))
        smtpServer =
            singleAttr.getJSONArray("attrvalue").getString(0);
    if (attrName.equalsIgnoreCase("smtp_port"))
        smtpPort =
            singleAttr.getJSONArray("attrvalue").getString(0);
    if (attrName.equalsIgnoreCase("from_addr"))
        fromAddr =
            singleAttr.getJSONArray("attrvalue").getString(0);
    if (attrName.equalsIgnoreCase("to_addr"))
        emailList = singleAttr.getJSONArray("attrvalue");
    if (attrName.equalsIgnoreCase("user"))
        user = singleAttr.getJSONArray("attrvalue").getString(0);
    if (attrName.equalsIgnoreCase("password"))
        password =
            singleAttr.getJSONArray("attrvalue").getString(0);
}

for (int i = 0; i < emailList.length(); i++) {
    Properties properties = System.getProperties();
    properties.setProperty("mail.smtps.host", smtpServer);
    properties.setProperty("mail.smtp.port", smtpPort);
    if (user != null && password != null) {
        properties.setProperty("mail.user", user);
        properties.setProperty("mail.password", password);
    }
    Session session = Session.getDefaultInstance(properties);
    MimeMessage message = new MimeMessage(session);
    message.setFrom(new InternetAddress(fromAddr));
    message.addRecipient(Message.RecipientType.TO, new InternetAddress(emailList.getString(i)));

    // Set the email subject and body */
    String subject =
        "OPAM Notification : " + resourceType + "" + operationName;
    String emailBody =
        "Target : " + targetName + "\nAccount : " + accountName + "\nOperation : " + operationName + "\nRequestor : " + requestor;
    message.setSubject(subject);
    message.setText(emailBody);

    /* Send the email */
```

Transport.send(message);
result.appendLog("Completed EmailNotificationPlugin successfully");
ctx.setPluginResult(result);
}
} catch (Exception e) {
    result.appendLog("Exception happened: " + e);
    result.setErrorCode(PluginResult.CODE_FAILURE);
    result.setErrorMsg(e.getMessage());
    ctx.setPluginResult(result);
}
}

### 16.6 Managing Plug-Ins

For information about managing plug-ins,

- Refer to Chapter 11, "Working with Plug-Ins" for information about managing plug-ins from the Console.
- Refer to Section A.7, "Working with Plug-Ins" for information about managing plug-ins from the command line.
This chapter describes how to configure Oracle Privileged Account Manager for integration with commonly used directory and identity management technologies. This chapter includes the following sections:

- Section 17.1, "Integrating with Oracle Identity Manager"
- Section 17.2, "Integrating with Oracle Access Management Access Manager"
- Section 17.3, "Integrating with the Credential Store Framework"

### 17.1 Integrating with Oracle Identity Manager

This section provides information about the Oracle Privileged Account Manager - Oracle Identity Manager integration process.

The topics include:

- Overview
- Before You Begin
- Setting Up Oracle Identity Manager for the Integration
- Running the opamSetup Script
- Creating the OPAM_TAGS UDF
- Tagging Catalog Entries with Oracle Privileged Account Manager Metadata

**Note:** If you are using Oracle Privileged Account Manager on IBM WebSphere, refer to "Differences When Integrating with Oracle Identity Manager" in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management for information about this topic.

#### 17.1.1 Overview

The integration of Oracle Privileged Account Manager and Oracle Identity Manager enables you to manage access to the LDAP groups that are also Oracle Privileged Account Manager grantees. Specifically, integrating these two products enables you to

- Manage the identity lifecycle from hiring to retirement
- Provide a native ability to automate adding and removing users to the proper LDAP groups based on their HR system updates
- Provide the ability to manually request access to accounts
- Support the ability to get approvals for requests
- Support reporting that you can use for attestation reporting; either to augment or in-lieu of Oracle Privileged Account Manager’s own reporting.

In addition, Oracle Privileged Account Manager leverages Oracle Identity Manager for workflow support. The integration points include:

- Access to privileged accounts granted to roles in Oracle Privileged Account Manager by an Oracle Privileged Account Manager administrator
- End users can request membership in these roles through Oracle Identity Manager
- Standard Oracle Identity Manager workflow are used to approve these requests
- Membership in the requested role results in end users getting access to the corresponding privileged accounts in Oracle Privileged Account Manager

*Figure 17–1 Oracle Identity Manager Workflow Topology*

To support this integration, Oracle Identity Manager

- Provides LDAP connector(s) to manage LDAP groups
- Populates the resource catalog with the proper enterprise roles and entitlements.

Oracle Privileged Account Manager target-accounts are entitlements because Oracle Identity Manager is not actually granting direct access to the actual account only a representation of that account.
17.1.2 Before You Begin

This section describes some tasks you must complete before starting the actual integration process. These tasks include:

- **Installing Oracle Identity Manager**
- **Configuring an Oracle Identity Manager Administrator**
- **Configuring the External Identity Stores**
- **Creating LDAP Groups**
- **Adding the Oracle Privileged Account Manager CA Certificate**

### 17.1.2.1 Installing Oracle Identity Manager

The instructions in this chapter assume you have already installed Oracle Identity Manager. If you have not yet installed Oracle Identity Manager, refer to the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management* for instructions.

### 17.1.2.2 Configuring an Oracle Identity Manager Administrator

When you configure an Oracle Identity Manager administrator for this integration, that administrator must be able to perform these tasks:

- Configure an Oracle Identity Manager rule that assigns new users to the proper LDAP groups based on a business rule. The rule should apply whether you assign the new users manually through the user screen or automatically by using an HR/text feed.

- Use Oracle Identity Manager’s native functionality to build requests for items in the Oracle Identity Manager resource catalog to ensure that the catalog is properly populated. Oracle Identity Manager enables users to request access to entitlements contained in the Oracle Identity Manager catalog.

- Set approver fields to the proper values. For example, in situations where one employee requests access to the email account of another employee who will be away from the office for an extended period of time.

- Handle "firecall" requests, where an Oracle Privileged Account Manager user must access a system that is outside the normal business process.

Firecall requests are handled based upon your business requirements and business rules. For example, if the Oracle Privileged Account Manager user is authorized for a target, but the access policy prevents that user from getting the password, then the Oracle Privileged Account Manager administrator can temporarily change the access policy for that target-account.

If the user cannot wait for Oracle Identity Manager, the Oracle Privileged Account Manager administrator can manually direct access (for example, add a specific grantee to the account) instead.

---

**Note:** If you are using Oracle Privileged Account Manager on IBM WebSphere, refer to "Differences in How Oracle Privileged Account Manager is Deployed in Oracle Fusion Middleware" in the *Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management* for information.
To review the steps for configuring an Oracle Identity Manager administrator, refer to "Managing Admin Roles" in the Oracle Fusion Middleware User’s Guide for Oracle Identity Manager.

17.1.2.3 Configuring the External Identity Stores

You must configure an external identity store as the main authentication source for Oracle Privileged Account Manager. Refer to Section 3.3.2, "Configuring an External Identity Store for Oracle Privileged Account Manager" for more information.

After configuring the Oracle Privileged Account Manager external identity store, you must configure Oracle Identity Manager to use that same identity store.

- You can configure Oracle Identity Manager to use the same LDAP server as Oracle Privileged Account Manager by using LDAPSync.
  
  Refer to "Completing the Prerequisites for Enabling LDAP Synchronization" in the Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management for instructions.

- Alternatively, you can use the Generic LDAP Connector to configure Oracle Identity Manager to use the same LDAP server as Oracle Privileged Account Manager.

  Refer to Section 17.1.3.1, "Installing and Configuring the Generic LDAP Connector" for more information about setting up and configuring the LDAP connector you need for the server.

---

**Note:** Best practice is to use LDAPSync for environments where Access Manager is also enabled. If Access Manager is not enabled in your environment, then you can use the LDAP Connector.

In general, most product deployments should use LDAPSync.

---

17.1.2.4 Creating LDAP Groups

Oracle Privileged Account Manager is optimized for managing shared and privileged accounts, such as root on an UNIX system.

Oracle Privileged Account Manager determines which users can check out passwords for accounts on a target, based on the grants those users have received. Grants can be made directly or through membership in groups. The groups themselves can be static or dynamic.

Ideally, these LDAP groups should match your enterprise roles. For example, if you have a "Data Center Product UNIX Administrators" enterprise role, you should have a corresponding LDAP group. The benefit of this match is that you can use these groups to control access to other applications besides Oracle Privileged Account Manager target-accounts.

---

**Note:** To create an LDAP group, contact your LDAP administrator.

---

17.1.2.5 Adding the Oracle Privileged Account Manager CA Certificate

You must configure Oracle Privileged Account Manager's Catalog Synchronization task to include the Oracle Privileged Account Manager server's web service Certificate authority (CA) certificate or HTTPS web service calls to the Oracle Privileged Account Manager server cannot succeed.
This process is done in two steps:

1. **Retrieve the CA Certificate**
2. **Import the CA Certificate**

---

**Note:** If you are using Oracle Privileged Account Manager on an IBM WebSphere server, these steps are slightly different. Refer to "Differences When Integrating with Oracle Identity Manager" in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management for instructions.

---

**Retrieve the CA Certificate**

To retrieve the Oracle Privileged Account Manager server’s CA certificate:

1. From your browser, connect to the Oracle Privileged Account Manager server web service:
   
   `https://hostname:sslport/opam`

2. Locate and save the CA certificate (.pem) file to the truststore. For example, from a Firefox browser
   
   a. Click the lock icon in the browser’s address bar.  
   
   ![Lock icon](https://)
   
   b. When the information dialog displays, click **More information**.
   
   c. On the Page Info dialog, click **View certificate**.
   
   d. On the Certificate Viewer dialog, select the Details tab to view the Certificate Hierarchy.
   
   e. Select the first (root) certificate in the Certificate Hierarchy list, and then click **Export**.
   
   f. When the Save Certificate to File dialog displays, navigate to the directory where you want to save the file. For example, `/tmp/opam.pem`
   
   g. Select **X.509 Certificate (PEM)** from the Save as type menu, enter **opam.pem** as the file name, and click **Save**.

**Import the CA Certificate**

Run the following command to import the CA certificate file, `opam.pem`, into the WebLogic truststore on the server where you are running Oracle Identity Manager:

```
keytool -import -file FILE_LOCATION -keystore TRUSTSTORE_LOCATION -storepass TRUSTSTORE_PASSWORD -trustcacerts -alias ALIAS
```

Where

- **FILE_LOCATION** is the full path and name of the certificate file.
- **ALIAS** with an alias for the certificate.
- **TRUSTSTORE_PASSWORD** is a password for the truststore.
- **TRUSTSTORE_LOCATION** is one of the following truststore paths:

<table>
<thead>
<tr>
<th>If You Are Using</th>
<th>Then Import the Certificate to the Keystore in this Directory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle jrockit_R27.3.1-jdk</td>
<td>JROCKIT_HOME/jre/lib/security</td>
</tr>
</tbody>
</table>
17.1.3 Setting Up Oracle Identity Manager for the Integration

**Note:** These instructions assume that you have already installed Oracle Identity Manager and that you are an Oracle Identity Manager administrator who can perform the different configuration tasks described in this section.

To prepare Oracle Identity Manager for the integration you must perform the tasks described in the following topics:

- Installing and Configuring the Generic LDAP Connector
- Creating an Application Instance

### 17.1.3.1 Installing and Configuring the Generic LDAP Connector

You must download and install a generic LDAP connector file that works with your LDAP identity store as a target.

For installation instructions, refer to "Installing Connectors" in the Oracle Fusion Middleware Administrator’s Guide for Oracle Identity Manager.

### 17.1.3.2 Creating an Application Instance

After installing the connector, you must create an application instance and make it available to Catalog.


### 17.1.4 Running the opamSetup Script

For the Oracle Privileged Account Manager-Oracle Identity Manager integration to become operational, you must run the Oracle Privileged Account Manager-Oracle Identity Manager integration setup script (opamSetup), which is available in the following directory:

```
<OIM Oracle Home>/server/bin
```

Run one of the following commands to start the script:

- For UNIX, use `opamSetup.sh`
- For Windows, use `opamSetup.bat`

```bash
opamSetup -oimUrl <OIM URL> -oimUser <OIM username> -oimPassword <OIM user password> -opamItResource <OPAM IT resource name> -opamServer <OPAM server name> -opamPort <OPAM server port> -opamUser <OPAM user> -opamPassword <OPAM user password> -idStoreItResource <ID Store IT resource name> [-ctxFactory <Initial context factory>] [-help]
```
where:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-oimUrl &lt;OIM URL&gt;</td>
<td>Provide the URL address for the Oracle Identity Manager server.</td>
</tr>
<tr>
<td>-oimUser &lt;OIM username&gt;</td>
<td>Provide a Oracle Identity Manager log-in user name.</td>
</tr>
<tr>
<td>-oimPassword &lt;OIM user password&gt;</td>
<td>Provide the Oracle Identity Manager log-in password.</td>
</tr>
<tr>
<td>-opamItResource &lt;OPAM IT resource name&gt;</td>
<td>Provide the Oracle Privileged Account Manager IT resource name.</td>
</tr>
<tr>
<td>-opamServer &lt;OPAM server name&gt;</td>
<td>Provide the path and directory name for the Oracle Privileged Account Manager server.</td>
</tr>
<tr>
<td>-opamPort &lt;OPAM server port&gt;</td>
<td>Provide the Oracle Privileged Account Manager server port.</td>
</tr>
<tr>
<td>-opamUser &lt;OPAM user&gt;</td>
<td>Provide a Oracle Privileged Account Manager log-in user name. Note: You must be an administrator with the User Manager Admin Role and the Security Administrator Admin Role to run this command.</td>
</tr>
<tr>
<td>-opamPassword &lt;OPAM user password&gt;</td>
<td>Provide the Oracle Privileged Account Manager log-in password.</td>
</tr>
<tr>
<td>-idStoreItResource &lt;ID Store IT resource name&gt;</td>
<td>Provide the name of the IT resource in the identity store.</td>
</tr>
<tr>
<td>-ctxFactory &lt;Initial context factory&gt;</td>
<td>Provide the name of the context factory (usually weblogic.jndi.WLInitialContextFactory).</td>
</tr>
<tr>
<td>-help</td>
<td>Optional. Display usage options for this command</td>
</tr>
</tbody>
</table>

Note: If you inadvertently omit a parameter, you will be prompted to provide it.

The opamSetup script performs the following tasks:

1. Creates the Oracle Privileged Account Manager IT resource with the opamServer, opamPort, opamUser, and opamPassword set-up script parameters.

2. Creates a UDF column named OPAM_TAGS in the Oracle Identity Manager catalog.

3. Creates an Oracle Privileged Account Manager synchronization scheduled job with the following characteristics:
   - **Name**: Oracle Privileged Account Manager Catalog Synchronization Job. If a job with this name already exists, the job appends a -1 to the name, then a -2, and so on.
   - **Schedule type**: Periodic, runs every 15 minutes.
   - **OPAMServerIdStoreItResource**: The idStoreItResource parameter of the set-up script.
   - **OpamServerItResource**: The opamItResource parameter of the set-up script.

4. Creates the OIM.OPAM.Integration system property (if it does not yet exist) and sets it to true.

If any of these tasks fail, the script automatically executes the next task.
17.1.5 Creating the **OPAM_TAGS** UDF

After setting up the Oracle Privileged Account Manager-Oracle Identity Manager integration environment, you must manually create an **OPAM_TAGS** user-defined field (**UDF**) in the Oracle Identity Manager catalog. The **OPAM_TAGS** UDF enables Oracle Privileged Account Manager to search the Oracle Identity Manager catalog.

To manually create the **OPAM_TAGS** UDF, perform the following steps:

1. Open the Oracle Identity Manager Admin Console and log in to Oracle Identity System Administration.
2. Create and activate a sandbox.

3. In the left pane, under **System Entities**, click **Catalog** to open the manage Catalog page.
4. Click the **Create a custom field** icon.
5. When the Select Field Type dialog box displays, select the **Text** field type to create a text field. Click **OK**.
6. When the page to create a custom field displays, specify the following settings:
   - **Appearance section**: Type **OPAM tags** in the **Display Label** field.
   - **Name section**: Type **OPAM_TAGS** in the **Name** field and type **OPAM metadata tags** in the **Description** field.
   - **Constraints section**: Check the **Searchable** box.
   - **Maximum length**: Type **256**.
   - **Default Value section**: Leave field blank.
   - **Advanced section**: Do not check any of the properties boxes.
7. Click **Save and Close**, then verify that the UDFs appear in the custom fields table.
8. Select the Manage Sandboxes tab and click **Publish Sandbox**.

---

**Note:** For detailed instructions about publishing a sandbox, refer to the "Managing Sandboxes" section in the *Oracle Fusion Middleware Developer’s Guide for Oracle Identity Manager*.

---

17.1.6 Tagging Catalog Entries with Oracle Privileged Account Manager Metadata

The Oracle Privileged Account Manager Catalog Synchronization Job created by the **opamSetup** script tags the catalog entries with the Oracle Privileged Account Manager metadata. This job automatically runs every 15 minutes.

If you need to run the job immediately, instead of waiting for the next cycle to begin, you can manually perform the following steps from the Oracle Identity Manager Admin Console:

1. Click **Scheduler**.
2. When the new screen displays, search for and select the **OPAM Catalog Synchronization** job.

3. Click **Run Now**.

4. After the job finishes, click **Refresh**.

5. To verify that the job ran successfully, check the **Job History** view.

---

**Note:** If you add new targets or accounts to Oracle Privileged Account Manager, you must run the Oracle Privileged Account Manager Catalog Synchronization Job again.

---

# 17.2 Integrating with Oracle Access Management Access Manager

This section explains how Oracle Access Management Access Manager (Access Manager) integrates with Oracle Privileged Account Manager. Using this integration scenario, you can protect Oracle Privileged Account Manager with Access Manager using a WebGate agent.

The topics in this section include:

- Before You Begin
- Enabling Single Sign-On

## 17.2.1 Before You Begin

Before starting the procedure described in Section 17.2.2, "Enabling Single Sign-On," be aware of the following:

- The instructions assume that you configured Oracle Internet Directory as the identity store; however, other component configurations are possible. Refer to the system requirements and certification documentation on Oracle Technology Network for more information about supported configurations.

- In addition, the instructions describe a specific example of using Access Manager to protect URLs. Although they outline the general approach for this type of configuration, you are not limited to using the exact steps and components described here. For example, Oracle Internet Directory is one of several identity stores certified with Access Manager 11g.

- You can use Oracle Adaptive Access Manager as an authentication option with Access Manager. Oracle Adaptive Access Manager provides strong-authentication and risk-based authorization that can be used to provide layered security for Oracle Privileged Account Manager.

To enable Oracle Adaptive Access Manager with Oracle Privileged Account Manager, select **Access Manager** as the authentication option for the WebGate that is protecting Oracle Privileged Account Manager.

- If you deployed Oracle Identity Navigator with Oracle Privileged Account Manager, and you are using Oracle Identity Navigator as the user interface for Oracle Privileged Account Manager, you can also protect Oracle Identity Navigator with Access Manager while enabling Single Sign-On.

Refer to "Integrating with Oracle Identity Navigator" in *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite* for instructions.

- Oracle Privileged Account Manager is protected by the domain agent out-of-the-box.
17.2.2 Enabling Single Sign-On

By default, the Access Manager 11g agent provides Single Sign On functionality for Oracle Privileged Account Manager and the following Identity Management consoles:

- Oracle Identity Manager
- Access Manager
- Oracle Adaptive Access Manager
- Oracle Authorization Policy Manager
- Oracle Identity Navigator

The Access Manager agent can only protect consoles in a single domain. If your environment spans multiple domains, you can use Access Manager 11g WebGate for Oracle HTTP Server 11g. Configuring Oracle Privileged Account Manager for WebGate-based single sign-on is the same as configuring Oracle Identity Navigator. Refer to "Integrating with Oracle Identity Navigator" in Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite.

You can use Access Manager to enable Single Sign On for the Oracle Privileged Account Manager's user interface by using any Access Manager authentication scheme as the challenge method.

The prerequisites are as follows:

- Oracle HTTP Server has been installed.
  When installing the Oracle HTTP Server, deselect Oracle WebCache and associated selected components with WebLogic domain (or WebSphere Cell).
- Access Manager 11g has been installed and configured properly.
- Oracle HTTP Server 11g has been installed and configured as a front-ending proxy web server for Oracle Privileged Account Manager.
- Access Manager 11g WebGate for Oracle HTTP Server 11g has been installed on the Oracle HTTP Server 11g.

See Also: Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management for details about installation of the listed components.

The steps for enabling Single Sign On in Oracle Privileged Account Manager are as follows:

1. Log in to the Oracle Access Management Console at http://oam_host:port/oamconsole

2. Click Application Domains in the Access Manager section.

3. When the Application Domains page is displayed, click Search.

4. In the search results, click IAM Suite.

5. Select the Resources tab.

6. Type /oinav/** in the Resource URL field, and then click Search.

7. Verify that the /oinav/** URL is listed as a resource in the search results and that its Authentication Policy is listed as a Protected Resource Policy.
- If the /oinav/** URL is not listed, then you must use the Access Management Console to configure a new resource for the agent under which the Oracle Privileged Account Manager URL is to be protected. Refer to Section 17.2.2.1, "Configure a New Resource for the Agent" for more information.

- If the /oinav/** URL is available in the list, then you can proceed to step 8.

8. Configure Oracle HTTP Server to point to the Access Manager domain which has the resources and policies configured. Refer to Section 17.2.2.2, "Configure Oracle HTTP Server for the Access Manager Domain" for more information.

9. Use the Access Management Console to add the following new identity providers:
   - Access Manager Identity Asserter
   - Oracle Internet Directory Authenticator

Refer to Section 17.2.2.3, "Add New Identity Providers" for more information.

10. Use a WLST command to enable access to more than one application using multiple tabs in a browser session. Refer to Section 17.2.2.4, "Configure Access to Multiple Applications" for more information.

### 17.2.2.1 Configure a New Resource for the Agent

After deploying Access Manager, if the /oinav/** URL is not available in the Agent IAMSuite's resource list, then you must configure that resource. The high-level steps for this process follow:

---

**Note:** For more information, refer to "Defining Resources in an Application Domain" in the Oracle Fusion Middleware Administrator's Guide for Oracle Access Management.

---

1. Log in to the Oracle Access Management Console at
   
   http://oam_host:port/oamconsole

2. Select the Policy Configuration tab.

3. Under Application Domains, select the agent under which the Oracle Privileged Account Manager URL is to be protected (for example, -OIMDomain).

4. Select the Resources tab and click the add icon to add a new resource. In the Resources table, enter the Resource Type, Host Identifier, and Resource URL value (/oinav/.../*) and click the Apply button.

5. Choose Protected Policy or the policy whose authentication schema is the LDAP schema. In the Resources table, click the add icon and choose the Oracle Privileged Account Manager URL (/oinav/.../*) from the drop-down list.


### 17.2.2.2 Configure Oracle HTTP Server for the Access Manager Domain

Perform these steps to ensure that Oracle HTTP Server front ends the Oracle WebLogic Server container where Oracle Privileged Account Manager is installed.

1. Navigate to the Oracle HTTP Server server config directory. For example, 
   
   /scratch/mydir1/oracle/product/11.1.1/as_1/instances/instance1/config
2. Locate the `<IfModule mod_weblogic.c>` block in the `mod_wl_ohs.conf` file. Add the host and the port number of the Oracle Privileged Account Manager URL to be protected. For example:

   ```
   MatchExpression /oinav* WebLogicHost=host WebLogicPort=port
   ```

   Where `host` and `port` correspond to the host and port on which the Oracle Privileged Account Manager Console was configured.

3. Restart the Oracle HTTP Server server in the OHS install `bin` directory, for example

   ```
   /scratch/mydir1/oracle/product/11.1.1/as_1/instances/instance1/bin
   ```

   by executing the following command:

   ```
   ./opmnctl restartproc ias=component=ohs1
   ```

17.2.2.3 Add New Identity Providers

Perform these steps to add two new identity providers:

1. Log into the Oracle Access Management Console and navigate to `Security Realms > myrealm > Providers`.

2. Add the following providers:
   - Access Manager Identity Asserter
   - Oracle Internet Directory Authenticator

3. Set the Control Flag of the Access Manager Identity Asserter to `Required`.

4. Update the following settings in the Oracle Internet Directory Authenticator:
   - Set the Control Flag to `Sufficient`
   - Select the `Provider specific` tab and make the necessary changes, supplying the host, port, and other credentials of the Oracle Internet Directory server. Configure the correct LDAP setting in the Oracle Internet Directory Authenticator.

The users and Groups in the LDAP will be reflected in the console.

5. Reorder the providers as follows:
   a. Access Manager Identity Asserter
   b. Authenticator
   c. Default Authenticator
   d. Default Identity Asserter


7. Enter the protected Oracle Privileged Account Manager URL, which uses the host and port from the Oracle HTTP Server install:

   ```
   http://OHSHost:OHSPort/oinav/faces/idmNag.jspx
   ```
17.2.2.4 Configure Access to Multiple Applications
When Single Sign On protection is provided by an 11g Access Manager Server, you must perform the following steps to configure access to applications using multiple tabs in a single browser session by changing to FORM cache mode.

1. Stop the Access Manager Managed Servers.
2. Execute the following online Access Manager WLST command:
   `configRequestCacheType(type='FORM')`
3. Restart the Access Manager Managed Servers.

17.3 Integrating with the Credential Store Framework
This section explains how Oracle Privileged Account Manager integrates with Credential Store Framework (CSF).

The topics include:
- Understanding Oracle Privileged Account Manager-Managed CSF Credentials
- Provisioning
- Lifecycle Management
- Application Consumption

17.3.1 Understanding Oracle Privileged Account Manager-Managed CSF Credentials
The Credential Store Framework (CSF) is an OPSS component that primarily provides secure storage for credentials. For example, many applications use CSF as a mechanism for storing application credentials.

Oracle Privileged Account Manager enables administrators to identify account credentials to be secured, shared, audited, and managed. In addition, Oracle Privileged Account Manager supports account lifecycle management activities such as periodic password modification.

Though many application developers use CSF to store application credentials for required targets (such as RDBMS and LDAP), there are certain aspects about how CSF is used that can potentially be improved, including:
- Applications storing their credentials in CSF do not expect these credentials to be shared. Therefore, a given instance of CSF can have multiple references to the same credential. For example, multiple applications could be relying on the same physical credential and yet have multiple logical references.
- Periodically modifying application credentials is necessary to satisfy compliance and internal IT policy requirements. However, modifying credentials (on the target and thereafter the CSF reference) remains a manual task, which is further complicated by the fact that there may be multiple references to the same credential in CSF. So, you must change the password or credential on the target and then manually update all references to that password in CSF.

Oracle Privileged Account Manager can automate this process, but automating the periodic modification of credentials is also complicated by the potential for multiple references that cannot be accurately traced.

Oracle Privileged Account Manager leverages its account lifecycle management feature to empower lifecycle management of application credentials stored in CSF.
17.3.2 Provisioning

If you decide that Oracle Privileged Account Manager will manage a particular account credential, then that credential must be provisioned through Oracle Privileged Account Manager. The following figure illustrates this provisioning process.

**Figure 17–2 Oracle Privileged Account Manager Provisioning Process**

The administrator

1. Adds an Oracle Privileged Account Manager target (if required).
2. Adds the Oracle Privileged Account Manager privileged account or credential to the target, which must include the necessary CSF mappings.

**Note:** CSF mappings are the mechanism by which a specific credential instance is uniquely identified within CSF.

The Oracle Privileged Account Manager server stores the CSF mappings along with its representation of the privileged account. The Oracle Privileged Account Manager server creates instances of the credential in CSF that correspond to the provided mappings.

**Figure 17–3 How Oracle Privileged Account Manager Uses CSF**
17.3.3 Lifecycle Management

An account provisioned as described in Section 17.3.2, "Provisioning" can have an associated Password Policy that governs password construction, periodic modification requirements, and so forth.

Oracle Privileged Account Manager normally honors and performs actions on the policy. However, whenever an administrator modifies an account credential that has associated CSF-mappings, Oracle Privileged Account Manager also updates the credential instances stored in CSF with those mappings (as shown in Figure 17–3). This update ensures that all relevant parties have access to the latest credential and allows the seamless management of password lifecycle events such as periodic modification.

17.3.4 Application Consumption

Using Oracle Privileged Account Manager to manage an application's credentials places no additional burden on that application. The only process change that occurs is that the credential must first be provisioned through Oracle Privileged Account Manager into Oracle Privileged Account Manager and CSF.

Oracle Privileged Account Manager pushes the credential to CSF with the administrator-provided mappings (as shown in Figure 17–3). If those mappings remain constant, the application can continue to access the credentials directly through CSF.
This part contains the following appendixes:

- Working with the Command Line Tool
- Working with Oracle Privileged Account Manager's RESTful Interface
- Troubleshooting Oracle Privileged Account Manager
- Glossary
Working with the Command Line Tool

You can use the Oracle Privileged Account Manager command line tool to perform many of the same tasks you perform by using the Oracle Privileged Account Manager Console. This appendix describes how to launch and work with the Oracle Privileged Account Manager command line tool.

This appendix includes the following sections:

- Section A.1, "Using the Command Line Tool"
- Section A.2, "Working with the Server"
- Section A.3, "Working with Policies"
- Section A.4, "Working with Targets"
- Section A.5, "Working with Accounts"
- Section A.6, "Working with Grantees"
- Section A.7, "Working with Plug-Ins"
- Section A.8, "Exporting and Importing Data"

Note:

- You can also use the Oracle Privileged Account Manager RESTful interface to perform many of these tasks. For more information, refer to Appendix B, "Working with Oracle Privileged Account Manager's RESTful Interface."

- The information provided in this appendix is essentially the same whether you are using Oracle Privileged Account Manager on WebLogic or on IBM WebSphere; however, there are a few minor differences.

Refer to "Differences When Using the Oracle Privileged Account Manager Command Line Tool and REST Interfaces on IBM WebSphere" in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management for more information.

- Globalization support for the Oracle Privileged Account Manager command line tool is not available for this release. The command line tool messages and help are only provided in English.
A.1 Using the Command Line Tool

This section describes how to launch and use the command line tool, and it contains the following sections:

- Section A.1.1, "Launching the Command Line Tool"
- Section A.1.2, "Issuing Commands"

A.1.1 Launching the Command Line Tool

Oracle Privileged Account Manager provides two methods for launching the command line tool:

- Launching the Command Line Tool from IAM_HOME
- Launching the Command Line Tool from Oracle Privileged Account Manager Client Archive

In most situations, you can use the instructions in Section A.1.1.1, "Launching the Command Line Tool from IAM_HOME" to launch the command line tool. However, if you want to use the Oracle Privileged Account Manager command line tool from machines other than the one where you set up Oracle Identity Management middleware, use the instructions in Section A.1.1.2, "Launching the Command Line Tool from Oracle Privileged Account Manager Client Archive."

---

**Note:** For security purposes, the Oracle Privileged Account Manager server only responds to SSL traffic.

When you provide the Oracle Privileged Account Manager server target to the Oracle Privileged Account Manager command line tool (or to Oracle Privileged Account Manager’s web-based Console), you must provide the SSL endpoint as https://hostname:sslport/opam.

By default, the WebLogic AdminServer (where the Oracle Privileged Account Manager Console runs) responds to SSL on port 7002 (In IBM WebSphere, the port is 8002). The default Oracle Privileged Account Manager server SSL port is 18102 for both WebLogic and IBM WebSphere. You can use the WebLogic console to check the port for your particular instance.

---

A.1.1.1 Launching the Command Line Tool from IAM_HOME

To launch the Oracle Privileged Account Manager command line tool:

1. Open a command window and set the $ORACLE_HOME$ and the $JAVA_HOME$ variables to the appropriate path.
   - Set $ORACLE_HOME$ to $IAM_HOME$.
   - Set $JAVA_HOME$ to the JRE location.
2. Change directory to $ORACLE_HOME$/opam/bin.
3. At the prompt, type one of the following commands:
   - On UNIX, type: opam.sh
   - On Windows, type: opam.bat

Invoking the command line tool, automatically connects you to the Oracle Privileged Account Manager server.
You can invoke the Oracle Privileged Account Manager command line tool from a remote client by providing the Oracle Privileged Account Manager server's URL (running on the same machine or on a different machine) in the `-url` option.

**A.1.1.2 Launching the Command Line Tool from Oracle Privileged Account Manager Client Archive**

The Oracle Privileged Account Manager client is also available as a standalone .zip file, located in the following directory of an Oracle Identity and Access Management suite installation:

IAM_HOME/opam/tools/opamclient.zip

Copy the archive and then follow these steps to launch the command line tool:

1. Unzip the archive on the machine where the Oracle Privileged Account Manager client is required.
   
   Unzipping the opamclient.zip file creates a top-level directory named opamclient.

2. Set the `OPAMCLIENT_HOME` variable to `<UNZIP_DIR>/opamclient` and set the `JAVA_HOME` variable to the JRE location.

3. At the prompt, type one of the following commands:
   
   - On UNIX, type: `opam.sh`
   - On Windows, type: `opam.bat`

   Invoking the command line tool, automatically connects you to the Oracle Privileged Account Manager server.

You can invoke the Oracle Privileged Account Manager command line tool by providing the Oracle Privileged Account Manager server's URL in the `-url` option.

**A.1.2 Issuing Commands**

Use the following syntax to issue any of the Oracle Privileged Account Manager commands:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x <opam-command>
```

where:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-url &lt;url&gt;</code></td>
<td>Provide the URL address for the Oracle Privileged Account Manager server.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> If you do not specify a URL for this option, it defaults to</td>
</tr>
<tr>
<td><code>-u &lt;username&gt;</code></td>
<td>Provide your log-in user name.</td>
</tr>
</tbody>
</table>
A.2 Working with the Server

The following sections contain information about the commands that you use to manage the Oracle Privileged Account Manager server.

- Section A.2.1, "getconfig Command"
- Section A.2.2, "getserverstatus Command"
- Section A.2.3, "modifyconfig Command"

A.2.1 getconfig Command

Use the getconfig command to view the OPAM Global Config configuration entry, which enables you to access and manage various Oracle Privileged Account Manager server properties.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x getconfig
- configtype <config type> <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-configtype &lt;global/session&gt;</td>
<td>Specify the configuration type.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

See Also:

modifyconfig Command
A.2.2 getserverstatus Command

Use the getserverstatus command to get the status for an Oracle Privileged Account Manager instance.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x getserverstatus <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

A.2.3 modifyconfig Command

Use the modifyconfig command to manage Oracle Privileged Account Manager server properties in the OPAM Global Config configuration entry. You can use this command to perform two types of configuration, global and session.

Global Configuration Type

The following properties are available for global configuration:

- **policyenforcerinterval**: Interval (in seconds) in which Oracle Privileged Account Manager checks accounts and then automatically checks-in the accounts that have exceeded the expiration time defined in the Usage Policy. (Default is 3600 seconds)
- **passwordcyclerinterval**: Interval (in seconds) in which Oracle Privileged Account Manager checks and then resets the password for any accounts that have exceeded the maximum password age defined in the Password Policy. (Default is 3600 seconds)
- **tdemode**: Flag to request that Oracle Privileged Account Manager use Transparent Data Encryption (TDE) mode or non-TDE mode. For more information, refer to Section 15.2, "Securing Data On Disk."

Session Configuration Type

The following properties are available for the session configuration:

- **updateinterval**: Interval (in seconds) in which the Oracle Privileged Session Manager server checks all of the checked out sessions for expiration and updates their transcripts.
- **opamserverurls**: List of Oracle Privileged Account Manager server URLs to which the Session Manager can connect.
- **maxrecordsize**: Maximum recording size that is allowed per session (in KB). When this quota is reached, the session is automatically terminated.

The following properties are SSH-specific:

- **opamListenPort**: The port on which Session Manager listens for incoming SSH connections.
- **sessioncheckoutinstructions**: The checkout instructions that are presented to users for SSH sessions.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x modifyconfig
-configtype <config type> <options>
```
The following table describes the options you can use with the `modifyconfig` command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;configtype &lt;global/session&gt;</code></td>
<td>Specify the configuration type.</td>
</tr>
<tr>
<td><code>&lt;propertyname &lt;property name&gt;</code></td>
<td>Specify the server property to be modified:</td>
</tr>
<tr>
<td></td>
<td>■ policyenforcerinterval</td>
</tr>
<tr>
<td></td>
<td>■ passwordcyclerinterval</td>
</tr>
<tr>
<td></td>
<td>■ tdemode</td>
</tr>
<tr>
<td><code>&lt;propertyvalue &lt;property value&gt;</code></td>
<td>Specify the property value to be modified.</td>
</tr>
<tr>
<td><code>&lt;help&gt;</code></td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

For example,

```
-x modifyconfig -configtype global -propertyname policyenforcerinterval
-propertyvalue 600
```

or

```
-x modifyconfig -configtype global -propertyname tdemode
-propertyvalue true
```

**See Also:**

*getconfig Command*

## A.3 Working with Policies

The following sections contain information about the commands that you use when working with Oracle Privileged Account Manager Password Policies and Usage Policies.

- Section A.3.1, "addpasswordpolicy Command"
- Section A.3.2, "addusagepolicy Command"
- Section A.3.3, "modifypasswordpolicy Command"
- Section A.3.4, "modifyusagepolicy Command"
- Section A.3.5, "removepasswordpolicy Command"
- Section A.3.6, "removeusagepolicy Command"
- Section A.3.7, "retrievepasswordpolicy Command"
- Section A.3.8, "retrieveusagepolicy Command"

### A.3.1 addpasswordpolicy Command

Use the `addpasswordpolicy` command to add a Password Policy.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x addpasswordpolicy <options>
```

The following table describes the options you can use with this command:
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-policyname &lt;policy name&gt;</code></td>
<td>Provide a name for the new Password Policy.</td>
</tr>
<tr>
<td><code>-policystatus &lt;active/disabled&gt;</code></td>
<td>Specify the Password Policy status.</td>
</tr>
<tr>
<td><code>[-passwordchagedurationunit &lt;minutes/hours/days&gt;]</code></td>
<td>Optional. Specify the password age unit.</td>
</tr>
<tr>
<td><code>[-passwordchagedurationvalue &lt;password change duration value&gt;]</code></td>
<td>Optional. Specify the password age value.</td>
</tr>
<tr>
<td><code>[-changeoncheckin &lt;true/false&gt;]</code></td>
<td>Optional. Specify whether to change the password when checking in the account using this Password Policy.</td>
</tr>
<tr>
<td><code>[-changeoncheckout &lt;true/false&gt;]</code></td>
<td>Optional. Specify whether to change the password when checking out the account using this Password Policy.</td>
</tr>
<tr>
<td><code>[-passwordcharsmin &lt;password minimum chars number&gt;]</code></td>
<td>Optional. Specify the minimum character length restriction for the Password Policy.</td>
</tr>
<tr>
<td><code>[-passwordcharsmax &lt;password maximum chars number&gt;]</code></td>
<td>Optional. Specify the maximum character length restriction for the Password Policy.</td>
</tr>
<tr>
<td><code>[-passwordalphabeticmin &lt;password minimum alphabetic chars number&gt;]</code></td>
<td>Optional. Specify the minimum number of alphabetic characters required for the Password Policy.</td>
</tr>
<tr>
<td><code>[-passwordnumericmin &lt;password minimum numeric chars number&gt;]</code></td>
<td>Optional. Specify the minimum number of numeric characters required for the Password Policy.</td>
</tr>
<tr>
<td><code>[-passwordalphanumericmin &lt;password minimum alphanumeric chars number&gt;]</code></td>
<td>Optional. Specify the minimum number of alphanumeric characters required for the Password Policy.</td>
</tr>
<tr>
<td><code>[-passworduniqueemin &lt;password minimum unique chars number&gt;]</code></td>
<td>Optional. Specify the minimum number of unique characters required for the Password Policy.</td>
</tr>
<tr>
<td><code>[-passworduppercasein &lt;password minimum uppercase chars number&gt;]</code></td>
<td>Optional. Specify the minimum number of uppercase characters required for the Password Policy.</td>
</tr>
<tr>
<td><code>[-passwordlowercasein &lt;password minimum lowercase chars number&gt;]</code></td>
<td>Optional. Specify the minimum number of lowercase characters required for the Password Policy.</td>
</tr>
<tr>
<td><code>[-passwordspecialmin &lt;password minimum special chars number&gt;]</code></td>
<td>Optional. Specify the minimum number of special characters required for the Password Policy.</td>
</tr>
<tr>
<td><code>[-passwordspecialmax &lt;password maximum special chars number&gt;]</code></td>
<td>Optional. Specify the maximum number of special characters allowed for the Password Policy.</td>
</tr>
<tr>
<td><code>[-passwordrepeatedmin &lt;password minimum repeated chars number&gt;]</code></td>
<td>Optional. Specify the minimum number of repeated characters allowed for the Password Policy.</td>
</tr>
<tr>
<td><code>[-passwordrepeatedmax &lt;password maximum repeated chars number&gt;]</code></td>
<td>Optional. Specify the maximum number of repeated characters allowed for the Password Policy.</td>
</tr>
<tr>
<td><code>[-startingchar &lt;true/false&gt;]</code></td>
<td>Optional. Specify whether the first character of the generated password can be a numeric character. If you specify <code>true</code>, then the password cannot start with a number.</td>
</tr>
<tr>
<td><code>[-isaccountnameallowed &lt;true/false&gt;]</code></td>
<td>Optional. Specify whether the generated password can be identical to the account name.</td>
</tr>
<tr>
<td><code>[-requiredchars &lt;required chars&gt;]</code></td>
<td>Optional. Specify characters that are required in the generated password. Use the comma (,) symbol to separate the characters. For example, a, b, c.</td>
</tr>
<tr>
<td><code>[-allowedchars &lt;allowed chars&gt;]</code></td>
<td>Optional. Specify characters that are allowed in the generated password. Use the comma (,) symbol to separate the characters. For example, a, b, c.</td>
</tr>
</tbody>
</table>
For example:

```bash
-url https://hostname:sslport/opam -u opamuser1 -p hr_password123 [-debug]
-x addpasswordpolicy -policyname password_policy_hr -policystatus active
-changeoncheckin true
```

**A.3.2 addusagepolicy Command**

Use the `addusagepolicy` command to add a Usage Policy.

**Command Syntax:**

```
[-url <url>] -u <username> [-p <password>] [-debug] -x addusagepolicy <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-policyname &lt;policy name&gt;</code></td>
<td>Provide a name for the new Usage Policy.</td>
</tr>
<tr>
<td><code>-policystatus &lt;active/disabled&gt;</code></td>
<td>Specify the Usage Policy status.</td>
</tr>
<tr>
<td><code>-description &lt;policy description&gt;</code></td>
<td>Optional. Provide a description of the Usage Policy.</td>
</tr>
<tr>
<td><code>-dateorduration &lt;date/duration&gt;</code></td>
<td>Set an expiration time based on date or duration.</td>
</tr>
</tbody>
</table>
| `-expireddatemutinutesfromcheckout <minutes to expiration>` | Optional. Specify the number of minutes until expiration. When a checked-out account with this Usage Policy exceeds the specified duration, Oracle Privileged Account Manager automatically checks-in that account.  
  **Note:** This field becomes a required field if you specify `duration` for the `-dateorduration` attribute. |
| `-expireddate <expiration date>` | Optional. Specify the expiration date. When an account with this Usage Policy meets this expiration date, Oracle Privileged Account Manager automatically checks-in that account.  
  **Note:** This field becomes a required field if you specify `date` for the `-dateorduration` attribute. |
| `-timezone <time zone>`         | Specify a time zone for the Usage Policy, including the timezone region.  
  For example, (GMT -6:00) America/Chicago. |
For example:
-url https://hostname:sslport/opam -u opamuser1 -p hr_password123 [-debug]
-x addusagepolicy -policyname usage_policy_fromPMtoAM -policystatus active
-dateorduration duration -expireddateminutesfromcheckout 120
-timezone (GMT -6:00) America/Chicago
monday:12:0:am|tuesday:1:15:am:2:35:pm

**A.3.3 modifypasswordpolicy Command**

Use the modifypasswordpolicy command to modify a Password Policy.

Command Syntax:

```bash
[-url <url>] -u <username> [-p <password>] [-debug] -x modifypasswordpolicy <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-policyname &lt;policy name&gt;</td>
<td>Specify the Password Policy to be modified.</td>
</tr>
<tr>
<td>-propertyname &lt;property name&gt;</td>
<td>Specify the property name that you want to modify.</td>
</tr>
<tr>
<td>-propertyvalue &lt;property value&gt;</td>
<td>Specify the property value that you want to modify.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

For example:
-url https://hostname:sslport/opam -u opamuser1 -p hr_password123 [-debug]
-x modifypasswordpolicy -policyname password_policy_hr
-propertyname changeoncheckin -propertyvalue true

**A.3.4 modifyusagepolicy Command**

Use the modifyusagepolicy command to modify a Usage Policy.

Command Syntax:

```bash
[-url <url>] -u <username> [-p <password>] [-debug] -x modifyusagepolicy <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-usagedates &lt;dates information of usage policy&gt;]</td>
<td>Specify the usage dates information for the policy by using the pipe (</td>
</tr>
</tbody>
</table>
| For example,                | monday:12:0:am|tuesday:1:15:am|Wednesday:12:0:am:12:0:am

**-enablerecording <true/false>**

Set this flag to enable (true) or disable (false) session recording when applying the Usage Policy to a session checkout. (Default is true.)

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

For example:
-url https://hostname:sslport/opam -u opamuser1 -p hr_password123 [-debug]
-x modifyusagepolicy -policyname usage_policy_fromPMtoAM -policystatus active
-dateorduration duration -expireddateminutesfromcheckout 120
-timezone (GMT -6:00) America/Chicago
monday:12:0:am|tuesday:1:15:am:2:35:pm
A.3.5 removepasswordpolicy Command

Use the removepasswordpolicy command to remove a Password Policy.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x removepasswordpolicy <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-policyname &lt;policy name&gt;</td>
<td>Specify the Password Policy to remove.</td>
</tr>
<tr>
<td></td>
<td>-help</td>
</tr>
</tbody>
</table>

For example:

```
-url https://hostname:sslport/opam -u opamuser1 -p hr_password123 [-debug]
-x removepasswordpolicy -policyname password_policy_hr
```

A.3.6 removeusagepolicy Command

Use the removeusagepolicy command to remove a Usage Policy.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x removeusagepolicy <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-policyname &lt;policy name&gt;</td>
<td>Specify the Usage Policy to remove.</td>
</tr>
<tr>
<td></td>
<td>-help</td>
</tr>
</tbody>
</table>

For example:

```
-url https://hostname:sslport/opam -u opamuser1 -p hr_password123 [-debug]
-x removeusagepolicy -policyname usage_policy_fromPMtoAM
propertyname changeoncheckin -propertyvalue true
```
A.3.7 retrievepasswordpolicy Command

Use the `retrievepasswordpolicy` command to retrieve a Password Policy.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x retrievepasswordpolicy <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-polycname &lt;policy name&gt;</td>
<td>Specify the Password Policy to be retrieved.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

For example:

```
-url https://hostname:sslport/opam -u opamuser1 -p hr_password123 [-debug]
-x retrievepasswordpolicy -polycname password_policy_hr
```

A.3.8 retrieveusagepolicy Command

Use the `retrieveusagepolicy` command to retrieve a Usage Policy.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x retrieveusagepolicy <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-polycname &lt;policy name&gt;</td>
<td>Specify the Usage Policy to be retrieved.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

For example:

```
-url https://hostname:sslport/opam -u opamuser1 -p hr_password123 [-debug]
-x retrieveusagepolicy -polycname usage_policy_hr
```

A.4 Working with Targets

The following sections contain information about the commands that you use when working with Oracle Privileged Account Manager targets.

- Section A.4.1, "addtarget Command"
- Section A.4.2, "displayalltargets Command"
- Section A.4.3, "modifytarget Command"
- Section A.4.4, "removetarget Command"
A.4.1 addtarget Command

Use the addtarget command to add a target.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x addtarget <options>
```

Oracle Privileged Account Manager supports multiple target types, and each target type has different required and optional parameters. You must specify the target type to see the target-specific options, as follows:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-targettype &lt;ldap</td>
<td>unix</td>
</tr>
</tbody>
</table>

**Note:** These options should be discovered at run time, before you execute the addtarget command.

The following examples illustrate the commands you can execute to list

- Example A–1, "Supported Target Types"
- Example A–2, "Required and Optional Parameters for a Specific Target Type"

**Example A–1  Supported Target Types**

```
sh opam.sh -url <OPAM url> -u <security admin user> -p <security admin user password> -x addtarget -help
```

For example, if https://hostname:sslport/opam is the Oracle Privileged Account Manager server URL, execute the following command:

```
sh opam.sh -url https://hostname:sslport/opam -u sec_admin -p welcome1 -x addtarget -help
```

**Example A–2  Required and Optional Parameters for a Specific Target Type**

```
sh opam.sh -url <OPAM url> -u <security admin user> -p <security admin user password> -x addtarget -targettype <any supported target type> -help
```

For example, if you are using the LDAP target type with https://hostname:sslport/opam as the Oracle Privileged Account Manager server URL, execute the following command:

```
sh opam.sh -url https://hostname:sslport/opam -u sec_admin -p welcome1 -x addtarget -targettype ldap -help
```
Refer to the following sections for a description of the parameters used with the different target types:

- Section A.4.1.1, "ldap Target Type Parameters"
- Section A.4.1.2, "database Target Type Parameters"
- Section A.4.1.3, "unix Target Type Parameters"
- Section A.4.1.4, "lockbox Target Type Parameters"

### A.4.1.1 ldap Target Type Parameters

The following table describes the ldap target type parameters that you can use with this command.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-targetname &lt;targetname&gt;</td>
<td>Provide a name for the target.</td>
</tr>
<tr>
<td>-domain &lt;domain&gt;</td>
<td>Provide a domain name.</td>
</tr>
<tr>
<td>-host &lt;host&gt;</td>
<td>Provide the host name.</td>
</tr>
<tr>
<td>-port &lt;port&gt;</td>
<td>Provide the TCP/IP port number used to communicate with the LDAP server.</td>
</tr>
<tr>
<td>[-ssl &lt;ssl&gt;]</td>
<td>Optional. Specify to connect to the LDAP server using SSL.</td>
</tr>
<tr>
<td>-principal &lt;principal&gt;</td>
<td>Provide the distinguished name with which to authenticate to the LDAP server.</td>
</tr>
<tr>
<td>-credentials &lt;credentials&gt;</td>
<td>Provide the principal's password.</td>
</tr>
<tr>
<td>[-passwordpolicy] &lt;password policy name&gt;</td>
<td>Optional. Identify a Password Policy to apply to the target. (See Note following table.)</td>
</tr>
<tr>
<td>[-passwordpolicyid] &lt;password policy ID&gt;</td>
<td>Optional. Identify a Password Policy to apply to the target. (See Note following table.)</td>
</tr>
<tr>
<td>-baseContexts &lt;baseContexts&gt; [Multi-Valued]</td>
<td>Specify one or more starting points in the LDAP tree to use when searching the tree. Searches are performed when discovering users from the LDAP server or when looking for groups in which the user is a member.</td>
</tr>
<tr>
<td>-accountNameAttribute &lt;accountNameAttribute&gt;</td>
<td>Identify the attribute that holds the account's user name.</td>
</tr>
<tr>
<td>[-organization &lt;organization&gt;]</td>
<td>Optional. Provide the organization name.</td>
</tr>
<tr>
<td>[-uidAttribute &lt;uidAttribute&gt;]</td>
<td>Optional. Provide the name of the LDAP attribute that is mapped to the UID attribute. (Defaults to uid)</td>
</tr>
<tr>
<td>[-accountSearchFilter &lt;accountSearchFilter&gt;]</td>
<td>Optional. Provide an LDAP filter to control which accounts are returned from the LDAP resource. If you do not specify a filter, then only accounts that include all specified object classes will be returned. (Defaults to (uid=*))</td>
</tr>
<tr>
<td>-passwordAttribute &lt;passwordAttribute&gt;</td>
<td>Optional. Identify the LDAP attribute that holds the password. When changing a user's password, Oracle Privileged Account Manager sets the new password to this attribute. (Defaults to userpassword)</td>
</tr>
</tbody>
</table>
A.4.1.2 database Target Type Parameters

The following table describes the database target type parameters that you can use with this command.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-targetname &lt;targetname&gt;</td>
<td>Provide a name for the target.</td>
</tr>
<tr>
<td>-domain &lt;domain&gt;</td>
<td>Provide a domain name.</td>
</tr>
<tr>
<td>-host &lt;host&gt;</td>
<td>Provide the host name.</td>
</tr>
<tr>
<td>-jdbcUrl &lt;jdbcUrl&gt;</td>
<td>Provide the JDBC URL that identifies the target system location.</td>
</tr>
<tr>
<td></td>
<td>Following are some example URL formats:</td>
</tr>
<tr>
<td></td>
<td>- For Oracle: jdbc:oracle:thin:@&lt;host&gt;:&lt;port&gt;:&lt;sid&gt;</td>
</tr>
<tr>
<td></td>
<td>- For MSSQL: jdbc:sqlserver://&lt;host&gt;:&lt;port&gt;;database=&lt;database&gt;</td>
</tr>
<tr>
<td></td>
<td>- For MySQL: jdbc:mysql://&lt;host&gt;:&lt;port&gt;/database</td>
</tr>
<tr>
<td></td>
<td>- For DB2: jdbc:db2://&lt;host&gt;:&lt;port&gt;/database</td>
</tr>
<tr>
<td></td>
<td>- For Sybase: jdbc:sybase:Tds:&lt;host&gt;:&lt;port&gt;/database</td>
</tr>
<tr>
<td>-loginUser &lt;loginUser&gt;</td>
<td>Provide the Admin User name.</td>
</tr>
<tr>
<td>-loginPassword &lt;loginPassword&gt;</td>
<td>Provide the Admin User's password.</td>
</tr>
<tr>
<td>-dbType &lt;dbType&gt;</td>
<td>Specify the database type for which the connector is being used. The</td>
</tr>
<tr>
<td></td>
<td>connector supports the Oracle, MSSQL, MySQL, DB2, and Sybase database types.</td>
</tr>
<tr>
<td>Note: You can also configure the connector to work against custom</td>
<td></td>
</tr>
<tr>
<td></td>
<td>database types.</td>
</tr>
<tr>
<td>-description &lt;description&gt;</td>
<td>Optional. Provide a description of the target.</td>
</tr>
</tbody>
</table>

Note:

- You can use either -passwordpolicy <password policy name> or -passwordpolicyid <policy ID> to apply a Password Policy to the target.
- You must specify all multi-valued attributes in this format: value1|value2|...
### A.4.1.3 unix Target Type Parameters

The following table describes the unix target type parameters that you can use with this command.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-targetname &lt;targetname&gt;</td>
<td>Provide a name for the target.</td>
</tr>
<tr>
<td>-domain &lt;domain&gt;</td>
<td>Provide a domain name.</td>
</tr>
<tr>
<td>-host &lt;host&gt;</td>
<td>Provide the host name.</td>
</tr>
<tr>
<td>-loginUser &lt;loginUser&gt;</td>
<td>Provide a user name with which to log into the target. For example, root.</td>
</tr>
<tr>
<td>-loginUserpassword &lt;loginUserpassword&gt;</td>
<td>Provide a password for the Login user.</td>
</tr>
<tr>
<td>-loginShellPrompt &lt;loginShellPrompt&gt;</td>
<td>Provide the shell prompt to display when you log into the target. For example, $ or #.</td>
</tr>
</tbody>
</table>
### Working with Targets

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-description &lt;description&gt;\&gt;</code></td>
<td><em>Optional.</em> Provide a description of the target.</td>
</tr>
<tr>
<td><code>-organization &lt;organization&gt;\&gt;</code></td>
<td><em>Optional.</em> Provide the organization name.</td>
</tr>
<tr>
<td><code>-passwordpolicy &lt;password policy name&gt;</code></td>
<td><em>Optional.</em> Specify a Password Policy to apply to the target. (See Note following table.)</td>
</tr>
<tr>
<td><code>-passwordpolicyid &lt;password policy ID&gt;</code></td>
<td><em>Optional.</em> Specify a Password Policy to apply to the target. (See Note following table.)</td>
</tr>
</tbody>
</table>
| `-passwordrollover <passwordrollover>` | *Optional.* Specify whether you want the target’s service account password to be rolled over according to the assigned Password Policy.  
  - **true:** Rollover the service account password, based on the assigned Password Policy.  
  - **false (default):** Do not rollover the service account password.  
  
  **Note:** Password rollover for target service accounts is similar to password expiration for privileged accounts. If a password has not been changed by the expiration date configured in the associated Password Policy, then Oracle Privileged Account Manager will automatically change the password to a randomized value. |
| `-sudoAuthorization <sudoAuthorization>` | *Optional.* Specify whether the user required sudo authorization.  
  - **true:** Do not require sudo authorization.  
  - **false (default):** Require sudo authorization for root user. |
| `-commandTimeout <commandTimeout>` | *Optional.* Specify the command timeout value in milliseconds. (Defaults to 120000) |
| `-passwordExpectExpressions <passwordExpectExpressions>` | *Optional.* Specify the expressions to be displayed on the target when setting the user’s password.  
  For example, if the expressions displayed on running the `passwd` command are, `Enter password:` and `Re-enter password:`, then you can enter the following value for this field:  
  `enter password,re-enter password`.  
  
  **Note:** You can use a regular expression, and the two expressions must be separated by a comma.  
  (Defaults to `new[\s](unix[\s])?password:,new[\s](unix[\s])?password ([\s]again)?:`) |
| `-prepasswdExpectExpression <prePasswdExpectExpression>` | *Optional.* Specify the prompt that can be displayed on some targets before the password prompts when running the `passwd` command.  
  You must provide the prompt expression and the expected input value for that expression, separated by a comma.  
  (Defaults to `None`) |
| `-sudopasswordExpectExpressions <sudoPasswordExpectExpressions>` | *Optional.* Specify the password prompt to be displayed when running a command in `sudo` mode. (Defaults to `password:`) |
| `-force <true/false>` | *Optional.* Enable or disable the requirement for connection validation.  
  - **true:** Skips connection validation.  
  - **false (default):** Enforces connection validation. |
| `-help` | *Optional.* Displays usage options for this command. |
A.4.1.4 lockbox Target Type Parameters

The following table describes the lockbox target type parameters that you can use with this command.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-targetname &lt;targetname&gt;</td>
<td>Provide a name for the target.</td>
</tr>
<tr>
<td>-domain &lt;domain&gt;</td>
<td>Provide a domain name.</td>
</tr>
<tr>
<td>-host &lt;host&gt;</td>
<td>Provide the host name.</td>
</tr>
<tr>
<td>[-organization &lt;organization&gt;]</td>
<td>Optional. Provide the organization name.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

Note:
- You can use either -passwordpolicy <password policy name> or -passwordpolicyid <policy ID> to apply a Password Policy to the target.
- You must specify all multi-valued attributes in this format: value1|value2|...

A.4.2 displayalltargets Command

Use the displayalltargets command to display a listing of all targets.

Note: You must be an administrator with the User Manager Admin Role, the Security Administrator Admin Role, or the Security Auditor Admin Role to successfully run this command.

Command Syntax:

[-url <url>] -u <username> [-p <password>] [-debug] -x displayalltargets <options>

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

A.4.3 modifytarget Command

Use the modifytarget command to modify a target.

Command Syntax:

[-url <url>] -u <username> [-p <password>] [-debug] -x modifytarget <options>

The following table describes the options you can use with this command:
A.4.4 removetarget Command

Use the removetarget command to remove a target.

Command Syntax:
```
[-url <url>] -u <username> [-p <password>] [-debug] -x removetarget <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-targetid &lt;targetid&gt;</td>
<td><code>Optional</code>. Specify the target GUID value of the target to be modified.</td>
</tr>
<tr>
<td></td>
<td><code>Note</code>: When you configure a target, Oracle Privileged Account Manager</td>
</tr>
<tr>
<td></td>
<td>automatically assigns a unique target GUID. Refer to Section 6.2, “Adding</td>
</tr>
<tr>
<td></td>
<td>Targets to Oracle Privileged Account Manager” for more information.</td>
</tr>
<tr>
<td>-targetname &lt;targetname&gt;</td>
<td><code>Optional</code>. Specify the name of the target to be modified.</td>
</tr>
<tr>
<td>-propertyname &lt;propertyname&gt;</td>
<td>Specify the name of the property that you want to modify.</td>
</tr>
<tr>
<td>-propertyvalue &lt;propertyvalue&gt;</td>
<td>Specify the property value that you want to modify.</td>
</tr>
<tr>
<td>-force &lt;true/false&gt;</td>
<td><code>Optional</code>. Enables or disables the requirement for connection validation.</td>
</tr>
<tr>
<td></td>
<td>- true: Skips connection validation.</td>
</tr>
<tr>
<td></td>
<td>- false (default): Enforces connection validation.</td>
</tr>
<tr>
<td>-help</td>
<td><code>Optional</code>. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

Note: You use either `<targetid>` or `<targetname>` to identify a target. Both values are unique.

A.4.5 resettargetpassword Command

Use the resettargetpassword command to manually reset a target service account password. When you execute this command, Oracle Privileged Account Manager returns the target service account details and prompts you to enter a new password.
Note:

- You must be an administrator with the *Security Administrator* Admin Role to execute this command.
- This command is not applicable for the lockbox or ldap target types and will return an "Operation not supported" error message.
- Refer to Chapter 7, "Working with Service Accounts" for information about service accounts.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] -x resettarggetpassword
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-targetid &lt;target id&gt;]</td>
<td>Optional. Identify the target to be reset.</td>
</tr>
<tr>
<td>[-targetname &lt;target name&gt;]</td>
<td>Optional. Identify the target to be reset.</td>
</tr>
<tr>
<td>[-password &lt;account password&gt;]</td>
<td>Optional. Provide a new password for the target.</td>
</tr>
<tr>
<td>[-autogen &lt;true/false&gt;]</td>
<td>Optional. Use to automatically generate a password, according to account Password Policy.</td>
</tr>
<tr>
<td></td>
<td><em>true</em>: Enable the system to automatically generate passwords.</td>
</tr>
<tr>
<td></td>
<td><em>false</em> (default): Disable the system’s ability to automatically generate passwords. Users must specify passwords.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

Note:

- You use either `<targetid>` or `<targetname>` to identify the target.
- You use either `<password>` or `<autogen>` to create a new password for the target.

**A.4.6 retrievetarget Command**

Use the retrievetarget command to get information about a target.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x retrievetarget <options>
```

The following table describes the options you can use with this command:
A.4.7 searchtarget Command

Use the searchtarget command to search for a target.

Command Syntax:

```bash
[-url <url>] -u <username> [-p <password>] [-debug] -x searchtarget <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-targettype &lt;ldap</td>
<td>Optional. Identify the type of target to search for as LDAP, Solaris, or Oracle DB.</td>
</tr>
<tr>
<td></td>
<td>-oracledb&gt;</td>
</tr>
<tr>
<td>-domain &lt;domain&gt;</td>
<td>Optional. Provide a domain to search.</td>
</tr>
<tr>
<td>-targetname &lt;target name&gt;</td>
<td>Optional. Provide the target name to search for.</td>
</tr>
<tr>
<td>-help</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

Note: You use either <targetid> or <targetname> to identify the target. Both values are unique.

A.4.8 showtargetpassword Command

Use the showtargetpassword command to view the password for a target service account. When you execute this command, Oracle Privileged Account Manager returns the target service account details and the password.

Note:

- You must be an administrator with the Security Administrator Admin Role to execute this command.
- This command is not applicable for the lockbox target type and will return an "Operation not supported" error message.
- Refer to Chapter 7, "Working with Service Accounts" for information about service accounts.

Command Syntax:

```bash
[-url <url>] -u <username> [-p <password>] -x showtargetpassword
```

The following table describes the options you can use with this command:
A.4.9 showtargetpasswordhistory Command

Use the `showtargetpasswordhistory` command to view the password history for a target where you have reset the password. When you execute this command, Oracle Privileged Account Manager returns the password history.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] -x showtargetpasswordhistory -targetid <targetid> <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-targetid &lt;target id&gt;]</td>
<td>Optional. Identify the target for which you are searching.</td>
</tr>
<tr>
<td>[-targetname &lt;target name&gt;]</td>
<td>Optional. Identify the name of the target for which you are searching.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

Note: You use either `<targetid>` or `<targetname>` to identify the target.

A.5 Working with Accounts

The following sections contain information about the commands that you use when working with Oracle Privileged Account Manager privileged accounts.

- Section A.5.1, "addaccount Command"
- Section A.5.2, "displayallaccounts Command"
- Section A.5.3, "checkin Command"
- Section A.5.4, "checkout Command"
- Section A.5.5, "displaycheckedoutaccounts Command"
- Section A.5.6, "modifyaccount Command"
- Section A.5.7, "removeaccount Command"
A.5.1 addaccount Command

Use the addaccount command to add a privileged account.

**Note:** You must never use the same account as the service account and as a privileged account to be managed by Oracle Privileged Account Manager. Refer to Chapter 7, "Working with Service Accounts" for information about service accounts.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x addaccount <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-targetid &lt;id&gt;]</td>
<td>Optional. Specify the target GUID value of a configured target.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> When you configure a target, Oracle Privileged Account Manager automatically assigns a unique target GUID. Refer to Section 6.2, &quot;Adding Targets to Oracle Privileged Account Manager&quot; for more information.</td>
</tr>
<tr>
<td>[-targetname &lt;name&gt;]</td>
<td>Optional. Specify the target name of a configured target.</td>
</tr>
<tr>
<td>[-password &lt;password&gt;]</td>
<td>Optional. Specify a default value for the account password.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> This field becomes a required field if the target type is lockbox.</td>
</tr>
<tr>
<td>-accountname &lt;name&gt;</td>
<td>Provide a name for the new account.</td>
</tr>
<tr>
<td>[-force &lt;true/false&gt;]</td>
<td>Optional. Enables or disables the requirement for connection validation.</td>
</tr>
<tr>
<td></td>
<td>- true: Skips connection validation.</td>
</tr>
<tr>
<td></td>
<td>- false (default): Enforces connection validation.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

**Note:**

- You use either `<targetid>` or `<targetname>` to identify the target. Both values are unique.
- You can use `-password` to set up an account password.
A.5.2 displayallaccounts Command

Use the displayallaccounts command to display a listing of all accounts.

Note: You must be an administrator with the User Manager Admin Role or the Security Administrator Admin Role to successfully run this command.

Command Syntax:

[-url <url>] -u <username> [-p <password>] [-debug] -x displayallaccounts <options>

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

A.5.3 checkin Command

Use the checkin command to check in privileged accounts.

Command Syntax:

[-url <url>] -u <username> [-p <password>] [-debug] -x checkin <options>

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-accountid &lt;account id&gt;]</td>
<td>Optional. Identify the account to be checked-in.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>{[-accountname &lt;account name&gt;] and [-targetname &lt;target name&gt;]}</td>
<td>Optional. Identify the account to be checked-in.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> The (&lt;accountname&gt; and &lt;targetname&gt;) combination forms a unique pair that can be used to identify a specific account.</td>
</tr>
<tr>
<td>[-checkoutid &lt;checkout ID&gt;]</td>
<td>Specify the checkout ID.</td>
</tr>
<tr>
<td>[-force &lt;true/false&gt;]</td>
<td>Optional. Enables or disables the ability to force check-in a privileged account.</td>
</tr>
<tr>
<td></td>
<td>A force check-in enables administrators with the User Manager Admin Role to check-in privileged accounts that have been checked-out by other users.</td>
</tr>
<tr>
<td></td>
<td>■ <strong>true:</strong> Enables force check-ins.</td>
</tr>
<tr>
<td></td>
<td>■ <strong>false:</strong> Disables force check-ins.</td>
</tr>
<tr>
<td>[-userid &lt;userid&gt;]</td>
<td>Optional. Specifies which user is to be force checked-in.</td>
</tr>
<tr>
<td></td>
<td>Oracle Privileged Account Manager allows multiple users to check out an account at the same time. By providing a userid, the force check-in only applies to the specified user.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>
A.5.4 checkout Command

Use the checkout command to check out privileged accounts.

Note: The checkout operation also provides a password for you to use.

Command Syntax:

```bash
[-url <url>] -u <username> [-p <password>] [-debug] -x checkout <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-accountid &lt;account id&gt;]</td>
<td>Optional. Identify the account to be checked-out.</td>
</tr>
<tr>
<td>([-accountname &lt;account name&gt;] and [-targetname &lt;target name&gt;])</td>
<td>Optional. Identify the account to be checked-out.</td>
</tr>
<tr>
<td></td>
<td>Note: The (&lt;accountname&gt; and &lt;targetname&gt;) combination forms a unique pair that can be used to identify a specific account.</td>
</tr>
<tr>
<td>[-checkouttype &lt;password/session&gt;]</td>
<td>Specify the type of checkout:</td>
</tr>
<tr>
<td></td>
<td>■ password (default): Allow users to only check out passwords.</td>
</tr>
<tr>
<td></td>
<td>■ session: Allow users to only check out sessions.</td>
</tr>
<tr>
<td>[-comment &lt;comment&gt;]</td>
<td>Optional. Provide a comment about the checkout.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

Note: You use either <accountid> or (<accountname> and <targetname>) to identify the account.

A.5.5 displaycheckedoutaccounts Command

Use the displaycheckedoutaccounts command to display a listing of a user's checked out accounts.

Command Syntax:

```bash
[-url <url>] -u <username> [-p <password>] [-debug] -x displaycheckedoutaccounts <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>
A.5.6 modifyaccount Command

Use the modifyaccount command to modify a privileged account.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x modifyaccount <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-accountid &lt;account id&gt;]</td>
<td>Optional. Identify the account to be modified.</td>
</tr>
<tr>
<td>([-accountname &lt;account name&gt;] and [-targetname &lt;target name&gt;])</td>
<td>Optional. Identify the account to be modified.</td>
</tr>
<tr>
<td>-propertyname &lt;propertyname&gt;</td>
<td>Specify the name of the property that you want to modify.</td>
</tr>
<tr>
<td>-propertyvalue &lt;propertyvalue&gt;</td>
<td>Specify the property value that you want to modify.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

**Note:**

- To identify an account, you can use either `accountid` or (`accountname` and `targetname`).
- To modify an account's Password Policy, you can use either `propertyname passwordpolicy -propertyvalue <policy name>` or `propertyname passwordpolicyid -propertyvalue <policy ID>`.

A.5.7 removeaccount Command

Use the removeaccount command to remove a privileged account.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x removeaccount <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-accountid &lt;account id&gt;]</td>
<td>Optional. Identify the account to be removed.</td>
</tr>
<tr>
<td>([-accountname &lt;account name&gt;] and [-targetname &lt;target name&gt;])</td>
<td>Optional. Identify the account to be removed.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>
A.5.8 **resetpassword Command**

Use the `resetpassword` command to manually reset the password for an account you have checked out. When you execute this command, Oracle Privileged Account Manager returns the account details and prompts you to enter a new password.

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-accountid &lt;account id&gt;</code></td>
<td>Optional. Identify the account to be reset.</td>
</tr>
<tr>
<td><code>([-accountname &lt;account name&gt;] and [-targetname &lt;target name&gt;])</code></td>
<td>Optional. Identify the account to be reset. <strong>Note:</strong> The <code>&lt;accountname&gt;</code> and <code>&lt;targetname&gt;</code> combination forms a unique pair that can be used to identify a specific account.</td>
</tr>
<tr>
<td><code>-password &lt;account password&gt;</code></td>
<td>Optional. Provide a new password for the account.</td>
</tr>
</tbody>
</table>
| `-autogen <true/false>` | Optional. Use to automatically generate a password, according to the account Password Policy.  
  - `true`: Enable the system to automatically generate passwords.  
  - `false (default)`: Disable the system’s ability to automatically generate passwords. Users must specify passwords. |
| `-help` | Optional. Displays usage options for this command. |

**Note:**
- You use either `<accountid>` or `( `<accountname>` and `<targetname>`) to identify the account.
- If you use `<accountid>` or `( `<accountname>` and `<targetname>`) , you must use `-password` or `-autogen`.

A.5.9 **retrieveaccount Command**

Use the `retrieveaccount` command to get information about a privileged account, such as which target the account is on. This information does not include passwords.
Command Syntax:

[-url <url>] -u <username> [-p <password>] [-debug] -x retrieveaccount <options>

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-accountid &lt;account id&gt;]</td>
<td>Optional. Identify the account to be retrieved.</td>
</tr>
<tr>
<td>[-accountname &lt;account name&gt;] and [-targetname &lt;target name&gt;]</td>
<td>Optional. Identify the account to be retrieved.</td>
</tr>
<tr>
<td></td>
<td>Note: The (&lt;accountname&gt; and &lt;targetname&gt;) combination forms a unique pair that can be used to identify a specific account.</td>
</tr>
<tr>
<td>[-targetname &lt;target name&gt;]</td>
<td>Optional. Identify the account to be retrieved.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

**Note:** You use either <accountid> or (<accountname> and <targetname>) to identify the account.

### A.5.10 searchaccount Command

Use the searchaccount command to search for an account.

Command Syntax:

[-url <url>] -u <username> [-p <password>] [-debug] -x searchaccount <options>

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-targettype &lt;ldap</td>
<td>unix</td>
</tr>
<tr>
<td>[-domain &lt;account domain&gt;]</td>
<td>Optional. Identify the account to search for.</td>
</tr>
<tr>
<td>[-targetname &lt;target name&gt;]</td>
<td>Optional. Identify the account to search for.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

**Note:** You can use any combination of -targettype, -domain, or -targetname to identify the account. If you do not provide any of these options, the search returns all accounts.

For example, the following search will return all targets:

https://<host name>:{port}/opam/target/search?

Whereas, the following search will return all targets whose type contains ldap and org:

https://<host name>:{port}/opam/target/search?type=ldap&org=us
A.5.11 searchcheckouthistory Command

Use the searchcheckouthistory command to search the checkouts for an account that you have checked out previously. When you execute this command, Oracle Privileged Account Manager returns the checkout history.

Note: You must be an administrator with the Security Administrator Admin Role or the User Manager Admin Role to successfully run this command.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x searchcheckouthistory -accountid <accountid> -fromtime <fromTime> -totime <toTime> <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-accountid &lt;account id&gt;]</td>
<td>Optional. Identify the account to search for.</td>
</tr>
<tr>
<td>[-accountname &lt;account name&gt;]</td>
<td>Optional. Identify the account to search for.</td>
</tr>
<tr>
<td>[-targetname &lt;target name&gt;]</td>
<td>Optional. Provide the name of the target.</td>
</tr>
<tr>
<td>-fromtime &lt;from time&gt;</td>
<td>Specify the time to start searching for checkouts by using one of the following formats:</td>
</tr>
<tr>
<td>-totime &lt;to time&gt;</td>
<td>Specify the time to stop searching for checkouts by using one of the following formats:</td>
</tr>
<tr>
<td>[-uid &lt;user id&gt;]</td>
<td>Identify the user to be searched.</td>
</tr>
<tr>
<td>[-event &lt;event&gt;]</td>
<td>Specify the command executed or a term in the log.</td>
</tr>
<tr>
<td>[-size &lt;size&gt;]</td>
<td>Specify the number of results to be returned.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

A.5.12 showpassword Command

Use the showpassword command to view the password for an account that you have checked out. When you execute this command, Oracle Privileged Account Manager returns the account details and the password.

Note: If the account has already been checked back in, you will get an error.

You must be an administrator with the Security Administrator Admin Role to successfully run this command.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] -x showpassword -accountid <accountid>
```
The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-accountid &lt;account id&gt;]</td>
<td>Optional. Identify the account for which the password is being retrieved.</td>
</tr>
<tr>
<td>([-accountname &lt;account name&gt;]</td>
<td>Optional. Identify the account for which the password is being retrieved.</td>
</tr>
<tr>
<td>and [-targetname &lt;target name&gt;])</td>
<td>Note: The (&lt;accountname&gt; and &lt;targetname&gt;) combination forms a unique pair that can be used to identify a specific account.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

Note: You use either <accountid> or (<accountname> and <targetname>) to identify the account.

A.5.13 showpasswordhistory Command

Use the showpasswordhistory command to view the password history for an account that you have checked out, checked in, or reset the password. When you execute this command, Oracle Privileged Account Manager returns the password history.

Note: You must be an administrator with the Security Administrator Admin Role to successfully run this command.

Command Syntax:

[-url <url>] -u <username> [-p <password>] -x showpasswordhistory [-accountid <accountid> <options>]

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-accountid &lt;account id&gt;]</td>
<td>Optional. Identify the account to search for.</td>
</tr>
<tr>
<td>[-accountname &lt;account name&gt;]</td>
<td>Optional. Provide the name of the account to search.</td>
</tr>
<tr>
<td>[-targetname &lt;target name&gt;]</td>
<td>Optional. Provide the name of the target to search.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

A.6 Working with Grantees

The following sections contain information about the commands that you use when working with Oracle Privileged Account Manager grantees.

- Section A.6.1, "displayallgroups Command"
- Section A.6.2, "displayallusers Command"
- Section A.6.3, "grantgroupaccess Command"
- Section A.6.4, "grantuseraccess Command"
- Section A.6.5, "removegroupaccess Command"
- Section A.6.6, "removeuseraccess Command"
**A.6.1 displayallgroups Command**

Use the `displayallgroups` command to display a listing of all groups.

**Command Syntax:**

```
[-url <url>] -u <username> [-p <password>] [-debug] -x displayallgroups <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

**Note:** You must be an administrator with the User Manager Admin Role to successfully run this command.

**A.6.2 displayallusers Command**

Use the `displayallusers` command to display a listing of all users.

**Command Syntax:**

```
[-url <url>] -u <username> [-p <password>] [-debug] -x displayallusers <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

**A.6.3 grantgroupaccess Command**

Use the `grantgroupaccess` command to give a group access to a privileged account.

```
[-url <url>] -u <username> [-p <password>] [-debug] -x grantgroupaccess <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-accountid &lt;account id&gt;]</td>
<td>Optional. Identify the account to which the group is granted access.</td>
</tr>
</tbody>
</table>
A.6.4 grantuseraccess Command

Use the grantuseraccess command to give a user access to a privileged account.

Command Syntax:

[-url <url>] -u <username> [-p <password>] [-debug] -x grantuseraccess <options>

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-accountid &lt;account id&gt;]</td>
<td>Optional. Identify the account to which the user is granted access.</td>
</tr>
<tr>
<td>([-accountname &lt;account name&gt;] and [-targetname &lt;target name&gt;])</td>
<td>Optional. Identify the account to which the user is granted access. Note: The (accountname and targetname) combination forms a unique pair that can be used to identify a specific account.</td>
</tr>
<tr>
<td>-userid &lt;user id&gt;</td>
<td>Identify the user to be given access.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

Note: You use either <accountid> or (accountname and targetname) to identify the account.

A.6.5 removegroupaccess Command

Use the removegroupaccess command to remove a group’s access to a privileged account.

Command Syntax:

[-url <url>] -u <username> [-p <password>] [-debug] -x removegroupaccess <options>

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-accountid &lt;account id&gt;]</td>
<td>Optional. Identify the account where access is being removed</td>
</tr>
</tbody>
</table>

Note: You use either <accountid> or (accountname and targetname) to identify the account.
A.6.6 removeuseraccess Command

Use the removeuseraccess command to remove a user's access to a privileged account.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x removeuseraccess <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-accountid &lt;account id&gt;]</td>
<td>Optional. Identify the account where access is being removed.</td>
</tr>
<tr>
<td>([-accountname &lt;account name&gt;] and [-targetname &lt;target name&gt;])</td>
<td>Optional. Identify the account where access is being removed. Note: The (&lt;accountname&gt; and &lt;targetname&gt;) combination forms a unique pair that can be used to identify a specific account.</td>
</tr>
<tr>
<td>-userid &lt;user id&gt;</td>
<td>Identify the user whose access is being removed.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

**Note:** You use either <accountid> or (<accountname> and <targetname>) to identify the account.

A.6.7 retrievegrantees Command

Use the retrievegrantees command to get information about the grantees on a privileged account.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x retrievegrantees <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-accountid &lt;account id&gt;]</td>
<td>Optional. Identify from which account the grantees are to be retrieved.</td>
</tr>
</tbody>
</table>

**Note:** You use either <accountid> or (<accountname> and <targetname>) to identify the account.
A.6.8 retrievegroup Command

Use the retrievegroup command to get information about a group.

Command Syntax:

[-url <url>] -u <username> [-p <password>] [-debug] -x retrievegroup <options>

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-groupname &lt;group name&gt;</td>
<td>Provide the name of the group to retrieve.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

A.6.9 retrieveuser Command

Use the retrieveuser command to get information about a user.

Command Syntax:

[-url <url>] -u <username> [-p <password>] [-debug] -x retrieveuser <options>

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-userid &lt;user id&gt;</td>
<td>Identify the user to be retrieved.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

A.6.10 searchgroup Command

Use the searchgroup command to search for a group.
### A.6.11 `searchuser` Command

Use the `searchuser` command to search for a user.

**Note:** You must be an administrator with the `User Manager Admin` Role to successfully run this command.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x searchuser <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-userid &lt;user id&gt;]</td>
<td>Optional. Search for the user by the user ID.</td>
</tr>
<tr>
<td>[-firstname &lt;first name&gt;]</td>
<td>Optional. Provide the user's first name.</td>
</tr>
<tr>
<td>[-lastname &lt;last name&gt;]</td>
<td>Optional. Provide the user's last name.</td>
</tr>
<tr>
<td>[-accountname &lt;account name&gt;]</td>
<td>Optional. Provide the name of the account to search.</td>
</tr>
<tr>
<td>[-targetname &lt;target name&gt;]</td>
<td>Optional. Provide the name of the target to search.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

### A.7 Working with Plug-Ins

The following sections describe the commands that you can use to configure and deploy Java plug-ins for Oracle Privileged Account Manager.

- Section A.7.1, "addplugin Command"
- Section A.7.2, "addplugincustomattr Command"
- Section A.7.3, "removeplugincustomattr Command"
- Section A.7.4, "retrieveplugin Command"
- Section A.7.5, "searchplugin Command"
A.7.1 addplugin Command

Use the addplugin command to add a plug-in to a resource.

**Note:** You must be an administrator with the Application Configurator Admin Role to execute this command.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x addplugin
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-pluginname &lt;plugin name&gt;</td>
<td>Specify a name for the new plug-in.</td>
</tr>
<tr>
<td>-resource &lt;target/account/server&gt;</td>
<td>Identify the resource on which the plug-in will perform.</td>
</tr>
<tr>
<td>-operation &lt;plugin operation&gt;</td>
<td>Specify the operation the plug-in will perform.</td>
</tr>
<tr>
<td>-timing &lt;pre/post&gt;</td>
<td>Specify the plug-in timing.</td>
</tr>
<tr>
<td></td>
<td><strong>Pre-plug-in:</strong> Performed before the Oracle Privileged Account Manager operation.</td>
</tr>
<tr>
<td></td>
<td><strong>Post-plug-in:</strong> Performed after the Oracle Privileged Account Manager operation.</td>
</tr>
<tr>
<td>-order &lt;plugin order&gt;</td>
<td>Specify the order in which the plug-in should be queued for execution. Where the smaller the number, the closer to the top (or beginning) of the queue. (Minimum value is 1.)</td>
</tr>
<tr>
<td>-classname &lt;plugin class name&gt;</td>
<td>Specify the plug-in’s class name.</td>
</tr>
<tr>
<td>-classpath &lt;plugin class path&gt;</td>
<td>Specify the path to the plug-in’s jar file.</td>
</tr>
<tr>
<td>[-status] &lt;active/disabled&gt;</td>
<td>Specify the plug-in execution status. Where</td>
</tr>
<tr>
<td></td>
<td><strong>active:</strong> Allows the plug-in to execute at runtime.</td>
</tr>
<tr>
<td></td>
<td><strong>disabled:</strong> Does not allow the plug-in to execute at runtime.</td>
</tr>
</tbody>
</table>

**Note:** Oracle Privileged Account Manager uses some of these options as filtering rules to decide whether to execute the plug-in. In addition, Oracle Privileged Account Manager evaluates these filtering rules in a certain order to decide one rule’s precedence over another.

For more information, about the filtering rules and creating plug-in configurations, refer to Section 11.2.8, "Filtering Rules." and Section 11.3, "Creating a Plug-In Configuration" respectively.
**Working with Plug-Ins**

**A.7.2 addplugincustomattr Command**

Use the `addplugincustomattr` command to add a plug-in custom attribute.

**Note:** You must be an administrator with the Security Administrator Admin Role or the Application Configurator Admin Role to execute this command.

**Command Syntax:**

```
[-url <url>] -u <username> [-p <password>] [-debug] -x addplugincustomattr
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-pluginname &lt;plugin name&gt;</code></td>
<td>Identify the plug-in on which to add the custom attribute.</td>
</tr>
<tr>
<td><code>-pluginattrname &lt;plugin custom attribute name&gt;</code></td>
<td>Specify the name of the custom attribute.</td>
</tr>
<tr>
<td><code>-pluginattrvalue &lt;plugin custom attribute value&gt;</code></td>
<td>Specify the value of the custom attribute. [Multi-Valued]</td>
</tr>
</tbody>
</table>
A.7.3 `removeplugincustomattr` Command

Use the `removeplugincustomattr` command to remove a custom attribute from a plug-in.

**Note:** You must be an administrator with the Security Administrator Admin Role or the Application Configurator Admin Role to execute this command.

**Command Syntax:**

```
[-url <url>] -u <username> [-p <password>] [-debug] -x removeplugincustomattr
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-pluginname</code> &lt;plugin name&gt;</td>
<td>Identify the plug-in from which the custom attribute should be removed.</td>
</tr>
<tr>
<td><code>-pluginattrname</code> &lt;plugin custom attribute name&gt;</td>
<td>Specify the name of the custom attribute to be removed.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

---

A.7.4 `retrieveplugin` Command

Use the `retrieveplugin` command to get information about a plug-in. This information does not include passwords.

**Note:** You must be an administrator with the Security Administrator Admin Role or the Application Configurator Admin Role to execute this command.

**Command Syntax:**

```
[-url <url>] -u <username> [-p <password>] [-debug] -x retrieveplugin <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-pluginname</code> &lt;plugin name&gt;</td>
<td>Identify the plug-in to retrieve.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>
**A.7.5 searchplugin Command**

Use the `searchplugin` command to search for a plug-in.

**Note:** You must be an administrator with the Security Administrator Admin Role, the User Manager Admin Role, or the Application Configurator Admin Role to execute this command.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x searchplugin <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-pluginname] &lt;plugin name&gt;</td>
<td>Optional. Identify the plug-in to search for.</td>
</tr>
<tr>
<td>[-description] &lt;plugin description&gt;</td>
<td>Optional. Identify the plug-in description to search for.</td>
</tr>
<tr>
<td>[-pluginstatus] &lt;active/disabled&gt;</td>
<td>Optional. Identify the plug-in status to search for.</td>
</tr>
<tr>
<td>[-resource] &lt;target/account/server&gt;</td>
<td>Optional. Identify the plug-in resource to search for.</td>
</tr>
<tr>
<td>[-operation] &lt;plugin operation&gt;</td>
<td>Optional. Identify the plug-in operation to search for.</td>
</tr>
<tr>
<td>[-timing] &lt;pre/post&gt;</td>
<td>Optional. Identify the plug-in timing to search for.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

You can use any combination of `-pluginname`, `-description`, `-pluginstatus`, `-resource`, `-operation` or `-timing` to identify the plug-in. If you do not provide any of these options, then the search returns all plug-ins.

For example, the following search returns all plug-ins:

https://<host name>:<port>/opam/plugin/search?

Whereas, the following search returns all plug-ins whose status is active and timing is pre:

https://<host name>:<port>/opam/plugin/search?pluginstatus=active&timing=pre

**A.7.6 modifyplugin Command**

Use the `modifyplugin` command to modify a plug-in.

**Note:** You must be an administrator with the Security Administrator Admin Role or the Application Configurator Admin Role to execute this command.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x modifyplugin <options>
```

The following table describes the options you can use with this command:

...
You can modify plug-in with the following property names:

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pluginStatus &lt;active/disabled&gt;</td>
<td>Modify the plug-in's status.</td>
</tr>
<tr>
<td>pluginDescription</td>
<td>Modify the plug-in description.</td>
</tr>
<tr>
<td>pluginResource &lt;target/account/server&gt;</td>
<td>Modify the resource on which the plug-in will perform.</td>
</tr>
<tr>
<td>pluginOperation</td>
<td>Modify the operation the plug-in performs.</td>
</tr>
<tr>
<td>pluginTiming &lt;pre/post&gt;</td>
<td>Modify the plug-in timing.</td>
</tr>
<tr>
<td>pluginOrder</td>
<td>Modify the plug-in order.</td>
</tr>
<tr>
<td>pluginClassName</td>
<td>Modify the plug-in's class name.</td>
</tr>
<tr>
<td>pluginClassPath [multi-valued]</td>
<td>Modify the plug-in's class path.</td>
</tr>
<tr>
<td>pluginEnableUser [multi-valued]</td>
<td>Modify the plug-in's enabled user list.</td>
</tr>
<tr>
<td>pluginDisableUser [multi-valued]</td>
<td>Modify the plug-in's disabled user list.</td>
</tr>
<tr>
<td>pluginEnableGroup [multi-valued]</td>
<td>Modify the plug-in's enabled group list.</td>
</tr>
<tr>
<td>pluginDisableGroup [multi-valued]</td>
<td>Modify the plug-in's disabled group list.</td>
</tr>
<tr>
<td>pluginVersion</td>
<td>Modify the plug-in's version.</td>
</tr>
<tr>
<td>pluginTimeout</td>
<td>Modify the plug-in's timeout.</td>
</tr>
</tbody>
</table>

**Note:** These property names are case-sensitive.

**A.7.7 removeplugin Command**

Use the removeplugin command to remove a plug-in.

```
Command Syntax:
[-url <url>] -u <username> [-p <password>] [-debug] -x removeplugin <options>
```

The following table describes the options you can use with this command:
A.8 Exporting and Importing Data

The following sections contain information about the commands that you use when exporting and importing Oracle Privileged Account Manager data.

- Section A.8.1, "export Command"
- Section A.8.2, "filedecryption Command"
- Section A.8.3, "import Command"

A.8.1 export Command

Use the export command to export data stored in Oracle Privileged Account Manager, such as targets and accounts, to XML format. This option and the "import Command" on page A-43 are useful for performing the following operations:

- Bulk operations, such as querying or loading large volumes of data
- Back-up and recovery operations, such as periodically backing up Oracle Privileged Account Manager data to XML
- Migration operations, such as exporting data from one Oracle Privileged Account Manager instance and importing it to another instance

**Note:** You must be an administrator with the Security Administrator Admin Role to use these commands.

The export command exports all Oracle Privileged Account Manager data; including targets, accounts, policies, and grants.

**Note:** Exporting accounts also exports the passwords for those accounts. For added security, you can export the passwords in an encrypted format by using the -encpassword and -enckeylen options. Be sure to note the encryption password and encryption key length because you must provide that same password for decryption during the import operation.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x export <options>
```

The following table describes the options you can use with the export command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-pluginname &lt;plugin name&gt;</td>
<td>Identify the plug-in to be removed.</td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
<tr>
<td>-f &lt;export file&gt;</td>
<td>Specify an export file name.</td>
</tr>
<tr>
<td>[-encpassword &lt;encryption password&gt;]</td>
<td>Optional. Specify a password to use when encrypting the account passwords to the exported file.</td>
</tr>
</tbody>
</table>
The XML schema for an export file is located in the following file:

```
ORACLE_HOME/opam/jlib/OPAMBulkTool.xsd
```

The following example shows some sample XML definitions of Oracle Privileged Account Manager elements.

### Example A–3 Sample XML Definition of Oracle Privileged Account Manager Elements

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<OPAMData xmlns="http://www.example.org/OPAMBulkTool"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.example.org/OPAMBulkTool OPAMBulkTool.xsd">
  <usagepolicy>
    <name value="Accounting Usage Policy"/>
    <status value="active"/>
    <description value="My Usage Policy"/>
    <globaldefault value="n"/>
    <dateorduration value="duration"/>
    <expiremin value="30"/>
    <expiredate value="08/08/2088"/>
    <expiretime value="11:30am"/>
    <timezone value="America/Los_Angeles"/>
    <usagedays>
      <day fromtime="12:00am" totime="12:00am" value="monday"/>
      <day fromtime="12:00am" totime="12:00am" value="tuesday"/>
      <day fromtime="12:00am" totime="12:00am" value="wednesday"/>
      <day fromtime="12:00am" totime="12:00am" value="thursday"/>
      <day fromtime="12:00am" totime="12:00am" value="friday"/>
      <day fromtime="12:00am" totime="12:00am" value="saturday"/>
      <day fromtime="12:00am" totime="12:00am" value="sunday"/>
    </usagedays>
  </usagepolicy>
  <passwordpolicy>
    <name value="Accounting Password Policy"/>
    <status value="active"/>
    <description value=""/>
    <globaldefault value="n"/>
    <changepassevery value="30-days"/>
    <changepasscheckout value="y"/>
    <changepasscheckin value="y"/>
    <passwordlength max="20" min="8"/>
    <minalphabets value="1"/>
  </passwordpolicy>
</OPAMData>
```
A.8.2 filedecryption Command

Use the filedecryption command to decrypt an encrypted Oracle Privileged Account Manager configuration file.

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x filedecryption
-f <encrypted file> -df <destination file> [-encpassword <decryption password>]
```

**Note:** This operation does not require any server connectivity when the `-offline true` option is provided.

The following table describes the options you can use with this command:
Exporting and Importing Data

A-43

For example, use the following command if you do not have server connectivity:

```
sh opam.sh -x filedecryption -f <encrypted file> -df <destination file> -offline true
```

### A.8.3 import Command

Use the `import` command to import data to Oracle Privileged Account Manager from an XML file. This option and the "export Command" on page A-40 are useful for performing the following operations:

- Bulk operations, such as querying or loading large volumes of data
- Back-up and recovery operations, such as periodically backing up Oracle Privileged Account Manager data to XML
- Migration operations, such as exporting data from one Oracle Privileged Account Manager instance and importing it to another instance

**Note:** You must be an administrator with both the `Security Administrator` Admin Role and the `User Manager` Admin Role to use these commands.

If the account status is checked-in, users do not have to provide status when importing data to Oracle Privileged Account Manager.

You can create an import XML file from previously exported data or you can manually create the file. If you previously exported the XML file with an encryption password, then you must provide the same password for decryption during import.

In addition to object creation, you can also use the `import` command to update and delete objects. Refer to reference for more information.
Exporting and Importing Data

Command Syntax:

```
[-url <url>] -u <username> [-p <password>] [-debug] -x import <options>
```

The following table describes the options you can use with this command:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-i &lt;import file&gt;</td>
<td>Specify an import file name.</td>
</tr>
<tr>
<td>[-encpassword &lt;encryption password&gt;]</td>
<td>Optional. Specify a password to use when decrypting account passwords from the exported file.</td>
</tr>
<tr>
<td>[-enckeylen &lt;key length for password encryption&gt;]</td>
<td>Optional. Specify the minimum key length for an encryption/decryption password. (Defaults to 128 bits)</td>
</tr>
<tr>
<td>[-force &lt;true/false&gt;]</td>
<td>Optional. Enables or disables the requirement for connection validation.</td>
</tr>
<tr>
<td>■ true: Skips connection validation.</td>
<td></td>
</tr>
<tr>
<td>■ false (default): Enforces connection validation.</td>
<td></td>
</tr>
<tr>
<td>[-log &lt;log file location&gt;]</td>
<td>Optional. Specify a file name and location for the log file. (Defaults to opamlog_&lt;timestamp&gt;.txt)</td>
</tr>
<tr>
<td>[-noencrypt &lt;true/false&gt;]</td>
<td>Optional. Specify whether to decrypt the imported file. (Defaults to false)</td>
</tr>
<tr>
<td>■ true: Skip the encryption password. The system will import the file in clear text.</td>
<td></td>
</tr>
<tr>
<td>■ false: Use the encryption password to decrypt the import file, and then load the decrypted data into the system.</td>
<td></td>
</tr>
<tr>
<td>[-help]</td>
<td>Optional. Displays usage options for this command.</td>
</tr>
</tbody>
</table>

The XML schema for an import file is located in the following file:

```
ORACLE_HOME/opam/jlib/OPAMBulkTool.xsd
```

The following examples show some sample XML definitions of Oracle Privileged Account Manager elements.

**Example A–4 Data Creation**

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
  <usagepolicy>
    <name value="Accounting Usage Policy"/>
    <status value="active"/>
    <description value="My Usage Policy"/>
    <globaldefault value="n"/>
    <dateordination value="duration"/>
    <expiremin value="30"/>
    <expiredate value="08/08/2088"/>
    <expiretime value="11:30am"/>
    <timezone value="America/Los_Angeles"/>
    <usagedays>
      <day fromtime="12:0am" totime="12:0am" value="monday"/>
      <day fromtime="12:0am" totime="12:0am" value="tuesday"/>
      <day fromtime="12:0am" totime="12:0am" value="wednesday"/>
      <day fromtime="12:0am" totime="12:0am" value="thursday"/>
      <day fromtime="12:0am" totime="12:0am" value="friday"/>
    </usagedays>
  </usagepolicy>
</OPAMData>
```
Example A–5  Data Modification: Modify An Account Password Policy

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<OPAMData xmlns="http://www.example.org/OPAMBulkTool"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.example.org/OPAMBulkTool OPAMBulkTool.xsd">
  <day fromtime="12:00am" totime="12:00am" value="saturday"/>
  <day fromtime="12:00am" totime="12:00am" value="sunday"/>
</usagepolicy>
<passwordpolicy>
  <name value="Accounting Password Policy"/>
  <status value="active"/>
  <description value=""/>
  <globaldefault value="n"/>
  <change passevery value="30-days"/>
  <changepasscheck out value="y"/>
  <changepasscheck in value="y"/>
  <passwordlength max="20" min="8"/>
  <minalphabets value="1"/>
  <minnumeric value="1"/>
  <minalphanumeric value="2"/>
  <specialchars max="5" min="1"/>
  <repeatedchars max="1" min="0"/>
  <minuniquechars value="1"/>
  <minuppercasechars value="1"/>
  <minlowercasechars value="1"/>
  <startwithchar value="n"/>
  <accountnamenamespace value="n"/>
  <passwordhistorydays value="30"/>
</passwordpolicy>
<target>
  <type name="database"/>
  <name value="AccountsDB"/>
  <attributes>
    <attributeName name="domain" value="Accounting"/>
    <attributeName name="host" value="localhost"/>
    <attributeName name="jdbcUrl" value="jdbc:oracle:thin:@dbhost:1521:orcl"/>
    <attributeName name="loginUser" value="system"/>
    <attributeName name="loginPassword" value="welcome1"/>
    <attributeName name="dbType" value="Oracle"/>
    <attributeName name="description" value="Accounting Database"/>
    <attributeName name="organization" value="Accounting"/>
    <attributeName name="connectionProperties" value=""/>
  </attributes>
</target>
<account>
  <name value="ACCT_DBA"/>
  <target name="AccountsDB"/>
  <description value="Accounting Database"/>
  <passwordpolicy name="Accounting Password Policy"/>
  <grantee>
    <user name="johndoe usagepolicy="Accounting Usage Policy"/>
    <user name="janedoe usagepolicy="Default Usage Policy"/>
  </grantee>
  <shared value="false"/>
</account>
</OPAMData>
```
Example A–6  Data Modification: Modify A Password Policy

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<OPAMData xmlns="http://www.example.org/OPAMBulkTool"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.example.org/OPAMBulkTool OPAMBulkTool.xsd">
<passwordpolicy operation="modify">
  <name value="test policy"/>
  <status value="active"/>
  <description value="test"/>
  <globaldefault value="n"/>
  <changepassevery value="45-hours"/>
  <changepasscheckout value="n"/>
  <changepasscheckin value="n"/>
  <passwordlength max="20" min="5"/>
  <minalphabets value="0"/>
  <minnumeric value="0"/>
  <minalphanumeric value="0"/>
  <specialchars max="5" min="0"/>
  <repeatedchars max="10" min="0"/>
  <minuniquechars value="0"/>
  <minlowercasechars value="0"/>
  <minuppercasechars value="0"/>
  <startwithchar value="y"/>
  <requiredchars value="a,b,c,d,e"/>
  <allowedchars value="a,b,c,d,e,f,g,h"/>
  <disallowedchars value="z,-,x"/>
  <accountnameaspass value="y"/>
</passwordpolicy>
</OPAMData>

Example A–7  Data Deletion: Delete a Target

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<OPAMData xmlns="http://www.example.org/OPAMBulkTool"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.example.org/OPAMBulkTool OPAMBulkTool.xsd">
<target operation="delete">
  <type name="lockbox"/>
  <name value="lockbox_target1"/>
</target>
</OPAMData>

Example A–8  Data Deletion: Delete an Account

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<OPAMData xmlns="http://www.example.org/OPAMBulkTool"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.example.org/OPAMBulkTool OPAMBulkTool.xsd">
<account operation="delete"/>
<name value="account3"/>
<target name="lockbox_target1"/>
</account>
<account operation="delete">
  <name value="account4"/>
  <target name="lockbox_target1"/>
</account>
</OPAMData>
B

Working with Oracle Privileged Account Manager's RESTful Interface

This appendix describes Oracle Privileged Account Manager's RESTful interface, including the specific APIs that are exposed through this interface.

This appendix includes the following sections:

- Section B.1, "Overview"
- Section B.2, "Server State Resource"
- Section B.3, "Configuration Resource"
- Section B.4, "Policy Resource"
- Section B.5, "Target Resource"
- Section B.6, "Account Resource"
- Section B.7, "UI Resource"
- Section B.8, "User Resource"
- Section B.9, "Group Resource"
- Section B.10, "Plug-In Resource"

B.1 Overview

While Oracle Privileged Account Manager can be consumed through several client interfaces, its fundamental access mechanism or layer is encapsulated in its RESTful interfaces.

Note: For information about using Oracle Privileged Account Manager's web-based Console or command line tool to perform tasks described in this appendix, refer to Chapter 4, "Starting and Using the Oracle Privileged Account Manager Console" or Appendix A, "Working with the Command Line Tool."

All interactions with Oracle Privileged Account Manager's server that are being used by external parties, such as a non-Oracle Privileged Account Manager server, are exposed through RESTful interfaces. All externally visible Oracle Privileged Account Manager resources are modeled by URIs, while standard HTTP operations are mapped to relevant Oracle Privileged Account Manager operations on those resources.
Note: The information provided in this appendix is essentially the same whether you are using Oracle Privileged Account Manager on WebLogic or on IBM WebSphere; however, there are a few minor differences.

For more information, refer to "Differences When Using the Oracle Privileged Account Manager Command Line Tool and REST Interfaces on IBM WebSphere" in the Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management.

B.2 Server State Resource

This section describes the Get Server State API.

B.2.1 Get Server State

Use this API to retrieve information about the status of a server.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of the Server State Resource</td>
</tr>
</tbody>
</table>

Example B–1 Example JSON Output of Server Status

```
{
    "RequestorGroups" : [
        "OPAM_APPLICATION_CONFIGURATOR",
        "OPAM_SECURITY_ADMIN",
        "OPAM_USER_MANAGER",
        "OPAM_SECURITY_AUDITOR"
    ],
    "ServerState" : {
        "Status" : "Oracle Privileged Account Manager Server is up!",
        "StatusCode" : 0
    },
    "Requestor" : "master_user",
    "version" : "11.1.2.2.0"
}
```

Where:

- **RequestorGroups** are groups assigned to the user who is making the request.
- **Requestor** is the user who is making the request.
- **StatusCode** indicates whether the server is working properly.
  - Returns a zero (0) if the server is working properly.
  - Returns a non-zero integral value if the server has encountered some issue.
- **Status** is an informative message about the state of the server.
- **version** is the Oracle Privileged Account Manager version.
B.3 Configuration Resource

This section describes the following configuration resource APIs:

- Global Configuration Resource
- Oracle Privileged Session Manager Configuration Resource

B.3.1 Global Configuration Resource

The APIs described in this section include:

- Get Configuration Resource
- Update Configuration Resource

B.3.1.1 Get Configuration Resource

Use this API to retrieve a configuration object for Oracle Privileged Account Manager.

---

**Note:** You must be an administrator with the User Manager Admin Role, the Security Administrator Admin Role, or the Application Configurator Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/config/configid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>200 and JSON representation of a config object</td>
</tr>
</tbody>
</table>

**Sample URI**

https://opam_server_host:opam_ssl_port/opam/config/globalconfig

**Example B–2 Sample JSON Representation of a config Object**

```json
{
    "config":{
        "configUID":"globalconfig",
        "configType":"config_globalconfig",
        "tdemode":[
            "true"
        ],
        "policyenforcerinterval":{
            "3600"
        },
        "passwordcyclerinterval":{
            "3600"
        }
    }
}
```

Where:

- **configUID** is a unique identifier for the config object.
- **configType** is the type of config object.
- **policyenforcerinterval** is the interval (in seconds) in which Oracle Privileged Account Manager checks accounts and then automatically checks-in the accounts that have exceeded the expiration time defined in the Usage Policy.

- **passwordcyclerinterval** is the interval (in seconds) in which Oracle Privileged Account Manager checks and then resets the password for any accounts that have exceeded the maximum password age defined in the Password Policy.

- **tdemode** is a flag to request that Oracle Privileged Account Manager use TDE or non-TDE mode.

### B.3.1.2 Update Configuration Resource

Use this API to modify a configuration object for Oracle Privileged Account Manager.

**Note:** You must be an administrator with the *Application Configurator Admin Role* to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/config/configid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td>JSON representation of Modification</td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>

**Example B–3** Example JSON Output of Modification

```json
{
  "modifications": [
    {
      "modification": {
        "tdemode": [
          "false"
        ]
      }
    }
  ]
}
```

Where:

- **modifications** are an array of modification JSON objects.
- **modification** is a JSON object representing the modification of a single configuration object.
- **tdemode** is a flag to request that Oracle Privileged Account Manager use TDE or non-TDE mode.

### B.3.2 Oracle Privileged Session Manager Configuration Resource

The APIs described in this section include:

- Get Configuration Resource
- Update Configuration Resource
B.3.2.1 Get Configuration Resource

Use this API to get a configuration object for Oracle Privileged Session Manager.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/config/sessionmgrconfig</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of a Session Manager config object</td>
</tr>
</tbody>
</table>

**Note:**
- You must be an administrator with the User Manager, the Security Administrator, or the Application Configurator Admin Role to use this API.
- You cannot run two instances of Oracle Privileged Session Manager on the same machine.

**Example B–4 Sample JSON Representation of Session Manager Config**

```json
{"config": {
  "updateinterval": 60,
  "pub-key": "ssh-dss AAAAB3NzaC1kc3MAAAAACCAA6279V8ozaK/s6x9ihSyIi3EziPtp0vN9dgeFq7VkP5vtj10HbYDk4/\n  \n  /MbcIlx9Ko/gDury2Y246+3URQE2zUN1FVZ146d+sm1VmhHq5M8zGZPjneFr3AFRE7RE0V/\n  \n  /D0hjncw84aLzSBU3p7Thb+b5pV7LnzAAzCID1cV4EB6T4U5u1QfD6x0A6wAAAAdEQJK1T6oiwzh+63x1i34ivbMKc
  Pqk7oi3FChKZS+NShnt1Rvd5cIDt8UY+WcwpWY\n  
  /hfafrKXxhC9OHFXXAI10R0WF/\n  
  /1YRa8cTFUA9AO0u8j701aqpm34P1ot1s8AHCkUjuy1Y\n  \n  /Vg8eJKHaYB5Uomd47skroVxc9K7bDwvAAAB2hcvPMcJARKtWFxtT8UkwwXowd3sae2dRmEUsir2bM108Hm1CV952n
  V3aeAFY+8dnQ9HFTF1M2t9cfmMnWX1LiAciw+Ex\n  
  /QSV7M5u9RBvCo+ixATSjypr6UmzmeMR6znnLYPduDmELttFx8kYt3RgpsdnfocyCmJK3Q==","prv-key": "MIIBugIBAAKBgQDetu/VfKM2iv7Onfy0usiJXyLmdM4j9T9RujfXYhau1ZD+b7Y9
  \n  /ntgH2A50zg2G37Mf3pqPqg7q8tmGLkyv7v7\n  
  /jptY1EUBNs1C7DrVWd+OnfrjijP2z3\n  /n6j0fMmT543ha9wBUR00RNPF07fw\n  
  /F442mNPOG18gqN3XE4W/\n  
  /m0qVey5wIWAIGbMIUIUK\n  
  /g0HppThm4jbjB80V0HE4BFa2QAdGBCSCPUp+gos4fut14QPN+Ir2zcCh6p06I\n  \n  /ncQ0SmvUvo/27z02db3exCA77FFav1nMGFK/\n  
  /4X2nOS8eVqVhXhXyysJN8EdFh55WEQX
  
  /t1GvQlDhi+1zpaqgs2t+92A8uLuGhwpF32h1YPH1ZBmBOVNaJneQ07JK6kPa
  
  /nP5u2v7GyZbeHcvPMcJARKtWFxtT8UkwwXowd3sae2dRmEUsir2bM108Hm1CV
  
  /n952nV3aeAFY+8dnQ9HFTF1M2t9cfmMnWX1LiAciw+Ex\n  
  /QSV7M5u9RBvCo+iX
  
  /nATSjyprK6UmzmeMR6znnLYPduDmELttFx8kYt3RgpsdnfocyCmJK3QIUIaexDoyJ1
  
  /nS6M10VqzYi1JwrEawl=", 
  "SSH": {
    "opamListenPort": 1222,
    "sessioncheckoutinstructions": "ssh -p <port> <opamuser>::<targetname>::<accountname>@<sessionmgrhost> \n Use opam password on password prompt"
  
  }
}
```

Where:
- **configUID** is a unique identifier for the config object.
- **configType** is the type of config object.
■ **updateinterval** is the interval (in seconds) in which the Oracle Privileged Session Manager server checks all of the checked-out sessions and updates their transcripts.

■ **opamserv urls** is an array of Oracle Privileged Account Manager server URLs to which Oracle Privileged Session Manager can connect.

■ **pub-key** is the Oracle Privileged Session Manager server’s public key.

■ **maxrecordsize** is the maximum recording size that is allowed per session (in KB). When this quota is reached, the session is automatically terminated.

■ **prv-key** is the Oracle Privileged Session Manager server’s private key.

Protocol-specific attributes include:

■ **opamListenPort** is the listener port for the protocol.

■ **sessionchkoutinstructions** is the session checkout instructions.

### B.3.2.2 Update Configuration Resource

Use this API to update a configuration object for Oracle Privileged Session Manager.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://[opam_server_host]:[opam_ssl_port]/opam/config/sessionmgrconfig</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td>JSON representation of Modification</td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>

---

**Note:** You must be an administrator with the Application Configurator Admin Role to use this API.

---

**Example B–5  Sample JSON Modification**

```json
{
"modifications": [
{
"modification": {
"updateinterval": 300
}
},
{
"modification": {
"opamserv urls": [
"https://localhost:7002/opam"
]
}
},
{
"modification": {
"SSH": {
"opamListenPort": 1222
}
}
},
{
"modification": {
```

B-6  Administrator’s Guide for Oracle Privileged Account Manager
"SSH": {
  "sessionchkoutinstructions": "ssh -p <port> <opamuser>:@<targetname>:<accountname>@<sessionmgrhost> \n Use opam password on password prompt"
}

**Note:** You can update all of these attributes, except
- **configUID** is a unique identifier for the config object.
- **configType** is the type of config object.

For the other attribute definitions, refer to Section B.3.2.1, "Get Configuration Resource."

### B.4 Policy Resource

This section describes the APIs you use when working with Oracle Privileged Account Manager policies.

The APIs described in this section include:
- **Search for Policies**
- **Get Default Policies**
- **Password Policy Resource**
- **Usage Policy Resource**

#### B.4.1 Search for Policies

Use this API to search for policies. This API is a search, using one or more of the following parameters:
- **policystatus**
- **policyname**

All of the parameters are optional.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/policy/search?param1=val1&amp;param2=val2</th>
</tr>
</thead>
</table>
| Method | GET
| Content-Type | |
| Body | |
| Returns on Success | Status code 200 and JSON representation of policies |

**Example B–6  Sample JSON Representation of Policies**

```json
{
  "usagepolicies": [
    {
      "policyname": "Default Usage Policy",
      "policyid": "usagepolicy1",
      "policystatus": "active",
    }
  ]
}
```
B.4.2 Get Default Policies

Use this API to get the Default Usage Policy and Default Password Policy.

Example B–7  Sample JSON Representation of Policies

```
{
  "usagepolicies": [
    {
      "policyname": "Default Usage Policy",
      "policyid": "usagepolicy1",
      "policystatus": "active"
    }
  ],
  "passwordpolicies": [
    {
      "policyname": "Default Password Policy",
      "policyid": "passwordpolicy2",
      "policystatus": "active"
    }
  ]
}
```

Where:

- **usagepolicies** is an array of Usage Policies.
- **passwordpolicies** is an array of Password Policies.
- **policyname** is the policy name.
- **policyid** is the policy’s unique identifier.
- **policystatus** is the policy status, where acceptable values are active or disabled.
■ **policyid** is the policy’s unique identifier.

■ **policystatus** is the policy status, where acceptable values are **active** or **disabled**.

This attribute only returns the default policies, Default Usage Policy and Default Password Policy.

### B.4.3 Password Policy Resource

The APIs described in this section include:

■ Retrieve a Password Policy

■ Update a Password Policy

■ Create a Password Policy

■ Get Accounts for Password Policy

■ Delete a Password Policy

#### B.4.3.1 Retrieve a Password Policy

Use this API to retrieve a Password Policy.

**Note:** You must be an administrator with the **User Manager Admin Role** or the **Security Administrator Admin Role** to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/passwordpolicy/{policyid}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of Password Policy</td>
</tr>
</tbody>
</table>

**Example B–8 Sample JSON Representation of Password Policy**

```json
{
    "passwordpolicy":{
        "policyid":"passwordpolicy2",
        "policystatus":"active",
        "policyname":"Default Password Policy",
        "description":"Default Password Policy",
        "globaldefault":"y",
        "passwordchangedurationunit":"days",
        "passwordchangedurationvalue":30,
        "passwordhistorydays":30,
        "changeoncheckin":"y",
        "changeoncheckout":"y",
        "passwordcharsmin":8,
        "passwordcharsmax":8,
        "passwordalphabeticmin":1,
        "passwordnumericmin":1,
        "passwordalphanumericmin":2,
        "passworduniquemin":1,
        "passworduppercasemin":1,
        "passwordlowercasemin":1,
        "passwordspecialmin":0,
        "passwordspecialmax":0,
```
"passwordrepeatedmin":0,
"passwordrepeatedmax":1,
"startingchar":"n",
"isaccountnameallowed":"n",
"requiredchars":[
  "a",
  "h",
  "j"
],
"allowedchars":[
  "b",
  "t",
  "y",
  "p",
  "u",
  "r",
  "o",
  "k",
  "j",
  "2",
  "s",
  "n",
  "a",
  "h",
  "j"
],
"disallowedchars":[
  "7",
  "8",
  "l"
],
}
}

Where:

- **passwordpolicy** is a passwordpolicy JSON object.
- **policyid** is the policy's unique identifier.
- **policystatus** is the policy's status, where acceptable values are **active** or **disabled**.
- **policyname** is the policy name.
- **description** is a description of the policy.
- **globaldefault** indicates whether the policy is a global default or not.
- **passwordchangedurationunit** and **passwordchangedurationvalue** determine the interval after which the account password must be changed. Where **passwordchangedurationunit** can have the values: **days**, **hours**, or **minutes**.
- **passwordhistorydays** indicates how many days to keep the password history.
- **changeoncheckin** indicates whether to change the password on check-in. (Valid values are **y** and **n**.)
- **changeoncheckout** indicates whether to change the password on checkout. (Valid values are **y** and **n**.)
- **startingchar** indicates the character with which the password should begin.
- **isaccountnameallowed** indicates whether the password can be the same as the account name.
- **requiredchars**, **allowedchars**, **disallowedchars** are characters that are required, allowed, and disallowed respectively.
- **passwordcharsmin** is the minimum number of characters required in the password.
- **passwordcharsmax** is the maximum number of characters allowed in the password.
- **passwordalphabeticmin** is the minimum number of alphabetic characters required in the password.
- **passwordnumericmin** is the minimum number of numeric characters required in the password.
- **passwordalphanumericmin** is the minimum number of alphanumeric characters required in the password.
- **passworduniquemin** is the minimum number of unique characters required in the password.
- **passworduppercasemin** is the minimum number of uppercase characters required in the password.
- **passwordlowercasemin** is the minimum number of lowercase characters required in the password.
- **passwordspecialmin** is the minimum number of special characters required in the password.
- **passwordspecialmax** is the maximum number of special characters allowed in the password.
- **passwordrepeatedmin** is the minimum number of repeated characters required in the password.
- **passwordrepeatedmax** is the maximum number of repeated characters allowed in the password.

### B.4.3.2 Update a Password Policy

Use this API to update a Usage Policy. You can update all of the attributes, except **policyid**, and you can update multiple attributes at a time.

**Note:** You must be an administrator with the Security Administrator Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/passwordpolicy/{policyid}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td>JSON representation for Password Policy modification</td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>

---

**Example B–9  Sample JSON Representation of Password Policy Modification**

```json
{
  "modifications":{
  }
  "modification":{
```
Where:

- **modifications** is an array of modification JSON objects.
- **modification** is a JSON object representing a single attribute.

### B.4.3.3 Create a Password Policy

Use this API to create a Password Policy.

**Note:** You must be an administrator with the Security Administrator Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/passwordpolicy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>POST</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td>JSON representation for Password Policy creation</td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 201</td>
</tr>
</tbody>
</table>

**Example B–10  Sample JSON Representation for Password Policy Creation**

```json
{
  "passwordpolicy":{
    "policystatus":"active",
    "policyname":"Default Password Policy",
    "description":"Default Password Policy",
    "passwordchangedurationunit":"days",
    "passwordchangedurationvalue":30,
    "passwordhistorydays":30,
    "changeoncheckin":"y",
    "changeoncheckout":"y",
    "passwordcharsmin":8,
    "passwordcharsmax":8,
    "passwordalphabetmin":1,
    "passwordnumericmin":1,
    "passwordalphanumericmin":2,
    "passworduniquemin":1,
    "passworduppercasemin":1,
    "passwordlowercasemin":1,
    "passwordspecialmin":0,
    "passwordspecialmax":0,
    "passwordrepeatedmin":0,
    "disallowedchars":[
      "4",
      "6"
    ],
    "modification":{
      "passwordalphabetmin":2
    }
  }
}
```
All attributes are optional, except policyname. For attribute definitions refer to Section B.4.3.1, "Retrieve a Password Policy."

B.4.3.4 Get Accounts for Password Policy

Use this API to retrieve a list of accounts for a Password Policy.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/passwordpolicy/{policyid}/accounts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of accounts</td>
</tr>
</tbody>
</table>

Example B–11 Sample JSON Representation of Accounts

```json
{
  "accounts": [
    {
      "account": {
        "accountUID": "5bb2c74e1655487c92ecf
```
For attribute definitions, refer to Section B.5, "Target Resource" and Section B.6, "Account Resource."

**B.4.3.5 Delete a Password Policy**

Use this API to delete a Password Policy.

**Note:** You must be an administrator with the Security Administrator Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/passwordpolicy/{policyid}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>DELETE</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status 200</td>
</tr>
</tbody>
</table>

**B.4.4 Usage Policy Resource**

The APIs described in this section include:

- Retrieve a Usage Policy
- Update a Usage Policy
- Create a Usage Policy
- Get Grants for Usage Policy
- Delete a Usage Policy

**B.4.4.1 Retrieve a Usage Policy**

Use this API to retrieve a Usage Policy.
Example B–12  Sample JSON Representation of Usage Policy

```json
{
  "usagepolicy":{
    "policyid":"usagepolicy1",
    "policystatus":"active",
    "policyname":"Default Usage Policy",
    "description":"Default Usage Policy",
    "globaldefault":"y",
    "dateorduration":"duration",
    "expireddateminutesfromcheckout":7200,
    "expireddate":"08\/08\/2088",
    "expireddatehour":0,
    "expireddateminutes":0,
    "expireddatetimeminpm":"am",
    "timezone":"America\/Los_Angeles",
    "usagedates":[
      {
        "day":"saturday",
        "fromhour":"12",
        "fromminutes":0,
        "frommorning":"am",
        "tohour":"12",
        "tominutes":0,
        "toamorpm":"am"
      },
      {
        "day":"wednesday",
        "fromhour":"12",
        "fromminutes":0,
        "frommorning":"am",
        "tohour":"12",
        "tominutes":0,
        "toamorpm":"am"
      },
      {
        "day":"sunday",
        "fromhour":"12",
        "fromminutes":0,
        "frommorning":"am",
        "tohour":"12",
        "tominutes":0,
        "toamorpm":"am"
      },
      {
        "day":"friday",
        "fromhour":"12",
        "fromminutes":0,
        "frommorning":"am",
        "tohour":"12",
        "tominutes":0,
        "toamorpm":"am"
      }
    ]
  }
}
```

URI | https://opam_server_host:opam_ssl_port/opam/usagepolicy/[policyid]
---|---
Method | GET
Content-Type |  
Body |  
Returns on Success | Status code 200 and JSON representation of Usage Policy
"toamorpm":"am",
},
{
"day":"tuesday",
"fromhour":"12",
"fromminutes":"0",
"fromamorpm":"am",
"tohour":"12",
"tominutes":"0",
"toamorpm":"am"
},
{
"day":"thursday",
"fromhour":"12",
"fromminutes":"0",
"fromamorpm":"am",
"tohour":"12",
"tominutes":"0",
"toamorpm":"am"
},
{
"day":"monday",
"fromhour":"12",
"fromminutes":"0",
"fromamorpm":"am",
"tohour":"12",
"tominutes":"0",
"toamorpm":"am"
}
],
"allowcheckouttype": "all"
}]

Where:
- **usagepolicy** is a usagepolicy JSON object.
- **policyid** is the Usage Policy's unique identifier.
- **policystatus** is set to active or disabled.
- **policyname** is a name of the policy
- **description** is a description of the policy.
- **globaldefault** indicates whether the policy is the global default policy or not.
- **dateorduration** indicates how the expiration time is calculated.
  - If set to date, then **expireddate, expireddatehour, expireddateminutes, and expireddateamorpm** are used.
  - If set to duration, then **expireddateminutesfromcheckout** is used.

Where:
- **expireddate** is the date of expiration. The format is *MM/dd/yyyy*.
- **expireddatehour.hour** are integer values between 0 and 12.
- **expireddateminutes.minutes** are integer values between 0 and 60.
- **expireddateamorpm** is am or pm.
- **expireddateminutesfromcheckout** are minutes from checkout.
■ timezone is a time zone for the Usage Policy.

■ usagedates is an array, where each value represents the check out time for individual days.

■ day is a day of the week, where acceptable values are 
sunday, monady, tuesday, wednesday, thursday, friday, and saturday.

Use the following attributes to indicate a range from and to:

■ fromhour is an integer value between 0 and 12.

■ fromminutes is an integer value between 0 and 60.

■ fromamorpm is am or pm.

■ tohour is an integer value between 0 and 12.

■ tonumtes is an integer value between 0 and 60.

■ toamorpm is am or pm.

■ allowcheckoutype indicates which type of checkout is permitted for the policy.
  - all: Choose this option to allow users to check out passwords and sessions.
  - password (default): Choose this option to allow users to only check out passwords.
  - session: Choose this option to allow users to only check out sessions.

B.4.4.2 Update a Usage Policy

Use this API to update a Usage Policy. You can update all attributes, except policyid, and you can update multiple attributes at a time.

Note: You must be an administrator with the User Manager Admin Role or the Security Administrator Admin Role to use this API.

---

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/usagepolicy/{policyid}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td>JSON representation of Usage Policy modification</td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>

Example B–13 Sample JSON Representation of Usage Policy Modification

```json
{
  "modifications": [
    {
      "modification": {
        "usagedates": [
          {
            "day": "saturday",
            "fromhour": "12",
            "fromminutes": "0",
            "fromamorpm": "am",
            "tohour": "12",
            "tonumtes": "0",
            "toamorpm": "am"
          }
        ]
      }
    }
  ]
}
```
Where:
- modifications are an array of modification JSON objects.
- modification is a JSON object representing a single attribute.

### B.4.4.3 Create a Usage Policy

Use this API to create a Usage Policy.

**Note:** You must be an administrator with the User Manager Admin Role or the Security Administrator Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/usagepolicy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>POST</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td>JSON representation for Usage Policy creation</td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 201</td>
</tr>
</tbody>
</table>

**Example B–14 Sample JSON Representation for Usage Policy Creation**

```json
{
    "usagepolicy":{
        "policystatus":"active",
        "policyname":"Default Usage Policy",
        "description":"Default Usage Policy",
        "dateorduration":"duration",
        "expireddateminutesfromcheckout":7200,
        "expireddate":"08/08/2088",
        "expireddatehour":0,
        "expireddateminutes":0,
        "expireddateamorpm":"am",
        "timezone":"America/Los_Angeles",
        "usagedates":{
            "day":"wednesday",
            "fromhour":"12",
            "fromminutes":"0",
            "fromamorpm":"am",
            "tohour":"12",
            "tominutes":"0",
            "toamorpm":"am"
        }
    }
}
```
'day': 'saturday',
'fromhour': '12',
'fromminutes': '0',
'fromamorpm': 'am',
'tohour': '12',
'tominutes': '0',
'toamorpm': 'am'
},

'day': 'wednesday',
'fromhour': '12',
'fromminutes': '0',
'fromamorpm': 'am',
'tohour': '12',
'tominutes': '0',
'toamorpm': 'am'
},

'day': 'sunday',
'fromhour': '12',
'fromminutes': '0',
'fromamorpm': 'am',
'tohour': '12',
'tominutes': '0',
'toamorpm': 'am'
},

'day': 'friday',
'fromhour': '12',
'fromminutes': '0',
'fromamorpm': 'am',
'tohour': '12',
'tominutes': '0',
'toamorpm': 'am'
},

'day': 'tuesday',
'fromhour': '12',
'fromminutes': '0',
'fromamorpm': 'am',
'tohour': '12',
'tominutes': '0',
'toamorpm': 'am'
},

'day': 'thursday',
'fromhour': '12',
'fromminutes': '0',
'fromamorpm': 'am',
'tohour': '12',
'tominutes': '0',
'toamorpm': 'am'
},

'day': 'monday',
'fromhour': '12',
'fromminutes': '0',
'fromamorpm': 'am',
'tohour': '12',
'tominutes': '0',
'toamorpm': 'am'
For attribute definitions, refer to Section B.4.1, "Retrieve a Usage Policy."

### B.4.4.4 Get Grants for Usage Policy

Use this API to retrieve a list of grants for a Usage Policy.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/usagepolicy/{policyid}/grantees</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of grants</td>
</tr>
</tbody>
</table>

**Example B–15 Sample JSON Representation of Grants**

```json
{
   "grantees": [
      {
         "grantee": {
            "accountUID": "16d245784350469cbe25229a7c45af22",
            "accountName": "oidperson10",
            "targetID": "75a23e9f30ba456b961a1f5d327e67ef",
            "targetName": "ldap1_target",
            "targetDomain": "needtofix",
            "targetType": "ldap",
            "grantee": "CrossDomainConnectors",
            "grantType": "role"
         }
      },
      {
         "grantee": {
            "accountUID": "3a7f105a1e45407284cd887f8774d00",
            "accountName": "openLDAPperson2",
            "targetID": "dd9d7a31b39348c79eb3246f04d40d",
            "targetName": "openldap_2.3_target",
            "targetDomain": "needtofix",
            "targetType": "ldap",
            "grantee": "opamuser2",
            "grantType": "user"
         }
      }
   ]
}
```

For attribute definitions, refer to Section B.5, "Target Resource" and Section B.6, "Account Resource."
B.4.4.5 Delete a Usage Policy

Use this API to delete a Usage Policy.

---

**Note:** You must be an administrator with the User Manager Admin Role or the Security Administrator Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/usagepolicy/[policyid]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>DELETE</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status 200</td>
</tr>
</tbody>
</table>

B.5 Target Resource

The APIs described in this section include:

- Get Target Attributes
- Add a Target
- Verify a Target
- Retrieve a Target
- Update a Target
- Remove a Target
- Search for Targets
- Get Available Accounts
- Retrieve Accounts Registered on a Target
- Get Target Types
- Reset Password
- Show Service Account Password
- Show Service Account Password (Deprecated)
- Show Service Account Password History

B.5.1 Get Target Attributes

Use this API to retrieve a list of the attributes that are associated with all of the target types.

You can use the list of supported target types, along with these attributes, to create the JSON object required to add a target. Refer to Section B.5.2, "Add a Target" for more information.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/target/attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
</tbody>
</table>
Sample URI

https://opam_server_host:opam_ssl_port/opam/target/attributes

Example B–16  JSON Output of Supported Target Types with Attributes

```json
{
    "TargetAttributes": [
        {
            "TargetType": "ldap",
            "DisplayName": "ldap",
            "BasicAttributes": [
                {
                    "name": "targetName",
                    "type": "string",
                    "description": "",
                    "label": "Target Name",
                    "mask": "false",
                    "array": "false",
                    "required": "true"
                },
                {
                    "name": "description",
                    "type": "string",
                    "description": "",
                    "label": "Description",
                    "mask": "false",
                    "array": "false",
                    "required": "false"
                },
                {
                    "name": "organization",
                    "type": "string",
                    "description": "",
                    "label": "Organization",
                    "mask": "false",
                    "array": "false",
                    "required": "false"
                },
                {
                    "name": "domain",
                    "type": "string",
                    "description": "",
                    "label": "Domain",
                    "mask": "false",
                    "array": "false",
                    "required": "true"
                },
                {
                    "name": "host",
                    "type": "string",
                    "description": "",
                    "label": "Host",
                    "mask": "false",
                    "array": "false",
                    "required": "true"
                }
            ]
        }
    ]
}
```
{
    'name': 'port',
    'type': 'int',
    'description': 'TCP/IP port number used to communicate with the LDAP server.',
    'label': 'TCP Port',
    'default': '',
    'mask': 'false',
    'array': 'false',
    'required': 'true'
},
{
    'name': 'ssl',
    'type': 'boolean',
    'description': 'Select the check box to connect to the LDAP server using SSL.',
    'label': 'SSL',
    'default': 'false',
    'mask': 'false',
    'array': 'false',
    'required': 'true'
},
{
    'name': 'principal',
    'type': 'string',
    'description': 'The distinguished name with which to authenticate to the LDAP server.',
    'label': 'Principal',
    'default': '',
    'mask': 'false',
    'array': 'false',
    'required': 'true'
},
{
    'name': 'credentials',
    'type': 'string',
    'description': 'Password for the principal.',
    'label': 'Password',
    'default': '',
    'mask': 'true',
    'array': 'false',
    'required': 'true'
},
{
    'name': 'baseContexts',
    'type': 'string',
    'description': 'One or more starting points in the LDAP tree that will be used when searching the tree. Searches are performed when discovering users from the LDAP server or when looking for the groups of which a user is a member.',
    'label': 'Base Contexts',
    'default': [
    ],
    'mask': 'false',
    'array': 'true',
    'required': 'true'
},
{
    'name': 'accountNameAttribute',
    'type': 'string',
    'description': 'Attribute which holds the account's user name.',
    'label': 'Account User Name Attribute',
}
"default":"uid",
"mask":"false",
"array":"false",
"required":"true"
}
],
"AdvancedAttributes":[
{
"name":"uidAttribute",
"type":"string",
"description":"The name of the LDAP attribute which is mapped to the Uid attribute.",
"label":"Uid Attribute",
"default":"uid",
"mask":"false",
"array":"false",
"required":"false"
},
{
"name":"accountSearchFilter",
"type":"string",
"description":"An optional LDAP filter to control which accounts are returned from the LDAP resource. If no filter is specified, only accounts that include all specified object classes are returned.",
"label":"LDAP Filter for Retrieving Accounts",
"default":"(uid=*)",
"mask":"false",
"array":"false",
"required":"false"
},
{
"name":"passwordAttribute",
"type":"string",
"description":"The name of the LDAP attribute which holds the password. When changing an user’s password, the new password is set to this attribute.",
"label":"Password Attribute",
"default":"userpassword",
"mask":"false",
"array":"false",
"required":"false"
},
{
"name":"accountObjectClasses",
"type":"string",
"description":"The object class or classes that will be used when creating new user objects in the LDAP tree. When entering more than one object class, each entry should be on its own line; do not use commas or semi-colons to separate multiple object classes. Some object classes may require that you specify all object classes in the class hierarchy.",
"label":"Account Object Classes",
"default":[
 "top",
 "person",
 "organizationalPerson",
 "inetOrgPerson" ]
},
"mask":"false",
"array":"true",
"required":"false" }
Where:

- **TargetAttributes** is an array of objects, where each object represents a target type.
- **TargetType** is the target type.
- **DisplayName** is how the target type name should display.
- **BasicAttributes** is an array of objects, where each object represents basic attributes for the target type.
- **AdvancedAttributes** is an array of objects, where each object represents advanced attributes for the target type.
- **name** is the attribute name to use when constructing the target JSON to create a target.
- **type** is the attribute type. Acceptable values include `string`, `int`, `boolean`, or `lov` (list of values).
- **description** is a helpful description of the attribute.
- **label** is how the attribute name should display.
- **default** is a default value for the attribute.
  - Specify a single value if the `array` parameter is `false` or specify an array of values if `array` is `true`.
- **mask** hides sensitive values, such as credentials.
  - Specify `true` to hide attributes.
  - Specify `false` if hiding attributes is not necessary.
- **array** indicates whether the attribute is single-valued or an array of multiple values.
  - Specify `true` if the attribute is an array of multiple values.
  - Specify `false` if the attribute is single-valued.
- **required** indicates whether the attribute is mandatory or optional.
  - Specify `true` for mandatory attributes.
  - Specify `false` for optional attributes.

### B.5.2 Add a Target

Use this API to add a target.

---

**Note:**

- You must be an administrator with the *Security Administrator* Admin Role to use this API.
- First, you must obtain a list of attributes for the target type as described in Section B.5.1, "Get Target Attributes." You use these attributes to create the JSON object sent in the body.
### Example B–17  Sample JSON Representation of Target for Addition (ldap TargetType)

```json
{
    "target": {
        "targetType": "ldap",
        "targetName": "ldap1-target",
        "host": "opam_server_host",
        "passwordpolicy": "passwordpolicy1",
        "domain": "berkeley",
        "description": "Ldap target",
        "organization": "ST-US",
        "credentials": "welcome",
        "uidAttribute": "uid",
        "port": "9876",
        "passwordAttribute": "userpassword",
        "principal": "cn=orcladmin",
        "accountSearchFilter": "(uid=*)",
        "baseContexts": [
            "cn=Users,c=US"
        ],
        "ssl": "false",
        "accountObjectClasses": [
            "top",
            "person",
            "organizationalPerson",
            "inetOrgPerson"
        ],
        "accountNameAttribute": "uid"
    }
}
```

### Example B–18  Sample JSON Representation of Target for Addition (database TargetType)

```json
{
    "target": {
        "targetType": "database",
        "targetName": "db1_target",
        "passwordpolicy": "passwordpolicy1",
        "passwordrollover": "true",
        "host": "afg1140282",
        "domain": "adc1140282Domain",
        "description": "Dbase target for the automation",
        "connectionProperties": ",",
        "dbType": "Oracle",
        "jdbcUrl": "jdbc:oracle:thin:@afg1140282.us.pk.com:11227:db5474",
        "loginPassword": "welcome1",
        "loginUser": "system"
    }
}
```
Example B–19  Sample JSON Representation of Target for Addition (unix TargetType)

```
{
  "target": {
    "targetType": "unix",
    "targetName": "BackUpUnixTarget",
    "passwordpolicy": "passwordpolicy1",
    "passwordrollover": "true",
    "host": "adc0345labc.us.mycompany.com",
    "domain": "US",
    "description": "Backup system",
    "organization": "IT",
    "port": "23",
    "sudoPasswdExpectExpression": "password",
    "commandTimeout": "120000",
    "passwordExpectExpressions": "new\[\s\]unix\[\s\]?password:,new\[\s\]unix\[\s\]?password\(\[\s\]again\)?:\",
    "loginShellPrompt": "$",
    "prePasswdExpectExpression": "None",
    "sudoAuthorization": "false",
    "loginUserpassword": "welcome1",
    "loginUser": "aime2"
  }
}
```

Sample Output

https://opam_server_host:opam_ssl_port/opam/target/9bbcb087174ad1900ea691a2573b61 as the Location.

Where:

- **target** is the target JSON object.
- **targetName** is the name of the target.
- **targetType** is the target type.
- **passwordpolicy** is the Password Policy identifier of the Password Policy applied to the target.
- **passwordrollover** is the flag that indicates whether to enable automatic password recycling for a target's service account.

If you set this flag to true, then Oracle Privileged Account Manager automatically resets the target's service account password based on the settings specified in the Password Policy that applies.

---

**Note:** The passwordrollover flag is currently not supported for ldap or lockbox targets.

All of the other attributes are dynamic and they correspond to the attributes in Section B.5.1, "Get Target Attributes."

**B.5.3 Verify a Target**

Use this API to verify a target.
Note: First, you must obtain a list of attributes for the target type. Refer to Section B.5.1, "Get Target Attributes," to create the JSON object to be sent in the body.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/target/test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td>JSON representation of target for addition/test</td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>

Example B–20  Sample JSON Representation of Target for Addition/Verification

```json
{
  "target": {
    "targetType": "ldap",
    "targetName": "ldap1-target",
    "host": "opam_server_host",
    "passwordPolicy": "passwordpolicy1",
    "domain": "berkeley",
    "description": "Ldap target",
    "organization": "ST-US",
    "credentials": "welcome",
    "uidAttribute": "uid",
    "port": "9876",
    "passwordAttribute": "userpassword",
    "principal": "cn=orcladmin",
    "accountSearchFilter": "(uid=*)",
    "baseContexts": [
      "cn=Users,c=US"
    ],
    "ssl": "false",
    "accountObjectClasses": [
      "top",
      "person",
      "organizationalPerson",
      "inetOrgPerson"
    ],
    "accountNameAttribute": "uid"
  }
}
```

Where:

- **target** is the target JSON object.
- **targetName** is the name of the target.
- **targetType** is the target type.
- **passwordPolicy** is the Password Policy identifier of the Password Policy applied to the target.

All of the other attributes are dynamic and they correspond to the attributes in Section B.5.1, "Get Target Attributes."
B.5.4 Retrieve a Target

Use this API to retrieve a target.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/target/{targetUID}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of target</td>
</tr>
</tbody>
</table>

**Example B–21**  **Sample JSON Representation of Target (ldap Target Type)**

```
{
  "target": {
    "targetUID": "9bbcbbbb087174ad1900ea691a2573b61",
    "targetType": "ldap",
    "targetName": "ldap1-target",
    "host": "opam_server_host",
    "domain": "berkeley",
    "description": "Ldap target",
    "organization": "ST-US",
    "credentials": "welcome",
    "uidAttribute": "uid",
    "port": 9876,
    "passwordAttribute": "userpassword",
    "principal": "cn=orcladmin",
    "accountSearchFilter": "(uid=*)",
    "baseContexts": [
      "cn=Users,c=US"
    ],
    "ssl": "false",
    "accountObjectClasses": [
      "top",
      "person",
      "organizationalPerson",
      "inetOrgPerson"
    ],
    "accountNameAttribute": "uid",
  }
}
```

**Example B–22**  **Sample JSON Representation of Target (database Target Type)**

```
{
  "target": {
    "targetUID": "62bcfb98f95d4966ab0ff9a44717a20a",
    "targetType": "database",
    "targetName": "db1_target",
    "passwordpolicy": "passwordpolicy1",
    "passwordrollover": "true",
    "host": "afg1140282",
    "domain": "adc1140282Domain",
    "description": "Dbase target for the automation",
    "connectionProperties": "",
    "dbType": "Oracle",
    "jdbcUrl": "jdbc:oracle:thin:@afg1140282.us.pk.com:11227:db5474",
    "loginPassword": "welcome1",
  }
}
```
"loginUser" : "system"
]
]

Example B–23  Sample JSON Representation of Target (unix Target Type)
{
  "target" : {
    "targetUID" : "a00075b4b7bb453c9482d02535989b53",
    "targetType" : "unix",
    "targetName" : "unix1-target",
    "passwordpolicy" : "passwordpolicy1",
    "passwordrollover" : "true",
    "host" : "adc0345labc.us.mycompany.com",
    "domain" : "US",
    "description" : "Backup system",
    "organization" : "IT",
    "port" : "23",
    "sudoPasswdExpectExpression" : "password",
    "commandTimeout" : "120000",
    "passwordExpectExpressions" :
      "new[\s](unix[\s])?password:,new[\s](unix[\s])?password\[\s\]again?:",
    "loginShellPrompt" : "$",
    "prePasswdExpectExpression" : "None",
    "sudoAuthorization" : "false",
    "loginUserpassword" : "welcome1",
    "loginUser" : "aime2"
  }
}

Where:

- **target** is the target JSON object.
- **targetUID** is the target's unique identifier.
- **targetName** is the name of the target.
- **targetType** is target type.
- **passwordrollover** is the flag that indicates whether to enable automatic password recycling for a target’s service account.

If you set this flag to **true**, then Oracle Privileged Account Manager automatically resets the target’s service account password based on the settings specified in the Password Policy that applies.

---

**Note:** The passwordrollover flag is currently not supported for ldap or lockbox targets.

---

All of the other attributes are dynamic and they correspond to the attributes in Section B.5.1, "Get Target Attributes."

B.5.5 Update a Target

Use this API to update a target.
You can change all of the attributes, except `targetType` and `targetUID`, and you can change multiple attributes at a time.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/target/[targetUID]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td>JSON representation of Target Modification</td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>

**Example B–24 Sample JSON Object to Modify Target**

```
"modifications": [
    {
      "modification": {
        "host": "opam_server_host"
      }
    },
    {
      "modification": {
        "port": "6000"
      }
    }
]
```

Where:
- `targetUID` is the target's unique identifier.
- `modifications` is an array of modification JSON objects.
- `modification` is a JSON object representing the modification of a single attribute.

**B.5.6 Remove a Target**

Use this API to delete a target.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/target/[targetUID]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>DELETE</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>
B.5.7 Search for Targets

Use this API to search for a target using any of the following request parameters:

- type
- name
- hostname
- domain
- description
- org

All of these parameters are optional.

---

**Note:** You must be an administrator with the User Manager Admin Role, Security Administrator Admin Role, or Security Auditor Admin Role to use this API.

---

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/target/search?param1=value1&amp;param2=value2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of Target Collection</td>
</tr>
</tbody>
</table>

---

**Sample URIs:**

- `https://opam_server_host:opam_ssl_port/opam/target/search?` - Returns all targets

---

**Example B–25 Sample JSON Representation of Target Collection**

```json
[
  "Target Collection":
  {
    "target":
    {
      "uri": "https:\/\/opam_server_host:opam_ssl_port\/opam\/target\1/9bbcb8b087174ad1900ea691a2573b61",
      "type": "ldap",
      "name": "person1-ldap",
      "host": "opam_server_host",
      "domain": "berkeley",
      "description": "Ldap target"
    },
    {
      "uri": "https:\/\/opam_server_host:opam_ssl_port\/opam\/target\1/ac246a162e948c7b1cdcc17dfe92c15",
      "type": "ldap",
      "name": "person1-ldap2",
      "domain": "berkeley",
      "description": "Ldap target"
    }
  }
]
```
Where:
- **Target Collection** is an array of target JSON objects.
- **target** is the target JSON object.
- **uri** is the target resource URI.
- **type** is the target type.
- **hostname** is the target's host name.
- **name** is the target name.
- **org** is the target’s organization.
- **domain** is the target’s domain.
- **description** is a description of the target system.

### B.5.8 Get Available Accounts

Use this API to retrieve all of the accounts present on the target system.

**Note:** You must be an administrator with the Security Administrator Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/target/{targetUID}/availableaccounts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 OK and JSON representation of account collection</td>
</tr>
</tbody>
</table>

**Example B–26  Sample JSON Representation of Account Collection**

```json
{
  "AvailableAccounts": [
    {
      "accountName": "SCOTT",
      "accountUid": "SCOTT"
    },
    {
      "accountName": "BLAKE",
      "accountUid": "BLAKE"
    },
    {
      "accountName": "JONES",
      "accountUid": "JONES"
    }
  ]
}
```
Where:
  ■ **AvailableAccounts** is an array of the accounts present on the target system.
  ■ **accountName** is the account name.
  ■ **accountUID** is the account's unique identifier.

### B.5.9 Retrieve Accounts Registered on a Target

Use this API to retrieve all the accounts on the target that are registered with Oracle Privileged Account Manager.

**Note:** You must be an administrator with the **User Manager Admin Role**, **Security Administrator Admin Role**, or **Security Auditor Admin Role** to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/target/{targetUID}/accounts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of URI collection of accounts</td>
</tr>
</tbody>
</table>

**Example B–27  Sample JSON Representation of URI Collection of Accounts**

```json
{
  "URI Collection": [
    {
      "account": {
        "uri": "https://opam_server_host:opam_ssl_port/opam/account/3740553e999a4f6aa8e8f9286d320cb4",
        "accountName": "sherlock"
      }
    },
    {
      "account": {
        "uri": "https://opam_server_host:opam_ssl_port/opam/account/c1106278022409aad758aec69d927d",
        "accountName": "root"
      }
    }
  ]
}
```

Where:
  ■ **URI Collection** is an array of accounts on a target that are registered with Oracle Privileged Account Manager.
  ■ **account** is the account JSON object.
  ■ **uri** is the account's URI.
  ■ **accountName** is the account name.
B.5.10 Get Target Types

Use this API to retrieve a list of all supported target types.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/target/types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of supported target types</td>
</tr>
</tbody>
</table>

Example B–28 Sample JSON Representation of Supported Target Types

```json
{
  "targettypes": [
    "ldap",
    "unix",
    "database",
    "lockbox"
  ]
}
```

Where:
- targettypes are the supported target types.

B.5.11 Reset Password

Use this API to reset the password on the target’s service account.

Note:
- You must be an administrator with the Security Administrator Admin Role to use this API.
- Refer to Chapter 7, "Working with Service Accounts" for information about service accounts.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/target/{targetUID}/resetpassword</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>

Example B–29 Sample JSON Representation of the New Password

```json
{
  "password": "welcome1"
}
```

or

```json
{
  "autogen": "true"
}
```
Where:
- **targetUID** is the target's unique identifier.
- **password** is the password to assign to the service account.
- **autogen** is the flag that controls whether to automatically generate the password or not. (Default is *false*.)

### B.5.12 Show Service Account Password

Use this API to retrieve and display the service account password.

**Note:**
- You must be an administrator with the *Security Administrator* Admin Role to use this API.
- Refer to Chapter 7, "Working with Service Accounts" for information about service accounts.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/target/{targetUID}/showpassword</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of service account</td>
</tr>
</tbody>
</table>

**Example B–30 Sample JSON Representation of Account Token**

```json
{
  "serviceAccount" : {
    "targetName" : "APILDAP",
    "targetUID" : "ad3163bfb37b4544a4c12ae06a39c2d9",
    "targetAccount" : "cn=admin",
    "targetPassword" : "welcome1",
    "targetPasswordChangeTime" : "2013-01-27 02:58:13.259"
  }
}
```

Where:
- **targetUID** is the target's unique identifier.
- **targetName** is the name of the target.
- **targetAccount** is the service account on the target.
- **targetPassword** is the service account password.
- **targetPasswordChangeTime** is the time when the password was modified.
B.5.13 Show Service Account Password (Deprecated)

**Note:** This API has been deprecated. Oracle recommends that you use the Show Service Account Password API in Section B.5.12, "Show Service Account Password."

Use this API to retrieve and display the service account password.

**Note:**
- You must be an administrator with the Security Administrator Admin Role to use this API.
- Refer to Chapter 7, "Working with Service Accounts" for information about service accounts.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/target/{targetUID}/showpassword</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of service account</td>
</tr>
</tbody>
</table>

**Example B–31 Sample JSON Representation of Account Token**

```json
{
  "serviceAccount" : {
    "targetName" : 'APILDAP',
    "targetUID" : 'ad3163bf3b37b4544a4c12ae06a39c2d9',
    "targetAccount" : 'cn=admin',
    "targetPassword" : 'welcome1',
  }
}
```

Where:
- **targetUID** is the target's unique identifier.
- **targetName** is the name of the target.
- **targetAccount** is the service account on the target.
- **targetPassword** is the service account password.
- **targetPasswordChangeTime** is the time when the password was modified.

B.5.14 Show Service Account Password History

Use this API to retrieve and display the service account password history.
Note:

- You must be an administrator with the Security Administrator Admin Role to use this API.
- Refer to Chapter 7, "Working with Service Accounts" for information about service accounts.

---

<table>
<thead>
<tr>
<th>URI</th>
<th>https:///opam_server_host:opam_ssl_port/opam/target/[targetUID]/showpasswordhistory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of service account</td>
</tr>
</tbody>
</table>

Example B–32  Sample JSON Representation of Target Token

```
{
  "targetToken": {
    "targetName": "SessionMgr_Target",
    "targetUID": "d5ac79483c2a4641adb97f2e72b17f28",
    "passwordHistory": [
      {
        "targetPassword": "welcome1",
        "modificationTime": "1383078344"
      },
      {
        "targetPassword": "4PkVerh7",
        "modificationTime": "1383078329"
      },
      {
        "targetPassword": "l9yAigqj",
        "modificationTime": "1383078314"
      },
      {
        "targetPassword": "welcome1",
        "modificationTime": "1383010874"
      }
    ]
  }
}
```

Where:

- **targetUID** is the target’s unique identifier.
- **targetName** is the name of the target.
- **passwordHistory** is the service account password history.
- **targetPassword** is the service account password.
- **modificationTime** (UTC time in seconds) is the time when the password was modified.

Password history results are sorted by modification time, where the most recent results will be at the top.
B.6 Account Resource

The APIs described in this section include:

- Add an Account to a Target
- Get Applicable Usage Policy for the Account
- Grant a User/Role Access to an Account
- Add or Remove a CSF Map-Key for an Account
- Search Accounts
- Search Assigned Accounts
- Retrieve an Account
- Retrieve Grantees on an Account
- Retrieve Users Who Checked Out an Account
- Check Out an Account
- Get All Checked Out Accounts
- Get Session Checkout Instructions
- Checkout History for an Account
- Checkout History
- Check In an Account
- Verify an Account
- Update an Account
- Remove an Account
- Remove a User's/Role's Access to an Account
- Show Password
- Show Password (Deprecated)
- Show Password History
- Show Password History (Deprecated)
- Reset Password

B.6.1 Add an Account to a Target

Use this API to add an account to the target. This API does not create an account on the target system, but it registers the existing account with the Oracle Privileged Account Manager target.

---

**Note:**

- You must never use the same account as the service account and as a privileged account to be managed by Oracle Privileged Account Manager.
- You must be an administrator with the Security Administrator Admin Role to use this API.
Example B–33  Sample JSON Representation of Account for Addition/Verification

```
{
   "account":{
      "accountName":"admin",
      "description": "maintenance account on the machine",
      "password": "welcome1",
      "passwordpolicy": "passwordpolicy2",
      "shared": "true",
      "targetUID": "9bbcbbbb087174ad1900ea691a2573b61"
   }
}
```

Where:

- **account** is the account JSON object.
- **accountName** is the name of the account.
- **description** is a description of the account. This attribute is optional.
- **password** is the account password. This attribute is optional.
- **passwordpolicy** is the policy ID of the Password Policy applicable to the account. This parameter is optional. By default, this parameter uses the global Default Password Policy.
- **shared** indicates the shared status of the account. This value is a Boolean and the default setting is false.
- **targetUID** is the target's unique identifier.

### B.6.2 Get Applicable Usage Policy for the Account

Use this API to get the applicable Usage Policy for an account.

#### Example B–34  Sample JSON Representation of the Usage Policy

```
{"usagepolicy":
   {
      "policyid": "bafdf3072b4b442db185dca18bd00e69",
      "policyname": "usage_policy_anytime"
   }
}
```

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account</th>
<th>Method</th>
<th>GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content-Type</td>
<td>application/json</td>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of the Usage Policy</td>
</tr>
</tbody>
</table>
Where:
- **usagepolicy** is the Usage Policy JSON object.
- **policyid** is the Usage Policy's unique identifier.
- **policyname** is a name of the policy

### B.6.3 Grant a User/Role Access to an Account

Use this API to grant a user or role access to an account. Multiple users and roles can be granted the access at a time.

**Note:** You must be an administrator with the **User Manager** Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th><code>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}</code></th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td><code>application/json</code></td>
</tr>
<tr>
<td>Body</td>
<td>JSON representation for adding grantees</td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>

**Example B–35  Sample JSON Representation for Adding Grantees**

```
{
    "modifications": [
        {
            "modification": {
                "usagepolicy": "usagepolicy1",
                "role": "opamgroup1",
                "operation": "add"
            }
        },
        {
            "modification": {
                "usagepolicy": "usagepolicy1",
                "user": "opamuser1",
                "operation": "add"
            }
        }
    ]
}
```

Where:
- **accountUID** is the account's unique identifier.
- **modifications** are an array of modification JSON objects.
- **modification** is a JSON object representing the modification of a single attribute.
- **role** indicates that a group has to be granted an access. This parameter value is the group name.
- **user** indicates that a user has to be granted an access. This parameter value is the user login id.
• **usagepolicy** indicates the Usage Policy identifier to be applied to the grant.
  
• **operation** indicates the type of operation to be performed. Acceptable values include:
  
  – **add** indicates grant.
  
  – **delete** indicates revocation.
  
  – **replace** indicates replacement of usagepolicy with a new value.

### B.6.4 Add or Remove a CSF Map-Key for an Account

Use this API to add a CSF map-key to an account or remove the map-key from an account. You can add or remove multiple map-keys at a time.

---

**Note:** You must be an administrator with the **Security Administrator** Admin Role to use this API.

---

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td>JSON representation for adding keymaps</td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>

---

**Example B–36  Sample JSON Representation for Map-Keys Addition/Removal**

```json
{
  "modifications": [
    {
      "modification": {
        "keymap": "[app1][sd45kjlf4g][t3:///localhost:7001][weblogic][password]",
        "operation": "add"
      }
    },
    {
      "modification": {
        "keymap": "[hrmap][hrkey2][t3:///localhost:7001][weblogic][password]",
        "operation": "delete"
      }
    }
  ]
}
```

Where:

• **accountUID** is the account’s unique identifier.

• **modifications** is an array of modification JSON objects.

• **modification** is a JSON object representing the modification of a single attribute.

• **keymap** is the map-key to be added or removed. The map-key must be in the following format:

  `{csfmap}[csfkey][Administration Server Url][username][password]`
■ operation indicates the type of operation to be performed. Acceptable values include:
  – add indicates addition of map-key.
  – delete indicates removal of map-key.

B.6.5 Search Accounts

Use this API to search accounts using one or more of the following search request parameters:

■ type
■ domain
■ description
■ name
■ accountname

All of these parameters are optional.

Note: You must be an administrator with the User Manager Admin Role or the Security Administrator Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/search?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of account collection</td>
</tr>
</tbody>
</table>

Example B–37  Sample JSON Representation of Account Collection

```
{
  "AccountCollection" : [
    {
      "account" : {
        "shared" : false,
        "passwordchangetime" : 1383072107,
        "targetUID" : "eadd96486e9a47b79bd23cf1167bd2b2",
        "domain" : "needtofix",
        "targetName" : "sunds_6.3_target",
        "targetType" : "ldap",
        "accountlevelstatus" : "checkedIn",
        "description" : ",",
        "accountName" : "dsperson1",
        "uri" : "https://localhost:7002/opam/account/35e2709edf0443edae8f67727d937bec",
        "accountUID" : "35e2709edf0443edae8f67727d937bec"
      },
    },
    {
      "account" : {
        "shared" : false,
        "passwordchangetime" : 1383072107,
```
Where:

- **account** is the account JSON object.
- **shared** indicates the shared status of the account. This value is a Boolean and the default setting is `false`.
- **accountlevelstatus** indicates whether the account has been checked in by anyone. Acceptable values are `checkedIn` and `checkedOut`.
- **description** is a description of the account. This attribute is `optional`.
- **accountName** is the name of the account.
- **accountUID** is the account's unique identifier.
- **passwordchangetime** is the time when the password was modified.

For all other attribute definitions, refer to Section B.5, "Target Resource."

### B.6.6 Search Assigned Accounts

Use this API to search assigned accounts using one or more of the following search request parameters:

- **type**
- **domain**
- **description**
- **name**
- **accountname**

All of these parameters are `optional`.

<table>
<thead>
<tr>
<th><strong>URI</strong></th>
<th><code>https://opam_server_host:opam_ssl_port/opam/account/myaccounts/search?</code></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
<td>GET</td>
</tr>
<tr>
<td><strong>Content-Type</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Body</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Returns on Success</strong></td>
<td>Status code 200 and JSON representation of account collection</td>
</tr>
</tbody>
</table>
Example B–38  Sample JSON Representation of Account Collection

```json
{
  "AccountCollection": [
    {
      "account": {
        "uri": "https://myhost:7002/opam/account/b0e7ae053afbb45658da4e3a0453bffe",
        "accountUID": "b0e7ae053afbb45658da4e3a0453bffe",
        "accountName": "dduck",
        "description": ",",
        "targetUID": "6e97271709c874c5897d7ea52071f0aac",
        "targetName": "unix1-target",
        "targetType": "unix",
        "domain": "US"
      }
    }
  ],
  "count": 1
}
```

Where:
- **account** is the account JSON object.
- **accountUID** is the account's unique identifier.
- **accountName** is the name of the account.
- **description** is a description of the account. This attribute is optional.

For all other attribute definitions, refer to Section B.5, "Target Resource."

B.6.7 Retrieve an Account

Use this API to retrieve an account.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of account</td>
</tr>
</tbody>
</table>

Example B–39  Sample JSON Representation of Account

```json
{
  "account": {
    "accountUID": "3f74a85e39e64432ba917a2e60fa15aa",
    "targetUID": "9bcbbb087174ad590ea91a2573b61",
    "accountName": "admin",
    "shared": true,
    "accountlevelstatus": "checkedIn",
    "passwordpolicy": "passwordpolicy2",
    "protocol": "ssh",
    "port": 22
  }
}
```

Where:
- **account** is the account JSON object.
■ **accountUID** is the account’s unique identifier.
■ **accountName** is the name of the account.
■ **passwordPolicy** is the policy ID of the Password Policy applicable to the account.
■ **shared** indicates the shared status of the account. This value is a Boolean and the default setting is **false**.
■ **targetUID** is target’s unique identifier.
■ **accountLevelStatus** indicates whether the account has been checked in by anyone. Acceptable values are **checkedIn** and **checkedOut**.
■ **protocol** is the protocol used to connect to the Oracle Privileged Session Manager server.
■ **port** is the port used to connect to the Oracle Privileged Session Manager server.

### B.6.8 Retrieve Grantees on an Account

Use this API to retrieve all the grantees of an account. A grantee can be a user or a role.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}/grantees</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td><strong>Returns on Success</strong></td>
<td>Status code 200 and JSON representation of Grantees</td>
</tr>
</tbody>
</table>

#### Example B–40  Sample JSON Representation of Grantees

```json
{
  "grantees":{
    "users":[
      "opamuser1"
    ],
    "roles":[
      "opamgroup1"
    ]
  }
}
```

Where:
- **grantees** are grantees of the account.
- **users** are the users who have been granted the account. Each value is the user’s login ID/UID.
- **roles** are the groups or roles who have been granted the account. Each value is a group name.

### B.6.9 Retrieve Users Who Checked Out an Account

Use this API to retrieve a list of all users who have currently checked out an account.
**Note:** You must be an administrator with the User Manager Admin Role or the Security Administrator Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}/whocheckedout</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of users who checked out the account.</td>
</tr>
</tbody>
</table>

**Example B–41  Sample JSON Representation of Users Who Checked Out the Account**

```json
{
    "users": [
        {
            "user": {
                "uid": "user_manager",
                "expiryTime": "1382147587",
                "checkoutTime": "1381715587",
                "checkoutUID": "f499b76719ba4d0aa30487e58316def3",
                "checkoutType": "password",
                "transcriptURL":"
            }
        },
        {
            "user": {
                "uid": "user_manager",
                "expiryTime": "1382147587",
                "checkoutTime": "1381715587",
                "checkoutUID": "f499b76719ba4d0aa30487e58316def3",
                "checkoutType": "session",
                "transcriptURL": "https://myhost:7002/opam_checkout/dee833184664ddfa09f454d0a9a023d/transcript"
            }
        }
    ]
}
```

Where:
- **transcriptURL** is the URL you use to access the session transcript.
- **checkoutType** indicates whether the checkout was a session checkout or a password checkout.
- **checkoutUID** is the unique ID for the checkout.

**B.6.10 Check Out an Account**

Use this API to check out an account.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}/checkout</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
</tbody>
</table>

---

Working with Oracle Privileged Account Manager's RESTful Interface  B-47
Example B–42  Sample JSON Representation of Account Token

```
{
  "accountToken": {
    "accountName": "admin",
    "accountUID": "3f74a85e39e64432ba917a2e60fa15aa",
    "accountPassword": "GJN8p2ol"
  }
}
```

Where:
- **accountUID** is the account’s unique identifier.
- **accountName** is the name of the account.
- **accountPassword** is the account password.

B.6.11 Get All Checked Out Accounts

Use this API to retrieve a list of all accounts that have been checked out by the logged in user.

**Example B–43 Sample JSON Representation of Account Collection**

```
{
  "Checkouts": [
    {
      "uri": "https://myhost:7002/opam/account/b0e7ae053af6b45658da4e3a0453bffec",
      "accountUID": "b0e7ae053af6b45658da4e3a0453bffec",
      "accountName": "dduck",
      "status": "checkedOut",
      "targetUID": "6e972170c874c5897d7ea52071f0aac",
      "targetName": "unix1-target",
      "targetType": "unix",
      "domain": "US",
      "expiryTime": "1371945854",
      "checkoutUID": "b97b2de6a80b40c48f873067027ac476",
      "checkoutType": "session",
      "transcriptURL": "https://myhost:7002/opam/account/checkout/b97b2de6a80b40c48f873067027ac476/\ntranscript"
    },
    {
      "uri": "https://myhost:7002/opam/account/b0e7ae053af6b45658da4e3a0453bffec",
      "accountUID": "b0e7ae053af6b45658da4e3a0453bffec",
      "accountName": "dduck",
      "status": "checkedOut",
      "targetUID": "6e972170c874c5897d7ea52071f0aac",
      "targetName": "unix1-target",
      "targetType": "unix",
      "domain": "US",
      "expiryTime": "1371945854",
      "checkoutUID": "b97b2de6a80b40c48f873067027ac476",
      "checkoutType": "session",
      "transcriptURL": "https://myhost:7002/opam/account/checkout/b97b2de6a80b40c48f873067027ac476/\ntranscript"
    }
  ]
}
```

Returns on Success

| Status code 200 and JSON representation of account token |

**Example B–44 URI and Method**

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/mycheckouts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of account collection</td>
</tr>
</tbody>
</table>
Where:

- **accountUID** is the account’s unique identifier.
- **accountName** is the name of the account.
- **checkoutUID** is the unique ID for the checkout.
- **checkoutType** indicates whether the checkout was a session checkout or a password checkout.
- **transcriptURL** is the URL to access the session transcript.

For all other attribute definitions, refer to Section B.5, "Target Resource."

### B.6.12 Get Session Checkout Instructions

Use this API to get information to help you perform a session checkout.

**Note:** For more information about password and session checkouts, refer to Section 8.5, "Checking Out Privileged Accounts" and Section 8.5.2, "Checking Out Privileged Account Sessions."

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}/checkout/session/instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of output</td>
</tr>
</tbody>
</table>

**Example B–44**  **Sample JSON Representation of Session Checkout Instructions**

```json
{
  "sessionCheckoutInstructions": {
    "accountName": "dduck",
    "targetName": "bkottaha-unix",
    "port": 1222,
    "instruction": "ssh -p <port> <opamuser>:<targetname>:<accountname>@<sessionmgrhost>\nUse opam password on password prompt"
  }
}
```

Where:

- **accountName** is the name of the account.
- **targetName** is the name of the target.
- **port** is the port that Session Manager listens to for connections.
■ **instruction** is the information required to perform a session checkout.

### B.6.13 Checkout History for an Account

Use this API to search for an account’s checkout history using one or more of the following parameters:

- **from**: Specify start time in seconds (UTC) *(required)*.
- **to**: Specify end time in seconds (UTC) *(required)*.
- **uid**: Specify the userID *(optional)*.
- **pattern**: Specify the command that was executed or a term in the log *(optional)*.
- **size**: Specify the number of array elements to be returned *(optional)*.

Use the `from` and `to` parameters to specify the time period in which the checkouts were running.

---

**Note:** You must be an administrator with the **User Manager** or **Security Administrator** Admin Role to access this query.

---

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}/checkouts/historical/search?param1=val1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of output</td>
</tr>
</tbody>
</table>

---

**Sample URL Output**

https://myhost:7002/opam/account/8d9e9ce750da4aedac3ffbea0d28a73a/checkouts/historical/search?from=123&to=1372893007&size=2&pattern=ls

---

**Example B–45 Sample JSON Representation of Account Checkout History**

```json
{
  "checkouts": [
    {
      "checkout": {
        "accountName": "dduck",
        "targetName": "unix1-target",
        "uid": "user_manager",
        "starttime": "1372883311",
        "endtime": "1372883323",
        "checkoutUID": "9c3c5d687d41a57b7dbda0692c9b06d",
        "checkoutType": "session",
        "transcriptURL": "https://myhost:7002/opam/checkout/9c3c5d687d41a57b7dbda0692c9b06d/transcript"
      }
    },
    {
      "checkout": {
        "accountName": "dduck",
        "targetName": "unix1-target",
        "uid": "user_manager",
```
"starttime": "1372812996",
"endtime": "1372813007",
"checkoutUID": "60f253f7c8a941309d64fe88787f90ee",
"checkoutType": "password",
"transcriptURL": ""
}
}
|
|
,"totalcount": 3,
"returncount": 2
}

Where:

- **transcriptURL** is the URL you use to access the session transcript.
- **checkoutType** indicates whether the checkout was a session checkout or a password checkout.
- **checkoutUID** is the unique ID for the checkout.
- **totalcount** is the number of actual search results.
- **returncount** is the number of search results that were actually returned (determined by size).

For all other attribute definitions, refer to Section B.6, "Account Resource."

### B.6.14 Checkout History

Use this API to search for the checkout history of all accounts, using one or more of the following parameters:

- **from**: Specify start time in seconds (UTC) *(required).*
- **to**: Specify end time in seconds (UTC) *(required).*
- **targetname**: Specify the name of a target on which to search *(optional).*
- **accountname**: Specify the name of an account to search *(optional).*
- **uid**: Specify the userID *(optional).*
- **pattern**: Specify the command that was executed or a term in the log *(optional).*
- **size**: Specify the number of array elements to be returned *(optional).*

Use the **from** and **to** parameters to specify the time period in which the checkouts were running.

**Note:** You must be an administrator with the Security Auditor Admin Role to access this query.

<table>
<thead>
<tr>
<th><strong>URI</strong></th>
<th>https://opam_server_host:opam_ssl_port/opam/checkout/historical/search?param1=val1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Method</strong></td>
<td>GET</td>
</tr>
<tr>
<td><strong>Content-Type</strong></td>
<td>discard</td>
</tr>
<tr>
<td><strong>Body</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Returns on Success</strong></td>
<td>Status code 200 and JSON representation of output</td>
</tr>
</tbody>
</table>
Sample URL
https://myhost:7002/opam/checkout/historical/search?from=123&to=1472816146&size=2&pattern=ls&accountname=a&targetname=h&uid=u

Example B–46 Sample JSON Representation of Checkout History

```
{
    "checkouts": [
    {
        "checkout": {
            "accountName": "dduck",
            "targetName": "unix1-target",
            "uid": "user_manager",
            "starttime": "1372883311",
            "endtime": "1372883323",
            "checkoutUID": "9c3c5d687d414a57b7dbda0692c9b06d",
            "checkoutType": "session",
            "transcriptURL": "https://myhost:7002/opam/checkout/9c3c5d687d414a57b7dbda0692c9b06d/transcript"
        }
    },
    {
        "session": {
            "accountName": "mmouse",
            "targetName": "unix1-target",
            "uid": "user_manager",
            "starttime": "1372880658",
            "endtime": "1372880667",
            "checkoutUID": "8d2a99d2b34a4e3297b051fb4028652f",
            "checkoutType": "password",
            "transcriptURL": ""
        }
    }
    ],
    "totalcount": 4,
    "returncount": 2
}
```

Where:

- **transcriptURL** is the URL you use to access the session transcript.
- **checkoutType** indicates whether the checkout was a session checkout or a password checkout.
- **checkoutUID** is the unique ID for the checkout.
- **totalcount** is the number of actual search results.
- **returncount** is the number of search results that were actually returned (determined by size).

For all other attribute definitions, refer to Section B.6, "Account Resource."

B.6.15 Check In an Account

Use this API to check in an account.

A checkout can be a *password* checkout or *session* checkout. You can individually check in each checkout by using its **checkoutUID** or you can check in all of the checkouts for
an account. (In this publication, the term "account checkout" generally refers to the latter case.)

---

**Note:** To do a force-check in, you must be an administrator with the User Manager Admin Role.

---

### Sample JSON Representations of Account Check Ins

The following examples illustrate different types of Force Check Ins

- **Example B–47, "Self Check In a Password or Session Checkout"**

```json
{
    "checkoutUID": "9c3c5d687d414a57b7dbda0692c9b06d"
}
```

- **Example B–48, "Force Account Check In (Both Password and Session) for All Users"**

```json
{
    "force": "true"
}
```

- **Example B–49, "Force Account Check In (Both Password and Session) for a Single User"**

```json
{
    "force": "true",
    "userid": "person1"
}
```

- **Example B–50, "Force Check In a Password or Session"**

```json
{
    "force": "true",
    "checkoutUID": "9c3c5d687d414a57b7dbda0692c9b06d"
}
```

---

**URI:**

```
https://opam_server_host:opam_ssl_port/opam/account/{accountUID}/checkin
```

**Method:** PUT

**Content-Type:** application/json

**Body**

**Returns on Success:** Status code 200
**Note:** If you want to perform an account check in (for both password or session), you do not have to provide any content in the JSON body.

Where:

- **force** is a flag that indicates a force check-in. (Default is *false*.)
- **userid** is the user who is to be force-checked in. (Default is to force-check in all users that have checked out the account.)
- **checkoutUID** is the unique identifier for a checkout.

### B.6.16 Verify an Account

Use this API to verify whether the account is present on the target system.

**Example B–51 Sample JSON Representation of Account Addition/Verification**

```json
{
    "account": {
        "accountName": "admin",
        "description": "maintenance account on the machine",
        "password": "welcome1",
        "passwordpolicy": "passwordpolicy2",
        "shared": "true",
        "targetUID": "9bcbcb087174ad1900ea691a2573b61"
    }
}
```

Where:

- **account** is the account JSON object.
- **accountName** is the name of the account.
- **description** is a description of the account. This attribute is *optional*.
- **password** is the account password. This attribute is *optional*.
- **passwordpolicy** is the policy ID of the Password Policy applicable to the account. This parameter is *optional*. By default, this parameters uses the global Default Password Policy.
- **shared** indicates the shared status of the account. This value is a Boolean and the default setting is *false*.
- **targetUID** is the target's unique identifier.

### B.6.17 Update an Account

Use this API to update an account. You can change multiple attributes at a time. Only passwordpolicy, description, and shared attributes can be updated.

**URI**

```
https://opam_server_host:opam_ssl_port/opam/account/test
```

**Method**

PUT

**Content-Type**

application/json

**Body**

JSON representation for account addition/verification

**Returns on Success**

Status code 200
Note: You must be an administrator with the Security Administrator Admin Role to use this API.

---

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td>JSON representation of account modifications</td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>

**Example B–52 Sample JSON Representation of Account Modifications**

```json
{
    "modifications": [
        {
            "modification": {
                "passwordpolicy": "passwordpolicy2"
            }
        },
        {
            "modification": {
                "shared": "false"
            }
        }
    ]
}
```

Where:
- **accountUID** is the account’s unique identifier.
- **modifications** are an array of modification JSON objects.
- **modification** is a JSON object representing the modification of a single attribute.

**B.6.18 Remove an Account**

Use this API to remove an account.

Note: You must be an administrator with the Security Administrator Admin Role to use this API.

---

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>DELETE</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>

Where:
- **accountUID** is the account’s unique identifier.
B.6.19 Remove a User’s/Role’s Access to an Account

Use this API to remove a user’s access or a role’s access to an account. You can revoke multiple user and role grants at a time.

**Note:** You must be an administrator with the User Manager Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/[accountUID]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td>JSON representation for removing grantees</td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>

**Example B–53 Sample JSON Representation for Removing Grantees**

```json
{
   "modifications": [
      {
         "modification": {
            "usagepolicy": "usagepolicy1",
            "role": "opamgroup1",
            "operation": "delete"
         }
      },
      {
         "modification": {
            "usagepolicy": "usagepolicy1",
            "user": "opamuser1",
            "operation": "delete"
         }
      }
   ]
}
```

Where:

- **accountUID** is the account’s unique identifier.
- **modifications** are an array of modification JSON objects.
- **modification** is a JSON object representing a single modification.
- **role** indicates that a group has to be granted an access. This parameter value is the group name.
- **user** indicates that a user has to be granted an access. This parameter value is the user login id.
- **usagepolicy** indicates the Usage Policy identifier to be applied to the grant.
- **operation** indicates the type of operation to be performed. Acceptable values include:
  - **add** indicates a grant.
  - **delete** indicates a revocation.
  - **replace** indicates the replacement of the usagepolicy with a new value.
B.6.20 Show Password

Use this API to retrieve and display the password associated with an account.

**Note:** You must be an administrator with the Security Administrator Admin Role or you must have checked out the account to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}/showpassword</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of account token</td>
</tr>
</tbody>
</table>

**Example B–54  Sample JSON Representation of Account Token**

```json
{
    "accountToken": {
        "accountName": "admin",
        "accountUID": "3f74a85e64432ba917a2e50fa15aa",
        "accountPassword": "GJN8p2ol"
    }
}
```

Where:
- **accountUID** is the account’s unique identifier.
- **accountName** is the name of the account.
- **accountPassword** is the account password.

B.6.21 Show Password (*Deprecated*)

**Note:** This API has been deprecated. Oracle recommends that you use the Show Password API in Section B.6.20, "Show Password."

Use this API to retrieve and display the password associated with an account.

**Note:** You must be an administrator with the Security Administrator Admin Role or you must have checked out the account to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}/showpassword</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of account token</td>
</tr>
</tbody>
</table>

**Example B–55  Sample JSON Representation of Account Token**

```json
{
}
```
"accountToken":{
    "accountName":"admin",
    "accountUID":"3f74a85e39e64432ba917a2e60fa15aa",
    "accountPassword":"GJN8p2ol"
}

Where:

- **accountUID** is the account's unique identifier.
- **accountName** is the name of the account.
- **accountPassword** is the account password.

### B.6.22 Show Password History

Use this API to retrieve and display the password history associated with an account.

**Note:** You must be an administrator with the Security Administrator Admin Role or you must have checked out the account to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}/showpasswordhistory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of account token</td>
</tr>
</tbody>
</table>

**Example B–56 Sample JSON Representation of Account Token**

```json
{
    "accountName": "opamuser1",
    "accountUID": "c1b054ed0f984e27bd68b8c28b985801",
    "passwordHistory": [
        {
            "accountPassword": "M7aGfNOR",
            "modificationTime": "1382996686"
        },
        {
            "accountPassword": "Dr3z5AGa",
            "modificationTime": "1382996412"
        }
    ]
}
```

Where:

- **accountUID** is the account's unique identifier.
- **accountName** is the name of the account.
- **passwordHistory** is the account password history.
- **accountPassword** is the account password.
- **modificationTime** is the time (in UTC seconds) when the password was modified.
B.6.23 Show Password History (*Deprecated*)

**Note:** This API has been deprecated. Oracle recommends that you use the Show Password History API in Section B.6.22, "Show Password History."

Use this API to retrieve and display the password history associated with an account.

**Note:** You must be an administrator with the Security Administrator Admin Role or you must have checked out the account to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}/showpasswordhistory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of account token</td>
</tr>
</tbody>
</table>

**Example B–57 Sample JSON Representation of Account Token**

```json
{
  "accountName":"admin",
  "accountUID":"3f74a85e39e64432ba917a2e60fa15aa",
  "passwordHistory": [
    {
      "accountPassword": "Ud2fykRx",
      "modificationTime": "2013-01-27 19:36:32.952"
    },
    {
      "accountPassword": "jgs21Z8w",
      "modificationTime": "2013-01-27 19:37:02.449"
    },
    {
      "accountPassword": "I3jDRaZb",
      "modificationTime": "2013-01-27 19:37:19.488"
    },
    {
      "accountPassword": "5VfKaYZT",
      "modificationTime": "2013-01-28 00:22:37.331"
    }
  ]
}
```

Where:
- `accountUID` is the account's unique identifier.
- `accountName` is the name of the account.
- `passwordHistory` is the account password history.
- `accountPassword` is the account password.
- `modificationTime` is the time when the password was modified.

B.6.24 Reset Password

Use this API to reset the password on the account.
Note: You must be an administrator with the Security Administrator Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/account/{accountUID}/resetpassword</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td>JSON representation of the new password</td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>

**Example B–58 Sample JSON Representation of the New Password**

```json
{
  "password":"welcome1"
}
```

Or,

```json
{
  "autogen":"true"
}
```

Where:
- **accountUID** is the account's unique identifier.
- **password** is the password assigned to the account.
- **autogen** is the a flag that controls whether to generate a password automatically or not. (Default is `false`.)

### B.7 UI Resource

The APIs described in this section include:
- **Search Accounts (Deprecated)**
- **Search Assigned Accounts (Deprecated)**
- **Get All Checked Out Accounts (Deprecated)**

#### B.7.1 Search Accounts *(Deprecated)*

**Note:** This API has been deprecated. Oracle recommends that you use the Search Accounts API in Section B.6, "Account Resource."

Use this API to search accounts using one or more of the following search request parameters:
- **type**
- **domain**
- **description**
- **name**
- **accountname**
All of these parameters are *optional*.

**Note:** You must be an administrator with the *User Manager* Admin Role or the *Security Administrator* Admin Role to use this API.

---

**Example B–59 Sample JSON Representation of Account Collection**

```json
{
    "AccountCollection" : [
        {
            "account" : {
                "shared" : false,
                "targetUID" : "eadd96486e9a47b79bd23cf11e67bd2b2",
                "domain" : "needtofix",
                "targetName" : "sunds_6.3_target",
                "targetType" : "ldap",
                "accountlevelstatus" : "checkedIn",
                "description" : ",",
                "accountName" : "dsperson1",
                "uri" : "https://localhost:7002/opam/account/35e2709edf043edae8f67727d937bec",
                "accountUID" : "35e2709edf043edae8f67727d937bec"
            }
        },
        {
            "account" : {
                "shared" : false,
                "targetUID" : "eadd96486e9a47b79bd23cf11e67bd2b2",
                "domain" : "needtofix",
                "targetName" : "sunds_6.3_target",
                "targetType" : "ldap",
                "accountlevelstatus" : "checkedIn",
                "description" : ",",
                "accountName" : "dsperson10",
                "uri" : "https://localhost:7002/opam/account/0a1ee2cb17e345cd537a2f05e11e93c",
                "accountUID" : "0a1ee2cb17e345cd537a2f05e11e93c"
            }
        }
    ],
    "count" : 2
}
```

For all other attribute definitions, refer to *Section B.5, "Target Resource"* and *Section B.6, "Account Resource."*
B.7.2 Search Assigned Accounts (Deprecated)

Note: This API has been deprecated. Oracle recommends that you use the Search Assigned Accounts API in Section B.6, "Account Resource."

Use this API to search assigned accounts using one or more of the following search request parameters:

- `type`
- `domain`
- `description`
- `name`
- `accountname`

All of these parameters are optional.

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of account collection</td>
</tr>
</tbody>
</table>

Example B–60 Sample JSON Representation of Account Collection

```json
{
  "AccountCollection" : [
    {
      "account" : {
        "status" : "checkedIn",
        "shared" : false,
        "targetUID" : "b7af920f673149f5b0f66da28fd8253",
        "domain" : "needtofix",
        "targetName" : "ldap1_target",
        "targetType" : "ldap",
        "accountlevelstatus" : "checkedIn",
        "description" : "",
        "accountName" : "person1",
        "uri" : "https://localhost:7002/opam/account/0d755f646bcf4fa08ca515ed3829aadf",
        "accountUID" : "0d755f646bcf4fa08ca515ed3829aadf"
      }
    },
    {
      "account" : {
        "status" : "checkedIn",
        "shared" : false,
        "targetUID" : "b7af920f673149f5b0f66da28fd8253",
        "domain" : "needtofix",
        "targetName" : "ldap1_target",
        "targetType" : "ldap",
        "accountlevelstatus" : "checkedIn",
        "description" : "",
        "accountName" : "person1",
        "uri" : "https://localhost:7002/opam/account/0d755f646bcf4fa08ca515ed3829aadf",
        "accountUID" : "0d755f646bcf4fa08ca515ed3829aadf"
      }
    }
  ]
}``

B-62  Administrator's Guide for Oracle Privileged Account Manager
"accountName": "person2",
"uri": "https://localhost:7002/opam/account/62c684c3821f4e11879oe815ee881e02",
"accountUID": "62c684c3821f4e11879oe815ee881e02"
}
],
"count": 2
}

Where:

- **status** indicates whether the requesting user has checked out the account or not.

For all other attribute definitions, refer to Section B.5, "Target Resource" and Section B.6, "Account Resource."

### B.7.3 Get All Checked Out Accounts (Deprecated)

**Note:** This API has been deprecated. Oracle recommends that you use the Get All Checked Out Accounts API in Section B.6, "Account Resource."

Use this API to retrieve a list of all accounts that have been checked out by the logged in user.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/ui/allaccounts/mycheckedout</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of account collection</td>
</tr>
</tbody>
</table>

#### Example B–61 Sample JSON Representation of Account Collection

```
{
  "AccountCollection": [
    {
      "account": {
        "uri": "https:\/\opam_server_host:opam_ssl_port\opam\account\3740553e999a4f6aa8e8f9286d320cb4",
        "accountUID": "3740553e999a4f6aa8e8f9286d320cb4",
        "accountName": "sherlock",
        "status": "checkedOut",
        "targetUID": "9bbcbdb087174ad1900ea691a2573b61",
        "targetName": "ldap1-target",
        "targetType": "ldap",
        "domain": "berkeley",
        "expiryTime": 1338765551,
      },
      "count": 1
    }
  ]
}
```

For attribute definitions, refer to Section B.5, "Target Resource" and Section B.6, "Account Resource."
B.8 User Resource

The APIs described in this section include:

- Get a User
- Get All Accounts Granted to a User
- Search Users from Identity Store
- Search for Assigned Users

B.8.1 Get a User

Use this API to retrieve a user.

---

**Note:** You must be an administrator with the User Manager Admin Role or the Security Administrator Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/user/{uid}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of user</td>
</tr>
</tbody>
</table>

**Example B–62 Sample JSON Representation of User**

```json
{
    "user": {
        "uid": "opamuser1",
        "lastname": "opamuser1",
        "usertype": "End-User",
        "opamrole": [
        ]
    },
    "dn": "uid=opamuser1,ou=people,ou=myrealm,dc=base_domain",
}
```

Where:

- **uid** is the login ID of the user.
- **lastname** is the last name of the user.
- **firstname** is the first name of the user.
- **dn** is the distinguished name of the user.
- **usertype** indicates whether the user has an Administrative Role.
- **opamrole** is the user's Admin Role.

B.8.2 Get All Accounts Granted to a User

Use this API to retrieve all of the accounts granted to a user.
Example B–63  Sample JSON Representation of Accounts Collection

```json
{
  "accounts": [
    {
      "account": {
        "accountUID": "16d245784350469cbe25229a7c45af22",
        "accountName": "oidperson10",
        "targetID": "75a23e9f30ba456b961a1f5d327e67ef",
        "targetName": "ldap1_target",
        "targetDomain": "needtofix",
        "targetType": "ldap"
      }
    },
    {
      "account": {
        "accountUID": "47671a7a4ebc44c496888aac5423dad1",
        "accountName": "oudperson11",
        "targetID": "488d6d656b2c4b96a5fd835c131b4c00",
        "targetName": "ldap_11.115_target",
        "targetDomain": "needtofix",
        "targetType": "ldap"
      }
    }
  ]
}
```

For attribute definitions, refer to Section B.5, "Target Resource" and Section B.6, "Account Resource."

### B.8.3 Search Users from Identity Store

Use this API to search for users. This API searches for the `searchKeyWord` in firstname, lastname, uid, and mail of the user.

**Note:** You must be an administrator with the User Manager Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/user/search/[searchKeyWord]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
</tbody>
</table>
Example B–64 Sample JSON Representation of Users

```
{
  "users": [
    {
      "user": {
        "uid": "opamenduser1",
        "firstname": "opamenduser1",
        "lastname": "opamenduser1",
        "dn": "uid=opamenduser1,ou=people,ou=myrealm,dc=base_domain"
      }
    },
    {
      "user": {
        "uid": "opamenduser2",
        "lastname": "opamenduser2",
        "dn": "uid=opamenduser2,ou=people,ou=myrealm,dc=base_domain"
      }
    },
    {
      "user": {
        "uid": "opamuser1",
        "lastname": "opamuser1",
        "dn": "uid=opamuser1,ou=people,ou=myrealm,dc=base_domain"
      }
    }
  ]
}
```

For attribute definitions, refer to Section B.8.1, "Get a User."

### B.8.4 Search for Assigned Users

Use this API to search for users. This API contains a search with the uid parameter. The uid parameter is **optional**.

**Note:** You must be an administrator with the User Manager Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/user/advancedsearch?param1=val1&amp;param2=val2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status 200 and JSON representation of users</td>
</tr>
</tbody>
</table>

Example B–65 Sample JSON Representation of Users

```
[
  
]
"users": [  
  {  
    "user": {  
      "uid": "OracleSystemUser",  
      "lastname": "OracleSystemUser",  
      "dn": "uid=OracleSystemUser,ou=people,ou=myrealm,dc=base_domain"  
    }  
  },  
  {  
    "user": {  
      "uid": "weblogic",  
      "dn": "uid=weblogic,ou=people,ou=myrealm,dc=base_domain"  
    }  
  },  
  {  
    "user": {  
      "uid": "app_config",  
      "lastname": "app_config",  
      "dn": "uid=app_config,ou=people,ou=myrealm,dc=base_domain"  
    }  
  },  
  {  
    "user": {  
      "uid": "sec_admin",  
      "lastname": "sec_admin",  
      "dn": "uid=sec_admin,ou=people,ou=myrealm,dc=base_domain"  
    }  
  },  
  {  
    "user": {  
      "uid": "user_manager",  
      "lastname": "user_manager",  
      "dn": "uid=user_manager,ou=people,ou=myrealm,dc=base_domain"  
    }  
  },  
  {  
    "user": {  
      "uid": "sec_auditor",  
      "lastname": "sec_auditor",  
      "dn": "uid=sec_auditor,ou=people,ou=myrealm,dc=base_domain"  
    }  
  },  
  {  
    "user": {  
      "uid": "opamenduser1",  
      "firstname": "opamenduser1",  
      "lastname": "opamenduser1",  
      "dn": "uid=opamenduser1,ou=people,ou=myrealm,dc=base_domain"  
    }  
  },  
  {  
    "user": {  
      "uid": "opamenduser2",  
      "lastname": "opamenduser2",  
      "dn": "uid=opamenduser2,ou=people,ou=myrealm,dc=base_domain"  
    }  
  },  
  {  
    "user": {  
      "uid": "opamuser1",  
      "lastname": "opamuser1",  
      "dn": "uid=opamuser1,ou=people,ou=myrealm,dc=base_domain"  
    }  
  }]

For attribute definitions, refer to Section B.8.1, "Get a User."

B.9 Group Resource

The APIs described in this section include:

- Get Group
- Get Member Users of a Group
- Get Member Groups of a Group
- Get All Accounts Granted to a Group
- Search Groups from Identity Store
- Advanced Search for Assigned Groups

B.9.1 Get Group

Use this API to retrieve a group.

**Note:** You must be an administrator with the **User Manager Admin** Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/group/{name}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of group</td>
</tr>
</tbody>
</table>

**Example B–66  Sample JSON Representation of Group**

```json
{
    "group":{
        "name": "opamgroup1",
        "dn": "cn=opamgroup1,ou=groups,ou=myrealm,dc=base_domain",
        "description": ""
    }
}
```

Where:

- **name** is the name of the group.
- **dn** is the distinguished name of the group.
- **description** is a description of the group.
B.9.2 Get Member Users of a Group

Use this API to retrieve the user members of a group.

Note: You must be an administrator with the User Manager Admin Role to use this API.

Example B–67  Sample JSON Representation of User Collection

```json
{
    "users": [
        {
            "user": {
                "uid": "master_user",
                "lastname": "master_user",
                "dn": "uid=master_user,ou=people,ou=myrealm,dc=base_domain"
            }
        },
        {
            "user": {
                "uid": "sec_admin",
                "lastname": "sec_admin",
                "dn": "uid=sec_admin,ou=people,ou=myrealm,dc=base_domain"
            }
        }
    ]
}
```

For attribute definitions, refer to Section B.8.1, "Get a User."

B.9.3 Get Member Groups of a Group

Use this API to retrieve the group members of a group.

Note: You must be an administrator with the User Manager Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/group/{name}/users</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of user collection</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/group/{name}/groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of group collection</td>
</tr>
</tbody>
</table>
Example B–68  Sample JSON Representation of Group Collection

```
{
  "groups": [
    {
      "group": {
        "name": "CrossDomainConnectors",
        "description": "CrossDomainConnectors can make inter-domain calls from foreign domains."
      }
    },
    {
      "group": {
        "name": "Deployers",
        "description": "Deployers can view all resource attributes and deploy applications."
      }
    }
  ]
}
```

For attribute definitions, refer to Section B.9.1, "Get Group."

B.9.4 Get All Accounts Granted to a Group

Use this API to retrieve the all of the accounts granted to a group.

**Note:** You must be an administrator with the User Manager Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/group/[name]/accounts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of accounts collection</td>
</tr>
</tbody>
</table>

Example B–69  Sample JSON Representation of Accounts Collection

```
{
  "accounts": [
    {
      "account": {
        "accountUID": "16d245784350469cbe25229a7c45af22",
        "accountName": "oidperson10",
        "targetID": "75a23e9f30ba456b961a1f5d327e67ef",
        "targetName": "ldap1_target",
        "targetDomain": "needtofix",
        "targetType": "ldap"
      }
    },
    {
      "account": {
        "accountUID": "47671a7a4ebc44c496888aac5423dad1",
        "accountName": "oudperson11",
        "targetID": "488d6d656b2c4b96a5fd835c131b4c00",
        "targetName": "oud_11.115_target",
        "targetDomain": "needtofix"
      }
    }
  ]
}
```
For attribute definitions, refer to Section B.5, "Target Resource" and Section B.6, "Account Resource."

B.9.5 Search Groups from Identity Store

Use this API to search for groups. This API searches for the searchKeyWord in the group names.

**Note:** You must be an administrator with the User Manager Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/group/search/{searchKeyWord}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status 200 and JSON representation of groups</td>
</tr>
</tbody>
</table>

**Example B–70 Sample JSON Representation of Groups**

```
{  
  "groups": [  
    {  
      "group": {  
        "name": "opamgroup1",  
        "description": "",  
      }  
    },  
    {  
      "group": {  
        "name": "opamgroup2",  
        "description": "",  
      }  
    },  
    {  
      "group": {  
        "name": "opamsubgroup1",  
        "description": "",  
      }  
    },  
    {  
      "group": {  
        "name": "opamsubgroup2",  
        "description": "",  
      }  
    },  
    {  
      "group": {  
        "name": "OPAM_APPLICATION_CONFIGURATOR",  
        "description": "OPAM_APPLICATION_CONFIGURATOR",  
      }  
    }  
  ]
```
For attribute definitions, refer to Section B.9.1, "Get Group."

B.9.6 Advanced Search for Assigned Groups

Use this API to search for groups who have been assigned an account. The request parameter is groupname, which is optional.

---

**Note:** You must be an administrator with the User Manager Admin Role to use this API.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/group/advancedsearch?param1=val1&amp;param2=val2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status 200 and JSON representation of groups</td>
</tr>
</tbody>
</table>

---

**Example B–71**  Sample JSON Representation of Groups

```json
{
    "groups": [
        {
            "group": {
                "name": "opamgroup1",
                "description": "",
            }
        },
        {
            "group": {
                "name": "opamgroup2",
                "description": "",
            }
        }
    ]
}
```
For attribute definitions, refer to Section B.9.1, "Get Group."

B.10 Plug-In Resource

The APIs described in this section include:

- Add Plug-In Configuration
- Verify Plug-In Configuration
- Search For Plug-In Configuration
- Retrieve Plug-In Configuration
- Update Plug-In Configuration
- Remove Plug-In Configuration

B.10.1 Add Plug-In Configuration

Use this API to add a plug-in configuration.
### Example B–72  Sample JSON Representation of Plug-In Configuration Creation

```
{
  "plugin": {
    "pluginName": "sampleplugin",
    "pluginDescription": "Sample Plugin",
    "pluginEnabled": "true",
    "pluginResource": "account",
    "pluginOperation": "checkout",
    "pluginTiming": "post",
    "pluginOrder": 10,
    "pluginClassName": "EmailNotifyPlugin",
    "pluginClassPath": "/u01/plugins/emailplugin.jar",
    "pluginEnableGroup": ["hrgroup", "itgroup"],
    "pluginEnableUser": ["admin"],
    "pluginEnableResult": 200,
    "pluginVersion": "1.0.0",
    "pluginCustomAttrs": [
      {
        "pluginCustomAttr": {
          "attrname": "notificationemail",
          "attrvalue": "abc@abc.com"
        }
      }
    ]
  }
}
```

### Sample Output

https://opam_server_host:opam_ssl_port/opam/plugin/9bbcbb087174ad1900ea691a2573b61

### B.10.2 Verify Plug-In Configuration

Use this API to validate a plug-in configuration, which includes

- Testing the uniqueness of the `pluginName`
- Testing the uniqueness of the `pluginResource`, `pluginOperation`, `pluginOrder` combination
- Validating attributes and allowed values
- Validating the loading of `pluginClassName` using the `pluginClassPath`

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/plugin/test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
</tbody>
</table>

Example B–73 Sample JSON Representation of Plug-In Configuration for Verification

```
{
  "plugin": {
    "pluginUID": "9bbcbbb087174ad1900ea691a2573b61",
    "pluginName": "sampleplugin",
    "pluginDescription": "Sample Plugin",
    "pluginEnabled": "true",
    "pluginResource": "account",
    "pluginOperation": "checkout",
    "pluginTiming": "post",
    "pluginOrder": "10",
    "pluginClassName": "EmailNotifyPlugin",
    "pluginClassPath": "/u01/plugins/emailplugin.jar",
    "pluginEnableGroup": ["hrgroup", "itgroup"],
    "pluginEnableUser": ["admin"],
    "pluginEnableResult": "200",
    "pluginVersion": "1.0.0",
    "pluginCustomAttrs": [
      {
        "pluginCustomAttr": {
          "attrname": "notificationemail",
          "attrvalue": "abc@abc.com"
        }
      }
    ]
  }
}
```

B.10.3 Search For Plug-In Configuration

Use this API, with any of the following parameters, to search for plug-in configurations:

- Name
- Description
- Enabled
- Resource
- Operation
- Timing

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/plugin/search?param1=value1&amp;param2=value2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
</tbody>
</table>
Returns on Success | Status code 200 and JSON representation of plug-in collection

Sample URI
https://opam_server_host:opam_ssl_port/opam/plugin
/search?name=email&enabled=true&timing=post

Example B–74  Sample JSON Representation of Plug-In Collection

```json
{"pluginCollection": [  
  "plugin": {  
    "pluginUID": "9bbcbbb087174ad1900ea691a2573b61"  
    "pluginDescription": "Sample Plugin"  
    "pluginName": "sampleplugin"  
    "pluginEnabled": true  
    "pluginResource": "account"  
    "pluginOperation": "checkout"  
    "pluginTiming": "post"  
    "pluginOrder": 10  
    "pluginClassName": "EmailNotifyPlugin"  
    "pluginClassPath": "/u01/plugins/emailplugin.jar"  
    "pluginEnableGroup": ["hrgroup", "itgroup"]  
    "pluginEnableUser": ["admin"]  
    "pluginEnableResult": 200  
    "pluginVersion": "1.0.0"  
    "pluginCustomAttrs": [  
      {  
        "pluginCustomAttr": {  
          "attrname": "notificationemail"  
          "attrvalue": "abc@abc.com"  
        }  
      }  
    ]  
  }  
]}
```

B.10.4 Retrieve Plug-In Configuration

Use this API to retrieve a plug-in configuration.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/plugin/plugin/{pluginUID}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>GET</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200 and JSON representation of a plug-in</td>
</tr>
</tbody>
</table>

Example B–75  Sample JSON Representation of Plug-In

```json
{"plugin": {  
  "pluginUID": "9bbcbbb087174ad1900ea691a2573b61"  
  "pluginName": "sampleplugin"  
  "pluginDescription": "Sample Plugin"  
  "pluginEnabled": true
```

---

B-76  Administrator’s Guide for Oracle Privileged Account Manager
B.10.5 Update Plug-In Configuration

Use this API to update a plug-in configuration.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/plugin/{pluginUID}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>PUT</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td>JSON representation of a plug-in modification</td>
</tr>
<tr>
<td>Returns on Success</td>
<td>Status code 200</td>
</tr>
</tbody>
</table>

Example B–76  Sample JSON Representation to Modify Plug-In

```json
{
  "modifications": [
    {
      "modification": {
        "pluginEnabled": "false",
        "pluginVersion": "1.0.1"
      }
    }
  ]
}
```

B.10.6 Remove Plug-In Configuration

Use this API to delete a plug-in configuration.

<table>
<thead>
<tr>
<th>URI</th>
<th>https://opam_server_host:opam_ssl_port/opam/plugin/{pluginUID}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>DELETE</td>
</tr>
<tr>
<td>Content-Type</td>
<td>application/json</td>
</tr>
<tr>
<td>Body</td>
<td></td>
</tr>
</tbody>
</table>

| **Returns on Success** | Status code 200 |
This appendix describes common problems that you might encounter when using Oracle Privileged Account Manager and explains how to solve them.

This appendix includes the following sections:

- Section C.1, "Introduction to Troubleshooting Oracle Privileged Account Manager"
- Section C.2, "Getting Started with Troubleshooting and Logging Basics for Oracle Privileged Account Manager"
- Section C.3, "Resolving Common Problems and Solutions"
- Section C.4, "Using My Oracle Support for Additional Troubleshooting Information"

In addition to this appendix, review the Oracle Fusion Middleware Error Messages Reference for information about the error messages you may encounter.

C.1 Introduction to Troubleshooting Oracle Privileged Account Manager

This section provides guidelines and a process for using the information in this chapter. Using the following guidelines and process will focus and minimize the time you spend resolving problems.

Guidelines

When using the information in this chapter, Oracle recommends:

- After performing any of the solution procedures in this chapter, immediately retrying the failed task that led you to this troubleshooting information. If the task still fails when you retry it, perform a different solution procedure in this chapter and then try the failed task again. Repeat this process until you resolve the problem.

- Making notes about the solution procedures you perform, symptoms you see, and data you collect while troubleshooting. If you cannot resolve the problem using the information in this chapter and you must log a service request, the notes you make will expedite the process of solving the problem.

Process

Follow the process outlined in Table C–1 when using the information in this chapter. If the information in a particular section does not resolve your problem, proceed to the next step in this process.
This section provides information about how to diagnose Oracle Privileged Account Manager problems. The topics include:

- Section C.2.1, "Increasing the Log Level"
- Section C.2.2, "Examining Exceptions in the Logs"

C.2.1 Increasing the Log Level

When an Oracle Privileged Account Manager error occurs, you can gather more information about what caused the error by generating complete logs that include debug information and connector logging. The following steps:

1. Set the Oracle Privileged Account Manager logging level to the finest level, which is `TRACE:32`.

   **Note:**
   - For more information about Oracle Privileged Account Manager logging, refer to Chapter 14, "Managing Oracle Privileged Account Manager Auditing and Logging."
   - For more information about setting logging levels, refer to "Implementing Java and Oracle Logging" in the Oracle Containers for J2EE Developer’s Guide.

2. Repeat the task or procedure where you originally encountered the error.
3. Examine the log information generated using the DEBUG level.

C.2.2 Examining Exceptions in the Logs

Examining the exceptions logged to the Oracle Privileged Account Manager log file can help you identify various problems.
You can access Oracle Privileged Account Manager’s diagnostic log in the following directories:

- `DOMAIN_HOME/servers/Adminserver/logs`
- `DOMAIN_HOME/servers/opamsserver/logs`

## C.3 Resolving Common Problems and Solutions

This section describes common problems and solutions. The topics include:

- Section C.3.1, "Console Cannot Connect to Oracle Privileged Account Manager Server"
- Section C.3.2, "Console Changes Are Not Reflected in Other, Open Pages"
- Section C.3.3, "Cannot Access Targets or Accounts"
- Section C.3.4, "Cannot Add Database Targets"
- Section C.3.5, "Cannot Add an Active Directory LDAP Target"
- Section C.3.6, "Grantee Cannot Perform a Checkout"
- Section C.3.7, "Cannot View Users or Roles from the Configured Remote Identity Store"
- Section C.3.8, "Group Membership Changes Are Not Immediately Reflected in Oracle Privileged Account Manager"
- Section C.3.9, "Cannot Use Larger Key Sizes for Export/Import"
- Section C.3.10, "Oracle Privileged Account Manager End Users Gain Privileges They Were Not Explicitly Granted"
- Section C.3.11, "Cannot Access MSSQL Server Targets and Accounts"
- Section C.3.12, "Troubleshooting Issues with Using Oracle Database TDE"
- Section C.3.13, "Cannot Open Session Recordings"
- Section C.3.14, "Session Checkout Does Not Work, Even After Granting the Account"
- Section C.3.15, "OPAM Console Login Does Not Work in Internet Explorer 11 Browser"

### C.3.1 Console Cannot Connect to Oracle Privileged Account Manager Server

Oracle Privileged Account Manager Console cannot connect to the Oracle Privileged Account Manager server.

**Cause**

If the Console cannot connect to the Oracle Privileged Account Manager server, then you might have a configuration problem with the Console or with Oracle Platform Security Services Trust.

**Solution**

To resolve this problem:

1. Verify that your host and port information is correct.
2. Confirm that the generated URL displayed on the Console is responsive.
3. Ensure that you correctly completed all of the configuration steps described in "Post-Installation Tasks" in the Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management.

---

**Note:** If you have configured a high availability instance, ensure that you correctly completed all of the Oracle Privileged Account Manager configuration steps described in the Oracle Fusion Middleware High Availability Guide.

---

### C.3.2 Console Changes Are Not Reflected in Other, Open Pages

When you have multiple browser windows or Console tabs open against the same Oracle Privileged Account Manager Console, updates made in one window or tab are not immediately reflected in the other windows or tabs.

**Cause**
The Oracle Privileged Account Manager Console does not proactively push updates to the browser.

**Solution**
To resolve this problem, refresh the browser window or tab.

### C.3.3 Cannot Access Targets or Accounts

Your attempts to access targets and privileged accounts are failing. You cannot check out, check-in, or test.

**Cause 1**
The ICF connector being used by Oracle Privileged Account Manager is having issues interacting with the target system.

**Solution 1**
To resolve this problem:

1. Verify that the target system is up, and that the privileged account of interest exists.
2. Increase Oracle Privileged Account Manager’s logging level to **TRACE:32** (its finest level) and review the trace logs to determine where the failure occurs.
   
   Problems are often caused by environmental issues that can be identified using the trace logs and remedied by fixing the configuration on the target system. Refer to Chapter 14, "Managing Oracle Privileged Account Manager Auditing and Logging" for more information.
3. You might have a connector issue. Submit a bug that includes a reproducible test case, target system details, and trace logs.

**Cause 2**
A user changed the target’s service account password out of band from Oracle Privileged Account Manager. For example, if the user changed the password by using the DB host or by using a different Oracle Privileged Account Manager instance in a different domain, the Show Password feature for the original Oracle Privileged Account Manager server does not reflect that change and any attempt to connect to that target will fail.
Solution 2
To resolve this problem, update the new password by editing the target through the Oracle Privileged Account Manager Console or the command line. Refer to Section 8.8, “Managing Privileged Account Passwords” or to Section A.5.8, "resetpassword Command" for more information.

C.3.4 Cannot Add Database Targets
This section describes issues that can prevent you from adding database targets:

■ Cannot Connect to Oracle Database with sysdba Role
■ Cannot Find Special Options for Adding a Database Target

C.3.4.1 Cannot Connect to Oracle Database with sysdba Role
Your attempts to connect to Oracle Database using the sysdba role are failing with the following error message:

Invalid Connection Details, see server log for details.

Cause
To connect to Oracle Database as a user with sysdba role, you must configure the Advanced Properties option with the value, internal_logon=sysdba.

You must also specify this setting for the Oracle Database SYS account, which must connect with the sysdba role. The Oracle Database SYS user is a special account and if you do not use this role, then the connection might fail. However, it is a better practice to create a service account instead of using SYS.

Solution
To resolve this problem:

1. Connect to Oracle Database as a user with the sysdba role.

   Note: These configuration steps are not necessary if you are connecting as a normal user.

2. Open the target’s General tab and expand Advanced Configuration to view the configuration options.

3. Enter the internal_logon=sysdba value into the Connection Properties field.

4. Click Test to retest the connection.

5. Save your changes.

C.3.4.2 Cannot Find Special Options for Adding a Database Target
You cannot find configuration options for connecting to database targets such as Oracle RAC Database or for using Secure Socket Layer (SSL).

Cause
Oracle Privileged Account Manager uses a Generic Database connector where special configuration options for specific database target systems are not exposed in a clean or intuitive manner.
Solution
To resolve this problem, define special connectivity options for database targets by modifying the **Database Connection URL** and **Connection Properties** parameter values.

---

**Note:**
- Refer to Section 6.2, "Adding Targets to Oracle Privileged Account Manager" for information about these parameters.
- Refer to the Oracle Identity Manager Connector Guide for Database User Management for information about which special options are supported.

---

C.3.5 Cannot Add an Active Directory LDAP Target

An LDAP target using Microsoft Active Directory fails when you test the connection, search for accounts, or check out passwords.

**Cause**
Active Directory defaults require specific configuration, so you must change the generic default values for the LDAP target. Oracle Privileged Account Manager uses a Generic LDAP connector where special or custom configuration options for specific LDAP target systems are not obvious. (Usually, only Active Directory LDAP targets cause issues.)

**Solution**
To resolve this problem, ensure the following when you add an LDAP target:

1. Use SSL to communicate with Active Directory.
   - Import the SSL certificates into the WebLogic instance running Oracle Privileged Account Manager. Refer to Section 15.1, "Configuring Oracle Privileged Account Manager to Communicate With Target Systems Over SSL" for more information.
   - From the Targets page, set the **TCP Port** to your Active Directory SSL port and enable the **SSL** checkbox.
2. Specify the following **Advanced Configuration** parameters:
   - Set **Password Attribute** to `unicodepwd`
   - Set **Advanced Configuration > Account Object Classes** to `top|person|organizationalPerson|user`.
3. Specify an attribute that is suitable for data in Active Directory, such as `uid` or `samaccountname`, for the **Account User Name Attribute**, **Uid Attribute**, and **LDAP Filter for Retrieving Accounts** configuration parameters.

---

**Note:** For more information about setting any of the following parameters, refer to Section 6.2.2, "ldap Target Type Parameters."

---

C.3.6 Grantee Cannot Perform a Checkout

A grantee’s attempt to checkout an account is failing with an **Insufficient Privileges** error.
Cause
The username is case-sensitive for Oracle Privileged Account Manager grants, but not always for WebLogic authentication.

Solution
To resolve this problem, be sure to enable the *Use Retrieved User Name As Principal* option for the authenticator being used for your production identity store. Refer to Section 3.3.2, "Configuring an External Identity Store for Oracle Privileged Account Manager" for more information.

C.3.7 Cannot View Users or Roles from the Configured Remote Identity Store

When you try to grant to a user or group, you cannot view all users and roles from the configured remote identity store.

**Cause 1**
The Control flag of the authenticator that corresponds to the identity store containing the user or role is not set to *SUFFICIENT*.

**Cause 2**
The user or role that you are searching for is not present in the first authenticator listed in the providers list.

**Solution**
To resolve this problem:

1. Set the Control flag for all necessary authenticators to *SUFFICIENT*.
2. By default, Oracle Privileged Account Manager searches for users and groups in the first authenticator in the Providers list. However, if you set the *virtualize* property in *jps-config.xml* to *true*, Oracle Privileged Account Manager fetches the entities from all LDAP authenticators. For example,

```xml
<serviceInstance name="idstore.ldap" provider="idstore.ldap.provider">
  <property name="idstore.config.provider" value="oracle.security.jps.wls.internal.idstore.WlsLdapIdStoreConfigProvider"/>
  <property name="CONNECTION_POOL_CLASS" value="oracle.security.idm.providers.stdldap.JNDIPool"/>
  <property name="virtualize" value="true"/>
</serviceInstance>
```

In WebLogic, the *jps-config.xml* file is located in the following location:

```
DOMAIN_HOME/config/fmwconfig
```

C.3.8 Group Membership Changes Are Not Immediately Reflected in Oracle Privileged Account Manager

You have an indirect grant through group membership and updates to that group membership are not immediately reflected in Oracle Privileged Account Manager. For example, if you assign a user to a Oracle Privileged Account Manager administration role or to a group granted with a Oracle Privileged Account Manager privileged account, you may not be able to view these changes right away.
Cause
WebLogic caches group memberships and identity assertions by default. Therefore, changes in the source location will not be reflected in Oracle Privileged Account Manager until the cache entries are recomputed.

Solution
To resolve this problem, modify the caching settings in your WebLogic Authenticator and Asserter configuration to suit your requirements.

Note: For more information, refer to
- "Optimizing the Group Membership Caches" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*
- "Configuring Identity Assertion Performance in the Server Cache" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*

C.3.9 Cannot Use Larger Key Sizes for Export/Import

You are unable to use key sizes larger than 128-bits for export or import operations.

Cause
The default JRE installation does not contain the Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 6.

Solution
To resolve this problem, apply the JCE patch, available for download from [http://www.oracle.com/technetwork/java/javase/downloads/jce-6-download-429243.html](http://www.oracle.com/technetwork/java/javase/downloads/jce-6-download-429243.html)

C.3.10 Oracle Privileged Account Manager End Users Gain Privileges They Were Not Explicitly Granted

An Oracle Privileged Account Manager end user can access all of the groups associated with a user, but was not explicitly granted access to those groups.

Cause 1
You granted an Oracle Privileged Account Manager end user access through an LDAP group that uses multiple values as its naming value.

For example, assume you configured an environment that uses CN as its naming attribute and that it contains two groups, A and B. Group A has only one CN value, cn=GroupA and group B has two CN values, cn=GroupA and cn=GroupB.

The Oracle Privileged Account Manager host container (WebLogic or WebSphere) will assert that actual members of GroupA are members of GroupA. However, the host container will also assert that the actual members of GroupB are also members of GroupA, which means that the members of GroupB will inadvertently get the privileges associated with GroupA.

Cause 2
You used nested group memberships.
If group B is a member of group A, and you grant group A access to an Oracle Privileged Account Manager resource, then you implicitly grant this privilege to group B.

**Solution**
To resolve this problem, you must ensure that group entries in LDAP have only a single value for the naming attribute being used.

**C.3.11 Cannot Access MSSQL Server Targets and Accounts**
Your attempts to access the MSSQL server database target and accounts are failing. You cannot test, check out, or check-in. Following are two reasons why this problem might occur:

**Cause 1**
The MSSQL driver sqljdbc4.jar is missing.

**Cause 2**

**Solution**
To resolve this problem:

1. Ensure MSSQL driver is available for the server as described by the note in Database Type description in Table 6.2.1, "database Target Type Parameters".
2. Use JAVA version 1.6.0_30 or higher to avoid encountering the referenced JAVA bug.

**C.3.12 Troubleshooting Issues with Using Oracle Database TDE**
This section describes issues you might encounter when you are attempting to set-up or to operate Oracle Privileged Account Manager in Oracle Database Transparent Data Encryption (TDE) mode. These issues include:

- **TDE Wallet Errors**
- **The TDE Wallet is Open, but Columns Are Not Encrypted**

**C.3.12.1 TDE Wallet Errors**
After enabling TDE mode, you see one of the following error messages:

- No TDE wallet found
- TDE wallet is closed
- TDE wallet is undefined
- TDE wallet is open but has no master key
- Columns are encrypted but TDE wallet is not open

**Cause**
The expected TDE wallet status is open.
Solution
To resolve a problem with the TDE wallet, refer to “Enabling Transparent Data Encryption” in the Oracle Database Advanced Security Administrator’s Guide.

C.3.12.2 The TDE Wallet is Open, but Columns Are Not Encrypted
After setting up TDE, you notice that the TDE wallet is open, but the columns are not encrypted.

Cause
The secure Oracle Privileged Account Manager columns are not encrypted.

Solution
To resolve this problem, perform the steps described in "Configuring Oracle Privileged Account Manager" of the Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management.

For example:
sqlplus DEV_OPAM/welcome1 @IAM_HOME/opam/sql/opamxencrypt.sql

C.3.13 Cannot Open Session Recordings
This section describes issues you might encounter when you are attempting to view session recording transcripts. These issues include:
- Cannot Access Recordings In Internet Explorer Browser
- Cannot Access Recordings in Any Browser

C.3.13.1 Cannot Access Recordings In Internet Explorer Browser
You used Internet Explorer to log in to the Oracle Privileged Account Manager Console, but when you tried viewing the Recording transcript from an account’s Checkout History page, the following message displayed:

There is a problem with this website's security certificate.

You cannot open the recording even after selecting the Continue to this website (not recommended) option.

Cause
Internet Explorer mandates key sizes that are greater than 1024 bits, but the out-of-the-box DemoCA and certificates that are generated by Oracle WebLogic Server are 512 bits.

Solution
To workaround this issue, you must generate a self-signed certificate with a key size that is greater than 1024 bits. Use the following steps:

1. Generate a self-signed certificate with a key size of 2048 bits.

Note: Refer to "Using the Oracle WebLogic Server Java Utilities" in the Oracle Fusion Middleware Command Reference for Oracle WebLogic Server for more information.
java utils.CertGen -keyfilepass <CAPassword> -certfile <hostname>-cert -keyfile <hostname>-key -cn <fully qualified hostname> -strength 2048 -selfsigned -keyusagecritical false -keyusage digitalSignature,nonRepudiation, keyEncipherment,dataEncipherment,keyAgreement,keyCertSign,cRLSign

For example:
java utils.CertGen -keyfilepass Welcome123 -certfile adc2120745-cert -keyfile adc2120745-key -cn adc2120745.mycompany.com -strength 2048 -selfsigned -keyusagecritical false -keyusage digitalSignature,nonRepudiation, keyEncipherment,dataEncipherment,keyAgreement,keyCertSign,cRLSign

2. Move the key with the demoidentity alias to demoidentityold.
   
cd MW_HOME/wlserver/server/lib

   keytool -list -keystore DemoIdentity.jks -storepass DemoIdentityKeyStorePassPhrase
   keytool -changealias -alias demoidentity -destalias demoidentityold -keypass DemoIdentityPassPhrase -keystore DemoIdentity.jks -storepass DemoIdentityKeyStorePassPhrase
   keytool -list -keystore DemoIdentity.jks -storepass DemoIdentityKeyStorePassPhrase

3. Update the DemoIdentityStore with the certificate and key that you generated in Step 1.

   Note: Refer to "Using the Oracle WebLogic Server Java Utilities" in the Oracle Fusion Middleware Command Reference for Oracle WebLogic Server for more information.

   cd MW_HOME/wlserver/server/lib

   java utils.ImportPrivateKey -keystore DemoIdentity.jks -storepass DemoIdentityKeyStorePassPhrase -keyfile <hostname>-key.pem -keyfilepass <CAPassword> -certfile <hostname>-cert.pem -alias demoidentity -keypass DemoIdentityPassPhrase

4. Import the certificate that you generated in Step 1 into the DemoTrust.jks file.

   keytool -importcert -v -trustcacerts -file <hostname>-cert.pem -keystore DemoTrust.jks -storepass DemoTrustKeyStorePassPhrase -alias <hostname>

5. Restart the Oracle WebLogic Server Domain.

   Note: For an environment hosted on multiple servers, you must repeat this step for each server. Most importantly, you must copy or duplicate the updates you performed on one server (in MW_HOME/wlserver/server/lib) on to the other servers.

C.3.13.2 Cannot Access Recordings in Any Browser

When you try to view a session recording, the error message "This web page is not available" displays and you are redirected to a URL that uses localhost as the host name.
Cause
The Oracle Privileged Account Manager server URL that was configured under the Oracle Privileged Account Manager Server Configuration has localhost defined in the URL. This host name is unresolvable from external hosts.

Solution
Use the Server Configuration page to change the Oracle Privileged Account Manager server URL to reflect the fully qualified host name for the Oracle Privileged Account Manager server.

C.3.14 Session Checkout Does Not Work, Even After Granting the Account
An end user has been granted access to an account. However, when that user tries to connect as that account through the Oracle Privileged Session Manager the connection is disallowed.

Cause
Although the end user has been granted access to the account, the effective Usage Policy does not include session as the Allowed checkout type. You must explicitly grant session access in the Usage Policy.

Solution
Modify the effective Usage Policy to also grant session access.

C.3.15 OPAM Console Login Does Not Work in Internet Explorer 11 Browser
You tried to log into Oracle Privileged Account Manager by using the Console in an Internet Explorer 11 browser. No error messages were reported, but the login did not go through.

Cause
The Oracle Privileged Account Manager Login does not work in an Internet Explorer 11 browser.

Workaround
Use a lower version of Internet Explorer or another browser.

Solution
Apply the Oracle Universal Installer (OUI) patch for bug number 18071063 as described in the downloaded patch Readme.

To download this patch, login to https://support.oracle.com. Select the Patches and Updates tab and search for patch number 18071063.

C.4 Using My Oracle Support for Additional Troubleshooting Information
You can use My Oracle Support (formerly MetaLink) to help resolve Oracle Fusion Middleware problems. My Oracle Support contains several useful troubleshooting resources, such as:

- Knowledge base articles
- Community forums and discussions
- Patches and upgrades
■ Certification information

**Note:** You can also use My Oracle Support to log a service request.

You can access My Oracle Support at [https://support.oracle.com](https://support.oracle.com).
This glossary contains terms that are specific to administering Oracle Privileged Account Manager.

**account**
An account on a target.

**ADF**
*Oracle Application Development Framework*. An end-to-end development framework, built on top of the Enterprise Java platform, that provides integrated infrastructure solutions for the various layers of an application and an easy way to develop on top of those layers.

**Application Configurator**
Administrative role with privileges to configure and manage Oracle Privileged Account Manager servers.

**Authentication provider**
A security provider that manages and enforces authentication rules.

For more detailed information, refer to "Configuring Authentication Providers" in the *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

**BI Publisher**
An Oracle reporting product that can create and manage formatted reports from different data sources.

**bootstrap user**
A default administrator (*weblogic* user) who is a member of the Administrators group. This user can create and assign users to Oracle Privileged Account Manager Admin Roles and can map users from the domain identity store to Oracle Privileged Account Manager Common Admin Roles.

**Credential Store Framework**
See *CSF*.

**CRUD**
*Create, Read, Update, and Delete*. Basic functions of persistent storage or a database.

**CSF**
*Credential Store Framework*. An OPSS component that primarily provides secure storage for credentials.
**DOMAIN_HOME**

An environment variable that is usually `MW_HOME/user_projects/domains/<domain_name>`

**Grantee**

A user, group, or role that has been granted access to a *privileged account*.

**ICF**

*Identity Connector FrameWork*. A component that provides basic provisioning, reconciliation, and other functions required by all Oracle Identity Manager and Oracle Waveset connectors.

**Identity Connector FrameWork**

See *ICF*.

**identity propagation**

Process in which the OPSS Trust Service Asserter examines and validates a token, and then asserts that the identity performing a RESTful call against the Oracle Privileged Account Manager server is the one contained in the token.

**JSON representation**

*JavaScript Object Notation*. A lightweight, human-readable data format that is taken from JavaScript and used to exchange information between a browser and a server.

**ldifmigrator tool**

*Oracle Internet Directory Data Migration Tool*. Converts LDIF files output from other directories or application-specific repositories into a format recognized by Oracle Internet Directory.

**lockbox targets**

A target type that does not interact with Oracle Privileged Account Manager, but still provides a secure mechanism for storing the passwords associated with privileged accounts in a deployment.

**Oracle Privileged Account Manager client**

Component that resides with the Oracle Privileged Account Manager target to provide passwords to the system for unattended connections.

**Oracle Privileged Account Manager target**

Component that has its privileged passwords managed by Oracle Privileged Account Manager.

**OPSS**


**Oracle Application Development Framework**

See *ADF*.

**Oracle Internet Directory Data Migration Tool**

See *ldifmigrator tool*. 
Oracle Platform Security Services
See OPSS.

Password Policy
Captures the password construction requirements enforced by a specific target on an associated privileged account. Administrators use this policy to construct the password value that Oracle Privileged Account Manager uses to reset a password on a privileged account. Every privileged account managed by Oracle Privileged Account Manager has an associated Password Policy.

privileged accounts
Accounts on a target that are deemed "privileged" in a deployment and are under Oracle Privileged Account Manager's purview. Accounts are usually privileged when

- They are associated with elevated privileges
- They are used by multiple end-users on a task-by-task basis
- Their use must be controlled and audited

Repository Creation Utility
Oracle Repository Creation Utility. An application that you can use to create a schema and load a repository into the database.

Representational State Transfer
See REST.

resources
Representation of targets and accounts.

REST
Representational State Transfer. Software architecture style for distributed hypermedia systems like the World Wide Web. Conforming to REST constraints is otherwise known as being RESTful.

SAML
Security Assertion Markup Language. An XML-based open standard product provided by the OASIS Security Services Technical Committee that enables the exchange of authentication and authorization data between security domains.

Security Assertion Markup Language
See SAML

service account
An account that Oracle Privileged Account Manager uses when it connects to a target system and to perform all Oracle Privileged Account Manager-related operations (such as discovering accounts, resetting passwords, and so forth) on that target system. Service accounts require some special privileges and properties. Service accounts are sometimes referred to as unattended accounts.

shiphome
The directory where you downloaded and extracted Oracle Privileged Account Manager.
target
A software system that contains, uses, and relies on accounts (user, system, or application).

unattended accounts
See service account.

Usage Policy
Defines the constraints around when and how a grantee can use a privileged account. Each privileged account managed by Oracle Privileged Account Manager has an associated Usage Policy.
A

access rights, 2-5, 4-2, C-8
accordions
  Administration, 4-4
  Configuration, 4-5, 5-5, 5-6
  Home, 4-3
  Reports, 4-4
accounts, attended, 1-5
accounts, privileged
  access issues, C-4
  access rights, 2-5, 2-6
  adding, 4-9, 8-4, A-22, B-39
  administration roles, 2-4
  assigning policies, 9-7
  auditing, 14-1
  checking in, A-23
  checking out, A-24
  checking out/in, 2-8, 8-10, 8-11, 8-13, B-47, B-52
  deployment report, 4-4
  description, 1-1, 8-1
  display listing, A-23
  forcing check-ins, 8-13
  granting to groups, 10-3, A-30
  granting to users, 10-2, A-31
  managing, 8-2, 17-13
  mapping, 8-5, 8-8
  modifying, A-25
  opening, 8-9, 12-2
  removing, 8-18, A-25, B-40, B-55
  removing access, A-31, A-32, B-56
  resetting passwords, 1-2, 8-13, 8-18, 9-4
  retrieving, A-26, B-33, B-45
  searching, 4-5, 8-9, A-27, B-43, B-60
  searching for assigned, B-44, B-62
  searching for checkout history, A-28
  securing shared, 2-8
  sharing, 2-8, 8-4, 8-6
  showing checked out, 12-3, A-24, B-48, B-63
  showing passwords, 8-16
  troubleshooting, C-4, C-9
  updating, B-54
  verifying, B-54
  viewing passwords, 8-13
accounts, service
  configuring, 6-2, 7-2
description, Glossary-3
reset password, B-35
show password, B-35
showing passwords, 8-13
accounts, unattended, 1-5, 6-2, 7-1, 7-2, Glossary-3
activating
  Password Policies, 9-6
  Usage Policies, 9-12
adding
  authenticators, 3-7
  CSF map-keys, B-42
  CSF mappings, 8-8
  custom connectors, 15-6
  custom plug-ins, 2-12
  grantees, 8-7, B-41
  identity providers, 17-12
  new connectors, 15-6
  OPAM server, 5-2, 5-3
  OPSM server, 5-6
  Password Policies, 9-6, A-6, B-12
  plug-ins, 11-2, A-35, B-73
  privileged accounts, 4-9, 8-4, A-22, B-39
  service accounts, 7-1
targets, 4-9, 6-1, 6-2, A-12, B-25, C-5
Usage Policies, 9-6, A-8, B-18
users and groups, 3-12
ADF
  authentication, 2-3
  definition/purpose, Glossary-1
  Oracle Privileged Account Manager Console, 1-8
Admin Roles, Common, 2-4
Administration accordion, 4-4
administrators
  configuring OIM, 17-4
  default, 2-6
agents, WebGate, 17-9
APIs, REST, B-1
application accounts
  managing, 8-2
targets, 6-1
Application Configurator role
  access rights, 2-5
  assigning, 3-15
Application Development Framework, Oracle
  See ADF
applications
configuring access to multiple, 17-13
default URLs, 3-2
deploying client, 2-3
roles, 2-4
storing credentials, 17-13
unattended, 1-5
writing custom, 1-7
architecture
diagram, 1-6
Oracle Privileged Account Manager server, 5-2
assigning policies, 9-7
attended accounts, 1-5
attributes
adding custom, A-36
removing custom, A-37
retrieving target, B-21
audit logs
default file location, 14-2
saving, 14-2
audit reports
configuring, 14-3
default report types, 14-10
deploying, 14-7
example, 14-11
audit schema, 14-6
auditing
CSF content, 14-11
event types, 14-1
example audit report, 14-11
file-based, 14-3
logging levels, 14-9
managing, 14-1
privileged accounts, 14-1
saving audit logs, 14-2
shared accounts, 2-8
authentication
ADF-based, 2-3
framework, 2-1
JAAS support, 1-8, 2-1
modes, 2-2
Oracle Privileged Account Manager command line
tool client, 2-4
Oracle Privileged Account Manager server, 2-4
SAML-based token, 2-2
schema, 17-11
user, 2-3
authenticators, adding, 3-7
authorization
Common Admin Roles, 2-4
default URLs, 3-2
end users/enterprise users, 2-6
framework, 2-1
mapping users to Admin Roles, 2-5, 2-6
weblogic or bootstrap user, 2-5, 2-6
Auto-Detect URL, 5-7
automation.log, 3-12
planning, 15-7
recovering data, 15-9
using Oracle Recovery Manager (RMAN), 15-9
basic logging, configuring, 14-12
BI Publisher
audit reports, 14-10
configuring connection to server, 14-9
deploying audit reports, 14-7
element audit report, 14-11
features, 1-4
BI_DOMAIN_HOME, setting, 3-3
bootstrap user, 2-6, Glossary-1
C
catalogs, 17-3
certificates
CA, retrieving, 17-4
SSL, importing, C-6
SSL, trusting, 15-1
channels, secure versus unsecure, 2-7
checking out
sessions, 8-11, 9-10, 9-12
checking out/in
accounts, 8-11
checkout date, 8-11
current, 8-11
clients, third-party, 1-7
current status, 8-14
current status, 8-11
command line tool
adding Oracle Privileged Account Manager
server, 5-2
authentication modes, 2-2, 2-4
command syntax, A-3
security, 2-4, 2-8
commandsidmConfigTool, 3-11
importing SSL certificates, 15-2
WLST, 17-13
Common Admin Roles, 2-4
configuration accordion, 4-5, 5-5, 5-6
configuration entry, A-4
configuration files
decrypting, A-42
encrypting, A-40
configuring
access to multiple applications, 17-13
audit reports, 14-3
data sources, 14-7
event types, 14-1
external identity store, 3-7
identity store, 3-13
OIM administrators, 17-3
Oracle HTTP Server, 17-11
Oracle Internet Directory authenticator, 3-7
plug-ins, 11-7
shared accounts, 8-6
connecting to
  Oracle Privileged Account Manager server, C-3
  Oracle Privileged Session Manager, 5-6
connectors
  adding new, 15-6
  bundle location, 3-4
  connecting to target systems, 2-7
  custom, 1-8
  deploying, 3-3
  description, 3-3
  developing ICF-compliant, 3-4
  Identity Connector FrameWork, 1-4
  installing, 3-3
  LDAP, 17-2
  opam-config.xml file, 3-4, 15-6
  opam-config.xsd file, 3-4, 3-5, 15-6
  shipped with Oracle Privileged Account
    Manager, 3-3
  storing, 3-4
  supported database types, 6-5
  writing, 3-3
Console
  description, 1-7
  securing, 2-7
  troubleshooting issues, C-4
  user authentication, 2-3
Control Flag attributes, 3-8
creating
  Password Policies, 9-6, B-12
  plug-in configurations, 11-7
  schema, 14-5, Glossary-3
  service accounts, 7-1
  Usage Policies, 9-11, B-18
  users/groups in identity store, 3-12
Credential Store Framework
  See CSF.
credentials
  managing application, 17-15
  provisioning through Oracle Privileged Account
    Manager, 17-14
  starting servers, 3-6
  storing, 8-8, 17-13
  using CSF, 17-13
CSF
  account mapping, 8-5, 8-8, 17-14
  adding/removing map-keys, B-42
  definition/purpose, Glossary-1
  enabling auditing, 14-11
  integration with, 17-13
Current Checkouts table, 8-11, 8-14
.custom applications, writing, 1-7
custom attributes, plug-in, A-36, A-37
custom code, security, 2-12
custom connectors
  adding, 15-6
  using, 1-8
custom keystores, 15-3
custom plug-ins
  adding, 2-12
customizing pages, 15-9

D
data
  exporting, A-40
  importing, A-43
data encryption, using, 2-10, 15-3, C-9
data sources
  configuring, 14-7
  defining JDBC, 14-8
data store, RDBMS, 2-10
data, purging, 15-9
databases, hardening back-end, 2-10
decrypting encrypted configuration files, A-42
default
  administrator, 2-6
  audit report types, 14-10
  password requirements, setting, 9-5
  ports, 3-2, A-2
  URLs, 3-2
Default Password Policy, 8-6, 9-2
Default Usage Policy, 9-2
defining
  JDBC connections and data sources, 14-8
  policies, 2-1
  roles, 2-1
deleting
  grantees, 10-4
  Password Policies, B-14
  plug-in configurations, 11-11
  plug-ins, B-77
  policies, 9-8, 9-14
  Usage Policies, B-21
deploying
  audit reports in BI Publisher, 14-7
  client applications, 2-3
  connectors, 3-3, 15-6
  Oracle Privileged Account Manager in Oracle
    Fusion Middleware, 1-9
Deployment Reports, 13-2
diagnosing problems, C-2
diagnostic logs, 14-12
disabling
  Password Policies, 9-6
  Usage Policies, 9-12
displaying
  checked out accounts, A-24, B-48, B-63
  group listing, A-30
  privileged accounts list, A-23
  target listing, A-17
  user listing, A-30
domain identity store, using Oracle Virtual
  Directory, 3-9

DOMAIN_HOME, 14-2, Glossary-2
DOMIAN_HOME, setting, 3-3
duration, password, 9-4

E
enchanting configuration files, A-40
end users
  privileges, 2-6, C-8
enterprise roles
  populating resource catalog, 17-2
entitlements
  populating resource catalog, 17-2
  requesting access, 17-3
environments, moving from test to production, 15-9
executing plug-ins, 3-16, 11-5, 11-8, 11-9, A-35
exporting
  troubleshooting, C-8
exporting data, A-40
extending, schema in Oracle Internet Directory, 3-10
external identity store, configuring, 3-7

F
Failure Reports, 13-3
file-based auditing, configuring, 14-3
files
  audit logs, 14-2
  connector bundles, 3-4
  mod_wl_ohs.conf file, 17-12
  opam_product_BIP11gReports_11_1_2_1_0.zip, 14-7
  opam-config.xml file, 3-4, 15-6
  opam-config.xsd file, 3-4, 3-5, 15-6
  Repository Creation Utility zip, 14-6
filtering rules, plug-in, 11-9, A-35
firecall requests, 17-3
forcing check-ins, 8-13
framework
  ADF, Glossary-1
  authentication and authorization, 2-1
  CSF, 8-5, 17-13
  ICF, 1-2, 1-4, 3-3, 8-3
  Oracle Privileged Account Manager, 2-1
plug-in, xvii, 2-12, 8-3, 11-2, 11-4, 16-1

G
generating audit reports, 14-3
generic logs, default location, 14-12
grantees
  adding to privileged accounts, 8-7
  avoiding multiple grant paths, 2-9
  granting accounts, 10-2, 10-3, A-30, B-41
  opening, 10-4
  removing, 8-8, 10-4
  retrieving, A-32, B-46
  searching, 10-3
  troubleshooting, C-8
groups
  creating in identity store, 3-12
  display listing, A-30
  granting accounts, 10-3
  retrieving, B-68, B-69, B-70
  retrieving information, A-33
  searching, A-33, B-71

H
Home accordion, 4-3
HTTP Basic-Authorization, 2-2, 2-4

I
IAM_HOME, setting, 3-3
ICF
description, Glossary-2
  developing compliant connectors, 3-4
  framework, 3-3
  integration with, 1-4
  managing application accounts, 8-3
  password management, 1-2
Identity Connector FrameWork
  See ICF.
identity propagation, 2-3, Glossary-2
identity providers, adding, 17-12
identity store
  configuring, 3-7, 3-13
  creating users/groups, 3-12
  Oracle Internet Directory, 3-7, 17-9
  Oracle Virtual Directory, 3-7
  preparing, 3-10
  seeding, 3-12
identity store, OPSS, 1-9
idmConfigTool command, 3-11
importing
  data, A-43
  SSL certificates, 15-2
  troubleshooting, C-8
integrating with
  CSF, 17-13
  Oracle Access Management Access Manager, 17-9
  Oracle Identity Manager, 17-1
  Oracle Identity Manager workflows, 17-2
  Oracle technologies, 1-3
interfaces
  Oracle Privileged Account Manager, 1-7
  REST API, 4
  securing, 2-7

J
JAAS authentication support, 1-8, 2-1
ejar files, connector, 3-4
JAVA_HOME, setting, 3-3
JavaScript Object Notation
  See JSON.
JDBC connections and data sources, 14-8
JSON Representations
description, Glossary-2
Oracle Privileged Account Manager
  architecture, 1-7
  RESTful APIs, B-1

K
key sizes, troubleshooting, C-8, C-10
Index-5

keystores
  custom, 15-3

L
  LDAP connectors, 17-2
  LDAP groups, 17-4
  ldifmigrator, Glossary-2
  Listener ports, 5-8
  loading audit schema, 14-6
  lockbox targets, 6-2, 6-7, 7-3, 7-4, A-19, A-20, Glossary-2
  logging
    audit logger, 14-1
    audit logs location, 14-2
    configuring basic, 14-12
    diagnosing problems, C-2
    exceptions, C-2
    generic logger, 14-12
    generic logs location, 14-12
    setting audit logging levels, 14-9
  Login page, rebranding, 15-9
  logs
    default locations, 14-12
    diagnostic, C-3
    generic, 14-12
    idmConfigTool automation.log, 3-12
    specifying name/location, A-41

M
  managing
    account credentials, 17-13
    application credentials, 17-15
    Oracle Privileged Account Manager audit
      logging, 14-1
      passwords, 1-2, 8-16
      server properties, A-4
    managing passwords, 8-16
    map-keys, CSF, B-42
    mapping, CSF, 8-5, 8-8, 17-14
    mod_wl_ohs.conf file, 17-12
  modifying
    Default Password Policy, 9-4
    Default Usage Policy, 9-10
    OPAM Global Config configuration entry, A-5
    Password Policies, A-9, B-11
    plug-ins, A-38, B-77
    policies, 9-2, B-17
    privileged accounts, A-25
    targets, A-17
    Usage Policies, A-9
    multiple grant paths, avoiding, 2-9
    MW_HOME, setting, 3-3

N
  network channel, securing, 2-6
  obfuscation, 2-11
  OPAM Global Config configuration entry, 5-4, A-5
  OPAM Service Account, 1-5
  OPAM service account
    description, 1-5
    managing passwords, 6-9, B-36, B-37
  OPAM service accounts
    creating, 7-1
    description, 7-1
    managing passwords, 7-2
  opam_product_BIP11gReports_11_1_2_1_0.zip file, 14-7
  opam-config.xml file, 3-4, 15-6
  opam-config.xsd file, 3-4, 3-5, 15-6
  opening
    grantees, 10-4
    plug-ins, 11-10
    policies, 9-3, 9-9
    privileged accounts, 8-9, 12-2
    targets, 6-9
  OPSS, 2-3
    description, Glossary-2
    identity store, 1-9
    Policy Store, 1-3
    providing authentication, 2-3
    security store, 1-9
    Trust Service, 1-3
  OPSS Trust Service, 2-3, Glossary-2
  OPSS-Trust Service Assertions, 2-3
  OPSS-Trust tokens, 2-1
  Oracle Access Management Access Manager integration with, 17-9
  Oracle Application Development Framework
    See ADF.
  Oracle Database
    backup and recovery, 15-7
    connecting to, 3-3, C-5
  Oracle Database TDE mode
    disabling
      from the command line, A-5
      from the Console, 5-5
      using REST API, B-4
    enabling
      from the command line, 15-4, A-5
      from the Console, 5-5
      using REST API, B-4
    securing OPAM database, 15-3, C-9
  Oracle Fusion Middleware
    deploying Oracle Privileged Account Manager, 1-9
  Oracle Fusion Middleware Audit Framework, 1-4
  Oracle HTTP Server
    configuring, 17-11
    using for Single Sign On, 17-10
  Oracle Identity Manager
    CA certificate, OPAM, 17-4
    configuring administrators, 17-4
    enterprise roles, 17-2
    entitlements, 17-2, 17-3
integration, 17-1, 17-2
resource catalog, 17-2
rules, 17-3
workflow support, 17-2
Oracle Internet Directory
configuring authenticator, 3-7
Data Migration Tool (ldifmigrator), Glossary-2
default store, 3-7, 17-9
Oracle Platform Security Services
See OPSS
Oracle Privileged Account Manager
architecture and topology, 1-6
command syntax, A-3
default connectors, 3-3
interfaces, 1-7
Managed Server, starting, 3-6
securing, 2-6
Oracle Privileged Account Manager Console
about, 1-7
adding Oracle Privileged Account Manager server, 5-2
ADF, 1-8
securing, 2-7
Oracle Privileged Account Manager server architecture, 5-2
authentication, 2-4
connecting to, C-3
Oracle Privileged Session Manager
configuring a connection, 5-6
managing, 5-6
Oracle Recovery Manager
See RMAN., 15-7, 15-9
Oracle Virtual Directory
identity store, 3-7
sample output from idmConfigTool, 3-11
using as domain identity store, 3-9
using non-OID directories, 3-10
ORACLE_HOME, setting, 3-3

P
packet sniffing, 2-7
pages, rebranding, 15-9
Password Complexity Rules, 9-5
password history, viewing, 7-4
Password Policies
activating, 9-6
adding, A-6
assigning to accounts, 9-7
creating, 9-6, B-12
deleting, B-14
description/purpose, 9-1
disabling, 9-6
modifying, 9-2, 9-4, A-9
removing, A-10
resetting passwords, 8-18, 9-4
retrieving, B-9
searching, 9-3, 9-9
specifying password durations, 9-4
updating, B-11
Password Policy, Default, 8-6
Password Rollover, 7-4
password rollover, 7-4
passwords
defining requirements, 9-5
managing, 1-2, 8-16
privileged, 1-2
propagating, 2-7
resetting, 2-9, 7-4, 8-18, A-18, A-26, B-35, B-59
resetting automatically, 1-2, 9-4
resetting manually, 8-18, 9-4
rollover, 7-4
service account, B-35, B-36, B-37
service accounts, 7-2
showing, 7-3, 8-16, 12-3, A-20, A-21, A-28, B-36, B-37, B-57
showing history, 7-3, 8-17, A-29, B-58, B-59
specifying duration period, 9-4
storing, 1-2
viewing password history, 4-9
viewing password reset history, A-21
Pattern fields, using, 4-6, 8-15
plug-in
filtering rules, 11-9, A-35
plug-in framework, xvii, 2-12, 8-3, 11-2, 11-4, 16-1
plug-ins, B-76
adding, A-35, B-73
adding custom, 2-12
adding custom attributes, A-36
creating configurations, 11-7
deleting configurations, 11-11
executing, 3-16, 11-5, 11-8, 11-9
modifying, A-38, B-77
opening, 11-10
overview, 11-1
post-operation, 11-5
pre-operation, 11-5
removing, A-39, B-77
removing custom attributes, A-37
required Admin Roles, 2-12
retrieving information, A-37
searching for, 11-10, A-38, B-75
verifying, B-74
policies
adding, A-6, A-8
assigning to accounts, 9-7
creating, 9-6, 9-11, B-12, B-18
default, 8-6
defining, 2-1
deleting, 9-8, 9-14, B-14, B-21
description/purpose, 9-1
disabling, 9-6, 9-12
getting default, B-8
making active, 9-6, 9-12
modifying, 9-4, 9-10
opening, 9-3, 9-9
retrieving, A-11, B-9, B-14
searching, 9-3, 9-9
searching for, B-7
types, 9-1
updating, B-11, B-17
verifying, 9-7, 9-8
viewing, 9-3, 9-9
Policy Store, OPSS, 1-3
ports
default, 3-2, A-2
Listener, 5-8
SSL, 5-2, A-2
post-operation plug-ins, 11-5
pre-operation plug-ins, 11-5
privileged accounts
access rights, 2-5, 2-6
adding, 8-4
administration roles, 2-4
assigning policies, 9-7
auditing, 14-1
checking out/in, 8-10, 8-11, 8-13
deployment report, 4-4
description, 1-1, 8-1
display listing, A-23
granting to groups, 10-3
granting to users, 10-2
managing, 8-2
mapping, 8-5, 8-8
opening, 8-9, 12-2
removing, A-25
removing from target, 8-18
removing group access, A-31
resetting passwords, 1-2, 8-18, 9-4
searching, 4-5, 8-9
searching for, A-27
searching for checkout history, A-28
securing shared, 2-8
sharing, 8-4, 8-6
showing checked out, 12-3, A-24, B-48, B-63
showing passwords, 8-16
viewing your accounts, 8-12, 12-2
privileged passwords, 1-2
privileged sessions
checking out, 8-11, 9-10, 9-12
recordings, 8-14
privileged sessions
checking out, 8-11, 9-10, 9-12
recordings, 8-14
privileges
administrators, 2-5
end users, 2-6
service accounts, 7-1
troubleshooting, C-8
propagating passwords, 2-7
propagation, identity, 2-3
properties
Session Manager, 5-7
protocol mappings, Listener, 5-8
provisioning
credentials, 17-14
process diagram, 17-14
purging data, 15-9
recordings
purging session, 15-9
recovering session, 15-9
troubleshooting, C-10
viewing session, 8-14
registered accounts, retrieving, B-34
removing
accounts from targets, 8-18
CSF map-keys, B-42
custom plug-in attributes, A-37
grantees, 10-4, A-31, A-32
Password Policies, A-10
plug-ins, A-39, B-77
policies, B-14, B-21
privileged accounts, A-25, B-40, B-55
required Admin Role, 2-5
targets, 6-10, A-18, B-31
Usage Policies, A-10
removing grantees, 8-8
reporting
BI Publisher, 14-10
eample audit report, 14-11
reports
audit, 14-7
Checkout History, 8-15, 13-4
configuring, 14-3
default audit, 14-10
Deployment, 13-2
eample audit, 14-11
Failure, 13-3
Usage, 13-3
viewing, 13-1
Reports accordion, 4-4
Repository Creation Utility, 14-6, Glossary-3
Representational state transfer service
See REST (Restful).
purging passwords, 1-2, 2-9, 8-18, 9-4, A-18, A-26, B-35, B-59
resource catalog, 17-2
REST (RESTful)
APIs, 3
calls, 3
definition/purpose, Glossary-3
interface, 4, B-1
service, 1-7, 7
retrieving, B-76
available accounts, B-33
grantees, A-32, B-46
group information, A-33
groups, B-68, B-69, B-70
Password Policies, B-9
plug-in information, A-37
plug-ins, B-76
policies, A-11
privileged accounts, A-26, B-45
registered accounts, B-34
target types, B-35
targets, A-19, B-29
Usage Policies, B-14
users, A-33, B-46, B-64
R
RDBMS data store, 2-10
rebranding pages, 15-9
target GUID
adding accounts, A-22
modifying targets, A-18
removing targets, A-18
retrieving targets, A-20
target service accounts, 7-3, 7-4, A-18, A-20, A-21
target types
lockbox, 6-2, 6-7, 7-3, 7-4, A-19, A-20, Glossary-2
retrieving, B-35
targets
adding, 4-9, 6-1, 6-2, A-12, B-25
connecting to, 2-7, C-4
display listing, A-17
lockbox, 6-2, 6-7, 7-3, 7-4, A-19, A-20, Glossary-2
modifying, A-17
opening, 6-9
removing, 6-10, A-18, B-31
removing accounts, 8-18
retrieving, A-19, B-29
searching for, 6-8, A-20, B-32
troubleshooting, C-4, C-5, C-6, C-9
updating, B-30
verifying, B-27
TDE mode
disabling
from the command line, A-5
from the Console, 5-5
using REST API, B-4
enabling
from the command line, 15-4, A-5
from the Console, 5-5
using REST API, B-4
securing OPAM database, 2-10, 15-3, C-9
troubleshooting, C-9
test to production, moving components from, 15-9
third-party clients, 1-7
tokens, OPSS Trust, 2-1
topology and architecture diagram, 1-6
Transparent Data Encryption mode
See TDE mode.
troubleshooting common problems, C-1
Trust Service, OPSS, 1-3

U
unattended
accounts, 1-5
applications, 1-5
unattended accounts, 7-1
unsecure channels, 2-7
unshared accounts, 2-8
updating
accounts, B-54
Password Policies, B-11
targets, B-30
Usage Policies, B-17
URIs, B-1
URLs, default application, 3-2
Usage Policies
activating, 9-12
adding, A-8
assigning to accounts, 9-7
creating, 9-11, B-18
deleting, B-21
description/purpose, 9-1
disabling, 9-12
modifying, 9-2, 9-10, A-9
removing, A-10
retrieving, B-14
searching, 9-3, 9-9
updating, B-17
Usage Reports, 13-3
user authentication, 2-3
User Manager role, 2-5
users
bootstrap, 2-6, Glossary-1
creating in identity store, 3-12
display listing, A-30
granting accounts, 10-2, B-41
removing access, A-32, B-56
retrieving, A-33, B-46, B-64
searching for, A-34, B-65, B-66
self-service, 3-16, 12-1
sharing accounts, 2-8, 9-11
utilities, Repository Creation Utility, 14-6
V
validating opam-config.xml, 15-6
verifying
OID configuration, 3-9
plug-in configurations, B-74
policies, 9-7, 9-8
privileged accounts, B-54
targets, B-27
viewing
accounts, 4-3
policies, 9-3, 9-9
reports, 13-1
your accounts, 8-12, 12-2
viewing passwords, 8-16
W
WebGate agents, 17-9
WebLogic
SSL port, 5-2, A-2
starting Admin Server, 3-6
weblogic user, 2-6
wildcards, in searches, 4-7
WLST commands, 17-13
workflows
administrator, 3-15
integrating with Oracle Identity Manager, 17-2
Oracle Identity Manager support, 17-2
self-service, 3-16, 12-1