Oracle® Fusion Middleware

Third-Party Application Server Guide for Oracle Identity and Access Management

11*g* Release 2 (11.1.2.1.0)

E28523-09

April 2014

ORACLE

Oracle Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management, 11*g* Release 2 (11.1.2.1.0)

E28523-09

Primary Author: Peter LaQuerre

Contributing Authors: Don Biasotti, Barbara Buerkle, Debapriya Datta, Gail Flanegin, Helen Grembowicz, Peter Jew, Mark Kennedy, Kevin Kessler, Liz Lynch, Robert May, Carlos Subi, Len Turmel, Nisha Singh, Priscilla Lee, KC Francis, Gina Cariaga, Michael Teger, Trish Fuzesy, Vinaye Misra, Prakash Hulikere, Showvik Roy Chowdhuri

Contributors: Mike Blevins, Robert Campbell, Dave Felts, Jeni Ferns, Nick Greenhalgh, Harry Hsu, Hareesh Kolpuru, Vasant Kumar, Dennis Leung, Dan MacKinnon, Mark Miller, Kevin Minder, Vinod Nimmagadda, Vijay Ramanathan, Michael Rubino, Roy Sandjaja, Reza Shafii, Vishal Sharma, Stephen Sherman, Payal Srivastra, Sitaraman Swaminathan, Ken Vincent, Prakash Yamuna, Lisa Zitek-Jones, Ayush Jindal, Lloyd Devraj

Contents

3 Managing Oracle Identity and Access Management Suite on IBM WebSphere

4 Managing Oracle Identity Manager on IBM WebSphere

5 Managing Access Manager on IBM WebSphere

8 Managing Oracle Adaptive Access Manager on IBM WebSphere

9 Managing Oracle Fusion Middleware Security on IBM WebSphere

10 Managing Oracle Entitlements Server on IBM WebSphere

11 Managing Oracle Privileged Account Manager on IBM WebSphere

12 Managing Oracle Identity Navigator on IBM WebSphere

13 Managing Oracle Access Management Mobile and Social on IBM WebSphere

14 Integrating Access Manager, Oracle Adaptive Access Manager, and Oracle Identity Manager on IBM WebSphere

A Fusion Middleware Control Page Reference

Preface

This preface contains the following sections:

Audience

This manual is intended for Oracle Fusion Middleware system administrators who are responsible for installing and managing Oracle Fusion Middleware on third-party application servers, such as IBM WebSphere.

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, see the following related documentation available in the Oracle Fusion Middleware 11*g* documentation library:

- *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Application Development Framework*

Conventions

The following text conventions are used in this document:

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

1

Introduction to Third-Party Application Servers

This chapter introduces the Oracle Identity and Access Management 11*g* support for third-party application servers.

This chapter contains the following sections:

1.1 What is a Third-Party Application Server?

A third-party application server is an application server provided by a vendor other than Oracle.

Oracle supports Oracle WebLogic Server as the primary platform for Oracle Fusion Middleware software components. However, to accommodate customers who want to run specific Oracle Fusion Middleware component software, such as Oracle Identity and Access Management Suite, on application servers other than Oracle WebLogic Server, Oracle supports the third-party application servers described in this document.

1.2 Oracle Fusion Middleware Components That Support Third-Party Application Servers

You can configure the following Oracle Fusion Middleware components on supported third-party application servers:

- Oracle Identity and Access Management
- Oracle SOA Suite
- Oracle WebCenter Portal
- Oracle WebCenter Content
- Oracle Application Development Framework (Oracle ADF)
- Oracle Application Developer Runtime

For this release of Oracle Fusion Middleware 11*g*, Oracle supports only IBM WebSphere Application Server as a third-party application server for these Oracle Fusion Middleware products.

Note: You can install and configure IBM WebSphere Application Server to work with Oracle Unified Directory. This is possible only if you are already managing Oracle Unified Directory using the graphical Oracle Directory Service Manager interface. For more information, see "Configuring IBM WebSphere for Oracle Directory Services Manager" in the *Oracle Fusion Middleware Installation Guide for Oracle Unified Directory*.

1.3 Overview of the Oracle Fusion Middleware IBM WebSphere Support

The following sections provide more detail about the supported Oracle Fusion Middleware features on IBM WebSphere:

- Supported IBM WebSphere Application Server
- Understanding the Topology of Oracle Fusion Middleware on IBM WebSphere Application Server - ND

1.3.1 Supported IBM WebSphere Application Server

Oracle supports **IBM WebSphere Application Server - Network Deployment (ND)** for Oracle Identity and Access Management Suite components.

For the most accurate and up-to-date information about the IBM WebSphere Application Server supported by Oracle Fusion Middleware, see the Certification information on the Oracle Technology Network (OTN), as described in Section 2.1, "Task 1: Review the System Requirements and Certification Information".

1.3.2 Understanding the Topology of Oracle Fusion Middleware on IBM WebSphere Application Server - ND

When you install and configure Oracle Identity and Access Management Suite with IBM WebSphere Application Server - ND, the configuration process automatically creates an IBM WebSphere cell that contains a special server, in addition to the Deployment Manager, called the OracleAdminServer.

This OracleAdminServer hosts the key infrastructure pieces of Oracle Identity and Access Management Suite products, including the Java Required Files (JRF) and Oracle Enterprise Manager product templates:

- The JRF template provides important Oracle libraries and other capabilities that support new versions of APIs that many Oracle Identity and Access Management Suite products and applications depend upon.
- The Oracle Enterprise Manager template provides Oracle Enterprise Manager Fusion Middleware Control, which you can use to manage the Oracle Identity and Access Management Suite products you install and configure.

Additional products are installed on additional servers in the newly created IBM WebSphere cell.

When you configure your IBM WebSphere cell for use with Oracle Identity and Access Management Suite, you can also include additional servers and clusters in your cell,

and you can configure the Oracle Identity and Access Management Suite products to work with an Oracle Real Application Clusters (Oracle RAC) database.

1.4 Documentation Resources for Using Oracle Identity and Access Management Suite Products on IBM WebSphere

You can refer to the following additional documentation resources for information about running Oracle Identity and Access Management Suite products on IBM WebSphere:

- The IBM WebSphere documentation available on the WebSphere Application Server Information Center for basic conceptual information about IBM WebSphere, as well details about installing IBM WebSphere.
- This document for an overview of the Oracle Identity and Access Management Suite support for IBM WebSphere, a summary of the overall steps required to install and configure Oracle Identity and Access Management Suite on IBM WebSphere, and a high-level listing of the features and tools available for installing and managing Oracle Identity and Access Management Suite on IBM WebSphere.
- *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server* for complete information on the capabilities of the Oracle Fusion Middleware Configuration Wizard, including information about creating and modifying cells, how to add additional servers and clusters to a cell, and how to configure Oracle Fusion Middleware products to support an Oracle Real Application Clusters (Oracle RAC) database.
- Specific sections of the Oracle Fusion Middleware documentation library for information about specific feature areas described in this guide. As you review this document, note the links to specific Oracle documentation that can help you successfully develop and administer your Oracle Identity and Access Management Suite products on IBM WebSphere.

2

Installing and Configuring Oracle Identity and Access Management on IBM WebSphere

The following sections describe how to install and configure Oracle Identity and Access Management on IBM WebSphere.

> **Note:** This chapter provides basic information about how to install and configure a single instance of Oracle Identity and Access Management on IBM WebSphere. If you are interested in configuring a high availability environment on IBM WebSphere, then review the content in this chapter, and then see Section 3.4, "Configuring Oracle Fusion Middleware High Availability on IBM WebSphere".

2.1 Task 1: Review the System Requirements and Certification Information

Before performing any upgrade or installation you should read the system requirements documentation to ensure that your environment meets the minimum installation requirements for the products you are installing.

The system requirements document covers information such as hardware and software requirements, minimum disk space and memory requirements, and required system libraries, packages, or patches:

http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-requirements-100147.html

In addition, you should read the certification document. The certification document covers supported installation types, platforms, operating systems, databases, JDKs, and third-party products:

http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-certification-100350.html

2.2 Task 2: Obtain the Necessary Software Media or Downloads

For this installation and configuration procedure, you will need to obtain the following software:

- IBM WebSphere Application Server - Network Deployment (ND)

  For more information, see Section 2.4.1, "IBM Online Resources for Obtaining and Installing the IBM WebSphere Software."

  For specific information the software requirements, refer to Section 2.1, "Task 1: Review the System Requirements and Certification Information".

- Oracle Database
- Oracle Fusion Middleware Repository Creation Utility 11*g* Release 2 (11.1.2.1.0)
- Oracle Identity and Access Management Suite 11*g* Release 2 (11.1.2.1.0)
- Oracle SOA Suite 11*g* (11.1.1.6.0)

  Note: Oracle Identity Manager requires Oracle SOA Suite. If you are installing Oracle Identity Manager, you must install Oracle SOA Suite 11*g* (11.1.1.6.0).

  After installing Oracle SOA Suite 11.1.1.6.0, you must apply mandatory SOA patches. For more information, see "SOA Patch Requirements for Oracle Identity Manager".

For information about where to download the software, refer to the *Oracle Fusion Middleware Download, Installation, and Configuration Readme Files* on the Oracle Technology Network (OTN):

http://download.oracle.com/docs/cd/E23104_01/download_readme.htm

2.3 Task 3: Identify a Database and Install the Required Database Schemas

The following Oracle Fusion Middleware products require a metadata repository with required schemas to be installed in a supported database:

- Oracle Identity and Access Management Suite
- Oracle SOA Suite

You cannot configure these products without first installing the required schemas in a supported database.

To create or update schemas in a database, use the Repository Creation Utility (RCU).

> **Note:** It is recommended that all metadata repositories reside on a database at the same site as the products to minimize network latency issues.

For information about identifying the schemas required for specific Oracle Fusion Middleware products, as well as information about the database requirements and running RCU, refer to *Oracle Fusion Middleware Repository Creation Utility User's Guide*.

For information on the databases supported by Oracle Fusion Middleware, see the certification information described in Section 2.1, "Task 1: Review the System Requirements and Certification Information".

Make a note of the database connection information along with the name and passwords for the schemas you create with the Repository Creation Utility. You will need these later when you configure the Oracle Fusion Middleware products.

2.4 Task 4: Install the IBM WebSphere Software

To install and configure Oracle Fusion Middleware with IBM WebSphere, you must first install (but not configure) IBM WebSphere Application Server -Network Deployment (ND).

Refer to the following sections for more information:

- IBM Online Resources for Obtaining and Installing the IBM WebSphere Software
- Important Considerations Before Installing the IBM WebSphere Software

2.4.1 IBM Online Resources for Obtaining and Installing the IBM WebSphere Software

Refer to the following IBM resources for more information.

Note that Oracle is not responsible for the content in the following links. These references are provided for convenience only. Be sure to refer to the IBM documentation provided with or referenced by your IBM WebSphere software distribution:

- To obtain and install the IBM WebSphere software, refer to the IBM WebSphere documentation. For more information, see Section 1.4, "Documentation Resources for Using Oracle Identity and Access Management Suite Products on IBM WebSphere".
- For more information about the Fix Packs available for IBM WebSphere 7.0, refer to the Fix list for IBM WebSphere Application Server V7.0 on the IBM Support Web site.
- You install the Fix Packs using the IBM WebSphere Update Installer. For more information, see the information about the Maintenance Download Wizard for WebSphere Application Server V7.0 on the IBM Support Web site.

2.4.2 Important Considerations Before Installing the IBM WebSphere Software

Before you perform the IBM WebSphere installation, note the following requirements for Oracle Fusion Middleware products:

- Using the Correct IBM WebSphere Installer for Your Platform
- About the Sample Applications and Default Profiles During the IBM WebSphere Installation
- About the WAS_HOME Directory Path

2.4.2.1 Using the Correct IBM WebSphere Installer for Your Platform

Note that like Oracle WebLogic Server, IBM WebSphere is available for different platforms. Some platforms, such as Linux 64-bit platforms, require unique IBM WebSphere installers.

Before you begin your IBM WebSphere installation, be sure you have obtained the correct IBM WebSphere installer for your platform.

2.4.2.2 About the Sample Applications and Default Profiles During the IBM WebSphere Installation

Do not install any sample applications or create any profiles during the IBM WebSphere installation process.

The goal is to install the IBM WebSphere software on disk in a directory available to the Oracle Fusion Middleware software installation, which you will perform later. You will use the Oracle Fusion Middleware Configuration wizard to configure the required IBM WebSphere profiles.

2.4.2.3 About the WAS_HOME Directory Path

When you install the IBM WebSphere software, you are prompted for the location where you want to install the software. For the purposes of this documentation, this location is later referred to as the WAS Home, or `WAS_HOME` in examples.

If you accept the default values that are provided during the installation, then the `WAS_HOME` is installed in the following directory structure:

```
DISK/IBM/WebSphere/AppServer
```

Create the `WAS_HOME` for the IBM WebSphere software on the same host where you plan to install the Oracle Fusion Middleware software. `WAS_HOME` should be at the same level as `MW_HOME`.

Make a note of this path. You will be asked to identify the location of the IBM WebSphere directory when you configure Oracle Fusion Middleware.

2.5 Task 5: Install Oracle SOA Suite (Oracle Identity Manager Users Only)

If you are installing Oracle Identity Manager, you must install Oracle SOA Suite 11*g* (11.1.1.6.0). Note that only Oracle Identity Manager requires Oracle SOA Suite. This step is required because Oracle Identity Manager uses process workflows in Oracle SOA Suite to manage request approvals.

Run the Oracle SOA Suite installer, as follows:

```
SOA_Installer_Home/Disk1/runInstaller -jreLoc WAS_HOME/java/jre
```

For more information about installing Oracle SOA Suite, see *Oracle Fusion Middleware Installation Guide for Oracle SOA Suite and Oracle Business Process Management Suite*.

SOA Patch Requirements for Oracle Identity Manager

After installing Oracle SOA Suite 11.1.1.6.0, you must apply mandatory SOA patches before installing Oracle Identity Manager. For information about the patches, refer to

the "Mandatory Patches Required for Installing Oracle Identity Manager" topic in the 11*g* Release 2 *Oracle Fusion Middleware Release Notes*.

2.6 Task 6: Install Oracle Identity and Access Management Suite

For instructions on installing Oracle Identity and Access Management on IBM WebSphere, refer to the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.

2.6.1 Special Instructions When Installing Oracle Identity and Access Management with IBM WebSphere

Note the following special instructions that apply when you are installing Oracle Fusion Middleware products on IBM WebSphere:

- When you run the Oracle Fusion Middleware installer, you must use the parameter `-DSHOW_APPSERVER_TYPE_SCREEN=true` to let the Oracle Universal Installer prompt for the IBM WebSphere home location.

  Example:

  ```
  diskname/iamsuite/Disk1/runInstaller -jreLoc
  diskname/IBM/WebSphere/AppServer/java/jre -DSHOW_APPSERVER_TYPE_SCREEN=true
  ```

- When you are prompted to specify a JRE/JDK location, you can specify the following directory in the IBM WebSphere home:

  On UNIX operating systems:

  ```
  WAS_HOME/java
  ```

  For example, if you are using the default location for a typical IBM WebSphere Application Server directory on a UNIX operating system:

  ```
  diskname/IBM/WebSphere/AppServer/java
  ```

- When you are prompted to provide a Middleware home, note that you can enter a new Middleware home directory path.

  When you install Oracle Fusion Middleware products on Oracle WebLogic Server, you create the Middleware home. This is because Oracle WebLogic Server is included in the Middleware home.

  In contrast, when you install Oracle Fusion Middleware on IBM WebSphere, you create the Middleware home when you install the Oracle Fusion Middleware software. This is because the IBM WebSphere software is not installed inside the Middleware home. It is installed in a separate directory structure.

- When you select IBM WebSphere as your application server and you are prompted for the Application Server Location, enter the path to the IBM WebSphere application server directory you created in Section 2.4, "Task 4: Install the IBM WebSphere Software".

  For example:

  ```
  diskname/IBM/WebSphere/AppServer/
  ```

2.7 Task 7: Optional: Enabling TDE in Oracle Privileged Account Manager Data Store (For Oracle Privileged Account Manager Users Only)

Oracle Privileged Account Manager can operate with Oracle Database TDE (Transparent Data Encryption) mode. You can choose to either enable or disable the TDE mode. Oracle strongly recommends to enable the TDE mode for enhanced security.

This section includes the following topics:

- Enabling TDE in the Database
- Enabling Encryption in OPAM Schema

2.7.1 Enabling TDE in the Database

For information about enabling Transparent Data Encryption (TDE) in the database for Oracle Privileged Account Manager, refer to the "Enabling Transparent Data Encryption" topic in *Oracle Database Advanced Security Administrator's Guide*.

For more information, see "Securing Stored Data Using Transparent Data Encryption" in the *Oracle Database Advanced Security Administrator's Guide*

After enabling TDE in the database for Oracle Privileged Account Manager, you must enable encryption in OPAM schema, as described in Section 2.7.2, "Enabling Encryption in OPAM Schema".

2.7.2 Enabling Encryption in OPAM Schema

To enable encryption in the OPAM schema, run the `opamxencrypt.sql` script with the OPAM schema user, using sqlplus or any other client.

IAM_HOME`/opam/sql/opamxencrypt.sql`

Example:

```
sqlplus DEV_OPAM/welcome1 @IAM_HOME/opam/sql/opamxencrypt.sql
```

2.8 Task 8: Configure Your Oracle Identity and Access Management Components in a New IBM WebSphere Cell

To configure Oracle Identity and Access Management components in an IBM WebSphere environment, you use the IBM WebSphere version of the Oracle Fusion Middleware Configuration Wizard.

This section includes the following topics:

- General Information About Using the Configuration Wizard on IBM WebSphere
- Configuring Oracle Identity and Access Management Components for Single-Node Setup

2.8.1 General Information About Using the Configuration Wizard on IBM WebSphere

Note the following information as you advance through the Configuration Wizard:

- Be sure to make a note of the values you enter on the Specify Cell, Profile, and Node Name Information screen. You will need these later when you are starting and managing the cell. In particular, make note of the values you enter in the

Deployment Manager Profile Name field and the **Application Server Profile Name** field.

- When the Add Products to Cell screen appears, refer to the following:

  "Fusion Middleware Product Templates" in the *Oracle Fusion Middleware Domain Template Reference* if you have questions about what capabilities are configured when you select each template.

- If you select a product that requires a database schema, you will be prompted for database connection information for each required schema. To fill out this screen, use the database and schema information you noted in Section 2.3, "Task 3: Identify a Database and Install the Required Database Schemas".

- When you are prompted for advanced options, you can click **Next** and use the default settings. Refer to Section 1.3.2, "Understanding the Topology of Oracle Fusion Middleware on IBM WebSphere Application Server - ND" for information on the topologies that will be created using the default settings.

  If you wish to modify the default settings (for example, if you want to target the products to different servers in the cell), refer to *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server*.

2.8.2 Configuring Oracle Identity and Access Management Components for Single-Node Setup

This section describes how to use the Configuration Wizard to configure your Oracle Identity and Access Management products in a simple IBM WebSphere cell. For complete information about using the Oracle Fusion Middleware Configuration Wizard, including information about adding servers and clusters to a cell, refer to the *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server*.

Note: The instructions here describe how to use the Configuration Wizard to configure your components. However, you can also use the WebSphere wsadmin command-line utility to configure your Oracle Fusion Middleware components.

- For more information about using the wsadmin command-line utility, see Section 3.1.3, "Using the Oracle Fusion Middleware wsadmin Commands".

- For more information about configuring components with wsadmin, see "Using wsadmin to Configure Oracle Fusion Middleware" in the *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server*.

To configure your Oracle Identity and Access Management product in a new IBM WebSphere cell, complete the following steps:

1. Start the Oracle Fusion Middleware Configuration Wizard by running the following command from the Oracle Identity and Access Management home:

   On UNIX operating systems:

   ```
   ORACLE_HOME/common/bin/was_config.sh
   ```

2. On the Select Configuration Option screen, select **Create and Configure Cell**. Then, click **Next**.

3. On the Specify Cell, Profile, and Node Name Information screen, provide the default name or a new name. The default names are:

   Cell Name: *host*Cell01

   Deployment Manager Profile Name: Dmgr01

   Deployment Manager Node name: *host*CellManager01

   Application Server Profile Name: Custom01

   Application Server Node Name: *host*Node01

4. On the Specify Deployment Manager Information screen, enter WebSphere Administration username and password. The WebSphere Administration user name and password provided here will be used for logging into the console and for performing certain configuration steps later. Click **Next**.

5. On the Add Products to Cell screen, select the required templates based on the components that you want to configure. Table 2–1 provides the name of the Oracle Identity and Access Management templates and their dependencies.

Table 2–1 Oracle Identity and Access Management Product Templates

Template Name	Dependency
Oracle Identity Manager for Webpshere ND - 11.1.2.0.0 [Oracle_IDM1]	■ Oracle SOA suite for WebSphere ND - 11.1.1.0 [Oracle_SOA1] ■ Oracle Enterprise Manager for WebSphere - 11.1.1.0 [oracle_common] ■ Oracle Workflow Client Extension - 11.1.1.0 [Oracle_SOA1] ■ Oracle WSM Policy Manager - 11.1.1.0 [oracle_common] ■ Oracle JRF Webservices Asynchronous services - 11.1.1.0 [oracle_common] ■ Oracle Platform Security Service - 11.1.1.0 [Oracle_IDM1] ■ Oracle JRF for WebSphere - 11.1.1.0 [oracle_common]
Oracle Access Management - 11.1.2.0.0 [Oracle_IDM1]	■ Oracle Platform Security Service - 11.1.1.0 [Oracle_IDM1] ■ Oracle JRF for WebSphere - 11.1.1.0 [oracle_common]
Oracle Adaptive Access Manager Admin Server - 11.1.2.0.0 [Oracle_IDM1] In addition, you can select the following: ■ **Oracle Adaptive Access Manager Offline - 11.1.2.0.0 [Oracle_IDM1]** ■ **Oracle Adaptive Access Manager - Server - 11.1.2.0.0 [Oracle_IDM1]**	■ Oracle Enterprise Manager for WebSphere - 11.1.1.0 [oracle_common] ■ Oracle Platform Security Service - 11.1.1.0 [Oracle_IDM1] ■ Oracle JRF for WebSphere - 11.1.1.0 [oracle_common] When you select the **Oracle Adaptive Access Manager - Server - 11.1.2.0.0 [Oracle_IDM1]** option, in addition to the templates mentioned above, the Oracle WSM Policy Manager - 11.1.1.0 [oracle_common] is also selected, by default.

Table 2–1 (Cont.) Oracle Identity and Access Management Product Templates

Template Name	Dependency
For Oracle Entitlements Server, the following templates are available: ■ **Oracle Entitlements Server for Admin Server- 11.1.1.0 [Oracle_IDM1]** ■ **Oracle Entitlements Server for Managed Server- 11.1.1.0 [Oracle_IDM1]**	■ Oracle Platform Security Service - 11.1.1.0 [Oracle_IDM1] ■ Oracle JRF for WebSphere - 11.1.1.0 [oracle_common]
For Oracle Privileged Account Manager, the following templates are available: ■ **Oracle Privileged Account Manager (Form auth-mode OINAV) - 11.1.2.1.0 [Oracle_IDM1]** ■ **Oracle Privileged Account Manager (Client-Cert auth-mode OINAV) - 11.1.2.1.0 [Oracle_IDM1]**	When you select the **Oracle Privileged Account Manager (Form auth-mode OINAV) - 11.1.2.1.0 [Oracle_IDM1]** option, the following options are also selected, by default: ■ Oracle Identity Navigator (Form auth-mode) - 11.1.2.1.0 [Oracle_IDM1] ■ Oracle Platform Security Service - 11.1.1.0 [Oracle_IDM1] ■ Oracle JRF for WebSphere - 11.1.1.0 [oracle_common] When you select the **Oracle Privileged Account Manager (Client-Cert auth-mode OINAV) - 11.1.2.1.0 [Oracle_IDM1]** option, the following options are also selected, by default: ■ Oracle Identity Navigator (Client-Cert auth-mode) - 11.1.2.1.0 [Oracle_IDM1] ■ Oracle Platform Security Service - 11.1.1.0 [Oracle_IDM1] ■ Oracle JRF for WebSphere - 11.1.1.0 [oracle_common]
For Oracle Identity Navigator, the following templates are available: ■ **Oracle Identity Navigator (Form auth-mode) - 11.1.2.1.0 [Oracle_IDM1]** ■ **Oracle Identity Navigator (Client-Cert auth-mode) - 11.1.2.1.0 [Oracle_IDM1]**	■ Oracle Platform Security Service - 11.1.1.0 [Oracle_IDM1] ■ Oracle JRF for WebSphere - 11.1.1.0 [oracle_common]

Select the required templates, and click **Next**.

6. On the Configure JDBC Component Schema screen, you can select the required component schemas to modify.

   You can set values for Schema Owner, Schema Password, Database and Service, Host Name, and Port. Click **Next**. The Test JDBC Component Schema screen appears. After the test succeeds, click **Next**. The Select Optional Configuration screen appears.

Note: In the Test JDBC Component Schema screen, if the schema test connectivity for Oracle Identity Manager fails, ignore the error message and proceed.

7. On the Select Optional Configuration screen, you can configure the following:

- Application Servers, Clusters and End Points
- Deployments and Services
- JDBC

Select the desired options, and click **Next**.

> **Note:** Steps 8-11 will appear based on the options that you select on the Select Optional Configuration screen.

8. Optional step: Configure the Application Server parameters.
9. Optional step: Configure Clusters, as required.
10. Optional step: Configure End Points, as required.
11. Optional step: Select Deployments, such as applications, libraries, and Services to target them to a particular cluster or server.
12. On the Configuration Summary screen, you can view summary of your configuration for deployments, application, and service. Review your configuration summary, and click **Create** to configure a new IBM WebSphere cell.

A new IBM WebSphere cell is created in the *WAS_HOME*/profiles/Dmgr01/config/cells directory (on UNIX).

> **Note:** If you are configuring Oracle Identity Manager, you must run the Oracle Universal Installer Configuration Assistant after configuring a WebSphere cell, to configure Oracle Identity Manager Server, Oracle Identity Manager Design Console, and Oracle Identity Manager Remote Manager. For more information, see Section 4.3.1, "Configuring Oracle Identity Manager for Single-Node Setup".

2.9 Task 9: Configure the Database Security Store

You must run the configureSecurityStoreWas.py script to configure the Database Security Store. The configureSecurityStoreWas.py script is located in the *IAM_HOME*/common/bin directory. You can use the -h option for help information about using the script.

The policy re-association migrates the OPSS security store from a source to a target LDAP- or DB-based store, and it resets the default policy and credential services to the target repository.

1. To perform policy re-association changes on IBM WebSphere, complete the following steps:
   a. Go to the *IAM_HOME*/common/bin directory.
   b. Run the following WSADMIN command to perform offline policy re-association:

      On UNIX operating systems:

      ```
      ./wsadmin.sh -lang jython -profileName DEPLOYMENT_MANAGER_PROFILE_NAME -f
      IAM_HOME/common/tools/configureSecurityStoreWas.py -d
      PATH_TO_DEPLOYMENT_MANAGER_CELL_DIRECTORY -t DB_ORACLE -j cn=jpsroot -m
      create --passcode OPSS_SCHEMA_PASSWORD --config IAM
      ```

For example:

```
./wsadmin.sh -lang jython -profileName Dmgr01 -f
IAM_HOME/common/tools/configureSecurityStoreWas.py -d
IAM_HOME/was/install/was6076/profiles/Dmgr01/config/cells/DefaultCell01 -t
DB_ORACLE -j cn=jpsroot -m create --passcode opsschemapassword --config IAM
```

Review the generated output and verify that no error is reported during policy re-association.

2. To verify the re-association of policy, complete the following steps:
   a. Log in to Oracle Identity System Administration. The log in must be successful.
   b. Log in to WAS Administration Console. The login must be successful.
   c. Log in to Oracle Enterprise Manager, and go to **Websphere Cell**, **Security**, **Security Provider Configuration**. Verify that the Store Type is Oracle Database pointing to jdbc/OPSSDBDS jndi.
   d. You can also run the following `WSADMIN` command to verify the re-association of policy

      On UNIX operating systems:

      ```
      ./wsadmin.sh -lang jython -profileName DEPLOYMENT_MANAGER_PROFILE_NAME -f
      IAM_HOME/common/tools/configureSecurityStoreWas.py -d
      PATH_TO_DEPLOYMENT_MANAGER_CELL_DIRECTORY -t DB_ORACLE -j cn=jpsroot -m
      validate --passcode OPSS_SCHEMA_PASSWORD
      ```

      For example:

      ```
      ./wsadmin.sh -lang jython -profileName Dmgr01 -f
      IAM_HOME/common/tools/configureSecurityStoreWas.py -d
      IAM_HOME/was/install/was6076/profiles/Dmgr01/config/cells/DefaultCell01 -t
      DB_ORACLE -j cn=jpsroot -m validate --passcode opsschemapassword
      ```

3. Stop the Node.

   Navigate to the following directory in the IBM WebSphere home and enter the following command:

   On UNIX operating systems:

   ```
   profiles/Server_profileName/bin/stopNode.sh
   ```

   For example:

   ```
   /disk01/IBM/WebSphere/AppServer/profiles
           /Custom01/bin/stopNode.sh
   ```

2.10 Task 10: Configure the Identity Store

On IBM WebSphere, Oracle Platform Security Services supports LDAP-based registries only. It does not support WebSphere's built-in file-based user registry.

You must complete the steps mentioned below to configure the OID store for Oracle Platform Security Services.

Note: The steps for configuring the Identity Store described below should be executed only once.

The steps provided in this section are not required if you are planning to integrate Oracle Access Manager and Oracle Identity Manager in the same WebSphere cell.

1. Start the Deployment Manager:

   Navigate to the following directory in the IBM WebSphere home and enter the following command:

   On UNIX operating systems:

   ```
   profiles/dmgr_profileName/bin/startManager.sh
   ```

   For example:

   ```
   /disk01/IBM/WebSphere/AppServer/profiles
          /Dmgr01/bin/startManager.sh
   ```

   Note: If you are running the `startManager.sh` (or `startManager.bat`) command from `WAS_HOME/bin` directory, you must specify the parameter `-profileName`.

   For example, on a UNIX operating system:

   ```
   WAS_HOME/bin/startManager.sh -profileName dmgr_profileName
   ```

2. `cd <oracle_common>/common/bin`

3. Run the following `wsadmin` command:

   ```
   ./wsadmin.sh -conntype SOAP -port <port_number> -user
   <username> -password <passwd>
   ```

   The port details are available in the `$WAS_HOME/profiles/Dmgr01/logs/AboutThisProfile.txt` file.

   You must use the same credentials that you provided when setting up the WAS cell.

4. `Opss.configureIdentityStore(propsFileLoc="<location of properties file>")`

   A sample properties file is provided below:

   ```
   user.search.bases=cn=Users,dc=myhost,dc=mycompany,dc=com
   group.search.bases=cn=Groups,dc=myhost,dc=mycompany,dc=com
   subscriber.name=dc=myhost,dc=mycompany,dc=com
   ldap.host=ldaphost.mycompany.com
   ldap.port=3333
   # admin.id must be the full DN of the user in the LDAP
   admin.id=cn=orcladmin,cn=Users,dc=myhost,dc=mycompany,dc=com
   admin.pass=welcome1
   user.filter=(&(uid=%v)(objectclass=person))
   group.filter=(&(cn=%v)(objectclass=groupofuniquenames))
   user.id.map=*:uid
   group.id.map=*:cn
   group.member.id.map=groupofuniquenames:uniquemember
   ssl=false
   ```

```
# primary.admin.id indicates a user who has admin permissions in the LDAP, must
be the name of the user, for example, for user "cn=tom", the primary.admin.id
is "tom"
primary.admin.id=orcladmin
# optional, default to "OID"
idstore.type=OID
# Optional properties for JPS LDAP identity store can also be configured in the
file.
username.attr=cn
user.object.classes=person
```

> **Note:** If you are an Oracle Privileged Account Manager user, then you must skip step 5 and continue with the steps described in Section 11.2.2, "Starting Oracle Privileged Account Manager on IBM WebSphere".

5. Stop and restart the Deployment Manager. While stopping the Deployment Manager, use the credentials used while setting up the WAS cell. While restarting the server, use the OID credentials as mentioned in `primary.admin.id` of the properties file.

2.11 Task 11: Start the IBM WebSphere Servers

After you finish configuring the Oracle Fusion Middleware software successfully, you can start the IBM WebSphere Deployment Manager, Node, and Servers.

The following procedure shows the sequence you must use to start the deployment manager, the node, and the servers in the cell.

> **Note:** If you have already started the Deployment Manager, then skip step 1.

In the following examples, replace the names of the deployment manager and profile name with the values you entered in the Configuration Wizard in Section 2.8, "Task 8: Configure Your Oracle Identity and Access Management Components in a New IBM WebSphere Cell":

1. Start the Deployment Manager:

   Navigate to the following directory in the IBM WebSphere home and enter the following command:

   On UNIX operating systems:

   `profiles/`*`dmgr_profileName`*`/bin/startManager.sh`

   For example:

   ```
   /disk01/IBM/WebSphere/AppServer/profiles
           /Dmgr01/bin/startManager.sh
   ```

Note: If you are running the `startManager.sh` (or `startManager.bat`) command from `WAS_HOME/bin` directory, you must specify the parameter `-profileName`.

For example, on a UNIX operating system:

```
WAS_HOME/bin/startManager.sh -profileName dmgr_profileName
```

2. Synchronize the node:

   Navigate to the following directory in the IBM WebSphere home and enter the following command:

   On UNIX operating systems:

   ```
   profiles/Server_profile_name/bin/syncNode.sh host_name SOAP_Port -username
   admin_user -password admin_password
   ```

   For example:

   ```
   /disk01/IBM/WebSphere/AppServer/profiles/Custom01/bin/syncNode.sh
   myhost.mycompany.com 8879 -username wasadmin -password welcome1
   ```

3. Start the node:

   Navigate to the following directory in the IBM WebSphere home and enter the following command:

   On UNIX operating systems:

   ```
   profiles/Server_profile_name/bin/startNode.sh
   ```

   For example:

   ```
   /disk01/IBM/WebSphere/AppServer/profiles/Custom01/bin/startNode.sh
   ```

   Note: If you are running the `startNode.sh` (or `startNode.bat`) command from `WAS_HOME/bin` directory, you must specify the parameter `-profileName`.

   For example, on a UNIX operating system:

   ```
   WAS_HOME/bin/startNode.sh -profileName Server_profileName
   ```

4. Start the OracleAdminServer server:

   Navigate to the following directory in the IBM WebSphere home and enter the following command:

   On UNIX operating systems:

   ```
   profiles/Server_profile_name/bin/startServer.sh OracleAdminServer
   ```

   For example:

   ```
   /disk01/IBM/WebSphere/AppServer/profiles/Custom01/bin/startServer.sh
            OracleAdminServer
   ```

Note: If you are running the `startServer.sh` (or `startServer.bat`) command from `WAS_HOME/bin` directory, you must specify the parameter `-profileName`.

For example, on a UNIX operating system:

```
WAS_HOME/bin/startServer.sh OracleAdminServer -profileName
Server_profileName
```

5. Start any additional servers that were configured as part of your IBM WebSphere cell.

   After you start the OracleAdminServer, you can start the other servers using the IBM WebSphere Administrative Console or Oracle Enterprise Manager Fusion Middleware Control. For more information, see Section 3.1, "Summary of the Oracle Fusion Middleware Management Tools on IBM WebSphere".

   Alternatively, you can use the startServer script, as follows:

   Navigate to the following directory in the IBM WebSphere home and enter the following command:

   On UNIX operating systems:

   ```
   profiles/Server_profile_name/bin/startServer.sh server_name
   ```

   For example, for an Oracle Access Manager cell on a UNIX operating system:

   ```
   /disk01/IBM/WebSphere/AppServer/profiles
           /Custom01/bin/startServer.sh oam_server1
   ```

   Note: If you are running the `startServer.sh` (or `startServer.bat`) command from `WAS_HOME/bin` directory, you must specify the parameter `-profileName`.

   For example, on a UNIX operating system:

   ```
   WAS_HOME/bin/startServer.sh server_name -profileName
   Server_profileName
   ```

   The typical servers that are configured for each of the Oracle Fusion Middleware components are listed in Table 2–2.

Table 2–2 Typical Oracle Fusion Middleware Component-Specific Managed Servers in an IBM WebSphere Cell

Component	Typical Managed Servers
Oracle SOA Suite	soa_server1, oam_server1
Oracle Identity and Access Management Suite	oam_server1, oim_server1, opam_server1

2.12 Task 12: Verify the Configuration of the IBM WebSphere Cell

To verify the installation, use the IBM WebSphere Administration Console and Oracle Enterprise Manager Fusion Middleware Control to verify that the management tools are working and the servers are up and running.

Refer to Section 3.1, "Summary of the Oracle Fusion Middleware Management Tools on IBM WebSphere" for more information on locating the URLs for these Web-based management tools.

> **Note:** For information about managing the Oracle Identity and Access Management products, see the component-specific chapters in this guide.

Special Instructions for Oracle Access Management Users

Oracle Access Management Administration Console will be deployed on `OracleAdminServer` and Oracle Access Management Runtime will be deployed on `oam_server1` by default.

You can access the Oracle Access Management console using the following URL:

`http://`*`WAS_HOST`*`:`*`OAM_AdminServer_Port`*`/oamconsole`

3

Managing Oracle Identity and Access Management Suite on IBM WebSphere

This chapter provides basic information about managing Oracle Identity and Access Management Suite on IBM WebSphere. This chapter contains the following topics:

- Summary of the Oracle Fusion Middleware Management Tools on IBM WebSphere
- Basic Administration Tasks on IBM WebSphere
- Deploying Applications on IBM WebSphere
- Configuring Oracle Fusion Middleware High Availability on IBM WebSphere

3.1 Summary of the Oracle Fusion Middleware Management Tools on IBM WebSphere

After you install and configure Oracle Identity and Access Management Suite on IBM WebSphere, you can verify the configuration, and monitor and manage the components of the Oracle Identity and Access Management Suite installation, using one of several management tools.

The following sections introduce the management tools:

- Using the WebSphere Administrative Console
- Using Oracle Enterprise Manager Fusion Middleware Control
- Using the Oracle Fusion Middleware wsadmin Commands

3.1.1 Using the WebSphere Administrative Console

This section contains the following topics:

- About the IBM WebSphere Administrative Console
- Locating the Port Number and URL of the IBM WebSphere Administrative Console

3.1.1.1 About the IBM WebSphere Administrative Console

The IBM WebSphere Administrative Console, also known as the IBM WebSphere Integrated Solutions Console, provides a Web-based interface for managing the IBM WebSphere environment.

You can use the IBM WebSphere Administrative Console to monitor and manage the cell and the servers on which the Oracle Identity and Access Management Suite products are deployed.

For more information about the IBM WebSphere Administrative Console, see the IBM WebSphere documentation, as well as the online help for the console.

3.1.1.2 Locating the Port Number and URL of the IBM WebSphere Administrative Console

Before you can display the IBM WebSphere Administrative Console, you must identify the port number on which is running.

To locate the port number and URL of the IBM WebSphere Administrative Console:

1. In a text editor, open the following properties file:

   WAS_HOME/profiles/deployment_mgr_name/properties/portdef.props

2. Locate the value of the WC_Adminhost property.
3. Open a browser and enter the following URL:

   http://*hostname*:*WC_Adminhost_port*/ibm/console

   For example:

   http://host42.example.com:9002/ibm/console

3.1.2 Using Oracle Enterprise Manager Fusion Middleware Control

This section contains the following topics:

- About Oracle Enterprise Manager Fusion Middleware Control
- Locating the Port Number and URL for Fusion Middleware Control
- Displaying Fusion Middleware Control
- Viewing an IBM WebSphere Cell from Fusion Middleware Control
- Viewing an IBM WebSphere Server from Fusion Middleware Control
- Viewing an IBM WebSphere Application Deployment from Fusion Middleware Control
- Performing Oracle Fusion Middleware-Specific Administration Tasks for the Cell
- Differences When Using Fusion Middleware Control on IBM WebSphere

3.1.2.1 About Oracle Enterprise Manager Fusion Middleware Control

Oracle Enterprise Manager Fusion Middleware Control is a Web browser-based, graphical user interface that you can use to monitor and administer Oracle Fusion Middleware.

Fusion Middleware Control organizes a wide variety of performance data and administrative functions into distinct, Web-based home pages for cells, servers, components, and applications. The Fusion Middleware Control home pages make it easy to locate the most important monitoring data and the most commonly used administrative functions from your Web browser.

For more information, refer to "Getting Started Using Oracle Enterprise Manager Fusion Middleware Control" in the *Oracle Fusion Middleware Administrator's Guide*.

Note that the information provided in the *Oracle Fusion Middleware Administrator's Guide* is specific to using Fusion Middleware Control on Oracle WebLogic Server. For more information, see Section 3.1.2.8, "Differences When Using Fusion Middleware Control on IBM WebSphere".

3.1.2.2 Locating the Port Number and URL for Fusion Middleware Control

To locate the port number for Fusion Middleware Control:

1. Use your Web browser to open the IBM WebSphere Administrative Console.
2. In the navigation panel, select **Servers > Server Types > WebSphere application servers**.
3. Click **OracleAdminServer** to display the configuration properties of the server.
4. In the Communications section of the resulting page, expand **Ports** to list the important port values for the OracleAdminServer.
5. Locate the value of the `WC_Defaulthost` port.

3.1.2.3 Displaying Fusion Middleware Control

To display Fusion Middleware Control, create a new Web browser window or tab, and enter the following URL:

`http://`*`hostname`*`:`*`WC_Defaulthost_port`*`/em`

For example:

```
http://host42.example.com:9002/em
```

Log in to Fusion Middleware Control using the same administration credentials you use when logging in to the IBM WebSphere Administrative Console.

3.1.2.4 Viewing an IBM WebSphere Cell from Fusion Middleware Control

From Fusion Middleware Control, you can manage the Oracle Fusion Middleware products that you have installed and configured as part of the IBM WebSphere cell.

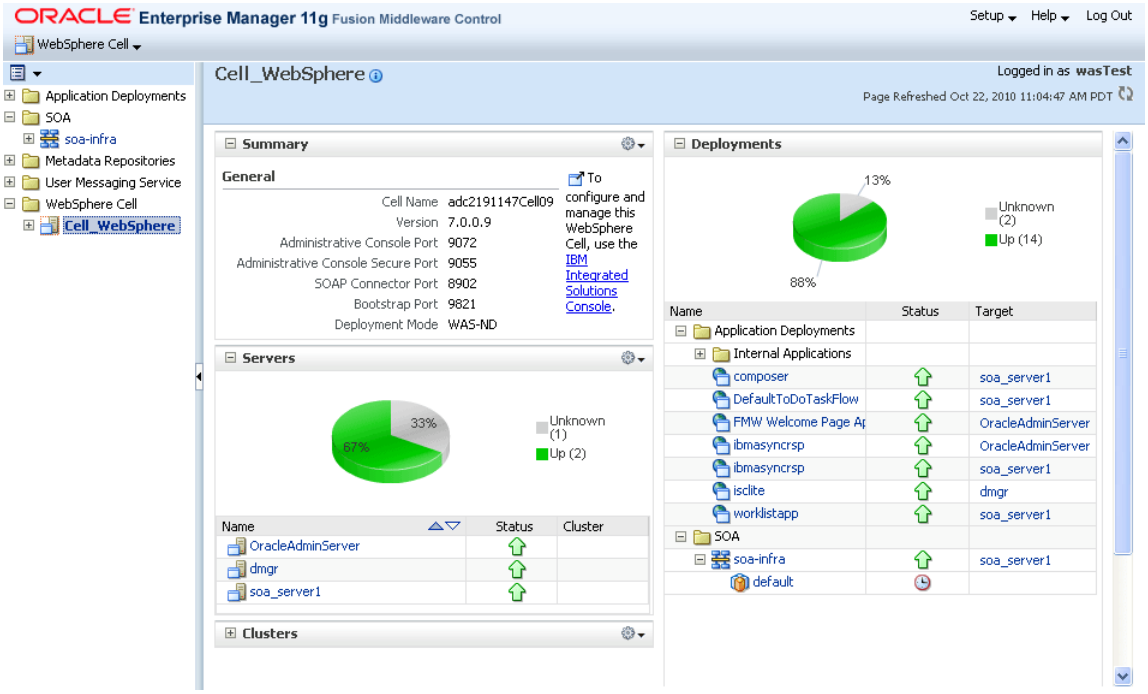
When you first log in to Fusion Middleware Control, the IBM WebSphere Cell home page appears (Figure 3–1). From this page, you can view the servers, applications, and clusters that are associated with the cell.

You can also navigate to the management pages for the Oracle Identity and Access Management Suite components you have installed and configured.

For more information about how to navigate within Oracle Enterprise Manager Fusion Middleware Control, see "Navigating Within Fusion Middleware Control" in the *Oracle Fusion Middleware Administrator's Guide*.

From the **WebSphere Cell** menu, you can perform Oracle Fusion Middleware administration functions. For help on a menu command, select the command, and then select **Enterprise Manager Help** from the **Help** menu on the resulting page.

Figure 3–1 Viewing the IBM WebSphere Cell from Fusion Middleware Control



3.1.2.5 Viewing an IBM WebSphere Server from Fusion Middleware Control

Each server in an IBM WebSphere cell has its own home page in Fusion Middleware Control.

To view the home page for a specific server:

1. In the Fusion Middleware Control Target Navigation Pane, expand the **WebSphere Cell** folder.
2. Expand the cell name, and click the server name.

From the WebSphere Application Server home page you can view general information about the server, display the IBM WebSphere Administrative Console, and view the status of the applications deployed to the server.

For a description of the features and options available on the IBM WebSphere Application Server home page, see Section A.1, "Understanding the Information on the IBM WebSphere Cell Home Page".

From the **WebSphere Application Server** menu, you can perform Oracle Fusion Middleware administration functions. For help on a menu command, select the command, and then--on the resulting page--select **Enterprise Manager Help** from the **Help** menu.

3.1.2.6 Viewing an IBM WebSphere Application Deployment from Fusion Middleware Control

Each application deployment in your IBM WebSphere cell has its own home page in Fusion Middleware Control.

An application deployment is an instance of a deployed application. For example, if you deploy the same application to two servers, then you have two deployments of the same application.

To view an application deployment in Fusion Middleware Control:

1. Navigate to the IBM WebSphere cell home page or an IBM WebSphere application server home page.
2. Locate the list of application deployments, and click the application name.

For a description of the features and options available on the IBM WebSphere Application Server home page, see Section A.3, "Understanding the Information on the IBM WebSphere Application Deployment Home Page".

From the **Application Deployment** menu, you can perform Oracle Fusion Middleware administration functions. For help on a menu command, select the command, and then--on the resulting page--select **Enterprise Manager Help** from the **Help** menu.

3.1.2.7 Performing Oracle Fusion Middleware-Specific Administration Tasks for the Cell

Oracle Enterprise Manager Fusion Middleware Control, when used with the IBM WebSphere Administrative Console, provides you with the tools you need to manage Oracle Fusion Middleware when it is installed and configured on IBM WebSphere.

You perform common IBM WebSphere administration tasks from the IBM WebSphere Administrative Console, and you can perform administration tasks that are specific to Oracle Fusion Middleware from the Fusion Middleware Control home pages.

3.1.2.8 Differences When Using Fusion Middleware Control on IBM WebSphere

When you use Oracle Enterprise Manager Fusion Middleware Control to manage Oracle Fusion Middleware products on IBM WebSphere, you will notice some differences from the features and functionality available when using it with Oracle WebLogic Server.

The differences vary, depending on whether you are using IBM WebSphere - Network Deployment (ND) or IBM WebSphere Application Server (AS).

Some specific menu commands and features available in an Oracle WebLogic Server environment are not available when you are managing Oracle Fusion Middleware in an IBM WebSphere environment. If a command or feature is not available, then it is not supported in the IBM WebSphere environment.

Table 3–1 describes some of the differences you might experience when managing Oracle Fusion Middleware on an IBM WebSphere cell, as opposed to an Oracle WebLogic Server domain.

Table 3–1 Summary of Differences When Managing IBM WebSphere As Opposed to Oracle WebLogic Server Domain

Feature or Functional Area	Differences on IBM WebSphere ND	Additional differences on IBM WebSphere AS
Managing an Oracle Fusion Middleware Farm	There is no concept of an Oracle Fusion Middleware farm when you are running on IBM WebSphere; instead, the first page that Fusion Middleware Control displays when you log in is the IBM WebSphere Cell home page. From the Cell home page, you can navigate to the other home pages that have monitoring and administrative features for the Oracle Fusion Middleware components. You can also link easily to the IBM WebSphere Administrative Console when necessary.	Same as ND.
Monitoring IBM WebSphere from Fusion Middleware Control	There are no IBM WebSphere performance metrics and no performance summary page for the IBM WebSphere cell or server pages.	Same as ND.
Deployment of Fusion Middleware Control in the cell	When you are managing an IBM WebSphere cell, Fusion Middleware Control runs on the OracleAdminServer, which is created when you configure Oracle Fusion Middleware products using the Configuration Wizard. You can then use Fusion Middleware Control to manage all the servers and applications deployed to the servers in the cell.	Single instance management only. Fusion Middleware Control must be running on the server that is being managed.
Application deployment from Fusion Middleware Control	You cannot deploy applications from Fusion Middleware Control on IBM WebSphere. Instead, you can use the IBM WebSphere Administrative Console or deploy directly from Oracle JDeveloper. For more information, see Section 3.3, "Deploying Applications on IBM WebSphere".	Same as ND.

3.1.3 Using the Oracle Fusion Middleware wsadmin Commands

The WebSphere Application Server wsadmin tool is a command-line utility that can be run in two modes:

- Interactive mode, where you enter commands directly in the shell
- Scripting mode, where you specify a Jython (.py) script on the command line

The examples in this chapter assume you are using interactive mode and the wsadmin command-line shell. For information about using scripting mode, refer to the IBM WebSphere documentation.

You can use the wsadmin tool to manage WebSphere Application Server as well as the configuration, application deployment, and server run-time operations.

Oracle Fusion Middleware provides a set of wsadmin commands that are used exclusively to manage the Oracle Fusion Middleware components that are configured in your IBM WebSphere cell.

For more information about the Oracle Fusion Middleware wsadmin commands and how to use them, refer to the following sections:

- Section 3.1.3.1, "About the Oracle Fusion Middleware wsadmin Command-Line Shell"
- Section 3.1.3.2, "Starting the Oracle Fusion Middleware wsadmin Command-Line Shell and Connecting to the Deployment Manager"
- Section 3.1.3.3, "Using the Oracle Fusion Middleware wsadmin Command-Line Online Help"
- Section 3.1.3.4, "Differences Between the wsadmin Commands and the WebLogic Scripting Tool (WLST) Commands"
- Section 3.1.3.5, "Differences Between Oracle Fusion Middleware wsadmin Commands and IBM WebSphere Wsadmin Commands"

3.1.3.1 About the Oracle Fusion Middleware wsadmin Command-Line Shell

A command-line shell is a command-line environment where a specific set of commands are available and supported. Within the shell, you can run these commands, obtain help on the commands, and perform administration tasks that are specific to the environment you are managing.

The Oracle Fusion Middleware wsadmin command-line shell is an Oracle Fusion Middleware-specific implementation of the wsadmin tool. From this shell, you can:

- Run the Oracle Fusion Middleware-specific wsadmin commands.
- List the available Oracle Fusion Middleware wsadmin commands.
- Obtain online help for the Oracle Fusion Middleware wsadmin commands.

3.1.3.2 Starting the Oracle Fusion Middleware wsadmin Command-Line Shell and Connecting to the Deployment Manager

Start the Oracle Fusion Middleware wsadmin command-line shell from `common/bin` directory of the Oracle home of the product you are managing.

For a complete list of the arguments you can use when starting wsadmin, refer to the IBM WebSphere documentation.

In a typical Oracle Fusion Middleware wsadmin session, you will want to specify the profile name and connect to the deployment manager of the cell you are managing.

> **Note:** The following examples assume you have already installed and configured an IBM WebSphere cell, using the instructions in Chapter 2, "Installing and Configuring Oracle Identity and Access Management on IBM WebSphere".
>
> Alternatively, if you want to run the wsadmin shell before configuring a cell, refer to "Prerequisite Environment Setup" in the *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server*.

The following example shows how you can start the wsadmin shell.

Note that this example assumes the IBM WebSphere Deployment Manager is on the local host and is using the default SOAP port. If the Deployment Manager is on a different host, then you will need to specify the host and port using additional command-line arguments. For more information, see the IBM WebSphere documentation and wsadmin command-line help.

To start the wsadmin shell, use this command syntax:

On UNIX operating systems:

```
ORACLE_HOME/common/bin/wsadmin.sh
-profileName profilename
-connType SOAP
-user admin_user
-password admin_password
```

The following example uses the complete path for the wsadmin script on a UNIX operating system:

```
/disk01/Oracle/Middleware/Oracle_SOA1/common/bin/wsadmin.sh -profileName soaDmgr05
```

Example 3–1 shows an example of starting the Oracle Fusion Middleware wsadmin command-line shell after you have changed directory to the `common/bin` directory in the Oracle Fusion Middleware product Oracle home on a UNIX system. The example also shows some typical output messages when you start the shell.

Example 3–1 Starting the Oracle Fusion Middleware Wsadmin Command-Line Shell

```
 ./wsadmin.sh -profileName soaDmgr05 -connType SOAP -user wasTest -password welcome1
IN SOA WsadminEnv.sh...
WSADMIN_CLASSPATH=:/scratch/wasTest/mwhome_soa_100719/oracle_common/soa/modules/oracle.soa.mgmt
_11.1.1/soa-infra-mgmt.jar:/scratch/wasTest/mwhome_soa_100719/ ...
    .
    .
    .
WASX7209I: Connected to process "dmgr" on node soaCellManager05 using SOAP connector;  The type of
process is: DeploymentManager

CFGFWK-24021:  OracleHelp loaded.
CFGFWK-24022:  For information on Oracle modules enter 'print OracleHelp.help()'
WASX7031I: For help, enter: "print Help.help()"
wsadmin>
```

3.1.3.3 Using the Oracle Fusion Middleware wsadmin Command-Line Online Help

The following sections describe some key features of the Oracle Fusion Middleware wsadmin command-line shell:

- Listing the Oracle Fusion Middleware wsadmin Command Categories
- Listing the Commands Within an Oracle Fusion Middleware wsadmin Command-Line Category
- Getting Help on a Specific Oracle Fusion Middleware wsadmin Command

3.1.3.3.1 Listing the Oracle Fusion Middleware wsadmin Command Categories To list the available categories of Oracle Fusion Middleware commands in the Oracle Fusion Middleware wsadmin command-line shell, use the following command:

```
wsadmin>print OracleHelp.help()
```

Example 3–2 shows an example of the output of the `print OracleHelp.help()` command when you run it from the Oracle Common home.

If you run the command from an Oracle Fusion Middleware component Oracle home (for example, an Oracle SOA Suite, Oracle WebCenter Portal, or Oracle WebCenter Content Oracle home), then the output will include information on the component-specific wsadmin commands.

Example 3–2 Listing the Available Commands from the Oracle Fusion Middleware wsadmin Command-Line Shell

```
wsadmin>print OracleHelp.help()

ADFMAdmin                   ADFM Lifecycle Management Commands.
MDSAdmin                    MDS Lifecycle Management Commands.
OracleDFW                   Lists commands for FMW diagnostic framework.
OracleDMS                   Lists commands for FMW performance metrics and
                            events.
OracleHelp                  Provides help for Oracle modules.
OracleJRF                   Commands for configuring Managed Servers with
                            Oracle Java Required Files (JRF)
OracleLibOVDConfig          List commands for managing OVD configuration
OracleMWConfig              Oracle Middleware Configuration Tool.
OracleMWConfigUtilities     Oracle Middleware Configuration Tool Utilities.
OracleODL                   Lists commands for FMW diagnostic logging.
URLConnection               List Commands for managing ADF Based URL
                            Connections
WebServices                 Lists commands for Oracle WebServices Management.
audit                       Lists commands for Common Audit Framework
igfconfig                   List commands for manageing IGF configuration
opss                        Oracle platform security services Commands.
wsmManage                   Lists commands for Oracle WSM Policy Management.

wsadmin>
```

3.1.3.3.2 Listing the Commands Within an Oracle Fusion Middleware wsadmin Command-Line Category To list the commands associated with a particular category, enter the category name inside single quotation marks within the parentheses. For example:

```
wsadmin>print OracleHelp.help('OracleODL.help')
```

Example 3–3 shows an example of listing the commands in a particular category.

Example 3–3 Listing a Specific Category of Oracle Fusion Middleware wsadmin Commands

```
wsadmin>print OracleHelp.help('OracleODL')

Commands for FMW diagnostic logging

configureLogHandler     Configure Java logging handlers.
displayLogs             Search and display the contents of diagnostic log
                        files.
getLogLevel             Returns the level of a given Java logger.
listLogHandlers         Lists Java log handlers configuration.
listLoggers             Lists Java loggers and their levels.
listLogs                Lists log files for FMW components.
setLogLevel             Sets the level of a given Java logger.

wsadmin>
```

3.1.3.3.3 Getting Help on a Specific Oracle Fusion Middleware wsadmin Command To get help on a specific Oracle Fusion Middleware wsadmin command:

```
wsadmin>print OracleHelp.help(category.command)
```

Example 3–4 shows an example of the online help output for a specific Oracle Diagnostic Logging command.

Example 3–4 Example of Online Help for a Specific Oracle Fusion Middleware wsadmin Command

```
wsadmin>print OracleHelp.help('OracleODL.listLogs')

Lists log files for FMW components.

Returns a PyArray with one element for each log. The elements of the
array are javax.management.openmbean.CompositeData objects describing
each log.

Syntax:
listLogs([options])
- options: optional list of name-value pairs.

  o target: the name of a Weblogic server, or an OPMN managed FMW component.
    For an OPMN managed component the syntax for the target is
    "opmn:<instance-name>/<component-name>".

    The target argument can be an array of strings containing one or more
    targets. In connected mode the default target includes all running
    Weblogic servers in the domain that have JRF enabled.
    In disconnected mode there is no default, the target option is required.

  o oracleInstance: defines the path to the ORACLE_INSTANCE (or Weblogic
    domain home). The command will be executed in disconnected mode when
    this parameter is used.

  o unit: defines the unit to use for reporting file size. Valid
    values are B (bytes), K (kilobytes), M (megabytes), G (gigabytes),
    or H (display size in a human-readable form, similar to Unix's "ls
    -h" option). The default value is H.

  o fullTime: a Jython Boolean value. If true, reports the full time
    for the log file last modified time. Otherwise displays a short
     version of the time. The default value is false.

Example:
1. listLogs()
2. listLogs(target="server1")
3. listLogs(target="opmn:instance1/ohs1")
4. listLogs(oracleInstance="/middleware/user_projects/domains/base_domain",
target="server1")
wsadmin>
```

3.1.3.4 Differences Between the wsadmin Commands and the WebLogic Scripting Tool (WLST) Commands

Many of the Oracle Fusion Middleware wsadmin commands that are supported for IBM WebSphere have equivalent WebLogic Scripting Tool (WLST) commands.

To find information about the equivalent WLST command, refer to the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

To list all the Oracle Fusion Middleware wsadmin command categories (or modules) use the `OracleHelp.help()` command, as shown in Example 3–2.

In many cases, the only difference between the WLST command and the wsadmin command is that you must prefix each wsadmin command with the category name. Example 3–6 shows how you might use the `listLoggers` command in WLST. Example 3–7 shows the same command in wsadmin.

Example 3–5 Using the ListLoggers Command in WLST

```
wls:/base_domain/serverConfig> listLoggers(pattern="oracle.dms.*")
-----------------------+-----------------
Logger                 | Level
-----------------------+-----------------
oracle.dms             | <Inherited>
oracle.dms.aggregator  | <Inherited>
oracle.dms.collector   | <Inherited>
oracle.dms.context     | <Inherited>
oracle.dms.event       | <Inherited>
oracle.dms.instrument  | <Inherited>
oracle.dms.jrockit.jfr | <Inherited>
oracle.dms.reporter    | <Inherited>
oracle.dms.trace       | <Inherited>
oracle.dms.translation | <Inherited>
oracle.dms.util        | <Inherited>
wls:/base_domain/serverConfig>
```

Example 3–6 Using the ListLoggers Command in Wsadmin

```
wsadmin>OracleODL.listLoggers(pattern="oracle.dms.*")
-----------------------+-----------------
Logger                 | Level
-----------------------+-----------------
oracle.dms             | WARNING:1
oracle.dms.aggregator  | NOTIFICATION:1
oracle.dms.collector   | NOTIFICATION:1
oracle.dms.context     | NOTIFICATION:1
oracle.dms.event       | NOTIFICATION:1
oracle.dms.instrument  | NOTIFICATION:1
oracle.dms.reporter    | NOTIFICATION:1
oracle.dms.trace       | NOTIFICATION:1
oracle.dms.translation | NOTIFICATION:1
oracle.dms.util        | NOTIFICATION:1
wsadmin>
```

3.1.3.5 Differences Between Oracle Fusion Middleware wsadmin Commands and IBM WebSphere Wsadmin Commands

Note the following difference between running Oracle Fusion Middleware wsadmin commands and the standard IBM WebSphere wsadmin commands:

- You must run the Oracle Fusion Middleware commands from the `common/bin` directory of the Oracle Fusion Middleware Oracle home.
- The Oracle Fusion Middleware wsadmin commands use the Jython scripting language exclusively.

3.2 Basic Administration Tasks on IBM WebSphere

The following sections provide information about some basic administration tasks you can perform when running Oracle Fusion Middleware on IBM WebSphere:

- Starting and Stopping Servers on IBM WebSphere
- Configuring Metadata Services (MDS) on IBM WebSphere
- Configuring Oracle Fusion Middleware Logging on IBM WebSphere
- Setting Up the Diagnostic Framework
- Creating a Data Source in an IBM WebSphere Cell

3.2.1 Starting and Stopping Servers on IBM WebSphere

There are two methods for starting and stopping the servers in your IBM WebSphere cell:

- Starting and Stopping IBM WebSphere Servers with Profile Scripts
- Starting and Stopping IBM WebSphere Servers with Fusion Middleware Control

3.2.1.1 Starting and Stopping IBM WebSphere Servers with Profile Scripts

Just as with any other IBM WebSphere cell, you can use profile scripts to start and stop the servers in a cell you configured for Oracle Fusion Middleware.

For example, to stop the OracleAdminServer, navigate to the following directory in the IBM WebSphere home, and enter the following command:

On UNIX operating systems:

```
profiles/profile_name/bin/stopServer.sh OracleAdminServer
         -profileName profileName
```

For example:

```
/disk01/IBM/WebSphere/ApplicationServer/profiles
        /Custom01/bin/stopServer.sh OracleAdminSErver
         -profileName Custom01
```

For examples of how to start the servers in your IBM WebSphere cell, see Section 2.11, "Task 11: Start the IBM WebSphere Servers".

For more information about the scripts that are generated for each profile, refer to the IBM WebSphere documentation.

3.2.1.2 Starting and Stopping IBM WebSphere Servers with Fusion Middleware Control

You can also stop and start IBM WebSphere servers from Oracle Enterprise Manager Fusion Middleware Control.

For example, to stop a server from Fusion Middleware Control:

1. Navigate to the Server home page.

   For more information, see Section 3.1.2.5, "Viewing an IBM WebSphere Server from Fusion Middleware Control".

2. From the **WebSphere Application Server** menu, select **Control**, and then select **Shut down**.

Fusion Middleware Control displays a confirmation dialog box.

3. Click **Shutdown**.

> **Note:** Fusion Middleware Control is deployed to the OracleAdminServer. As a result, if you stop the OracleAdminServer, then Fusion Middleware Control will be stopped, and you must use the profile scripts to start the servers.
>
> For more information, see Section 3.2.1.1, "Starting and Stopping IBM WebSphere Servers with Profile Scripts".

3.2.2 Configuring Metadata Services (MDS) on IBM WebSphere

On IBM WebSphere, you can manage Oracle Fusion Middleware Metadata Services (MDS) using Oracle Enterprise Manager Fusion Middleware Control and the `wsadmin` command-line utility, just as you can other Oracle Fusion Middleware components.

Refer to the following sections for more information about the differences from configuring MDS on Oracle WebLogic Server:

- Differences in MDS Command-Line Features on IBM WebSphere
- Differences in MDS Fusion Middleware Control Pages on IBM WebSphere

3.2.2.1 Differences in MDS Command-Line Features on IBM WebSphere

All the wsadmin commands you use to manage MDS on IBM WebSphere have equivalent WebLogic Scripting Tool (WLST) commands, which are documented in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

In addition, refer to the wsadmin online help for information about any differences between the MDS commands available in WLST and in wsadmin.

For example, note the following differences when using the `registerMetadataDBRepository` command on IBM WebSphere:

- The command has an additional parameter on IBM WebSphere (`authAlias`).
- The existing `targetServers` parameter allows you to specify a target WebSphere server or cluster for the repository, rather than a Oracle WebLogic Server instance.

For more information, see the following:

- Using the registerMetadataDBRepository authAlias parameter on IBM WebSphere.
- Using the registerMetadataDBRepository targetServers Parameter on IBM WebSphere
- More Information About the registerMetadaDBRepository Command on IBM WebSphere

3.2.2.1.1 Using the registerMetadataDBRepository authAlias parameter on IBM WebSphere

Use the `authAlias` argument to create or use an existing authentication alias for connecting to the database where the MDS schema resides. For example:

- If you do not provide a value for the `authAlias` parameter, then Oracle Fusion Middleware assumes that the authentication alias name is the same as the metadata repository name.

- If you provide a user name and password, then Oracle Fusion Middleware creates a new authentication alias either by using the value of the `authAlias` parameter as the alias name if it is provided, or by using the name of the metadata repository as alias name if the `authAlias` parameter is not provided.
- If you do not provide a user name and password, then Oracle Fusion Middleware assumes you want to connect to the database using the existing authentication alias, which is either the value of the `authAlias` parameter or the name of the metadata repository if the `authAlias` parameter is not provided.

3.2.2.1.2 Using the registerMetadataDBRepository targetServers Parameter on IBM WebSphere

Use the `targetServers` parameter to specify the WebSphere servers or clusters to which this repository will be registered. If this argument is not specified, then the repository will be registered only to the DeploymentManager.

The server or cluster must be specified in the form of specifying a configuration object in the wsadmin scripting tool. A configuration object can be specified as multiple `/type:name/` value pairs in the containment path string. For example:

```
'/Cell:myCell/Node:myNode/Server:myServer/'
```

The containment path must be a path that contains the correct hierarchical order.

To specify multiple servers or clusters, separate the names with a comma.

Note that if you later add additional servers or clusters to the cell, you must do one of the following to ensure that the repository is available from the new servers or clusters that were added after the initial registration of the repository:

- Use the `deregisterMetadataDBRepository` command to deregister the repository from all the initial targets, then run the `registerMetadataDBRepository` command again to reregister the repository with more targets. Note that the repository will be unavailable on all servers until you run the `registerMetadataDBRepository` command the second time.
- Manually create the repository on the new servers or clusters, using the exact same properties as the repository you created with the `registerMetadataDBRepository` command.

3.2.2.1.3 More Information About the registerMetadaDBRepository Command on IBM WebSphere

For more information about using the `registerMetadataDBRepository` command on IBM WebSphere, review the wsadmin online help for the command:

```
wsadmin> print MDSAdmin.help('registerMetadataDBRepository')
```

For more information about using wsadmin command-line online help, see Section 3.1.3.3, "Using the Oracle Fusion Middleware wsadmin Command-Line Online Help".

3.2.2.2 Differences in MDS Fusion Middleware Control Pages on IBM WebSphere

When you are using Fusion Middleware Control to manage the MDS repository on IBM WebSphere, there are some differences in the Fusion Middleware Control pages. These differences are due to the differences in the basic administration functions for Oracle WebLogic Server and IBM WebSphere.

For example:

- On Oracle WebLogic Server, the Metadata Repository home page includes a Targeted Servers region, which identifies Oracle WebLogic Server servers that can access the repository. This region is not available on IBM WebSphere.
- On IBM WebSphere, the Register Database-Based Metadata Repository page provides the ability to specify an authentication alias, which can be used to represent the credentials required to connect to the repository database.

3.2.3 Configuring Oracle Fusion Middleware Logging on IBM WebSphere

There are several ways to change the configuration of log files for the Oracle Fusion Middleware products when running with IBM WebSphere.

Consider the following when modifying the log configuration:

- To change the log levels, you can use the IBM WebSphere Administrative Console, Fusion Middleware Control, or the `OracleODL` commands in the Oracle Fusion Middleware wsadmin command-line shell.

  Note that in IBM WebSphere, `java.util.logging` is implemented differently than in Oracle WebLogic Server; specifically, child loggers do not inherit the log level property from the parent. However, you can change the log levels for a logger and its descendants, by using the wsadmin commands shown in Example 3–7.

  Note that in Example 3–7, the two spaces before the `OracleODL.setLogLevel` command are required. The spaces indicate that this line is a continuation of the previous line.
- To change other configuration properties, you can use Fusion Middleware Control, or the `OracleODL` commands in the wsadmin command line.
- The name of the log configuration file is `websphere-logging.xml`. Note, however, that you should not edit the file directly; you should use Fusion Middleware Control, the wsadmin command line, or the IBM WebSphere Administrative Console to modify the file.
- The main diagnostic log file is located in the following directory:

  SERVER_LOG_ROOT`/`*server_name*`-diagnostic.log`

  For more information about the SERVER_LOG_ROOT environment variable, see the IBM WebSphere documentation.

  Note that some Oracle Fusion Middleware components also generate their own logs, which are also stored in this location.

Example 3–7 Sample Oracle Fusion Middleware Wsadmin Script that Sets Logging Levels

```
wsadmin>myLoggers = OracleODL.listLoggers(pattern="oracle.dms.*")
----------------------+-----------------
Logger                | Level
----------------------+-----------------
oracle.dms            | WARNING:1
oracle.dms.aggregator | NOTIFICATION:1
oracle.dms.collector  | NOTIFICATION:1
oracle.dms.context    | NOTIFICATION:1
oracle.dms.event      | NOTIFICATION:1
oracle.dms.instrument | NOTIFICATION:1
oracle.dms.reporter   | NOTIFICATION:1
oracle.dms.trace      | NOTIFICATION:1
```

```
oracle.dms.translation | NOTIFICATION:1
oracle.dms.util        | NOTIFICATION:1
wsadmin>print myLoggers
{'oracle.dms.translation': 'NOTIFICATION:1', 'oracle.dms.context':
 'NOTIFICATION:1', 'oracle.dms.event': 'NOTIFICATION:1', 'oracle.dms':
 'NOTIFICATION:1', 'oracle.dms.util': 'NOTIFICATION:1', 'oracle.dms.aggregator':
 'NOTIFICATION:1', 'oracle.dms.reporter': 'NOTIFICATION:1', 'oracle.dms.trace':
 'NOTIFICATION:1', 'oracle.dms.instrument': 'NOTIFICATION:1',
 'oracle.dms.collector': 'NOTIFICATION:1'}
wsadmin> for loggerName in myLoggers.keys():
wsadmin>  OracleODL.setLogLevel(target="OracleAdminServer", logger=loggerName,
level="FINE")
wsadmin>
wsadmin>OracleODL.listLoggers(pattern="oracle.dms.*")
----------------------+-----------------
Logger                | Level
----------------------+-----------------
oracle.dms            | WARNING:1
oracle.dms.aggregator | TRACE:1
oracle.dms.collector  | TRACE:1
oracle.dms.context    | TRACE:1
oracle.dms.event      | TRACE:1
oracle.dms.instrument | TRACE:1
oracle.dms.reporter   | TRACE:1
oracle.dms.trace      | TRACE:1
oracle.dms.translation | TRACE:1
oracle.dms.util       | TRACE:1
```

3.2.4 Setting Up the Diagnostic Framework

Because the Automatic Diagnostic Repository (ADR) binaries are not automatically installed when Oracle Fusion Middleware is installed on IBM WebSphere, the Diagnostic Framework cannot access the ADR to store incidents.

To allow incident creation on IBM WebSphere, you must install the ADR binaries and configure each WebSphere server to point to those binaries.

Perform the following steps:

1. Download and install the Oracle Database Instant Client binaries version 11.2.0.1 from Oracle Technology Network (OTN).

   http://www.oracle.com/technology/software/tech/oci/instantclient/index.html

   Select your operating system, then select **Basic.**

2. Install the downloaded files on the host on which IBM WebSphere is running.

3. Configure the IBM Websphere server to set the system property `oracle.adr.home` to the location of the installed Oracle Database Instant Client binaries, using the WebSphere Integrated Solutions Console.

   For example, to set the property on distributed platforms:

   a. Expand **Servers,** then **Server Types.** Select **WebSphere application servers.**

   b. On the Application servers page, select the server.

   c. In the Server Infrastructure section of the server page, expand **Java and process management,** then select **Process Definition.**

   d. In the Process Definition page, select **Java Virtual Machine.**

e. Select **Custom Properties,** then click **New.**

f. For **Name,** enter `oracle.adr.home`.

g. For **Value,** enter the location of the installed files.

h. Click **Apply,** then **Save.**

3.2.5 Creating a Data Source in an IBM WebSphere Cell

Creating a data source is a common administration task, which is required when configuring certain aspects of your Oracle Fusion Middleware environment.

Data sources that connect to the product schemas installed by the Repository Creation Utility are created when you run the Configuration Wizard. However, there are other scenarios where you might need to create a data source--for example, you might need a data source for the applications you deploy.

To create a data source on IBM WebSphere, you can use the IBM WebSphere Administrative Console.

The following example shows how to create an IBM WebSphere data source for an Oracle database. Creating the database involves the following tasks:

- Task 1, "Create an authentication alias for the Oracle database you want to access"
- Task 2, "Create a JDBC data provider for the Oracle database"
- Task 3, "Modify the JDBC data provider to use the latest Oracle database classes"
- Task 4, "Create a JDBC data source that uses the Oracle database JDBC provider"
- Task 5, "Test the Data Source Connection"

Task 1 Create an authentication alias for the Oracle database you want to access

1. Log in to the IBM WebSphere Administrative Console and navigate to **Security** > **Global Security**.
2. On the Global Security page, select **Java Authentication and Authorization Service** > **J2C Authentication Data**.
3. Click **New**.
4. On the General Properties page enter the information shown in Table 3–2.
5. Save the new authentication alias to the master configuration.

Table 3–2 Authentication Alias General Properties for an Oracle Database Data Source

Element	Description
Alias	Enter a name for the alias. Use a name that identifies the purpose of the credentials assigned to the alias. For example, `OracleDBalias`.
User ID	Enter the Oracle database user name you will use to connect to the database. **Note:** Where required, also include the role. For example, if you are connecting as SYS, then enter the following in this field: `SYS as SYSDBA`
Password	Enter the password for the database user.

Table 3–2 (Cont.) Authentication Alias General Properties for an Oracle Database Data

Element	Description
Description	Optionally, enter a description that describes the purpose of the authentication alias.

Task 2 Create a JDBC data provider for the Oracle database

1. Log in to the IBM WebSphere Administrative Console and navigate to **Resources > JDBC > JDBC Providers**.
2. Select the appropriate **Scope** for the data provider you are about to create.
3. Click **New**.

   The IBM WebSphere Administrative Console displays a three-step wizard to guide you through the JDBC provider creation process.
4. In Step 1 of the JDBC provider wizard, make the selections shown in Table 3–3.
5. In Step 2 of the JDBC provider wizard, accept the default values.

   Note: You will modify these later in the procedure.
6. In Step 3 of the JDBC provider wizard, verify the values you entered and selected so far.
7. Click **Finish** to create the initial provider and return to the JDBC Providers page.

Table 3–3 Recommended Values to Select When Creating an IBM WebSphere Data Source for an Oracle Database

Element	Recommended Value
Database Type	Select **Oracle** from the drop-down menu.
Provider Type	Select **Oracle JDBC Driver** from the drop-down menu.
Implementation Type	Select **Connection pool data source** from the drop-down menu.
Name	Provide a unique name for the JDBC provider, or use the default name.
Description	Optionally, provide a description of the JDBC provider. This can be useful if you are creating multiple data sources for specific purposes.

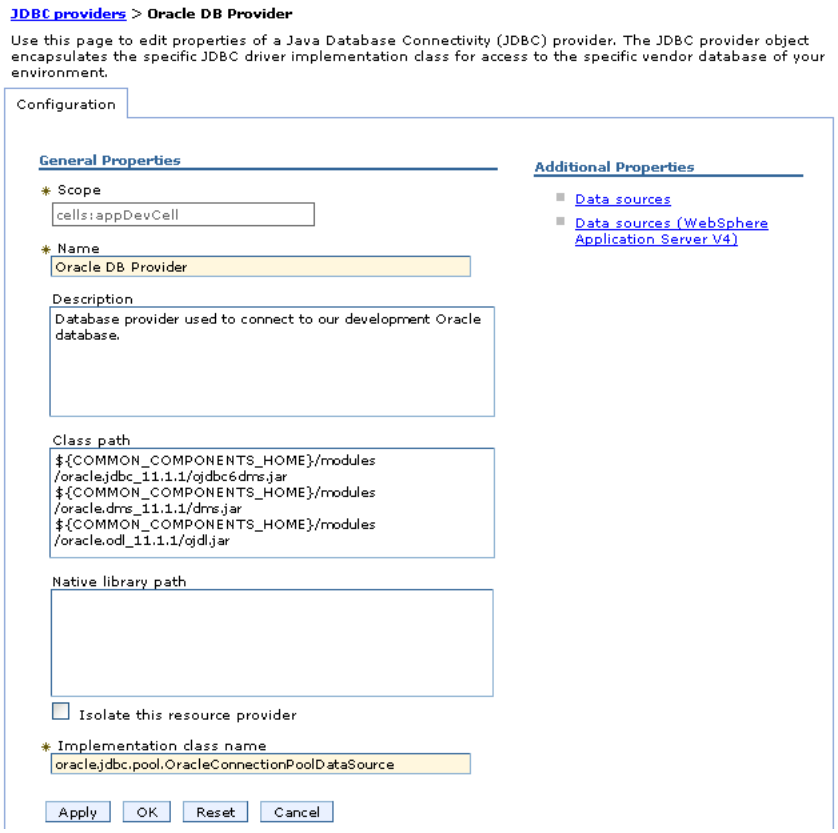
Task 3 Modify the JDBC data provider to use the latest Oracle database classes

1. Click the name of the database provider in the list of JDBC providers.
2. In the General properties section of the page, replace the value in the **Class path** field with the following:

   ```
   ${COMMON_COMPONENTS_HOME}/modules/oracle.jdbc_11.1.1/ojdbc6dms.jar
   ${COMMON_COMPONENTS_HOME}/modules/oracle.dms_11.1.1/dms.jar
   ${COMMON_COMPONENTS_HOME}/modules/oracle.odl_11.1.1/ojdl.jar
   ```

   Press Enter to separate the path locations so they appear on one line each, as shown in Figure 3–2.
3. Click **OK** to return to the JDBC Providers page.
4. Click **Save** to save your changes to the master configuration.

Figure 3–2 Summary of IBM WebSphere JDBC Provider Values for an Oracle Database



Task 4 Create a JDBC data source that uses the Oracle database JDBC provider

1. Log in to the console and navigate to **Resources > JDBC > Data Sources**.
2. Select the appropriate **Scope** for the data source you are about to create.
3. Click **New**.

   The IBM WebSphere Administrative Console displays a five-step wizard to guide you through the data source creation process.
4. In Step 1 of the data source wizard, enter a name for the data source and a JNDI location.

   For example, use `myOracleDS` as the data source name and `jdbc/myOracleDS` as the JNDI location.
5. In Step 2 of the data source wizard, select **Select an existing JDBC provider** and select the JDBC provider you created earlier in this procedure from the drop-down menu.
6. In Step 3 of the data source wizard, do the following:
   a. In the **URL** field, enter the connection string for the Oracle database, using the following format:

      `jdbc:oracle:thin:@`*`hostname`*`:`*`port`*`:`*`SID`*

      For example:

      `jdbc:oracle:thin:@host42.example.com:1521:DB43`

b. From the **Data store helper class name** menu, select the appropriate class name, based on whether you are connecting to a 10*g* or 11*g* Oracle database.

c. Optionally, select **Use this data source in container managed persistence (CMP)**.

   See the IBM WebSphere Administrative Console online help for information about the purpose of this option.

7. In Step 4 of the data source wizard, use the **Component-managed authentication alias** menu to select the authentication alias you created for the Oracle database earlier in this procedure.

   See the IBM WebSphere Administrative Console online help for information about the other options on the page.

8. In Step 5 of the wizard, review your changes. If they are accurate, click **Finish** to return to the Data Sources page.

9. Save the configuration changes, as directed in the console.

Task 5 Test the Data Source Connection

On the Data Sources page, select the data source and click **Test Connection** to verify your data source configuration.

3.3 Deploying Applications on IBM WebSphere

Refer to the following sections for information on deploying your Oracle Fusion Middleware applications on IBM WebSphere:

- Preparing to Deploy Oracle Fusion Middleware Applications on IBM WebSphere
- Methods for Deploying Oracle Fusion Middleware Applications on IBM WebSphere
- Deploying Applications that Require MDS Deployment Plan Customizations on IBM WebSphere

3.3.1 Preparing to Deploy Oracle Fusion Middleware Applications on IBM WebSphere

Before you can deploy Oracle Fusion Middleware applications (such as ADF, Oracle SOA Suite, Oracle WebCenter Portal, or Oracle WebCenter Content applications) to IBM WebSphere, you must follow certain steps for preparing the environment.

For example, you must be sure the Java Required Files (JRF) template has been applied to the IBM WebSphere servers. This can be accomplished by configuring the environment using the Oracle Fusion Middleware Configuration Wizard, as described in Chapter 2, "Installing and Configuring Oracle Identity and Access Management on IBM WebSphere" and in the *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server*.

3.3.2 Methods for Deploying Oracle Fusion Middleware Applications on IBM WebSphere

The primary methods for deploying your Oracle Fusion Middleware applications to IBM WebSphere are as follows:

- If you are working in a development or testing environment, you can deploy your applications directly from Oracle JDeveloper.

For information about configuring Oracle JDeveloper with an IBM WebSphere environment, see "Deploying the Application" in the *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*.

For information about deploying Oracle SOA Suite, Oracle WebCenter Portal, or Oracle WebCenter Content applications, refer to the corresponding chapter in this guide and the appropriate product development guide.

- If you are working in a testing or production environment, you can deploy application archives--for example, Enterprise Archive (EAR) files--from the IBM WebSphere Administration Console.

3.3.3 Deploying Applications that Require MDS Deployment Plan Customizations on IBM WebSphere

To deploy an application that requires MDS Deployment Plan customizations, you must use Oracle JDeveloper, unless you use the MDS wsadmin commands to customize the MDS deployment plan.

After you customize the deployment plan, you can then deploy the application archive from the IBM WebSphere Administrative Console.

3.4 Configuring Oracle Fusion Middleware High Availability on IBM WebSphere

The following sections provide information on configuring Oracle Fusion Middleware components for high availability on IBM WebSphere:

- Documentation Resources for Configuring Oracle Fusion Middleware High Availability on IBM WebSphere
- Configuring Java Object Cache for Oracle Fusion Middleware on IBM WebSphere

3.4.1 Documentation Resources for Configuring Oracle Fusion Middleware High Availability on IBM WebSphere

When configuring a high availability environment for the Oracle Fusion Middleware components that you install and configure on IBM WebSphere, refer to the following resources:

- The IBM WebSphere documentation available on the WebSphere Application Server Information Center.
- The *Oracle Fusion Middleware High Availability Guide*, which describes basic high availability concepts for Oracle Fusion Middleware components on Oracle WebLogic Server.
- The Oracle Fusion Middleware Enterprise Deployment Guides, which provide specific reference topologies for configuring the various Oracle Fusion Middleware components in a Oracle WebLogic Server-based production environment.
- The *Oracle Fusion Middleware Release Notes* for your platform, for information about known issues and workarounds when configuring Oracle Fusion Middleware components on IBM WebSphere.

In addition, refer to "Using wsadmin to Configure Oracle Fusion Middleware" in the *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server*, which provides examples of how you can use the wsadmin command-line to:

- Create servers, clusters, and cluster members on IBM WebSphere
- Create data sources for communicating with an Oracle Real Application Clusters database
- Federate remote nodes to an existing cell

3.4.2 Configuring Java Object Cache for Oracle Fusion Middleware on IBM WebSphere

When configuring high availability for Oracle Fusion Middleware, the *Oracle Fusion Middleware High Availability Guide* and Oracle Fusion Middleware Enterprise Deployment Guides suggest using Java Object Cache (JOC) to increase the performance of Oracle Web Services Manager and Oracle WebCenter Portal.

To configure JOC in this scenario, Oracle Fusion Middleware provides a custom script called `configure-joc.py`. This script is not supported on IBM WebSphere.

As an alternative, you can use the following procedure to configure JOC for Oracle Fusion Middleware on IBM WebSphere:

1. Locate and edit the `javacache.xml` file for each server in the cluster.

   The javacache.xml file is located in the Deployment Manager directory for each server:

   ```
   WAS_HOME/profiles/dmgr_proile_name/config
           /cells/cell_name
           /nodes/node_name
           /servers/server_name
           /fmwconfig/javacache.xml
   ```

   For example, if you have configured a cluster called WC_Spaces, and the cluster contains two servers, WC_Spaces and WC_Spaces2, then you can locate the `javacache.xml` file as follows:

   ```
   WebSphere/AppServer/profiles/Dmgr01/config/cells/Cell01/nodes/Node01/servers/WC
   _Spaces/fmwconfig/javacache.xml

   WebSphere/AppServer/profiles/Dmgr01/config/cells/Cell01/nodes/Node01/servers/WC
   _Spaces2/fmwconfig/javacache.xml
   ```

2. Make the following changes to the `javacache.xml` file:
   - Set the `enabled` attribute of the `<communicationService>` element to TRUE.
   - Remove the `outOfProc="false"` attribute from the `<packet-distributer>` element.
   - Add the `<distributor-location>` elements with the host and port of the servers in the cluster

   Example 3–8 provides a sample `javacache.xml` file that has been modified for use on IBM WebSphere. In the example, replace host with the host address and replace port with the port used for JOC communication. You can select any free port.

3. Login to the IBM WebSphere Administrative Console and navigate to the Nodes page (**System administration > Nodes**).
4. Select all nodes in the cluster and click on **Full Resynchronize**.
5. Restart all servers in the cluster.

Example 3–8 Sample javacache.xml File - Modified for IBM WebSphere

```
<?xml version="1.0" encoding="UTF-8"?>
<cache-configuration
    xmlns="http://www.oracle.com/oracle/ias/cache/configuration11"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" max-objects="5000"
    max-size="10" private="false"
    cache-dump-path="jocdump" system="false" clean-interval="60"
    version="11.1.1.2.0"
    internal-version="110000">
  <communicationService enabled="true">
    <v2 ssl-config-file=".sslConfig" init-retry="300" init-retry-delay="2000"
        enable-ssl="false" auto-recover="false">
      <packet-distributor enable-router="false" startable="true"
        dedicated-coordinator="false" >
        <distributor-location host="myhost1.example.com"
          port="9988" ssl="true"/>
       <distributor-location host="myhost2.exmaple.com" port="9988"
          ssl="true"/>
      </packet-distributor>
    </v2>
  </communicationService>
  <diskCache size="10" count="5000" ping-interval="60"/>
  <logging override-parent="false" location="javacache.log"
     default-level="SEVERE"/>
  <dms enabled="false"/>
</cache-configuration>
```

4

Managing Oracle Identity Manager on IBM WebSphere

This chapter contains information about managing Oracle Identity Manager on IBM WebSphere Application Server. It contains the following sections:

4.1 Conventions Used in this Document

Table 4–1 lists and describes conventions used in this document:

Table 4–1 Conventions Used in this Document

Convention	Description
OIM_HOME	Represents the directory where the Oracle Identity Manager server is installed.
OIM_ORACLE_HOME	Represents an environment variable that identifies the directory where Oracle Identity Manager is installed. This variable is used for various Oracle Identity Manager scripts.
WAS_HOME	Represents the directory where the IBM WebSphere Application Server is installed.
WAS_CLIENT_HOME	Represents the directory where the IBM WebSphere Application Client is installed.
MW_HOME	Represents the directory where Oracle Fusion Middleware is installed.
COMMON_COMPONENTS_HOME	The Common Components home contains the binary and library files required for Fusion Middleware Control and Java Required Files (JRF). For example, *MW_HOME*/oracle_common.
Custom01 | *Custom02*	Represents the name of a custom profile.

Table 4–1 (Cont.) Conventions Used in this Document

Convention	Description
Dmgr01 | *Dmgr02*	Represents the name of a Deployment Manager profile.
OIM_DC_HOME	Represents the directory where the Oracle Identity Manager Design Console is installed.
OIM_RM_HOME	Represents the directory where the Oracle Identity Manager Remote Manager is installed.
OIM_CELL_NAME	Represents the IBM WebSphere Application Server cell where the Oracle Identity Manager Server is located.
JAVA_HOME	Represents the location of the IBM Java Runtime directory for the Oracle Identity Manager server. Note that in some procedures, *JAVA_HOME* can represent the location of the IBM Java Runtime directory for the Oracle Identity Manager Remote Manager.
RBACX_HOME	Represents the directory where Oracle Identity Analytics is installed.
ANT_HOME	Represents the directory where Apache Ant is installed.

4.2 System Requirements and Certified Components

Before deploying and using Oracle Identity Manager, you must ensure that your environment meets the minimum installation requirements. For information about hardware and software requirements, minimum disk space and memory requirements, and required system libraries, packages, or patches, review the system requirements document at the following URL:

`http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-requirements-100147.html`

The following URL contains information about supported installation types, platforms, operating systems, databases, JDKs, and third-party products for Oracle Fusion Middleware:

`http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-certification-100350.html`

In addition, see "Patch Requirements" in the *Oracle Fusion Middleware Release Notes* for information about the patches required for Oracle Identity Manager.

Note:

- Minimum memory requirement for setting up Oracle Identity Manager on IBM WebSphere Application Server is 8 GB.
- BI Publisher reports on WebSphere are not certified for Oracle Identity Manager 11*g* Release 2 (11.1.2.1.0).

4.3 Installing Oracle Identity Manager on IBM WebSphere

This section describes how to install Oracle Identity Manager on IBM WebSphere in the following configurations:

- Configuring Oracle Identity Manager for Single-Node Setup
- Installing Oracle Identity Manager for a Clustered Configuration

- Performing Oracle Identity Manager Clustered Scale Out Configuration

4.3.1 Configuring Oracle Identity Manager for Single-Node Setup

As a part of installing Oracle Identity Manager, after cell configuration is performed as described in Chapter 2, "Installing and Configuring Oracle Identity and Access Management on IBM WebSphere", you must configure Oracle Identity Manager. To do so:

1. Use the Oracle Fusion Middleware Configuration Wizard to create the Oracle Identity Manager cell, as described in Section 2.8, "Task 8: Configure Your Oracle Identity and Access Management Components in a New IBM WebSphere Cell".
2. Run the copy_jars.sh script. For example:

```
cd $OIM_HOME/server/wasconfig
./copy_jars.sh
```

> **Note:** Before you run the copy_jars.sh script, ensure the *WAS_HOME*, *COMMON_COMPONENTS_HOME*, and *OIM_ORACLE_HOME* variables are set. *COMMON_COMPONENTS_HOME* represents the location of the Oracle Fusion Middleware common directory, such as *MW_HOME*/oracle_common and; *OIM_ORACLE_HOME* represents the location where the Oracle Identity Manager Server is installed, such as *MW_HOME*/Oracle_IDM1. *WAS_HOME* represents the location where WebSphere is installed, such as IBM/WebSphere/AppServer.

3. Start, stop, and synchronize the Node Agent as follows:

```
$WAS_HOME/profiles/Custom01/bin/stopNode.sh -username USER_NAME -password
PASSWORD
$WAS_HOME/profiles/Dmgr01/bin/startManager.sh
$WAS_HOME/profiles/Custom01/bin/syncNode.sh DMGR_HOST DMGR_SOAP_PORT -username
USER_NAME -password PASSWORD
$WAS_HOME/profiles/Custom01/bin/startNode.sh
```

> **Note:** Make sure that Node Manager and Deployment Manger are up and running without issues.

4. Stop the Node Manager and Deployment Manger for configuring DB policy store, as shown:

```
$WAS_HOME/profiles/Custom01/bin/stopNode.sh -username USER_NAME -password
PASSWORD
$WAS_HOME/profiles/Dmgr01/bin/stopManager.sh -username USER_NAME -password
PASSWORD
```

5. Perform database policy migration by referring to Section 2.9, "Task 9: Configure the Database Security Store".
6. Start the Deployment Manager. To do so, run the following command in the IBM WebSphere home:

For UNIX, run:

```
profiles/dmgr_profileName/bin/startManager.sh
```

For example, on the UNIX operating system, run:

```
/disk01/IBM/WebSphere/AppServer/profiles/Dmgr01/bin/startManager.sh
```

7. Start the Node Manager by running the following command:

```
$WAS_HOME/profiles/Custom01/bin/startNode.sh
```

8. Run the seed_opss_permission.sh script as follows:

```
cd OIM_HOME/server/wasconfig/
sh seed_opss_permission.sh
```

Note:

- Before you run the seed_opss_permission.sh script, ensure the *WAS_HOME*, *COMMON_COMPONENTS_HOME*, and *OIM_ORACLE_HOME* variables are set. *COMMON_COMPONENTS_HOME* represents the location of the Oracle Fusion Middleware common directory, such as *IDM_HOME*/oracle_common/ and; *OIM_ORACLE_HOME* represents the location where the Oracle Identity Manager Server is installed.
- The script will prompt you to enter values for the following:

```
Enter Deployment Manager Profile Name [Ex: Dmgr01]:
Enter Deployment Manager host name:
Enter Deployment Manager SOAP Port:
Enter WebpSphere Administrator username:
Enter the WebpSphere Administrator password:
```

- On running the seed_opss_permission.sh script, you might encounter the following warning message that you can ignore:

```
Failed to import script libraries modules:
COMMON_COMPONENTS_HOME/common/wsadmin/wsmAgent.py; Examine the
wsadmin log file to determine the problem.
```

---

9. Stop, synchronize, and start the node, and start the SOA server. For example:

```
$WAS_HOME/profiles/Custom01/bin/stopNode.sh -username WAS_ADMIN_USER_NAME
-password WAS_ADMIN_PASSWORD
$WAS_HOME/profiles/Custom01/bin/syncNode.sh DMGR_HOST DMGR_SOAP_PORT
WAS_ADMIN_USER_NAME -password WAS_ADMIN_PASSWORD
$WAS_HOME/profiles/Custom01/bin/startNode.sh
$WAS_HOME/profiles/Custom01/bin/startServer.sh soa_server1
```

10. Use the Oracle Universal Installer Configuration Assistant to configure the Oracle Identity Manager Server, Design Console, and Remote Manager.

    Start the configuration assistant as follows:

```
cd $OIM_HOME/bin
./config.sh -jreLoc LOCATION_OF_IBM_JRE -DSHOW_APPSERVER_TYPE_SCREEN=true
```

Note: You must run the Configuration Assistant on each machine where you installed an Oracle Identity Manager component. For example, on the machine hosting the Oracle Identity Manager server, the machine hosting the Oracle Identity Manager Design Console, and the machine hosting the Oracle Identity Manager Remote Manager.

On the Components to Configure screen, select the components that you want to configure. On the Database screen, provide the connect string and user names and passwords for Oracle Identity Manager and MDS schema.

Table 4–2 provides information about *specific* Configuration Assistant screens and appropriate information to enter on those screens—the table does not cover self-explanatory, standard screens.

Table 4–2 Information for Specific Configuration Assistant Screens

Screen Name	Input Description
Application Server	Be sure to select **WebSphere**

WebSphere AS Details	▪ The WAS Cell home location is:
	`$WAS_HOME/profiles/Dmgr01/config/cells/CELL_NAME`
	▪ You can identify the WAS Admin URL port from the `Management bootstrap port` entry in the following file:
	`$WAS_HOME/profiles/Dmgr01/logs/AboutThisProfile.txt`
	▪ You can identify the WAS Admin Soap Port from the following file:
	`$WAS_HOME/profiles/Dmgr01/logs/AboutThisProfile.txt`
	▪ The WAS Admin Name and WAS Admin Password are the same as you used to create the cell.

OIM Server	Use the default value provided in the OIM HTTP URL field.

11. On OIM Server Details page, enter the OIM Server admin password, keystore password, and the URL information. Continue until configuration has finished.
12. Copy wf_client_config.xml.template from $*OIM_HOME*/server/wasconfig/ directory to $*WAS_HOME*/lib/ext as wf_client_config.xml. For example:

    cp $*OIM_HOME*/server/wasconfig/wf_client_config.xml.template $WAS_HOME/lib/ext/wf_client_config.xml

    Update the wf_client_config.xml file with SOA Server hostname and its bootstrap port under <serverURL> tag. For example:

    ```
    <serverURL>corbaloc:iiop:localhost:2800</serverURL>
    ```

Tip: You can identify the SOA bootstrap port by performing the following steps:

1. Log in to IBM WebSphere Administrative Console.
2. Select **Servers**, **Server Types**, **Web Application Servers**.
3. Click the SOA Server name.
4. In the Communications Group area, click **Ports**.

   The value of BOOTSTRAP_ADDRESS is the SOA Server bootstrap port.

13. Stop the servers if they are running. For example:

```
$WAS_HOME/profiles/Custom01/bin/stopServer.sh soa_server1 -username
WAS_ADMIN_USERNAME -password WAS_ADMIN_PASSWORD
$WAS_HOME/profiles/Custom01/bin/stopNode.sh -username WAS_ADMIN_USERNAME
-password WAS_ADMIN_PASSWORD
$WAS_HOME/profiles/Dmgr01/bin/stopManager.sh -username WAS_ADMIN_USERNAME
-password WAS_ADMIN_PASSWORD
```

14. Start the servers. For example:

Note: Be sure to execute the syncNode script, as this will transfer xldatabasekey to *Custom01* profile.

```
$WAS_HOME/profiles/Dmgr01/bin/startManager.sh
$WAS_HOME/profiles/Custom01/bin/syncNode.sh DMGR_HOST DMGR_SOAP_PORT -username
WAS_ADMIN_USERNAME -password WAS_ADMIN_PASSWORD
$WAS_HOME/profiles/Custom01/bin/startNode.sh
$WAS_HOME/profiles/Custom01/bin/startServer.sh soa_server1
$WAS_HOME/profiles/Custom01/bin/startServer.sh oim_server1
$WAS_HOME/profiles/Custom01/bin/startServer.sh OracleAdminServer
```

15. If Oracle Identity Manager administrator user is different than WebSphere administrator user, then perform the following steps:

    a. In the navigator pane of Enterprise Fusion Middleware Control, expand WebSphere Cell to view the cells.

    b. Select the cell on which Oracle Identity Manager and SOA are configured.

    c. Right-click the cell name, and select **Web Services**, **Platform Policy Configuration**.

    d. In the Add New Configure Property window, specify the following values, and then click **OK**.

       – In the Name field, enter `jndi.lookup.csf.key`.

       – In the Value field, enter `admin-csf-key`.

    e. Crete a .py file, for example was_admin.py, with the following content:

```
Opss.createCred (map='oracle.wsm.security', key='admin-csf-key',
user='ADMIN_USER_NAME', password='ADMIN_PASSWORD',
desc='wsm-pm admin user csf-key')
AdminApp.edit ('wsm-pm', '[-MapRolesToUsers [[policy.Updater
AppDeploymentOption.No AppDeploymentOption.No ADMIN_USER_NAME ""
AppDeploymentOption.No "user:ADMIN_USER_NAME" "" ]]]')
AdminApp.edit('wsm-pm', '[ -MapRolesToUsers [[policy.Accessor
AppDeploymentOption.No AppDeploymentOption.No ADMIN_USER_NAME ""
```

```
AppDeploymentOption.No " |user:ADMIN_USER_NAME" "" ]]]' )
AdminApp.edit('wsm-pm', '[ -MapRolesToUsers [[policy.User
AppDeploymentOption.No AppDeploymentOption.No ADMIN_USER_NAME ""
AppDeploymentOption.No "user:ADMIN_USER_NAME" "" ]]]' )
AdminApp.edit('wsm-pm', '[ -MapRolesToUsers [[policyViewer
AppDeploymentOption.No AppDeploymentOption.No ADMIN_USER_NAME ""
AppDeploymentOption.No " |user:ADMIN_USER_NAME" "" ]]]' )
AdminConfig.save()
```

Replace *ADMIN_USER_NAME* and *ADMIN_PASSWORD* with admin user credentials.

f. Run the following script:

```
$COMMON_COMPONENTS_HOME/common/bin/wsadmin.sh
-profileName DMGR_PROFILE_NAME -conntype SOAP -host DMGR_HOSTNAME -port
DMGR_SOAP_PORT -user WEBSPHERE_ADMIN -password WEBSPHERE_ADMIN_PASSWORD -f
was_admin.py
```

g. Restart all the servers.

16. For additional postinstallation configuration of Oracle Identity Manager, perform the steps described in Section 4.4, "Performing Postinstallation Configuration on IBM WebSphere" and Section 4.6.1, "URL Changes Related to Oracle Identity Manager".

4.3.1.1 Installing and Configuring the Design Console

Perform the following step after the Design Console installs, but before you start it:

To install the Design Console on Microsoft Windows:

1. Install the App client by referring to IBM documentation.

2. Install fix packs by referring to IBM documentation.

3. Update the following properties in the *WAS_CLIENT_HOME*/properties/sas.client.props file.

Edit the values as follows. Note that `com.ibm.CORBA.securityServerPort` represents the Oracle Identity Manager bootstrap port:

```
com.ibm.CORBA.securityServerHost=OIM_HOSTNAME
com.ibm.CORBA.securityServerPort=OIM_BOOTSTRAP_PORT
com.ibm.CORBA.loginSource=none
```

4. Install Design Console on Microsoft Windows. To do so:

Note: Make sure that Appclient is installed.

---

a. Install Oracle Identity Manager by running the installer. To do so, open a command prompt in Windows, and run the Oracle Identity Manager installer, as shown:

```
c:\setup.exe -jreLoc LOCATION_OF_IBM_JDK
```

b. Start the configuration assistant as follows:

```
cd $OIM_HOME/bin >config.bat -jreLoc LOCATION_OF_IBM_JDK -enableWAS
```

c. Configure the following:

- Select Design Console.
- Enter the Oracle Identity Manager host name and port number.

Tip: The port number is Oracle Identity Manager server bootstrap address. To check this:

1. Login to WebSphere Network Deployment Manager Console.
2. Go to **Server**, **Server types**, **Websphere Application server**, *oim_server*, **Expand Port**.
3. Check for BOOTSTRAP_ADDRESS port.

d. Continue and finish the wizard.

4.3.1.2 (OPTIONAL) Installing the Oracle Identity Manager Remote Manager on a Separate System

When you install the Oracle Identity Manager Remote Manager as a part of the Oracle Identity Manager installation, the Remote Manager is installed on the same host as Oracle Identity Manager. In typical Oracle Identity Manager environments, the Remote Manager is deployed on a separate host, not on the same host as Oracle Identity Manager.

If desired, you can perform the following steps to install the Remote Manager on a separate system:

Note: Make sure that WebSphere Application Server is installed. In addition, ensure that the separate system for the Remote Manager has the IBM JRE installed on it. If it does not, then install it.

1. Start the installer using the following command:

```
cd iamsuite/Disk1
./runInstaller -jreLoc LOCATION_OF_IBM_JRE
```

Note: When the Install Software Updates installer screen is displayed, you must select the Skip Software Updates option.

2. Start the configuration assistant as follows:

```
cd $OIM_HOME/bin >config.bat -jreLoc LOCATION_OF_IBM_JDK -enableWAS
```

3. In the Components to Configure page, select **Remote Manager**.
4. Select WebSphere as the application server.
5. Provide OIM server host and port details. Enter values for Remote Name, Listening Port number, and RMI Port number.
6. Continue and finish the wizard.

4.3.1.3 Installing the Diagnostic Dashboard

To install the Diagnostic Dashboard:

1. Login to IBM WebSphere Administrative Console.
2. Expand **Applications**, and click **WebSphere enterprise applications**.

3. Click **Install**.
4. Select **Remote file system**.
5. Enter the complete path to the XIMDD.ear file. The XIMDD.ear file is available in the $*OIM_HOME*/server/webapp/optional/ directory. Then, click **Next**.
6. Choose Fast Path to install application.
7. Click **Next** in the Select installation options.
8. Check the **Select** option in the Map modules to servers page, and click **Next**.
9. Click **Next** in the Map virtual hosts for Web modules page.
10. Click **Finish** in the Summary page.
11. Save the changes.

4.3.2 Installing Oracle Identity Manager for a Clustered Configuration

This section describes how to install Oracle Identity Manager on IBM WebSphere in a clustered configuration. By performing the steps in this section, you will create a configuration as described in Table 4–3.

Table 4–3 Overview of Clustered Configuration

Deployment Manager Machine	WebSphere Node 2 Machine	Design Console Machine
■ WebSphere Deployment Manager ■ WebSphere Node1 ■ OracleAdminServer ■ *OIM_SERVER_1* ■ *SOA_SERVER_1*	■ WebSphere Node2 ■ *OIM_SERVER_2* ■ *SOA_SERVER_2*	■ Oracle Identity Manager Design Console

To install Oracle Identity Manager on IBM WebSphere in a clustered configuration:

1. Install and patch Oracle Database. For Oracle Database patch requirements, see "Patch Requirements" in the *Oracle Fusion Middleware Release Notes*.
2. Install the Oracle Fusion Middleware Repository Creation Utility (RCU). For more information, see "Obtaining and Running Repository Creation Utility" in the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.
3. Create and load the **Identity Management - Oracle Identity Manager** schema into the database using the Oracle Fusion Middleware Repository Creation Utility (RCU). For more information, refer to the following documents:
   - *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*
   - *Oracle Fusion Middleware Repository Creation Utility User's Guide*
4. Make sure to have IBM HTTP Server (IHS) available. To install and setup IHS:
   a. Install IHS on the Deployment Manager Machine with appropriate HTTP host Admin port.
   b. Provide `webserver1` as the webserver name.

c. The IHS setup prompts to configure/generate the default plug-in configuration. Select **Yes** to generate the default plug-in configuration.

d. After the setup is complete, start IHS by running the following command:

```
IHS_INSTALL_DIRECTORY/bin/apachectl start
```

e. Verify that the IHS Welcome page is displayed by navigating to the following URL:

http://*IHS_HOSTNAME*:*PORT_NUMBER*

Note: See Section 4.4.6, "Performing Postinstallation Configuration of IHS (Optional)" for post-installation configuration of IHS.

---

5. On Deployment Manager Machine and WebSphere Node 2 Machine, install IBM WebSphere Application Server Network Deployment 7.0 with fix pack 23 by referring to IBM documentation.
6. On Design Console Machine, install IBM WebSphere Application Client 7.0 with fix pack 23 to host the Oracle Identity Manager Design Console. Refer to IBM documentation for more information about installing IBM WebSphere Application Client.
7. On Deployment Manager Machine and WebSphere Node 2 Machine, install Oracle SOA Suite 11.1.1.6.0. For more information, refer to the "Installing Oracle SOA Suite (Oracle Identity Manager Users Only)" section of the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*. In addition, apply the SOA patches listed in "Mandatory Patches Required for Installing Oracle Identity Manager" of the *Oracle Fusion Middleware Release Notes*.

   Note: Make sure to use WebSphere Application Server JRE when installing SOA.

   ---

8. On Deployment Manager Machine and WebSphere Node 2 Machine, install Oracle Identity Manager. For more information about installing Oracle Identity Manager, refer to the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.

   To start the installer, run:

   ```
   cd iamsuite/Disk1
   ./runInstaller -jreLoc LOCATION_OF_IBM_JRE -DSHOW_APPSERVER_TYPE_SCREEN=true
   ```

   Note: When the **Install Software Updates** installer screen appears, you must select the **Skip Software Updates** option.

   ---

9. On the Deployment Manager Machine, use the Oracle Fusion Middleware Configuration Wizard to create the Oracle Identity Manager cell. By default, the Configuration Wizard is located at:

   MW_HOME/Oracle_IDM1/common/bin/was_config.sh

   For more information, refer to the *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server*.

Table 4–4 provides information about specific Configuration Wizard screens and appropriate information to enter on those screens—the table does not cover self-explanatory, standard screens.

Table 4–4 Information for Specific Configuration Wizard Screens

Screen Name	Input Description
Select Configuration Option	Create and configure cell.
Add Products to Cell	Select **Oracle Identity Manager for WebSphere ND**. The SOA/EM template and other dependent templates should also be selected.
Select Optional Configuration	At a minimum, you must select the **Application Servers, Clusters and End Points** option—this is a required option.
Configure Application Servers	Perform the following steps: 1. In the Name field, enter a name for the Oracle Identity Manager server, for example, OIM_SERVER_1. 2. In the Node Name list, select the Node Agent for OIM_SERVER_1. For example: WebSphere Node1. 3. In the Name field, enter a name for the Oracle SOA Suite server, for example, SOA_SERVER_1. 4. In the Node Name list, select the Node Agent for SOA_SERVER_1. For example: WebSphere Node1.
Configure Clusters Screen	Perform the following steps: 1. Click **Add**. 2. Enter a name for the cluster in the cluster name field, for example: SOACluster. 3. Select the appropriate SOA server from the First cluster member list. 4. Click **Add**. 5. Enter a name for the cluster in the cluster name field, for example: OIMCluster. 6. Select the appropriate Oracle Identity Manager server from the First cluster member list.
Configure Additional Cluster Members	Click **Next**, or optionally, add servers to an existing system in the cluster.

10. On the Deployment Manager Machine, execute the copy_jars.sh script. For example:

```
cd $OIM_HOME/server/wasconfig
./copy_jars.sh
```

Note: Before you execute the copy_jars.sh script, ensure the *WAS_HOME*, *COMMON_COMPONENTS_HOME*, and *OIM_ORACLE_HOME* variables are set. *COMMON_COMPONENTS_HOME* represents the location of the Oracle Fusion Middleware common directory, such as *MW_HOME*/oracle_common and; *OIM_ORACLE_HOME* represents the location where the Oracle Identity Manager Server is installed, such as *MW_HOME*/Oracle_IDM1. *WAS_HOME* represents the location where WebSphere is installed, such as IBM/WebSphere/AppServer.

11. On the Deployment Manager Machine, start, stop, and synchronize the IBM WebSphere nodes as follows:

```
$WAS_HOME/profiles/Custom01/bin/stopNode.sh
$WAS_HOME/profiles/Dmgr01/bin/startManager.sh
$WAS_HOME/profiles/Custom01/bin/syncNode.sh DMGR_HOST DMGR_SOAP_PORT
$WAS_HOME/profiles/Custom01/bin/startNode.sh
```

For specifying the port number for *DMGR_SOAP_PORT*, refer to the $WAS_HOME/profiles/*Dmgr01*/logs/AboutThisProfile.txt file that contains information about the ports.

Note: When you start, stop, and synchronize the IBM WebSphere nodes, you must:

- Use the user name and password that you used to create the cell.
- Execute syncNode.sh. If you do not, some applications will not be deployed correctly.
- Execute syncNode.sh from the following directory:

  ```
  $WAS_HOME/profiles/Custom01/bin
  ```

12. On the Deployment Manager Machine, perform database policy migration by referring to step 1 of Section 2.9, "Task 9: Configure the Database Security Store".
13. Start, stop, and synchronize the WebSphere nodes by running the following commands:

```
$WAS_HOME/profiles/Custom01/bin/stopNode.sh
$WAS_HOME/profiles/Dmgr01/bin/stopManager.sh
$WAS_HOME/profiles/Dmgr01/bin/startManager.sh
$WAS_HOME/profiles/Custom01/bin/syncNode.sh DMGR_HOST DMGR_SOAP_PORT
$WAS_HOME/profiles/Custom01/bin/startNode.sh
```

14. On the Deployment Manager Machine, execute the seed_opss_permission.sh script as follows:

```
cd OIM_HOME/server/wasconfig/
sh seed_opss_permission.sh
```

Note:

- Before you execute the seed_opss_permission.sh script, ensure the *WAS_HOME*, *COMMON_COMPONENTS_HOME*, and *OIM_ORACLE_HOME* variables are set. *COMMON_COMPONENTS_HOME* represents the location of the Oracle Fusion Middleware common directory, such as *IDM_HOME*/oracle_common/ and; *OIM_ORACLE_HOME* represents the location where the Oracle Identity Manager Server is installed.
- The script will prompt you to enter values for the following:

```
Enter Deployment Manager Profile Name [Ex: Dmgr01]:
Enter Deployment Manager host name:
Enter Deployment Manager SOAP Port:
Enter WebpSphere Administrator username:
Enter the WebpSphere Administrator password:
```

- On running the seed_opss_permission.sh script, you might encounter following error message:

```
Failed to import script libraries modules:
COMMON_COMPONENTS_HOME/common/wsadmin/wsmAgent.py; Examine the
wsadmin log file to determine the problem.
```

  When you encounter this error, check the system-jazn-data.xml file to ensure that permission has been granted to oim_customreg.jar. If permission is not granted, then you must add the permission manually. To do so:

  i) Open the *WAS_HOME*/profiles/*Dmgr01*/config/cells/*OIM_CELL_NAME*/fmwconfig/system-jazn-data.xml file.

  ii) Search for following entry. If this entry does not exist in system-jazn-data.xml, then manually add it. Make sure to replace *OIM_ORACLE_HOME* with the actual path.

```
<grant>
<grantee>
<codesource>
<url>file:OIM_ORACLE_HOME/server/loginmodule/was/oim_customreg.
jar</url>
</codesource>
</grantee>
<permissions>
<permission>
<class>oracle.security.jps.service.credstore.CredentialAccessPe
rmission</class>
<name>context=SYSTEM,mapName=oim,keyName=*</name>
<actions>read,write,delete</actions>
</permission>
<permission>
<class>oracle.security.jps.service.credstore.CredentialAccessPe
rmission</class>
<name>context=SYSTEM,mapName=oracle.wsm.security,keyName=*</nam
e>
<actions>read,write,delete</actions>
</permission>
</permissions>
</grant>
```

15. Add the following properties by logging in to the IBM WebSphere Administrative Console and clicking **System Administration**, **Node Agents**, ***NAME_OF_NODE_AGENT_ON_DEPLOYMENT_MANAGER_MACHINE***, **Java and Process Management**, **Process Definition**, **Java Virtual Machine**, **Custom Properties**.

    Note: When you create the properties:

    - An example location for the *PATH_TO_jps-config.xml_IN_THE_fmwconfig_DIRECTORY* is: *WAS_HOME*/profiles/*Dmgr01*/config/cells/*HOST_NAME_Cell01*/fmwconfig/jps-config.xml
    - An example location for the *PATH_TO_THE_fmwconfig_DIRECTORY* is: *WAS_HOME*/profiles/*Dmgr01*/config/cells/*HOST_NAME_Cell01*/fmwconfig

    ```
    Name: oracle.security.jps.config
    Value: PATH_TO_jps-config.xml_IN_THE_fmwconfig_DIRECTORY
    Description (optional): Adding the jpsconfig location using OPSS System
    Property

    Name: oracle.domain.config.dir
    Value: PATH_TO_THE_fmwconfig_DIRECTORY
    Description (optional): Setting the Key Store Domain Config directory
    ```

    Click **OK** and save the changes.

16. On the Deployment Manager Machine, stop, synchronize, and start the Node Agent. For example:

    ```
    $WAS_HOME/profiles/Custom01/bin/stopNode.sh
    $WAS_HOME/profiles/Custom01/bin/syncNode.sh DMGR_HOST DMGR_SOAP_PORT
    $WAS_HOME/profiles/Custom01/bin/startNode.sh
    $WAS_HOME/profiles/Custom01/bin/startServer.sh soa_server1
    ```

17. On the Deployment Manager Machine, configure the Oracle Identity Manager server (and optionally the Oracle Identity Manager Remote Manager) using the Oracle Universal Installer Configuration Assistant.

    Note: You do not need to run the Configuration Assistant on the WebSphere Node 2 Machine.

    Start the configuration assistant as follows:

    ```
    cd $OIM_HOME/bin
    ./config.sh -jreLoc LOCATION_OF_IBM_JRE -DSHOW_APPSERVER_TYPE_SCREEN=true
    ```

    Table 4–5 provides information about specific Configuration Assistant screens and appropriate information to enter on those screens—the table does not cover self-explanatory screens.

Table 4–5 Information for Specific Configuration Assistant Screens

Screen Name	Input Description
Application Server	Be sure to select **WebSphere**

WebSphere AS Details	▪ The WAS Cell home location is:
	`$WAS_HOME/profiles/Dmgr01/config/cells/CELL_NAME`
	▪ You can identify the WAS Admin URL port from the `Management bootstrap port` entry in the following file:
	`$WAS_HOME/profiles/Dmgr01/logs/AboutThisProfile.txt`
	▪ You can identify the WAS Admin Soap Port from the following file:
	`$WAS_HOME/profiles/Dmgr01/logs/AboutThisProfile.txt`
	▪ The WAS Admin Name and WAS Admin Password are the same as you used to create the cell.

OIM Server	In the OIM HTTP URL field, enter the HTTP URL for the IBM HTTP Server.

18. On the Deployment Manager Machine, stop the SOA server, the Node Agent, and the Deployment Manager if they are running. For example:

```
$WAS_HOME/profiles/Custom01/bin/stopServer.sh soa_server1
$WAS_HOME/profiles/Custom01/bin/stopNode.sh
$WAS_HOME/profiles/Dmgr01/bin/stopManager.sh
```

19. On the Deployment Manager Machine, start the Deployment Manager, synchronize the Node Agent, and start the Node Agent.

> **Note:** Be sure to execute the syncNode script, as this will transfer the required configuration information to *Custom01* profile.

For example:

```
$WAS_HOME/profiles/Dmgr01/bin/startManager.sh
$WAS_HOME/profiles/Custom01/bin/syncNode.sh DMGR_HOST DMGR_SOAP_PORT
$WAS_HOME/profiles/Custom01/bin/startNode.sh
```

20. On the WebSphere Node 2 Machine, launch the Oracle Fusion Middleware Configuration Wizard to federate the machine and configure its cell. By default, the Configuration Wizard is located at:

    MW_HOME/Oracle_IDM1/common/bin/was_config.sh

    For more information, refer to the *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server*.

    Table 4–6 provides information about specific Configuration Wizard screens and appropriate information to enter on those screens—the table does not cover self-explanatory, standard screens.

Table 4–6 Information for Specific Configuration Wizard Screens

Screen Name	Input Description
Select Configuration Option	Select the **Federate Machine and Configure Cell** option.
Specify Profile and Node Name Information	Enter information about the profile and node names you want to create for the WebSphere Node 2 Machine.
Specify Deployment Manager Information	Enter information about the existing Deployment Manager system.
Select Optional Configuration	Be sure to select the **Application Servers, Clusters and End Points** option—this is a required option.
Configure Additional Cluster Members	Perform the following steps: 1. Click **Add**. 2. In the Name field, enter a name for the second server in the SOACluster. For example: *SOA_SERVER_2*. 3. In the Node Name list, select the Node Agent for *SOA_SERVER_2*. For example: WebSphere Node2. 4. In the Cluster Name list, select the SOACluster. 5. Click **Add**. 6. In the Name field, enter a name for the second server in the OIMCluster. For example: *OIM_SERVER_2*. 7. In the Node Name list, select the Node Agent for *OIM_SERVER_2*. For example: WebSphere Node2. 8. In the Cluster Name list, select the OIMCluster.

21. On the WebSphere Node 2 Machine, execute the copy_jars.sh script. For example:

```
cd $OIM_HOME/server/wasconfig
./copy_jars.sh
```

> **Note:** Before you execute the copy_jars.sh script, ensure the *WAS_HOME*, *COMMON_COMPONENTS_HOME*, and *OIM_ORACLE_HOME* variables are set. *COMMON_COMPONENTS_HOME* represents the location of the Oracle Fusion Middleware common directory, such as *MW_HOME*/oracle_common and; *OIM_ORACLE_HOME* represents the location where the Oracle Identity Manager Server is installed, such as *MW_HOME*/Oracle_IDM1. *WAS_HOME* represents the location where WebSphere is installed, such as IBM/WebSphere/AppServer.

22. On the Deployment Manager Machine, stop the SOA server, the Node Agent, and the Deployment Manager if they are running. For example:

```
$WAS_HOME/profiles/Custom01/bin/stopServer.sh soa_server1
$WAS_HOME/profiles/Custom01/bin/stopNode.sh
$WAS_HOME/profiles/Dmgr01/bin/stopManager.sh
```

23. On the Deployment Manager Machine, start the Deployment Manager, synchronize the Node Agent, and start the Node Agent.

Note: Be sure to execute the syncNode script, as this will transfer the required configuration information to *Custom01* profile.

For example:

```
$WAS_HOME/profiles/Dmgr01/bin/startManager.sh
$WAS_HOME/profiles/Custom01/bin/syncNode.sh DMGR_HOST DMGR_SOAP_PORT
$WAS_HOME/profiles/Custom01/bin/startNode.sh
```

24. On the WebSphere Node 2 Machine, stop, synchronize, and start the IBM WebSphere nodes as follows:

```
$WAS_HOME/profiles/Custom01/bin/stopNode.sh
$WAS_HOME/profiles/Custom01/bin/syncNode.sh DMGR_HOST DMGR_SOAP_PORT
$WAS_HOME/profiles/Custom01/bin/startNode.sh
```

25. Add the following properties by logging in to the IBM WebSphere Administrative Console and clicking **System Administration**, **Node Agents**, ***NAME_OF_NODE_AGENT_ON_WEBSPHERE_NODE2_MACHINE***, **Java and Process Management**, **Process Definition**, **Java Virtual Machine**, **Custom Properties**.

Note: When you create the properties:

- An example location for the *PATH_TO_jps-config.xml_IN_THE_fmwconfig_DIRECTORY* is: *WAS_HOME*/profiles/*Custom01*/config/cells/*HOST_NAME_Cell01*/fmwconfig/jps-config.xml
- An example location for the *PATH_TO_THE_fmwconfig_DIRECTORY* is: *WAS_HOME*/profiles/*Custom01*/config/cells/*HOST_NAME_Cell01*/fmwconfig

```
Name: oracle.security.jps.config
Value: PATH_TO_jps-config.xml_IN_THE_fmwconfig_DIRECTORY
Description (optional): Adding the jpsconfig location using OPSS System
Property

Name: oracle.domain.config.dir
Value: PATH_TO_THE_fmwconfig_DIRECTORY
Description (optional): Setting the Key Store Domain Config directory
```

Click **OK** and save the changes.

26. Copy wf_client_config.xml.template from $OIM_HOME/server/wasconfig directory to $WAS_HOME/lib/ext as wf_client_config.xml. For example, cp $OIM_HOME/server/wasconfig/wf_client_config.xml.template $WAS_HOME/lib/ext/wf_client_config.xml.

Note: Perform this step in both Deployment Manager Machine and WebSphere Node 2 Machine.

Update the wf_client_config.xml file with SOA Server hostname and its bootstrap port under <serverURL> tag. For example:

```
<serverURL>corbaloc:iiop:host1:bootstrap_port1,:host2:bootstrap_port2
</serverURL>
```

Tip: You can identify the SOA bootstrap port by performing the following steps:

1. Log in to IBM WebSphere Administrative Console.
2. Select **Servers**, **Server Types**, **Web Application Servers**.
3. Click the SOA Server name.
4. In the Communications Group area, click **Ports**.

   The value of BOOTSTRAP_ADDRESS is the SOA Server bootstrap port.

27. Perform the following steps to enable load balancing of JMS message processing by MDBs:
    a. Log in to IBM WebSphere Administrative Console.
    b. Click **Resources**, **JMS**, **Activation Specifications**, ***NAME_OF_OIM_ACTIVATION_SPECIFICATION***. Then select **Always activate MDBs in all servers**.
    c. Click **OK** and **Save** the configuration.

Note: You must perform this step individually for each of the following Oracle Identity Manager Activation Specifications:

- oimAttestationQueueMDBActivationSpec
- oimAuditQueueMDBActivationSpec
- oimDefaultQueueMDBActivationSpec
- oimKernelQueueMDBActivationSpec
- oimProcessQueueMDBActivationSpec
- oimReconQueueMDBActivationSpec
- oimSODQueueMDBActivationSpec

28. On Deployment Manager Machine and WebSphere Node 2 Machine, stop, synchronize, and start the Node Agents. For example:

```
$WAS_HOME/profiles/Custom01/bin/stopNode.sh
$WAS_HOME/profiles/Custom01/bin/syncNode.sh DMGR_HOST DMGR_SOAP_PORT
$WAS_HOME/profiles/Custom01/bin/startNode.sh
```

29. On the Deployment Manager Machine, start the servers as follows:

```
$WAS_HOME/profiles/Custom01/bin/startServer.sh SOA_SERVER_1
$WAS_HOME/profiles/Custom01/bin/startServer.sh OIM_SERVER_1
$WAS_HOME/profiles/Custom01/bin/startServer.sh OracleAdminServer
```

30. On the WebSphere Node 2 Machine, start the servers as follows:

```
$WAS_HOME/profiles/Custom01/bin/startServer.sh SOA_SERVER_2
$WAS_HOME/profiles/Custom01/bin/startServer.sh OIM_SERVER_2
```

31. If Oracle Identity Manager administrator user is different than WebSphere administrator user, then perform the following steps:

    a. In the navigator pane of Enterprise Fusion Middleware Control, expand WebSphere Cell to view the cells.

    b. Select the cell on which Oracle Identity Manager and SOA are configured.

    c. Right-click the cell name, and select **Web Services**, **Platform Policy Configuration**.

    d. In the Add New Configure Property window, specify the following values, and then click **OK**.

       – In the Name field, enter `jndi.lookup.csf.key`.

       – In the Value field, enter `admin-csf-key`.

       Note: If the property is not persisted after saving the changes, then perform the following steps:

       1. On the Deployment Manager Machine, go to the Dmgr profile. For example, go to the directory path /profiles/Dmgr01/config/cells/*CELL_NAME*/fmwconfig/policy-accessor-config.xml.
       2. In the policy-accessor section, uncomment the jndi.lookup.key property, and replace the value `{papCsfKey}` value with `admin-csf-key`. This value is the lookup key for admin-user and its password in the credential store.
       3. Save and close the policy-accessor-config.xml file.
       4. Login to the IBM WebSphere Administrative Console, and perform a node synchronization to ensure that the changed configuration is propagated across all nodes of the cluster.
       5. To verify, connect to the nodes of the cluster and check the fmwconfig/policy-accessor-config.xml file in the nodes. The file must be updated with the new values for jndi.lookup.csf.key.

    e. Create a .py file, for example was_admin.py, with the following content:

```
Opss.createCred (map='oracle.wsm.security', key='admin-csf-key',
user='ADMIN_USER_NAME', password='ADMIN_PASSWORD',
desc='wsm-pm admin user csf-key')
AdminApp.edit ('wsm-pm', '[-MapRolesToUsers [[policy.Updater
AppDeploymentOption.No AppDeploymentOption.No ADMIN_USER_NAME ""
AppDeploymentOption.No "user:ADMIN_USER_NAME" "" ]]]')
AdminApp.edit('wsm-pm', '[ -MapRolesToUsers [[policy.Accessor
AppDeploymentOption.No AppDeploymentOption.No ADMIN_USER_NAME ""
AppDeploymentOption.No " |user:ADMIN_USER_NAME" "" ]]]' )
AdminApp.edit('wsm-pm', '[ -MapRolesToUsers [[policy.User
AppDeploymentOption.No AppDeploymentOption.No ADMIN_USER_NAME ""
AppDeploymentOption.No "user:ADMIN_USER_NAME" "" ]]]' )
AdminApp.edit('wsm-pm', '[ -MapRolesToUsers [[policyViewer
AppDeploymentOption.No AppDeploymentOption.No ADMIN_USER_NAME ""
AppDeploymentOption.No " |user:ADMIN_USER_NAME" "" ]]]' )
AdminConfig.save()
```

       Replace *ADMIN_USER_NAME* and *ADMIN_PASSWORD* with admin user credentials.

    f. Run the following script:

```
$COMMON_COMPONENTS_HOME/common/bin/wsadmin.sh
-profileName DMGR_PROFILE_NAME -conntype SOAP -host DMGR_HOSTNAME -port
DMGR_SOAP_PORT -user WEBSPHERE_ADMIN -password WEBSPHERE_ADMIN_PASSWORD -f
was_admin.py
```

g. Restart all the servers.

32. On the Design Console Machine, install the Oracle Identity Manager Design Console. For example:

To start the installer:

```
cd iamsuite\Disk1
setup.exe -jreLoc LOCATION_OF_IBM_JRE
```

Note: When the **Install Software Updates** installer screen appears, you must select the **Skip Software Updates** option.

33. On Design Console Machine, configure the Oracle Identity Manager Design Console using the Oracle Universal Installer Configuration Assistant.

Start the configuration assistant as follows:

```
cd $OIM_HOME\bin
config.bat -jreLoc LOCATION_OF_IBM_JRE
```

Table 4–7 provides information about *specific* Configuration Assistant screens and appropriate information to enter on those screens—the table does not cover self-explanatory screens.

Table 4–7 Information for Specific Configuration Assistant Screens

Screen Name	Input Description
Application Server	Be sure to select **WebSphere**
OIM Server Host and Port	■ The WAS Client Home Location is $*WAS_CLIENT_HOME*. ■ The OIM Server Hostname is the host where *OIM_SERVER_1* was created. ■ You can identify the OIM Server Port and OIM Server Bootstrap Port by performing the following steps: 1) Log in to the IBM WebSphere administrative console. 2) Click **Servers** > **Server Types** > **Web Application Servers**. 3) Click ***OIM_SERVER_1***. 4) Click **Ports** in the Communications Group area. For the OIM Server Port, use the value from WC_defaulthost. For the OIM Server Bootstrap Port, use the value from BOOTSTRAP_ADDRESS.

34. On Design Console Machine, perform the following steps after the Design Console installs, but before you start it:

a. Update the following properties in the *WAS_CLIENT_HOME*/properties/sas.client.props file.

Edit the values as follows. Note that com.ibm.CORBA.securityServerPort represents the Oracle Identity Manager bootstrap port:

```
com.ibm.CORBA.securityServerHost=OIM_SERVER1_HOSTNAME|OIM_SERVER2_HOSTNAME
com.ibm.CORBA.securityServerPort=OIM_SERVER1_BOOTSTRAP_PORT|OIM_SERVER2_BOO
TSTRAP_PORT
com.ibm.CORBA.loginSource=none
```

b. Open the xlconfig.xml file for the Design Console and change the following values:

Set ApplicationURL to:
http://*WEBSERVER_HOSTNAME*:*WEBSERVER_PORT*/

Set java.naming.provider.url to:corbaloc:iiop:*OIM_SERVER1_HOSTNAME*:*OIM_SERVER1_BOOTSTRAP_PORT*,:*OIM_SERVER2_HOSTNAME*:*OIM_SERVER2_BOOTSTRAP_PORT*

35. For additional postinstallation configuration of Oracle Identity Manager, perform the steps described in Section 4.4, "Performing Postinstallation Configuration on IBM WebSphere" and Section 4.6.1, "URL Changes Related to Oracle Identity Manager".

Note: After the installation is complete, the Segregation of Duties (SoD) Check application is successfully enabled on the primary node, but the application fails to deploy on the second node. For more information about this issue, refer to My Oracle Support web site at the following URL:

https://support.oracle.com/

4.3.3 Performing Oracle Identity Manager Clustered Scale Out Configuration

Perform the procedure described in this section to add additional Oracle Identity Manager and SOA server to existing Oracle Identity Manager on IBM WebSphere clustered environment.

By performing the following steps, you will create a configuration as described in Table 4–3, " Overview of Clustered Configuration".

The additional node machines required are:

- WebSphere Node3
- OIM_SERVER_3
- SOA_SERVER_3

To add additional Oracle Identity Manager and SOA server, perform the following steps on the additional node machines:

1. Install IBM WebSphere Application Server Network Deployment 7.0 with fix pack 23 by referring to IBM documentation.

2. Install Oracle SOA Suite 11.1.1.6.0. For more information, refer to the "Installing Oracle SOA Suite (Oracle Identity Manager Users Only)" section of the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.

3. Install Oracle Identity Manager 11*g* Release 2 (11.1.2.1.0). For more information about installing Oracle Identity Manager, refer to the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.

To start the installer, run the following commands:

```
cd iamsuite/Disk1
```

```
./runInstaller -jreLoc LOCATION_OF_IBM_JRE -DSHOW_APPSERVER_TYPE_SCREEN=true
```

4. Start the Oracle Fusion Middleware Configuration Wizard to federate the machine and configure its cell. By default, the Configuration Wizard is located at:

   MW_HOME/Oracle_IDM1/common/bin/was_config.sh

   For more information, refer to the Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server.

   Table 4–8 provides information about specific Configuration Wizard screens and appropriate information to enter on those screens. The table does not cover self-explanatory, standard screens.

Table 4–8 Information for Specific Configuration Wizard Screens

Screen Name	Input Description
Select Configuration Option	Select the Federate Machine and Configure Cell option.
Specify Profile and Node Name Information	Enter information about the profile and node names you want to create for Additional Node machine.
Specify Deployment Manager Information	Enter information about the existing Deployment Manager system.
Select Original Configuration	Be sure to select the Application Servers, Clusters and End Points option. This is a required option.
Configure Additional Cluster Members	Perform the following steps: 1. Click **Add**. 2. In the Name field, enter a name for the second server in the SOA cluster. For example: SOA_SERVER_3. 3. In the Node Name list, select the Node Agent for SOA_SERVER_3. For example: WebSphere_Node3. 4. In the Cluster Name list, select the SOA cluster. 5. Click **Add**. 6. In the Name field, enter a name for the second server in the OIM cluster. For example: OIM_SERVER_3. 7. In the Node Name list, select the Node Agent for OIM_SERVER_3. For example: WebSphere Node3. 8. In the Cluster Name list, select the OIMCluster.

5. Run the copy_jars.sh script. For example:

```
cd $OIM_HOME/server/wasconfig
./copy_jars.sh
```

Note: Before you execute the copy_jars.sh script, ensure the *WAS_HOME*, *COMMON_COMPONENTS_HOME*, and *OIM_ORACLE_HOME* variables are set. *COMMON_COMPONENTS_HOME* represents the location of the Oracle Fusion Middleware common directory, such as *MW_HOME*/oracle_common. *OIM_ORACLE_HOME* represents the location where the Oracle Identity Manager Server is installed, such as *MW_HOME*/Oracle_IDM1.

6. Add the following properties by logging in to the IBM WebSphere Administrative Console and clicking **System Administration**, **Node Agents**, ***NAME_OF_NODE_AGENT_ON_ADDITIONAL_NODE_MACHINE***, **Java and Process Management**, **Process Definition**, **Java Virtual Machine**, **Custom Properties**.

   Note: When you create the properties:

   - An example location for the *PATH_TO_jps-config.xml_IN_THE_fmwconfig_DIRECTORY* is: *WAS_HOME*/profiles/*Custom01*/config/cells/*HOST_NAME_Cell01*/fmwconfig/jps-config.xml
   - An example location for the *PATH_TO_THE_fmwconfig_DIRECTORY* is: *WAS_HOME*/profiles/*Custom01*/config/cells/*HOST_NAME_Cell01*/fmwconfig

   ---

   - **Name:** oracle.security.jps.config
   - **Value:** *PATH_TO_jps-config.xml_IN_THE_fmwconfig_DIRECTORY*
   - **Description (optional):** Adding the jpsconfig location using OPSS System Property
   - **Name:** oracle.domain.config.dir
   - **Value:** *PATH_TO_THE_fmwconfig_DIRECTORY*
   - **Description (optional):** Setting the Key Store Domain Config directory

   Click **OK** and save the changes.

7. Copy wf_client_config.xml.template from OIM_HOME/server/wasconfig directory to WAS_HOME/lib/ext as wf_client_config.xml. For example: cp $OIM_HOME/server/wasconfig/wf_client_config.xml.template $*WAS_HOME*/lib/ext/wf_client_config.xml.

   Update the wf_client_config.xml file with SOA Server hostname and its bootstrap port under <serverURL> tag. For example:

   ```
   <serverURL>corbaloc:iiop:host1:port1,:host2:port2,:host3:port3 </serverURL>
   ```

   Tip: You can identify the SOA bootstrap port by performing the following steps:

   1. Log in to IBM WebSphere Administrative Console.
   2. Select **Servers**, **Server Types**, **Web Application Servers**.
   3. Click the SOA Server name.
   4. In the Communications Group area, click **Ports**.

      The value of BOOTSTRAP_ADDRESS is the SOA Server bootstrap port.

8. Stop, synchronize, and start the Node Agents, SOA Server and OIM Server. For example:

   ```
   $WAS_HOME/profiles/Custom01/bin/stopNode.sh
   $WAS_HOME/profiles/Custom01/bin/syncNode.sh DMGR_HOST DMGR_SOAP_PORT
   $WAS_HOME/profiles/Custom01/bin/startNode.sh
   ```

```
$WAS_HOME/profiles/Custom01/bin/startServer.sh SOA_SERVER_3
$WAS_HOME/profiles/Custom01/bin/startServer.sh OIM_SERVER_3
```

4.4 Performing Postinstallation Configuration on IBM WebSphere

This section describes the following postinstallation configuration tasks on IBM WebSphere:

- Configuring Transaction Timeout Properties
- Updating SOA Server Default Composite (Cluster Only)
- Accessing the Dynamic Monitoring Service Application (Optional)
- Seeding LDAP Reconciliation Scheduled Jobs into the Database Schema
- Changing Memory Settings for Oracle Identity Manager
- Performing Postinstallation Configuration of IHS (Optional)
- Adjusting Email Notification WSUrl (Cluster Only)

4.4.1 Configuring Transaction Timeout Properties

To change the transaction timeout properties to 10 minutes:

1. Log in to IBM WebSphere Administrative Console.
2. Navigate to the Transaction service panel by selecting **Servers**, **Server Types**, **WebSphere application servers**, *oim_server_name*, **Container Services**, **Transaction Service**.
3. Change the value of Total transaction lifetime timeout to `600`.

   The default value is 120.
4. Change the value of Maximum transaction timeout to 600 seconds.

   The default value is 300.
5. Stop and restart WebSphere Application Server. In a clustered deployment, this must be done on all Oracle Identity Manager servers.

4.4.2 Updating SOA Server Default Composite (Cluster Only)

In an integrated environment, Oracle Identity Manager is front ended by HTTP Server. Therefore, all SOA server default composites must be updated.

To update the SOA server default composite:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control Console.
2. Navigate to **SOA**, **soa-infra** (SOA server name), **default**.

   The following default composites are available: DefaultRequestApproval, DefaultOperationalApproval, DefaultRoleApproval, DefaultSODApproval, BeneficiaryManagerApproval, RequesterManagerApproval, CertificationProcess, DisconnectedProvisioning.
3. For each default composite, perform the following steps:
   a. Click the composite name.
   b. From Component Metrics, click on task with Component Type as Human Workflow.

c. Select the Administration tab and update the fields as follows:

   Host Name: HTTP Server host

   HTTP Port: If SSL mode, leave blank. If non-SSL mode, enter HTTP Server port.

   HTTPS Port: If SSL mode, enter HTTPS server port. If non-SSL mode, leave blank.

d. Click **Apply**.

4.4.3 Accessing the Dynamic Monitoring Service Application (Optional)

To access the Dynamic Monitoring Service (DMS) application on IBM WebSphere:

1. Log in to IBM WebSphere Administrative Console as the administrator.
2. On the left pane, go to **Applications**, **Application Types**, **WebSphere enterprise applications**.
3. On the right pane, click **Dmgr DMS Application_11.1.1.1.0**.
4. Click **Security role to user/group mapping**.
5. Select the **Admin** role, and click **Map Users**.
6. Type `wasadmin` in the search string, and click **Search**.
7. Select **wasadmin** in the Available box, and click the right arrow.
8. Click **OK** to go back. Click **OK** again.
9. Click **Save directly to the master configuration**.
10. Start Dmgr DMS Application_11.1.1.1.0.
11. Repeat steps 3 to 10 for DMS Application_11.1.1.1.0.
12. Stop all servers and the Deployment Manager. Start the Deployment Manager, synchronize the nodes, start nodes, and start all servers.

You can access the DMS application from the following URL:

http://*OIM_HOST*:*OIM_PORT*/dms/Spy

4.4.4 Seeding LDAP Reconciliation Scheduled Jobs into the Database Schema

While configuring postinstallation LDAP synchronization for Oracle Identity Manager, perform the following steps to load the LDAP reconciliation scheduled jobs into the Quartz table of the Oracle Identity Manager database schema by performing the following steps:

> **See Also:** "Enabling LDAP Synchronization in Oracle Identity Manager" in the *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite* for information about postinstallation configuration of LDAP synchronization for Oracle Identity Manager

1. As a prerequisite, set the OIM_ORACLE_HOME environment variable. For example:

   For UNIX, run the following command:

   ```
   setenv OIM_ORACLE_HOME /u01/mwhome/Oracle_IDM
   ```

2. Seeding the LDAP reconciliation scheduled jobs can be performed in any one of the following ways:

   Seeding LDAP reconciliation scheduled jobs with parameters:

   a. Go to the $*OIM_ORACLE_HOME*/server/setup/deploy-files directory.

   b. Set ant home. The following is a sample command to set ant home in UNIX:

   ```
   setenv ANT_HOME /u01/mwhome/modules/org.apache.ant_1.7.1
   ```

   Note: If ANT is not installed, then download and ANT from Oracle Technology Network (OTN) web site by navigating to the following URL:

   http://www.oracle.com/technetwork/index.html

   Install ANT and set the ANT_HOME. Make sure that ant executable file exists in the $ANT_HOME/bin/ant/ directory.

   c. Run the following ant command with parameters:

   ```
   $ANT_HOME/bin/ant -f setup.xml seed-ldap-recon-jobs
   -DoperationsDB.driver=oracle.jdbc.OracleDriver
   -DoperationsDB.user=SCHEMA_OWNER -DOIM.DBPassword=SCHEMA_OWNER_PASSWORD
   -DoperationsDB.host=SCHEMA_HOST_ADDRESS
   -DoperationsDB.port=SCHEMA_PORT_NUMBER
   -DoperationsDB.serviceName=SCHEMA_SERVICE_NAME -Dssi.provisioning=ON
   -Djta.location=WAS_INSTALATION_DIR/plugins/javax.j2ee.jta.jar
   -Dojdbc.location=OJDBC_LOCATION -Dwork.dir=seed_logs
   ```

   For example:

   ```
   $ANT_HOME/bin/ant -f setup.xml seed-ldap-recon-jobs
   -DoperationsDB.driver=oracle.jdbc.OracleDriver
   -DoperationsDB.user=schemaowner1_OIM -DOIM.DBPassword=SCHEMA_OWNER_PASSWORD
   -DoperationsDB.host=myhost.mycompany.com -DoperationsDB.port=1234
   -DoperationsDB.serviceName=oimdb.regress.rdbms.mycompany.com
   -Dssi.provisioning=ON
   -Djta.location=WAS_INSTALATION_DIR/plugins/javax.j2ee.jta.jar
   -Dojdbc.location=MW_HOME/oracle_common/inventory/Scripts/ext/jlib/ojdbc6.ja
   r -Dwork.dir=seed_logs
   ```

   Seeding LDAP reconciliation scheduled jobs with the profile file:

   a. Set the following environment variables:

   – OIM_ORACLE_HOME to the OIM_HOME directory.

   – Set ANT_HOME to the directory on which ANT is installed.

   Note: If ANT is not installed, then download and ANT from Oracle Technology Network (OTN) web site by navigating to the following URL:

   http://www.oracle.com/technetwork/index.html

   Install ANT and set the ANT_HOME. Make sure that ant executable file exists in the $ANT_HOME/bin/ant/ directory.

b. Go to the $*OIM_ORACLE_HOME*/server/bin/ directory.

c. Create a property file with the properties listed in Table 4–9.

Note: You can also use the appserver.profile file instead of creating a new property file. Make sure that the properties listed in this step are present with the values.

Table 4–9 Parameters of the Property File

Parameter	Description
operationsDB.user	Oracle Identity Manager database schema owner.
operationsDB.driver	Constant value of oracle.jdbc.OracleDriver.
operationsDB.host	Oracle Identity Manager database schema host address.
OIM.DBPassword	Oracle Identity Manager database schema owner's password.
operationsDB.serviceName	Oracle Identity Manager database schema service name, for example, oimdb.regress.rdbms.mycompany.com.
operationsDB.port	Oracle Identity Manager database schema port number.
ssi.provisioning	Value must be ON.
jta.location	Value is *WAS_INSTALLATION_DIRECTORY*/plugins/javax.j2ee.jta.jar.
ojdbc.location	Directory on which JDBC is installed, for example, *MW_HOME*/oracle_common/inventory/Scripts/ext/jlib/ojdbc6.jar.
work.dir	Any preferred directory on which log files will be created After successful completion of target, you can check logs at the $*WORK_DIR*/seed_logs/ldap/SeedSchedulerData.log file.

d. Go to the $*OIM_ORACLE_HOME*/server/setup/deploy-files/ directory.

e. Run the following command:

```
$ANT_HOME/bin/ant -f setup.xml seed-ldap-recon-jobs -propertyfile
$OIM_ORACLE_HOME/server/bin/PROPERTY_FILE_NAME
```

4.4.5 Changing Memory Settings for Oracle Identity Manager

For staging and test deployments of Oracle Identity Manager, the maximum heap size of 2 GB is recommended. For the maximum heap size in production deployments, refer to *Oracle Fusion Middleware Performance and Tuning Guide*.

To change the heap setting for Oracle Identity Manager on WebSphere:

1. Log in to the WebSphere Administrative Console.
2. Navigate to **Servers**, **Server Types**, **WebSphere application servers**, *server_name*, **Java & Process Management**, **Process Definition**, **Java Virtual Machine**.
3. Set the value of Maximum heap size to `2048`.
4. Save the changes, and restart the server.

4.4.6 Performing Postinstallation Configuration of IHS (Optional)

If IHS configuration is used in your deployment, then perform the following steps for postinstallation configuration of IHS:

1. Configure virtual host alias for IHS. To do so:
   a. Login to IBM WebSphere Administrative Console.
   b. Select the **default_host** virtual host.
   c. Create the virtual host alias for IHS by providing values for IHS host and port.
2. Configure IHS with WebSphere as follows:
   a. Copy *IHS_INSTALL_DIRECTORY*/Plugins/bin/configurewebserver1.sh to the *WAS_HOME*/bin/ directory.
   b. Run the configurewebserver1.sh script from the WAS_HOME/bin/ directory as follows:

      ```
      configurewebserver1.sh -user WAS_ADMIN_USER -password WAS_ADMIN_PASSWORD
      -ihsAdminPassword IHS_ADMIN_PASSWORD
      ```

      The script generates the port bindings and creates the plugin-cfg.xml file for use by WebSphere and IHS.
   c. In the IBM WebSphere Administrative Console, go to **Servers**, **Web server**. The new webserver1 is displayed in the list.
   d. Select webserver1, and click **Propagate** to propagate the plug-in to IHS. Verify that the updated plugin-cfg.xml file is propagated to the *IHS_INSTALL_DIR*/Plugins/config/webserver1/ directory.
3. Configure IHS port and URL as follows:
   a. Configure SOA composites to point to IHS as described in Section 4.4.2, "Updating SOA Server Default Composite (Cluster Only)".
   b. Configure Oracle Identity Manager frontend ports to point to IHS as described in Section 4.6.1.3, "Oracle Identity Manager Host and Port Changes".
4. Restart all servers.
5. Verify the Oracle Identity Manager URL by navigating to:

   http://*HOST_NAME*:*PORT*/identity

4.4.7 Adjusting Email Notification WSUrl (Cluster Only)

In a clustered deployment of Oracle Identity Manager on IBM WebSphere, perform the following steps to adjust email notification WSUrl to point to IHS:

1. Login to Oracle Enterprise Manager.
2. Click **Application Deployments**.
3. Right-click **OIMAppMetadata(*OIM_SERVER_NAME*)**, and select **System MBean Browser**.
4. In the System MBean Browser, navigate to **Application Defined MBeans**, **oracle.iam**, **Server: *OIM_SERVER_NAME***, **Application: oim**, **IAMAppRuntimeMBean**, and select **UMSEmailNotificationProviderMBean**.
5. In the Attributes tab, locate WSUrl, and replace the existing host name and port number with the host name and port number of IHS.

4.5 Upgrading Oracle Identity Manager on IBM WebSphere

This section describes the steps required to upgrade and configure Oracle Identity Manager Release 9.x to Oracle Identity Manager 11*g* Release 2 (11.1.2.1.0) on IBM WebSphere. It contains the following sections:

- Prerequisites for the Upgrade
- Installing Oracle Identity Manager
- Upgrading Oracle Identity Manager Schema
- Configuring Oracle Identity Manager
- Upgrading Features Using MT Upgrade Utility in Post-Config Mode
- Performing Postupgrade Configuration

4.5.1 Prerequisites for the Upgrade

Before upgrading Oracle Identity Manager Release 9.x to 11*g* Release 2 (11.1.2.1.0) on IBM WebSphere, make sure that:

- A WAS_HOME where IBM WebSphere Application Server 7.0.0 with fixpack 19 has been installed.
- A Middleware home location exists with SOA installed on it.
- Oracle Database 11*g* with Oracle Identity Manager dependent schemas, such as MDS, SOAINFRA, OPSS, and ORASDPM, are created.

Perform the following prerequisites steps:

1. **Run the PreUpgradeReport utility.**

   You must run the PreUpgradeReport utility to analyze your Oracle Identity Manager environment before you begin the upgrade process. Address all issues listed as part of this report with the solution provided. After fixing the issues, run the report until no pending issues are listed in the report. See "Pre-Upgrade" in the *Oracle Fusion Middleware Upgrade and Migration Guide for Oracle Identity and Access Management* for information about running the PreUpgradeReport utility.

2. **Install IBM WebSphere Application Server.**

   Follow the instructions in Section 2.4, "Task 4: Install the IBM WebSphere Software" for installing IBM WebSphere Application Server 7.0 and applying the latest Fix Pack for IBM WebSphere 7.0.

3. **Install Oracle SOA Suite (11.1.1.6.0).**

   See "Installing Oracle SOA Suite 11.1.1.6.0 (Oracle Identity Manager Users Only)" in the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management* for information about installing SOA Suite.

4. **Create the database schema.**

   You must create and load the appropriate Oracle Fusion Middleware schemas in the database using Repository Creation Utility (RCU) before installing and configuring Oracle Identity Manager. See "Creating Database Schema Using the Oracle Fusion Middleware Repository Creation Utility (RCU)" in the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management* for details.

4.5.2 Installing Oracle Identity Manager

Install Oracle Identity Manager as a part of Oracle Identity and Access Management 11*g* by running the Oracle Identity and Access Management Installer. To do so, follow the instructions in the following sections of the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*:

- "Starting the Oracle Identity and Access Management Installer"
- "Installing Oracle Identity and Access Management (11.1.2)"

4.5.3 Upgrading Oracle Identity Manager Schema

Before you begin:

- Create a backup of Oracle Identity Manager Release 9.x Schema.
- Run the OSI Data Upgrade using the OSI Data Upgrade Utility. For more information about running the OSI Data Upgrade Utility, see the technote "OSI Data Upgrade Utility for Upgrading OIM 9.1.0.x to OIM 11g Version" with ID 1303215.1 at the following URL:

  `https://support.oracle.com`
- Set the JAVA_HOME environment variable.

To upgrade Oracle Identity Manager Release 9.x schema to 11*g* Release 2 (11.1.2.1.0):

1. Start the Oracle Fusion Middleware Upgrade Assistant by running the following command:

   ```
   ./ua
   ```

   The Welcome page of the Oracle Fusion Middleware Upgrade Assistant wizard is displayed.
2. Click **Next**. The Specify Operation page of the wizard is displayed.
3. Select the **Upgrade Oracle Identity Manager Schema** option, and then click **Next**.
4. In the Prerequisites page, select all the checkboxes to specify that the prerequisites have been met. Click **Next**.
5. In the Specify OIM Database page, enter the following connection details for the source Oracle Identity Manager database, and then click **Next**.
   - **Host:** Name of the host on which the database is deployed.
   - **Port:** Port number to connect to the host identified in the Host field.
   - **Service Name:** A string that is the global database name, a name comprised of the database name and domain name, entered during installation or database creation.
   - **OIM Schema:** Name of the Oracle Identity Manage schema.
   - **SYS Password:** Database system administrator password.
6. In the Upgrading Components page, a progress bar shows the progress of the schema upgrade. The status of the upgrade components are also displayed. When finished, click **Next**.
7. In the Upgrade Summary page, expand the upgrade component names to display the summary information of the upgrade. When finished, click **Upgrade**.

Note: After the schema upgrade is performed, you must disable workflow upgrade before proceeding to the next step. To disable workflow upgrade, run the following SQL command:

```
update upgrade_feature_state set
IS_FEATURE_UPGRADED='Y',FEATURE_UPGRADE_STATE='UPGRADED' where
FEATURE_ID = 'OIM91UPG.Workflow'
```

4.5.4 Configuring Oracle Identity Manager

You must manually perform the following steps to configure Oracle Identity Manager:

- Creating and Configuring a Cell
- Performing Manual Configuration Steps
- Upgrading CSF Seeding
- Upgrading Oracle Identity Manager Components

4.5.4.1 Creating and Configuring a Cell

To create and/or extend a cell with the Oracle Identity Manager 11*g* Release 2 (11.1.2.1.0) components:

1. Start the Fusion Middleware Configuration Wizard by running the following command:

   ```
   cd $OIM_ORACLE_HOME/common/bin
   ./was_config.sh -log=config.log -log_priority=debug
   ```

   The Select Configuration Option page of the Fusion Middleware Configuration Wizard is displayed.

2. Select the Create and Configure Cell option, and click **Next**.

3. In the Specify Cell, Profile and Node Information page, you can specify the default names, or you can provide new names. Enter the following values, and then click **Next**.

   - **Cell Name:** *HOST_NAME*Cell01
   - **Deployment Manager Profile Name:** Dmgr01
   - **Deployment Manager Node Name:** *HOST_NAME*CellManager01
   - **Application Server Profile Name:** Custom01
   - **Application Server Node Name:** *HOST_NAME*Node01

4. In the Specify Deployment Manager Information page, enter WebSphere administrator username and password. The WebSphere administrator username and password provided here will be used for logging into Oracle Identity Manager UI and for later configuration steps.

   Click **Next**.

5. In the Add Products to Cell page, select the products that you want to add to the cell. Make sure to select the correct WAS ND template for the WAS ND install and not the WAS AS template. When finished, click **Next**.

6. In the Configure JDBC Component Schema page, note that the connection test must succeed. If the Configuration Wizard cannot contact the database, then the

Configuration Wizard might not generate the WAS files correctly, though an error might not be displayed.

Click **Next**.

7. Continue with the installation steps by clicking Next until the Test JDBC Component Schema page is displayed.

   The Oracle Identity Manager template and dependent templates create three servers: oim_server1, soa_server1, and OracleAdminServer. The oim, Nexaweb, OIMMetadata, and XIMDD applications are deployed on oim_server1.

4.5.4.2 Performing Manual Configuration Steps

Before you run the copy_jars.sh, seed_opss_permission.sh, and configure_nodeagent.sh scripts, ensure that the following variables are set to avoid or to bypass the prompting for environment variable:

- DMGR_PROFILE_ROOT: WebSphere Deployment Manager profile directory, for example, /opt/softwares/IBM/WebSphere/AppServer/profiles/Dmgr01/.
- OIM_ORACLE_HOME : See Table 4–1, " Conventions Used in this Document".
- WEBSPHERE_ADMIN : WebSphere administrator username.
- WEBSPHERE_ADMIN_PASSWORD : WebSphere administrator password.
- CELL_HOME_LOCATION : Location of the WebSphere cell home directory, for example, /opt/softwares/IBM/WebSphere/AppServer/profiles/Dmgr01/config/cells/*HOST_NAME*Cell01.
- DMGR_PROFILE_NAME : WebSphere Deployment Manager profile name, for example, Dmgr01.
- DMGR_HOSTNAME : WebSphere Deployment Manager hostname.
- DMGR_SOAP_PORT : WebSphere Deployment Manager SOAP port.
- WAS_HOME: See Table 4–1, " Conventions Used in this Document".
- COMMON_COMPONENTS_HOME : Oracle Middleware common directory, for example, /opt/softwares/IBM/WebSphere/oracle_common.

To perform the manual configuration steps before you use the Configuration Assistant:

1. Copy the JAR files to Copy the jar files to $WAS_HOME/lib/ext/ by running the following command:

   ```
   ./copy_jars.sh
   ```

2. Start, stop, and synchronize the Node Agent as follows:

   ```
   $WAS_HOME/profiles/Custom01/bin/stopNode.sh -username USER_NAME -password
   PASSWORD
   $WAS_HOME/profiles/Dmgr01/bin/startManager.sh
   $WAS_HOME/profiles/Custom01/bin/syncNode.sh DMGR_HOST DMGR_SOAP_PORT -username
   USER_NAME -password PASSWORD
   $WAS_HOME/profiles/Custom01/bin/startNode.sh
   ```

   Use the username and password that you used for cell creation. The port numbers to be used during sync node are available in the $*WAS_HOME*/profiles/*DMGR*/logs/AboutThisProfile.txt file.

3. Go to the *OIM_HOME*/server/wasconfig/ directory, and run the following commands:

```
sh seed_opss_permission.sh
```

And:

```
sh configure_nodeagent.sh
```

Note: The following error message is generated on running the seed_opss_permission.sh script:

```
WASX7487E: Failed to import script libraries modules:
/u02/Oracle/Middleware/oracle_common/common/wsadmin/wsmAgent.py;
Examine the wsadmin log file to determine the problem.
```

This is a benign error and can be ignored.

---

4. Stop, synchronize, and start the node, as shown:

```
$WAS_HOME/profiles/Custom01/bin/stopNode.sh -username USER_NAME -password
PASSWORD
$WAS_HOME/profiles/Custom01/bin/syncNode.sh DMGR_HOST DMGR_SOAP_PORT -username
USER_NAME -password PASSWORD
$WAS_HOME/profiles/Custom01/bin/startNode.sh
$WAS_HOME/profiles/Custom01/bin/startServer.sh SOA_SERVER
```

Use the same username and password that you used for cell creation.

4.5.4.3 Upgrading CSF Seeding

To upgrade CSF seeding by running the MT upgrade script in pre-config mode:

1. Perform the following as prerequisites:
   - Copy .xldatabasekey to the *WAS_HOME*/profiles/*Dmgr*/config/cells/*HOST_NAMECell*/fmwconfig/ directory.
   - Populate the *MW_HOME*/Oracle_IDM1/server/bin/upgrade_was.properties file with the correct input properties. Table 4–10 lists the input properties and sample values.

Table 4–10 Sample Input Values for upgrade_was.properties

Input Property	Sample Value
Oracle Identity Manager Release 9.x home location	`oim91Home=/scratch/oim9101was/xellerate`
Oracle Identity Manager schema JDBC URL	`oimSchemaJDBCURL=HOST_NAME:PORT:oimdb`
Oracle Identity Manager schema name	`oimSchemaName=oim91011`
Oracle Identity Manager MDS schema JDBC URL	`mdsSchemaURL=HOST_NAME:PORT:oimdb`
Oracle Identity Manager MDS schema	`mdsSchemaName=Sample_MDS`

Table 4–10 (Cont.) Sample Input Values for upgrade_was.properties

Input Property	Sample Value
Oracle OIM home	`oracleOIMHome=/scratch/wasr2install/mw/Oracle_IDM1`
Middleware home	`middleWareHome=/scratch/wasr2install/mw`
SOA host name	`soaHost=soahost.mycompany.com`
SOA port number	`soaPort=`*`PORT_NUMBER`*
SOA user name	`soaUserName=wasadmin`
WAS domain manager cell home	`wasCellHome=/scratch/wasr2install/was/profiles/Dmgr03/config/cells/`*`HOST_NAME`*`Cell02`

MT in pre-config mode	`CSFSeed=true`
	When CSFSeed=true, MT is run is pre-config mode, and the following properties are set:
	`PRE_OIM_CONFIG=true`
	`POST_OIMCONFIG=false`

Note: The *WAS_HOME*/profiles/Dmgr03/properties/portdef.props file contains all the port numbers relevant to the particular cell.

– Set the JAVA_HOME and APPSERVER_TYPE environment variables. JAVA_HOME must point to IBM_JDK.

2. Go to the *MW_HOME*/Oracle_IDM1/server/bin/ directory, and run the OIMMTUpgrade_WS.sh script, as shown:

```
export JAVA_HOME=/scratch/wasr2install/was/java/
export APPSERVER_TYPE='was'
./OIMMTUpgrade_WS.sh
```

Note: The log file for the script is *MW_HOME*/Oracle_IDM1/server/upgrade/logs/MT/ua-*TIME_STAMP*.log.

4.5.4.4 Upgrading Oracle Identity Manager Components

To upgrade Oracle Identity Manager components by running the Configuration Assistant:

1. Start the Configuration Assistant by running the following command:

```
cd $OIM_HOME/bin
./config.sh -jreLoc LOCATION_OF_IBM_JDK -DSHOW_APPSERVER_TYPE_SCREEN=true
```

2. In the Components to Configure page of the Oracle Identity Management Configuration wizard, expand **Oracle Identity Manager**, and select **OIM Server**. Then, click **Next**.
3. In the Database page, enter the database connect string and schema details. When finished, click **Next**.
4. In the Application Server page, verify that **WebSphere** is selected. Then, click **Next**.

   Note: The application server type is selected by default if a SOA home has already been installed and the type has been set to WebSphere. If not, then select **WebSphere** as the application server type.

5. In the WebSphere Details page, specify values for the following:
   - **Cell Path:** This is the WebSphere cell home location, which is $*WAS_HOME*/profiles/*Dmgr01*/config/cells/*CELL_NAME*. The default cellname is *HOST_NAME*Cell01.
   - **Admin URL:** The WebSphere Admin URL port can be obtained from the Management bootstrap port entry in the $*WAS_HOME*/profiles/*Dmgr01*/logs/AboutThisProfile.txt file.
   - **Admin Soap Port:** This is the Admin SOAP port for the WebSphere Application Server.
   - **Admin UserName:** The same user name provided for domain creation.
   - **Admin Password:** The password provided for domain creation.
6. In the OIM Server page, enter the Oracle Identity Manager server admin password, keystore password, and the URL information. Then, click **Next**.
7. Continue with the steps of the wizard by clicking **Next** until the configuration completes.
8. Copy wf_client_config.xml.template from the *OIM_HOME*/server/wasconfig/ directory to the *WAS_HOME*/lib/ext/ directory as wf_client_config.xml.
9. Update the wf_client_config.xml file with the SOA Server hostname and its bootstrap port under the <serverURL> tag. The tag is in the following format:

   ```
   <serverURL>corbaloc:iiop:SOA_SERVER_HOSTNAME:SOA_SERVER_BOOTSTRAP_PORT</serverURL>
   ```

   For example:

   ```
   <serverURL>corbaloc:iiop:myhost.mycompany.com:2800</serverURL>
   ```

10. Stop the node, start manager, and sync nodes, as shown:

    ```
    $WAS_HOME/profiles/Custom01/bin/stopServer.sh SOA_SERVER -username USER_NAME
    -password PASSWORD
    $WAS_HOME/profiles/Custom01/bin/stopNode.sh -username USER_NAME -password
    PASSWORD
    $WAS_HOME/profiles/Dmgr01/bin/stopManager.sh -username USER_NAME -password
    PASSWORD
    $WAS_HOME/profiles/Dmgr01/bin/startManager.sh
    $WAS_HOME/profiles/Custom01/bin/syncNode.sh DMGR_HOST DMGR_SOAP_PORT -username
    USER_NAME -password PASSWORD
    ```

```
$WAS_HOME/profiles/Custom01/bin/startNode.sh
$WAS_HOME/profiles/Custom01/bin/startServer.sh OracleAdminServer
$WAS_HOME/profiles/Custom01/bin/startServer.sh SOA_SERVER
$WAS_HOME/profiles/Custom01/bin/startServer.sh OIM_SERVER
```

Note: The username and password are the same that you used during cell creation.

When finished, make sure that you start the respective managed servers.

4.5.5 Upgrading Features Using MT Upgrade Utility in Post-Config Mode

After Oracle Identity Manager configuration is complete and all the servers including OIM server is up for populating default metadata, you can upgrade all the features using the MT upgrade utility in the post-config mode.

To upgrade the features by using the MT upgrade utility in the post-config mode:

1. Perform the following prerequisites:
   - Shut down Oracle Identity Manager after populating the default metadata.
   - Make sure that the Admin and SOA servers are up and running.
   - Populate the $*MW_HOME*/Oracle_IDM1/server/bin/upgrade_was.properties file with the correct input properties. Table 4–11 lists the input parameters with sample values.

Table 4–11 Input Parameters for upgrade_was.properties

Input Parameter	Sample Value
Oracle Identity Manager Release 9.x home location	`oim91Home=/scratch/oim9101was/xellerate`
Oracle Identity Manager schema JDBC URL	`oimSchemaJDBCURL=HOST_NAME:PORT:oimdb`
Oracle Identity Manager schema name	`oimSchemaName=oim91011`
Oracle Identity Manager MDS schema JDBC URL	`mdsSchemaURL=HOST_NAME:PORT:oimdb`
Oracle Identity Manager MDS schema name	`mdsSchemaName=Sample_MDS`
Oracle OIM home	`oracleOIMHome=/scratch/wasr2install/mw/Oracle_IDM1`
Middleware home	`middleWareHome=/scratch/wasr2install/mw`
SOA host name	`soaHost=soahost.mycompany.com`
SOA port number	`soaPort=SOA_PORT`
SOA user name	`soaUserName=wasadmin`

Table 4–11 (Cont.) Input Parameters for upgrade_was.properties

Input Parameter	Sample Value
WebSphere domain manager cell home	`wasCellHome=/scratch/wasr2install/was/profiles/Dmgr03/config/cells/HOST_NAMECell02`

MT in post-config mode	`CSFSeed=false`
	When CSFSeed=false, MT is run is pre-config mode, and the following properties are set:
	`PRE_OIM_CONFIG=false`
	`POST_OIMCONFIG=true`

Note: The *WAS_HOME*/profiles/Dmgr03/properties/portdef.props file contains the port numbers relevant to the particular cell.

- Set the JAVA_HOME and APPSERVER_TYPE environment variables. JAVA_HOME must point to IBM_JDK.

2. Go to the *MW_HOME*/Oracle_IDM1/server/bin/ directory, and run the OIMMTUpgrade_WS.sh script, as shown:

```
export JAVA_HOME=/scratch/wasr2install/was/java/
export APPSERVER_TYPE='was'
./OIMMTUpgrade_WS.sh
```

Note: The log file for the script is *MW_HOME*/Oracle_IDM1/server/upgrade/logs/MT/ua-*TIME_STAMP*.log.

3. Analyze the Feature Upgrade Summary Report. Start the Oracle Identity Manager Managed Servers, and access the application.

4.5.6 Performing Postupgrade Configuration

After upgrading Oracle Identity Manager Release 9.x to Release 11*g* Release 2 (11.1.2.1.0), perform the following postupgrade configuration:

- Customizing the UI to Mark Attributes as Required

4.5.6.1 Customizing the UI to Mark Attributes as Required

After upgrading Oracle Identity Manager Release 9.x to 11*g* Release 2 (11.1.2.1.0), the upgraded metadata files have certain attributes as mandatory. But, these attributes are not marked as required in the UI. For example, the upgraded metadata files for the create user operation, such as CreateUserDataSet.xml and User.xml, have first name and user login attributes as mandatory, but these attributes are not marked as required in the UI.

For fields that you want to retain as required, such as First Name and User Login, on a screen, perform the following steps:

1. Create a sandbox and activate it.
2. Go to the specific screen, for example Create User, enter values in the existing mandatory fields, and then click **Customize** at the top.
3. On the Composer menu, select **View**, **Source**.
4. Click the field, and then confirm to edit taskflow. Click **Edit** to open the Component Properties dialog box.
5. On the Component Properties dialog box, select the option for the Show Required property.
6. For the Required property, open the Expression Editor and enter true as the value.
7. Click **Apply**, and then click **OK**.
8. On the Composer toolbar, click **Close**, and test your changes.
9. Export and publish the sandbox.

4.6 Handling Lifecycle Management Changes on IBM WebSphere

Because of integrated deployment of Oracle Identity Manager with other applications, such as Oracle Access Management (OAM), and configuration changes in those applications, various configuration changes might be required in Oracle Identity Manager and IBM WebSphere Application Server. These configuration changes are described in the following sections:

- URL Changes Related to Oracle Identity Manager
- Password Changes Related to Oracle Identity Manager
- Configuring SSL for Oracle Identity Manager

4.6.1 URL Changes Related to Oracle Identity Manager

Oracle Identity Manger uses various hostname and port in its configuration because of the architectural and middleware requirements. This section describes ways to make the corresponding changes in Oracle Identity Manager and IBM WebSphere Application Server configuration for any change in the integrated and dependent applications.

This section contains the following topics:

- Oracle Identity Manager Database Host and Port Changes
- Oracle Virtual Directory Host and Port Changes
- Oracle Identity Manager Host and Port Changes
- SOA Host and Port Changes
- OAM Host and Port Changes

4.6.1.1 Oracle Identity Manager Database Host and Port Changes

This section describes the configuration areas where database hostname and port number are used.

After installing Oracle Identity Manager, if there are any changes in the database hostname or port number, then the following changes are required:

Note: Before making changes to the database host and port, shutdown the managed servers hosting Oracle Identity Manager. But you can keep IBM WebSphere Administrative Server running.

- **To change datasource oimJMSStoreDS configuration:**
  1. Navigate to **Resources**, **JDBC**, **Data Sources**, and then **oimJMSStoreDS**.
  2. Modify the values of the URL to reflect the changes to database host and port.
- **To change datasource ApplicationDB configuration:**
  1. Navigate to **Resources**, **JDBC**, **Data Sources**, and then **applicationDB**.
  2. Modify the values of the URL to reflect the changes to database host and port.
- **To change datasource oimOperationsDB configuration:**
  1. Navigate to **Resources**, **JDBC**, **Data Sources**, and then **oimOperationsDB**.
  2. Modify the values of the URL to reflect the changes to database host and port.
- **To change the datasource related to Oracle Identity Manager Meta Data Store (MDS) configuration:**
  1. Navigate to **Services**, **JDBC**, **Data Sources**, and then **mds-oim**.
  2. Modify the values of the URL and Properties fields to reflect the changes in the database host and port.
- **To change Custom Registry configuration:**
  1. In IBM WebSphere Administrative console, navigate to **Security**, **Global security**.
  2. Click **Configure** next to the Standalone custom registry.
  3. Select **DBUrl**, and then click **Edit**.
  4. Modify the value of the DBUrl field to reflect the change in hostname and port.

  Note: If Service Oriented Architecture (SOA) and Oracle Web Services Manager (OWSM) undergo configuration changes, then you must make similar changes for datasources related to SOA or OWSM.

After making changes in the datasources, restart the IBM WebSphere Application Server, and start the Oracle Identity Manager Managed WebSphere servers.

Note: Whenever Oracle Identity Manager application configuration information is to be changed by using OIM App Config MBeans from the Enterprise Management (EM) console, at least one of the Oracle Identity Manager Managed Servers must be running. Otherwise, you cannot figure out any of the OIM App Config MBeans from the EM console.

- **To change DirectDB configuration:**
  1. Log in to Enterprise Manager by using the following URL:

http://*ORACLE_ADMIN_SERVER*/em

2. Navigate to **Websphere Cell**, **OIM server**.
3. Right-click **OIM server**, and select to **System MBean Browser**.
4. In the System MBean Browser, navigate to **Application Defined MBeans**.
5. Navigate to **oracle.iam**, **Application:oim**, **XMLConfig**, **Config**, **XMLConfig.DirectDBConfig**, and then **DirectDB**.
6. Enter the new value for the URL attribute to reflect the changes to host and port, and then apply the changes.

Note: When Oracle Identity Manager single instance deployment is changed to Oracle Real Application Clusters (Oracle RAC) or Oracle RAC is changed to single instance deployment, change the oimJMSStoreDS, oimOperationsDB, and mds-oim datasources. In addition to the generic changes to make these datasources to multidatasource configuration, change the Custom Registry and domain credential store configurations to reflect the Oracle RAC URL. For information about these generic changes, see *Oracle Fusion Middleware High Availability Guide*.

4.6.1.2 Oracle Virtual Directory Host and Port Changes

When LDAP synchronization is enabled, Oracle Identity Manager connects with directory servers through Oracle Virtual Directory (OVD). This connection takes place by using LDAP/LDAPS protocol.

To change OVD host and port:

1. Log in to Oracle Identity System Administration.
2. Under Configuration, click **IT Resource**.
3. From the IT Resource Type list, select **Directory Server** , and click **Search**.
4. Edit the Directory Server IT resource. To do so:
   a. If the value of the Use SSL field is set to `False`, then edit the Server URL field. If the value of the Use SSL field is set to `True`, then edit the Server SSL URL field.
   b. Click **Update**.

4.6.1.3 Oracle Identity Manager Host and Port Changes

This section consists of the following topic:

- Changing OimFrontEndURL in Oracle Identity Manager Configuration

---

Note: When additional Oracle Identity Manager nodes are added or removed, perform the procedures described in this section to configure Oracle Identity Manager host and port changes.

4.6.1.3.1 Changing OimFrontEndURL in Oracle Identity Manager Configuration The OimFrontEndURL is the URL used to access the Oracle Identity Manager UI. This can be a load balancer URL or Web server URL depending on the application server is fronted with loan balancer or Web server, or single application server URL. This is

used by Oracle Identity Manager in the notification e-mails as well as the callback URL for SOA calls.

The change may be necessary because of change in Web server hostname or port for Oracle Identity Manager deployment in a clustered environment, or WebSphere managed server hostname or port changes for Oracle Identity Manager deployment in a nonclustered environment.

To change the OimFronEndURL in Oracle Identity Manager configuration:

1. Log in to Enterprise Manager by using the following URL when the Oracle Admin Server and Oracle Identity Manager managed servers, at least one of the servers in case of a clustered deployment, are running:

   http://*ORACLE_ADMIN_SERVER*/em

2. Navigate to **WebSphere Cell**, **OIM server**.
3. Right-click **OIM server**, and navigate to **System MBean Browser**.
4. Under Application Defined MBeans, navigate to **oracle.iam**, **Application:oim**, **XMLConfig**, **Config**, **XMLConfig.DiscoveryConfig**, and then **Discovery**.
5. Enter new value for the OimFrontEndURL attribute, and click **Apply** to save the changes. Example values can be:

   http://myoim.mydomain.com

   https://myoim.mydomain.com

   http://myserver.mydomain.com:7001

> **Note:** SPML clients store Oracle Identity Manager URL for invoking SPML and sending callback response. Therefore, changes are required corresponding to this. In addition, if Oracle Identity Manager is integrated with OAM, OAAM, or Oracle Identity Navigator (OIN), there may be corresponding changes necessary. For more information, refer to OAM, OAAM, and OIN documentation in the Oracle Technology Network (OTN) Web site.

4.6.1.4 SOA Host and Port Changes

To change the SOA host and port:

> **Note:** When additional SOA nodes are added or removed, perform this procedure to change the SOA host and port.

1. Log in to Enterprise Manager by using the following URL when the Oracle Admin Server and Oracle Identity Manager managed servers, at least one of the servers in case of a clustered deployment, are running:

   http://*ORACLE_ADMIN_SERVER*/em

2. Navigate to **Websphere Cell**, **OIM server**.
3. Right-click **OIM server**, and navigate to **System MBean Browser**.
4. Under Application Defined MBeans, navigate to **oracle.iam**, **Application:oim**, **XMLConfig**, **Config**, **XMLConfig.SOAConfig**, **SOAConfig**.

5. Change the values of the Rmiurl and Soapurl attributes, and click **Apply** to save the changes.

   The Rmiurl attribute is used for accessing SOA EJBs deployed on SOA managed servers. This is the application server URL. Example values for this attribute can be:

```
corbaloc:iiop:mysoa1.mydomain.com:2800
corbaloc:iiop:mysoa1.mydomain.com:2800,: mysoa2.mydomain.com:2801
corbaloc:iiop:mysoa1.mydomain.com:2800,: mysoa2.mydomain.com:2801,:
mysoa3.mydomain.com:2802
```

Note: The $WAS_HOME/lib/ext/wf_client_config.xml file must be modified with similar changes.

4.6.1.5 OAM Host and Port Changes

To change the OAM host and port:

1. Log in to Enterprise Manager by using the following URL when the Oracle Admin Server and Oracle Identity Manager managed servers, at least one of the servers for a clustered deployment, are running:

   http://*ORACLE_ADMIN_SERVER*/em

2. Navigate to **Websphere Cell**, and then to **OIM server**.
3. Right-click **OIM server**, and navigate to **System MBean Browser**.
4. Under Application Defined MBeans, navigate to **oracle.iam**, **Application:oim**, **XMLConfig**, **Config**, **XMLConfig.SSOConfig**, and then **SSOConfig**.
5. Change the values of the AccessServerHost and AccessServerPort attributes and other attributes as required, and click **Apply** to save the changes.

4.6.2 Password Changes Related to Oracle Identity Manager

Various passwords are used for Oracle Identity Manger configuration because of the architectural and middleware requirements. This section describes the default passwords and ways to make the changes to the password in Oracle Identity Manger and Oracle WebLogic configuration for any change in the dependent or integrated products.

This section consists of the following topics:

- Changing IBM WebSphere Administrator Password
- Changing Oracle Identity Manager Administrator Password
- Changing Oracle Identity Manager Database Password
- Changing Oracle Identity Manager Passwords in the Credential Store Framework
- Changing OVD Password

4.6.2.1 Changing IBM WebSphere Administrator Password

To change IBM WebSphere administrator password:

1. Log in to Oracle Identity Self Service as System Administrator.
2. Search for WebSphere Administrator User.

3. Click **Reset Password**.
4. Enter new password and confirm new password.
5. Click **Reset Password**.

4.6.2.2 Changing Oracle Identity Manager Administrator Password

During Oracle Identity Manager installation, the installer prompts for the Oracle Identity Manager administrator password. If required, you can change the administrator password after the installation is complete. To do so, you must log in to Oracle Identity Self Service as the System Administrator. In addition, change the password in CSF for entry sysadmin under the map 'oim'.

Note: If OAM or OAAM is integrated with Oracle Identity Manager, then you might have to make corresponding changes in those applications. For more information, refer to OAM and OAAM documentation in the Oracle Documentation web site by using the following URL:

http://docs.oracle.com/

Tip: To ensure optimum performance during password reset in Oracle Identity Manager on WebSphere, update the following JVM args for oim_server by using IBM WebSphere Administrative Console:

```
-Doracle.dms.transtrace.level=NONE
-Doracle.dms.transtrace.uri=NONE
-Doracle.dms.context.dumbasastump=true
-Doracle.dms.sensors=none
-Doracle.dms.context=OFF
```

4.6.2.3 Changing Oracle Identity Manager Database Password

Oracle Identity Manager uses two database schemas for storing Oracle Identity Manager operational and configuration data. It uses Oracle Identity Manager MDS schema for storing configuration-related information and Oracle Identity Manager schema for storing other information. Any change in the schema password requires changes on Oracle Identity Manager configuration.

Changing Oracle Identity Manager database password involves the following:

Note: Before changing the database password, shutdown the managed servers that host Oracle Identity Manager.

- **To change datasource oimJMSStoreDS configuration:**
  1. Navigate to **Resources**, **JDBC**, **Data Sources**, **oimJMSStoreDS**.
  2. Click the JAAS - J2C authentication data link.
  3. Click the *CELL_NAME*/oimJMSStoreDS_alias link.
  4. In the Password field, enter the new Oracle Identity Manager database schema password.
  5. Click **Apply** to save the changes.

- **To change datasource oimOperationsDB configuration:**
  1. Navigate to **Resources**, **JDBC**, **Data Sources**, **oimJMSStoreDS**.
  2. Click the JAAS - J2C authentication data link.
  3. Click the *CELL_NAME*/oimOperationDB_alias link.
  4. In the Password field, enter the new Oracle Identity Manager database schema password.
  5. Click **Apply** to save the changes.
- **To change datasource related to Oracle Identity Manager MDS configuration:**
  1. Navigate to **Resources**, **JDBC**, **Data Sources**, **mds-oim**.
  2. Click the JAAS - J2C authentication data link.
  3. Click the *CELL_NAME*/oimJMSStoreDS_alias link.
  4. In the Password field, enter the new Oracle Identity Manager database schema password.
  5. Click **Apply** to save the changes.

  Note:

  - For Oracle Identity Manager deployments with Oracle Real Application Clusters (Oracle RAC) configuration, you might have to make changes in all the datasources under the respective multi-datasource configurations.
  - You might have to make similar changes for datasources related to SOA or OWSM, if required.

- **To change cell credential store configuration:**
  1. Log in to Enterprise Manager by using the following URL:

     http://*ADMIN_SERVER*/em
  2. Click **WebSphere Cell**, **Security**, and then click **Credentials**.
  3. Expand **oim**, and select **OIMSchemaPassword**, and click **Edit**.
  4. In the Password field, enter the new password, and click **OK**.

After changing the Oracle Identity Manager database password, restart the WebSphere Administrative Server. Start the Oracle Identity manager Managed WebSphere Server as well.

4.6.2.4 Changing Oracle Identity Manager Passwords in the Credential Store Framework

Oracle Identity Manager installer stores several passwords during the install process. Various values are stored in Credential Store Framework (CSF) as key and value. Table 4–12 lists the keys and the corresponding values:

Table 4–12 CSF Keys

Key	Description
DataBaseKey	The password for the key used to encrypt database. The password is the user input value in the installer for the Oracle Identity Manager keystore.
.xldatabasekey	The password for keystore that stores the database encryption key. The password is the user input value in the installer for the Oracle Identity Manager keystore.
xell	The password for key 'xell', which is used for securing communication between Oracle Identity Manager components. Default password generated by Oracle Identity Manager installer is xellerate.
default_keystore.jks	The password for the default_keystore.jks JKS keystore in the *CELL_HOME*/config/fmwconfig/ directory. The password is the user input value in the installer for the Oracle Identity Manager keystore.
SOAAdminPassword	The password is user input value in the installer for SOA Administrator Password field.
OIMSchemaPassword	The password for connecting to Oracle Identity Manager database schema. Password is user input value in the installer for OIM Database Schema Password field.
JMSKey	The password is the user input value in the installer for the Oracle Identity Manager keystore.

To change the values of the CSF keys:

1. Log in to Enterprise Manager.
2. Click **WebSphere Cell**.
3. Navigate to **Security**, and then **Credential**.
4. Expand **oim**. The list of all the key and value pairs for Oracle Identity Manager are displayed. You can edit and change the values.

4.6.2.5 Changing OVD Password

To change the OVD password:

1. Log in to Oracle Identity System Administration.
2. Under Configuration, click **IT Resource**.
3. From the IT Resource Type list, select **Directory Server**.
4. Click **Search**.
5. Edit the Directory Server IT resource. To do so, in the Admin Password field, enter the new OVD password, and click **Update**.

4.6.3 Configuring SSL for Oracle Identity Manager

This section describes the procedure for generating keys, signing and exporting certificates, setting up SSL Configuration for Oracle Identity Manager and for the components with which Oracle Identity Manager interacts, and establish secure communication between them. It includes the following topics:

- Enabling SSL for Oracle Identity Manager and SOA Servers

- Enabling SSL for Oracle Identity Manager DB
- Enabling SSL for LDAP Synchronization
- Securing the Remote Manager with SSL

> **Note:** Before configuring SSL for Oracle Identity Manager, you must generate keys, sign the certificates, and export and import the certificates. For more information about these procedures, refer to IBM WebSphere documentation, or contact IBM support.

4.6.3.1 Enabling SSL for Oracle Identity Manager and SOA Servers

You need to perform the following configurations in Oracle Identity Manager and SOA servers to enable SSL:

- Enabling SSL for Oracle Identity Manager
- Securing the Design Console with SSL
- Configuring SSL for Oracle Identity Manager Utilities
- Configuring SSL for MDS Utilities

4.6.3.1.1 Enabling SSL for Oracle Identity Manager Enabling SSL for Oracle Identity Manager is described in the following sections:

- Enabling SSL for Oracle Identity Manager By Using Default Setting
- Enabling SSL for Oracle Identity Manager By Using Custom Keystore

4.6.3.1.2 Enabling SSL for Oracle Identity Manager By Using Default Setting

By default, SSL ports are enabled for all the WebSphere Application Servers.

To check SSL port:

1. Log in to IBM WebSphere Administrative Console.
2. Navigate to **Servers**, **Server Types**, and click the WebSphere application servers link.
3. Click the oim servers link.
4. Expand Ports link. WC_defaulthost_secure is the SSL port.

4.6.3.1.3 Enabling SSL for Oracle Identity Manager By Using Custom Keystore

Refer to IBM WebSphere documentation for information about changing default keystores. Otherwise, contact IBM support.

After enabling SSL on Oracle Identity Manager and SOA Servers, change OimFrontEndURL and SOA server URL to use SSL port. For details, refer to IBM WebSphere documentation.

4.6.3.1.4 Securing the Design Console with SSL To secure the Design Console with SSL:

1. Open the *WAS_CLIENT_HOME*/properties/sas.client.props file.
2. Ensure the following properties are configured with values of **true**. If they are not set to **true**, update them to have values of **true**.

```
com.ibm.CSI.performTransportAssocSSLTLSRequired
com.ibm.CSI.performTransportAssocSSLTLSSupported
```

Note:

- Setting com.ibm.CSI.performTransportAssocSSLTLSRequired to **true** configures the Design Console to server connection over SSL.
- You can change the default keystore for IBM WebSphere by referring to WebSphere documentation provided by IBM.

4.6.3.1.5 Configuring SSL for Oracle Identity Manager Utilities Oracle Identity Manager client utilities include PurgeCache, GenerateSnapshot, UploadJars, and UploadResources.

To configure SSL for Oracle Identity Manager utilities:

1. Open the *WAS_SERVER_HOME*/profiles/*DMGR_PROFILE*/properties/sas.client.props file.
2. Ensure the values of the following properties are set to true:

   com.ibm.CSI.performTransportAssocSSLTLSRequired

   com.ibm.CSI.performTransportAssocSSLTLSSupported

4.6.3.1.6 Configuring SSL for MDS Utilities The following options must be added to all Oracle Identity Manager MDS Utilities that contains wsadmin script:

-Dcom.ibm.SSL.ConfigURL=file:DMGR_PROFILE\properties\ssl.client.props

4.6.3.2 Enabling SSL for Oracle Identity Manager DB

You need to perform the following configurations to enable SSL for Oracle Identity Manager DB:

- Setting Up DB in Server-Authentication SSL Mode
- Creating KeyStores and Certificates
- Updating Oracle Identity Manager
- Updating WebSphere Server

4.6.3.2.1 Setting Up DB in Server-Authentication SSL Mode To set up DB in Server-Authentication SSL mode:

1. Stop the DB server and the listener.
2. Configuring the listener.ora file as follows:
   a. Navigate to the path:

      $DB_ORACLE_HOME/network/admin directory

      For example:

      /scratch/user1/production-database/product/11.1.0/db_1/network/admin

   b. Edit the listener.ora file to include SSL listening port and Server Wallet Location.

      The following is the sample listener.ora file:

      ```
      # listener.ora Network Configuration File:
      ```

```
/scratch/rbijja/production-database/product/11.1.0/db_1/network/admin/liste
ner.ora
# Generated by Oracle configuration tools.

SSL_CLIENT_AUTHENTICATION = FALSE

WALLET_LOCATION =
  (SOURCE =
    (METHOD = FILE)
    (METHOD_DATA =
      (DIRECTORY =
/scratch/rbijja/production-database/product/11.1.0/db_1/bin/server_keystore
_ssl.p12)
    )
  )

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCPS)(HOST = server1.mycompany.com)(PORT =
2484))
    )
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP)(HOST = server1.mycompany.com)(PORT =
1521))
    )
  )

TRACE_LEVEL_LISTENER = SUPPORT
```

3. Configure the sqlnet.ora file as follows:

   a. Navigate to the path:

      $DB_ORACLE_HOME/network/admin directory

      For example:

      /scratch/user1/production-database/product/11.1.0/db_1/network/admin

   b. Edit sqlnet.ora file to include:

      – TCPS Authentication Services
      – SSL_VERSION
      – Server Wallet Location
      – SSL_CLIENT_AUTHENTICATION type (either true or false)
      – SSL_CIPHER_SUITES that can be allowed in the communication (optional)

      The following is the sample sqlnet.ora file:

```
# sqlnet.ora Network Configuration File:
/scratch/rbijja/production-database/product/11.1.0/db_1/network/admin/sqlne
t.ora
# Generated by Oracle configuration tools.

SQLNET.AUTHENTICATION_SERVICES= (BEQ, TCPS)

SSL_VERSION = 3.0

SSL_CLIENT_AUTHENTICATION = FALSE
```

```
WALLET_LOCATION =
  (SOURCE =
    (METHOD = FILE)
    (METHOD_DATA =
      (DIRECTORY =
/scratch/rbijja/production-database/product/11.1.0/db_1/bin/server_keystore
_ssl.p12)
    )
  )
```

4. Configure the tnsnames.ora file as follows:

   a. Navigate to the path:

      $DB_ORACLE_HOME/network/admin directory

      For example:

      /scratch/user1/production-database/product/11.1.0/db_1/network/admin

   b. Edit the tnsnames.ora file to include SSL listening port in the description list of the service.

      The following is the sample tnsnames.ora file:

```
# tnsnames.ora Network Configuration File:
/scratch/user1/production-database/product/11.1.0/db_1/network/admin/tnsnam
es.ora
# Generated by Oracle configuration tools.

PRODDB =
 (DESCRIPTION_LIST =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCPS)(HOST = server1.mycompany.com)(PORT =
2484))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = proddb)
    )
  )
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = server1.mycompany.com)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = proddb)
    )
  )
 )
```

5. Start/Stop utilities for DB server.
6. Start the DB server.

4.6.3.2.2 Creating KeyStores and Certificates You can create server side and client side KeyStores using the orapki utility. This utility will be shipped as a part of Oracle DB installation.

KeyStores could be of any format such as JKS and PKCS12. The format of keystore changes based on the provider implementation. For example, JKS is the implementation provided by Sun Oracle where as PKCS12 is implemented by OraclePKIProvider.

Only JKS client KeyStore is used in Oracle Identity Manager for DB server. This is because using non-JKS KeyStores format such as PKCS12 requires significant changes on the installer side at the critical release time. However, Oracle Identity Manager already has a KeyStore named default-KeyStore.jks, which is in JKS format.

The following are the KeyStores that you can create using orapki utility:

- Creating a Root CA Wallet
- Creating DB Server Side Wallet
- Creating Client Side Wallet

Note: Wallets and KeyStores are interchangeably used and they both mean the same. These refer to a repository of public/private keys and self-signed/trusted certificates.

Creating a Root CA Wallet

To create a root certification authority (CA) wallet:

1. Navigate to the following path:

   $DB_ORACLE_HOME/bin directory

2. Create a wallet by using the command:

   ```
   ./orapki wallet create -wallet CA_keystore.p12 -pwd welcome1
   ```

3. Add a self signed certificate to the CA wallet by using the command:

   ```
   ./orapki wallet add -wallet CA_keystore.p12 -dn 'CN=root_test,C=US' -keysize
   2048 -self_signed -validity 3650 -pwd welcome1
   ```

4. View the wallet using the command:

   ```
   ./orapki wallet display -wallet CA_keystore.p12 -pwd welcome1
   ```

5. Export the self signed certificate from the CA wallet using the command:

   ```
   ./orapki wallet export -wallet CA_keystore.p12 -dn 'CN=root_test,C=US' -cert
   self_signed_CA.cert -pwd welcome1
   ```

Creating DB Server Side Wallet

To create a DB server side wallet:

1. Create a server wallet using the command:

   ```
   ./orapki wallet create -wallet server_keystore_ssl.p12 -auto_login -pwd
   welcome1
   ```

2. Add a certificate request to the server wallet using the command:

   ```
   ./orapki wallet add -wallet server_keystore_ssl.p12/ -dn
   'CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US' -keysize 2048 -pwd
   welcome1
   ```

3. Export the certificate request to a file, which will be used later for getting it signed using the root CA signature:

   ```
   ./orapki wallet export -wallet server_keystore_ssl.p12/ -dn
   'CN=Customer,OU=Customer,O=Customer,L=City,ST=NY,C=US' -request server_creq.csr
   -pwd welcome1
   ```

4. Get the server wallet's certificate request signed using the CA signature:

```
./orapki cert create -wallet CA_keystore.p12 -request server_creq.csr -cert
server_creq_signed.cert -validity 3650 -pwd welcome1
```

5. View the signed certificate using the command:

```
/orapki cert display -cert server_creq_signed.cert -complete
```

6. Import the trusted certificate in to the server wallet using the command:

```
./orapki wallet add -wallet server_keystore_ssl.p12 -trusted_cert -cert
self_signed_CA.cert -pwd welcome1
```

7. Import this newly created signed certificate (user certificate) to the server wallet using the command:

```
./orapki wallet add -wallet server_keystore_ssl.p12 -user_cert -cert
server_creq_signed.cert -pwd welcome1
```

Creating Client Side Wallet

To create a client side (Oracle Identity Manager server) wallet:

1. Create a client keystore using default-keystore.jks keystore which is populated in the following path:

   DMGR_PROFILE/config/cells/*CELL_NAME*/fmwconfig

   Note: You can also use Oracle PKCS12 wallet as the client keystore.

2. Import the self-signed CA trusted certificate that you have already exported using the server side commands, to the client keystore (default-keystore.jks) by using the command:

```
keytool -import -trustcacerts -alias dbtrusted -noprompt -keystore
default-keystore.jks -file self_signed_CA.cert -storepass xellerate
```

4.6.3.2.3 Updating Oracle Identity Manager You need to perform the following steps in Oracle Identity Manager to enable Oracle Identity Manager and Oracle Identity Manager DB in SSL mode for a secure communication:

1. Import the trusted certificate into the default-keystore.jks keystore of Oracle Identity Manager.
2. Log in to Enterprise Manager.
3. Click **WebSphere Cell**, and select **System MBean Browser**.
4. Under Application Defined MBeans, navigate to oracle.iam, Application:oim, XMLConfig, Config, XMLConfig.DirectDBConfig, and DirectDB.
5. Change the values for attributes "Sslenabled", "Url" and click **Apply**. If SSL mode is enabled for DB, then "Url" should contain TCPS enables and SSL port in it.

   For example:

   url="jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCPS)(HOST=my.domain.com)(PORT=2484))(CONNECT_DATA=(SERVICE_NAME=proddb)))"

6. Restart the Oracle Identity Manager server.

4.6.3.2.4 Updating WebSphere Server After enabling SSL for Oracle Identity Manager DB, you need to change the following Oracle Identity Manager datasources and custom registry to use DB SSL port:

- Configuring Datasource
- Updating Datasource oimJMSStoreDS Configuration
- Updating Datasource oimOperationsDB Configuration
- Updating Datasource Related to Oracle Identity Manager MDS Configuration
- Updating Oracle Identity Manager Custom Registry

Configuring Datasource

To configure the datasource:

1. Log in to IBM WebSphere Administrative Console.
2. Perform the datasource changes.

> **Note:** Before performing changes to the datasource, you must shutdown the managed servers hosting Oracle Identity Manager application.

Updating Datasource oimJMSStoreDS Configuration

To update the datasource oimJMSStoreDS configuration:

1. Log in to IBM WebSphere Administrative Console.
2. Navigate to **Resources**, **JDBC**, **Data Sources**, **oimJMSStoreDS**.
3. Change the value of the URL. The following is an example URL:

```
jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCPS)(HOST=my.domain.com)(POR
T=2484))(CONNECT_DATA=(SERVICE_NAME=proddb)))
```

4. Click **Apply** and make sure to save the change.
5. Go to **Additional Properties**, **Custom Properties**, and add a custom property with the following details:
   - **Name:** connectionProperties
   - **Value:** javax.net.ssl.trustStore=*CELL_HOME*/fmwconfig/default-keystore.jks;javax.net.ssl.trustStoreType=JKS;javax.net.ssl.trustStorePassword=Welcome1;oracle.net.ssl_version=3.0
   - **Type:** java.lang.String

Updating Datasource oimOperationsDB Configuration

To update the Change Datasource oimOperationsDB Configuration:

> **Note:** To add a custom property, see "Updating Datasource oimJMSStoreDS Configuration" on page 4-53.

1. Log in to IBM WebSphere Administrative Console.

2. Navigate to **Resources**, **JDBC**, **Data Sources**, **oimOperationsDB**.
3. Change the value of the URL. The following is an example URL:

```
jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCPS)(HOST=my.domain.com)(POR
T=2484))(CONNECT_DATA=(SERVICE_NAME=proddb)))
```

4. Click **Apply** and make sure to save the change.

Updating Datasource Related to Oracle Identity Manager MDS Configuration

To update datasource related to Oracle Identity Manager MDS configuration:

> **Note:** To add a custom property, see "Updating Datasource oimJMSStoreDS Configuration" on page 4-53.

1. Log in to IBM WebSphere Administrative Console.
2. Navigate to **Resources**, **JDBC**, **Data Sources**, **mds-oim**.
3. Change the value of the URL.
4. Click **Apply** and make sure to save the changes.

> **Note:** You might have to perform similar updates for SOA/OWSM related datasources if required.

Updating Oracle Identity Manager Custom Registry

The existing Oracle Identity Manager custom registry in WebSphere Application Server is configured against non-SSL DB details. In order to user SSL DB details, you must perform the following:

1. Log in to IBM WebSphere Administrative Console.
2. Expand **Security**.
3. Click the **Global security** link.
4. Click **Configure**.
5. Edit DBUrl.
6. Click **Apply** and make sure to save the change.

4.6.3.3 Enabling SSL for LDAP Synchronization

You need to perform the following configurations to enable Oracle Identity Manager to use SSL enabled Oracle Virtual Directory (OVD):

- Enabling OVD-OID with SSL
- Updating Oracle Identity Manager for OVD Host/Port

4.6.3.3.1 Enabling OVD-OID with SSL To enable OVD-OID with SSL:

1. Log in to the OVD EM console.
2. Expand **Identity and Access** and navigate to ovd1, Administration, Listeners.
3. Click **Create** and enter all the required fields.

Note: You must select the Listener Type as LDAP.

4. Click **OK**.
5. Select the newly created LDAP listener and click **Edit**.
6. In the Edit Listener - OIM SSL ENDPOINT page, edit the newly created LDAP listener.
7. Click **OK**. The SSL Configuration page opens.
8. Select the **Enable SSL** checkbox.
9. In the Advanced SSL Settings section, for SSL Authentication, select **No Authentication**.
10. Click **OK**.
11. Stop and start the OVD server for the changes to take effect.

Note: You must not use the restart option.

4.6.3.3.2 Updating Oracle Identity Manager for OVD Host/Port When LDAP synchronization is enabled on Oracle Identity Manager, Oracle Identity Manager connects with directory servers through OVD. It connects using ldap/ldaps protocol.

To change OVD host/port:

1. Log in to Oracle Identity Manager Administrative and User console.
2. Navigate to Advanced and click **Manage IT Resource**.
3. Select IT Resource Type as **Directory Server** and click **Search**.
4. In the IT Resource Directory Server, edit server URL to include SSL protocol and SSL port details.
5. Ensure that Use SSL is set to true and click **Update**.

4.6.3.4 Securing the Remote Manager with SSL

This section describes how to configure SSL for the Oracle Identity Manager Remote Manager on IBM WebSphere. This section includes the following topics:

- Overview
- Configuring One-way SSL Authentication
- Configuring Two-way SSL Authentication

4.6.3.4.1 Overview SSL authentication can be one-way or two-way:

- **One-way**: The Oracle Identity Manager Server (the SSL client application) verifies the identity of the Oracle Identity Manager Remote Manager (the SSL server application).
- **Two-way**: The Oracle Identity Manager Server (the SSL client application) verifies the identity of the Remote Manager (the SSL server application) *and* the Remote Manager verifies the identity of the Oracle Identity Manager Server.

To establish an SSL trust relationship, you import the SSL server's (CA signed) certificate in to the SSL client's keystore. When you installed the Remote Manager, a

keystore and public certificate were created. The Remote Manager's keystore is located in the *OIM_RM_HOME*/config/default-keystore.jks file. The certificate is located in the *OIM_RM_HOME*/config/xlserver.cert file.

> **Note:** The Remote Manager does not support non-SSL communication. By default, one-way SSL authentication is supported. Two-way SSL authentication can be enabled by performing the steps in the appropriate section below.

4.6.3.4.2 Configuring One-way SSL Authentication One-way SSL authentication allows the Oracle Identity Manager Server to verify the identity of the Remote Manager. To configure one-way SSL authentication, the Remote Manager's certificate must be trusted in the Oracle Identity Manager Server's keystore, which is located at:

WAS_HOME/profiles/*Dmgr01*/config/cells/*OIM_CELL_NAME*/fmwconfig/default-keystore.jks

To configure one-way SSL authentication using CA certificates:

1. Copy the Remote Manager's certificate, *OIM_RM_HOME*/config/xlserver.cert, to the Oracle Identity Manager Server system.

   > **Note:** The Oracle Identity Manager Server certificate is also named xlserver.cert. Make sure that you do not unintentionally overwrite the server's certificate.

2. Import the Remote Manager certificate that you copied to the Oracle Identity Manager Server's system in step 1 into the Server's keystore by executing the following shell command:

   ```
   JAVA_HOME/jre/bin/keytool -import -alias TRUSTED_SERVER_CERTIFICATE \
   -file RM_CERT_LOCATION/xlserver.cert \
   -keystore
   WAS_HOME/profiles/Dmgr01/config/cells/OIM_CELL_NAME/fmwconfig/default-keystore.
   jks \
   -trustcacerts -storepass OIM_SERVER_KEYSTORE_PASSWORD
   ```

   Note that *JAVA_HOME* represents the location of the IBM Java Runtime directory for the Oracle Identity Manager Server and *RM_CERT_LOCATION* represents the location where you copied the Remote Manager's certificate step 1.

3. When prompted, enter **Y** (for Yes) to trust the certificate being imported.
4. Restart the application servers, including the Deployment Manager.

4.6.3.4.3 Configuring Two-way SSL Authentication Two-way SSL authentication allows the Oracle Identity Manager Server and the Remote Manager to verify each other's identities. To configure two-way SSL authentication, the Remote Manager's certificate must be trusted in the Oracle Identity Manager Server's keystore and Oracle Identity Manager Server's certificate must be trusted in Remote Manager's keystore.

The Oracle Identity Manager Server's keystore is located at:

WAS_HOME/profiles/*Dmgr01*/config/cells/*OIM_CELL_NAME*/fmwconfig/default-keystore.jks

The Oracle Identity Manager Server's certificate is located in:

WAS_HOME/profiles/*Dmgr01*/config/cells/*OIM_CELL_NAME*/fmwconfig/xlserver.cert

The Remote Manager's keystore is located in:

OIM_RM_HOME/config/default-keystore.jks

The Remote Manager's (CA signed) certificate is located in:

OIM_RM_HOME/config/xlserver.cert

To configure two-way SSL authentication using CA certificates:

1. Copy the Remote Manager's certificate, *OIM_RM_HOME*/config/xlserver.cert, to the Oracle Identity Manager Server system.

   Note: The Oracle Identity Manager Server's certificate is also named xlserver.cert. Be sure you do not unintentionally overwrite the server's certificate.

2. Import the Remote Manager's certificate that you copied to the Oracle Identity Manager Server's system in step 1 into the server's keystore by executing the following shell command:

   ```
   JAVA_HOME/jre/bin/keytool -import -alias TRUSTED_SERVER_CERTIFICATE \
   -file RM_CERT_LOCATION/xlserver.cert \
   -keystore
   WAS_HOME/profiles/Dmgr01/config/cells/OIM_CELL_NAME/fmwconfig/default-keystore.
   jks \
   -trustcacerts -storepass OIM_SERVER_KEYSTORE_PASSWORD
   ```

   Note that *JAVA_HOME* represents the location of the IBM Java Runtime directory for the Oracle Identity Manager Server and *RM_CERT_LOCATION* represents the location where you copied the Remote Manager's certificate step 1.

3. When prompted, enter **Y** (for Yes) to trust the certificate being imported.
4. Restart the application servers, including the Deployment Manager.
5. Copy the Oracle Identity Manager Server's certificate to the Remote Manager system. The Oracle Identity Manager Server's keystore is located at:

   WAS_HOME/profiles/*Dmgr01*/config/cells/*OIM_CELL_NAME*/fmwconfig/xlserver.cert

   Note: The Remote Manager's certificate is also named xlserver.cert. Be sure you do not unintentionally overwrite the server's certificate.

6. Import the Oracle Identity Manager Server's certificate that you copied to the Remote Manager system in step 5 into the Remote Manager's keystore by executing the following shell command:

   ```
   JAVA_HOME/jre/bin/keytool -import -alias TRUSTED_SERVER_CERTIFICATE \
   -file OIM_SERVER_CERT_LOCATION/xlserver.cert \
   -keystore OIM_RM_HOME/config/default-keystore.jks -trustcacerts \
   -storepass RM_KEYSTORE_PASSWORD
   ```

   Note that *JAVA_HOME* represents the location of the IBM Java Runtime directory for the Remote Manager and *OIM_SERVER_CERT_LOCATION* is the location where you copied the Oracle Identity Manager Server's certificate in step 5.

7. When prompted, enter **Y** (for Yes) to trust the certificate being imported.

8. Open the Remote Manager configuration file, *OIM_RM_HOME*/config/xlconfig.xml.
9. Change the value of the `<RMSecurity>.<ClientAuth>` configuration parameter to **true** and save the file.
10. Restart the Remote Manager.

4.7 Using Oracle Identity Manager Utilities on IBM WebSphere

This section describes how to use Oracle Identity Manager utilities on IBM WebSphere:

- Prerequisites for Using Oracle Identity Manager Utilities on IBM WebSphere
- Using Oracle Enterprise Manager to Export Metadata Files from the MDS Database
- Using Oracle Enterprise Manager to Import Metadata Files into the MDS Database
- Using the PurgeCache, UploadJars, DownloadJars, DeleteJars, UploadResourceBundles, and DownLoadResourceBundles Utilities
- Using the Plugin Registration and Unregistration Utility
- Registering a SOA Composite with Oracle Identity Manager on IBM WebSphere
- Using the Form Version Control Utility

4.7.1 Prerequisites for Using Oracle Identity Manager Utilities on IBM WebSphere

Before running Oracle Identity Manager utilities on WebSphere, set the following environment variables:

- *OIM_ORACLE_HOME*: The environment variable to identify the directory on which Oracle Identity Manager is installed.
- *JAVA_HOME*: The location of the IBM Java Runtime directory for the Oracle Identity Manager server.
- *WAS_HOME*: The directory on which WebSphere Application Server is installed.
- *APP_SERVER*: The allowed values are weblogic or websphere. Here, it must be set to `websphere`.
- *MW_HOME*: The directory path for Middleware home.
- *PROFILE_NAME*: The name of the profile.
- *WAS_CELL_HOME*: The location of the cell on which Oracle Identity Manager is deployed.

4.7.2 Using Oracle Enterprise Manager to Export Metadata Files from the MDS Database

To export metadata files from the MDS database using Oracle Enterprise Manager:

1. Log in to Oracle Enterprise Manager using the IBM WebSphere administrator's credentials.
2. Select **System MBean Browser** from the WebSphere Cell list.
3. Expand the following entries: **Application Defined MBeans**, **oracle.mds.lcm**, **Server:*NAME_OF_OIM_SERVER***, **Application: oim**, **MDSAppRuntime**.

4. Click **MDSAppRuntime**.
5. Click the **Operations** tab.
6. Click **exportMetadata**.
7. Enter a value for the **toLocation** property, which identifies the destination directory to which XML files will be exported. For example: /home/user/temp.
8. Click **Edit** for the Docs parameter.
9. Click **Add** and enter the path to the metadata file(s) you want to export. For example: /db/oim-config.xml.
10. Click **Invoke**.

4.7.3 Using Oracle Enterprise Manager to Import Metadata Files into the MDS Database

To import metadata files into the MDS database using Oracle Enterprise Manager:

1. Copy the metadata files you want to import to a temporary location. For example:

```
/home/user/temp/file/ProvisionResourceADUser.xml
/home/user/temp/file/ModifyResourceADUser.xml
```

2. Log in to Oracle Enterprise Manager using the IBM WebSphere administrator's credentials.
3. Select **System MBean Browser** from the WebSphere Cell list.
4. Expand the following entries: **Application Defined MBeans**, **oracle.mds.lcm**, **Server:*NAME_OF_OIM_SERVER***, **Application: oim**, **MDSAppRuntime**.
5. Click **MDSAppRuntime**.
6. Click the **Operations** tab.
7. Click **importMetadata**.
8. Enter a value for the **fromLocation** property, which identifies the source directory from which XML files will be imported. For example: /home/user/temp.
9. Click **Edit** for the Docs parameter.
10. Click **Add** and enter the location of the metadata file(s) to import. For example: /file/*.xml.
11. Click **Invoke**.

4.7.4 Using the PurgeCache, UploadJars, DownloadJars, DeleteJars, UploadResourceBundles, and DownLoadResourceBundles Utilities

This section describes how to use the following Oracle Identity Manager utilities on IBM WebSphere:

- **PurgeCache.sh**: Purges all elements in the cache.
- **UploadJars.sh**: Uploads JAR files into the database.
- **DownloadJars.sh**: Downloads JAR files from the database.
- **DeleteJars.sh**: Deletes JAR files from the database.
- **UploadResourceBundles.sh:** Uploads the connector or custom resource bundle to the database.

- **DownLoadResourceBundles.sh:** Downloads the resource bundle from the database.

To use these Oracle Identity Manager utilities on IBM WebSphere:

1. Set the following environment variables:
   - *OIM_ORACLE_HOME*: Identifies the directory where you installed the Oracle Identity Manager Server.
   - *WAS_HOME*: Identifies the directory where you installed IBM WebSphere Network Deployment Manager.
   - *JAVA_HOME*: Identifies the IBM Java Runtime directory for the Oracle Identity Manager Server.
2. Table 4–13 shows values you must set in the *OIM_ORACLE_HOME*/server/bin/websphere.properties file before using the utilities:

Table 4–13 Values to Set in the websphere.properties File for Utilities

Property	Value
com.ibm.ws.scripting.port	The SOAP port of the IBM WebSphere Server where Oracle Identity Manager is installed. To identify the SOAP port: 1. Log in to the WebSphere Administrative console: 2. Click **Server**, **Server Types**, **Websphere application servers**, ***NAME_OF_OIM_SERVER***. 3. Expand the **Ports** entry in the Communications section. 4. Use the value listed in the **SOAP_CONNECTOR_ADDRESS** entry.
com.ibm.ws.scripting.host	The host name of the system where Oracle Identity Manager is installed.
was_servername	The name of the IBM WebSphere Server where Oracle Identity Manager is installed.
was_nodename	The name of the IBM WebSphere node where Oracle Identity Manager is installed. To identify the node name: 1. Log in to the WebSphere Administrative console: 2. Click **System Administration** > **Nodes**.
application_name	The name of the application, enter `oim`.

3. Open the *OIM_ORACLE_HOME*/server/bin/setEnv.sh file with an editor.
4. Edit the APP_SERVER=@appserver parameter to become: APP_SERVER=websphere.
5. Edit the PROFILE_NAME=@profilename parameter to point to the appropriate profile, for example: PROFILE_NAME=*Dmgr01*.
6. Use an editor to open the sas.client.props file of the profile where Oracle Identity Manager is installed. For example:

   WAS_HOME/profiles/*Dmgr01*/properties/sas.client.props.

7. Edit the following properties to become:

> **Note:** You can identify the bootstrap address for Oracle Identity Manager by performing the following steps:
>
> 1. Log in to the WebSphere Administrative console.
> 2. Click **Server**, **Server Types**, **Websphere application servers**, ***NAME_OF_OIM_SERVER***.
> 3. Expand the **Ports** entry in the Communications section.
> 4. Use the value listed in the **BOOT_STRAP_ADDRESS** entry.

```
com.ibm.CORBA.securityServerHost=OIM_HOSTNAME
com.ibm.CORBA.securityServerPort=OIM_BOOTSTRAP_ADDRESS
com.ibm.CORBA.loginSource=none
```

8. Execute the utility. For example:

```
./PurgeCache.sh CATEGORY_NAME
./UploadJars.sh
./DownloadJars.sh
./DeleteJars.sh
./UploadResourceBundles.sh
./DownLoadResourceBundles.sh
```

When prompted, enter information for the following:

- Oracle Identity Manager administrator user name
- Oracle Identity Manager administrator password
- The service URL. For example:

  corbaloc:iiop:*OIM_HOSTNAME*:*OIM_SERVER_BOOTSTRAP_ADDRESS*
- The context Factory:

  com.ibm.websphere.naming.WsnInitialContextFactory

> **Note:** Some of the utilities, such as Upload, Download, and Delete JARs, and UploadResourceBundles will prompt you for additional information, such as the type and name of the JAR file to execute or location of the custom resource bundle to execute on.

4.7.5 Using the Plugin Registration and Unregistration Utility

You can use the Plugin Registration Utility for registration and unregistration related tasks. The Plugin Registration Utility is located in the *OIM_HOME*/plugin_utility/ directory and uses the following files:

- pluginregistration.xml
- ant.properties

Before Using the Plugin Registration Utility:

1. Set the following environment variables:
   - *JAVA_HOME*: Identifies the IBM Java Runtime directory for the Oracle Identity Manager Server.

- *ANT_HOME*: Identifies the directory where Apache Ant version 1.7 or higher is installed.

  Note: The Plugin Registration Utility requires Apache Ant version 1.7 or higher.

- *WAS_CELL_HOME*: Represents the WebSphere cell.
- *PROFILE_NAME*: Represents the custom profile name.

2. Edit the ant.properties for *WAS_HOME* and *OIM_HOME*. For example:

```
was.home=/test/WAS110912/IBM/WebSphere/AppServer
oim.home=/test/WAS110912/Oracle_IDM1/server
login.config=${oim.home}/config/authws.conf
```

Registering a Plug-in:

To register a plug-in, execute the ant target register command. For example:

```
ant -f  pluginregistration.xml register
```

You will be prompted for the following information:

- Oracle Identity Manager administrator user name and password.
- The service URL, for example:

  `corbaloc:iiop:`*`OIM_HOSTNAME`*`:`*`OIM_SERVER_BOOTSTRAP_ADDRESS`*

- The Context Factory, for example:

  `com.ibm.websphere.naming.WsnInitialContextFactory`

- The full path to and complete name of the plug-in file, for example:

  `/test/pluginsfolder/plugins.zip`

  Note: After providing the information for the plug-in file, you will be prompted for additional information, such as the oimrealm.

Unregistering a Plug-in:

To unregister a plug-in, execute the ant target unregister command. For example:

```
ant -f  pluginregistration.xml unregister
```

You will be prompted for the following information:

- Oracle Identity Manager administrator user name and password.
- The service URL, for example:

  `corbaloc:iiop:`*`OIM_HOSTNAME`*`:`*`OIM_SERVER_BOOTSTRAP_ADDRESS`*

- The Context Factory, for example:

  `com.ibm.websphere.naming.WsnInitialContextFactory`

- The complete class name with package of the plug-in, for example:

```
oracle.iam.scheduler.LongJob
```

Note: After providing the information for the class name with package, you will be prompted for additional information, such as the oimrealm.

4.7.6 Registering a SOA Composite with Oracle Identity Manager on IBM WebSphere

Oracle SOA suite composites must be registered with Oracle Identity Manager before they can be used as an approval process. The procedure to register SOA composites is documented in the "Registering a SOA Composite with Oracle Identity Manager" section of the *Oracle Fusion Middleware Developer's Guide for Oracle Identity Manager*. However, this procedure was developed for Oracle Identity Manager on Oracle WebLogic Server. To use that information for Oracle Identity Manager on IBM WebSphere:

Before Registering

1. Open the *OIM_ORACLE_HOME*/server/bin/setEnv.sh file with an editor.
2. Edit the APP_SERVER=@appserver parameter to become: APP_SERVER=websphere.
3. Edit the MW_HOME=@mwhome parameter to point to the directory where Oracle Fusion Middleware is installed.

Executing the ant Script

Execute *WAS_HOME*/bin/ws_ant.sh. For example:

```
$WAS_HOME/bin/ws_ant.sh -f registerworkflows-mp.xml register
```

4.7.7 Using the Form Version Control Utility

For detailed information about using the Form Version Control (FVC) utility, see "Using the Form Version Control Utility" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Manager*. Running the FVC utility on IBM WebSphere has the following differences:

- Ensure that the prerequisites described in Section 4.7.1, "Prerequisites for Using Oracle Identity Manager Utilities on IBM WebSphere" are met.
- The FVC utility script for WebSphere is:

  fvcutil_websphere.sh for UNIX

4.8 Understanding Identity Certification on IBM WebSphere

This section discusses identity certification tasks that need to be completed by an Oracle Identity Manager Certification Administrator. Prior to creating certifications, refer to Section 4.8.8, "Pre-Requisites for Identity Certifications" and the chapter on Access Catalog administration for more information on how to configure the business metadata of artifacts in the Access Catalog.

This section contains certification information about Oracle Identity Manager on IBM WebSphere Application Server. It contains the following topics:

- Identity Certification Configuration

- Multi-Phased Review and Advanced Delegation
- Understanding How Risk Summaries are Calculated
- Creating Certifications
- Scheduling Certifications
- Understanding Closed-Loop Remediation and Remediation Tracking
- Installing ADFDi Plug-in for Excel-Based Certification Sign-Off
- Pre-Requisites for Identity Certifications

4.8.1 Identity Certification Configuration

Prior to creating a new certification, certain global configuration settings that apply to all certifications created can be applied. These configuration settings can be applied by clicking the checkboxes and then clicking the Save button. Table 4–14 lists the general configuration settings. Table 4–15 lists the global configuration settings.

Note: The general configuration settings does not impact the existing certification when modified.

Table 4–14 General Configuration Settings

Name	Description
Password required on sign-off	This option when checked requires a reviewer of the certification to enter their credentials once they click the sign-off button or complete the review of the certification.
Allow comments on certify operations	This option, when checked, allows a reviewer to enter a comment in a text box after a certify decision has been made on the access details of the user, the reviewer is certifying.
Allow comments on all non-certify operations	This option, when checked, allows a reviewer to enter a comment in a text box after a non-certify decision (that is, Revoke, Unknown or Exception Allowed) has been made on the access details of the user, the reviewer is certifying.
Verify employee access	This option, when checked, causes the user certification page 1 summary view to be displayed. If it is not checked, then page 1 is not displayed to the reviewer and all users are claimed by default.
Prevent self certification	This option, when checked, ensures that the reviewers' access rights are not a part of the certification population. If indeed the reviewer is a part of the certification population, an alternative reviewer can be selected, and that reviewers access rights are automatically routed to the alternate reviewer who gets a new certification.
User and Account Selections	This option controls the presentation of users and accounts in the certification with three possible options that can be selected: 1. Include only active users and active accounts 2. Include any user with active accounts 3. Include all users and all accounts

Table 4–14 (Cont.) General Configuration Settings

Name	Description
Allow advanced delegation	This option, when selected, allows the reviewer of the certification to Delegate the users to an alternate reviewer. If this option is not selected, then advanced delegation option such as Delegate is not available to the reviewer. See Section 4.8.2.2, "Advanced Delegation" for more details.
Allow multi-phased review	This option, when selected, creates the ability to generate a multi-phased certification review campaign. This option only applies to user certifications. See Section 4.8.2, "Multi-Phased Review and Advanced Delegation" for more details.
Allow reassignment	This option, when selected, allows the reviewer of the certification to Re-assign the users to an alternate reviewer. If this option is not selected, then advanced delegation option such as Re-assign is not available to the reviewer.
Allow auto-claim	This option, when selected, automatically claims all users in the first step of the certification. It applies to more than users, Roles in Role certification, Application Instances in application instance certification, Entitlements in entitlement certification, and users in user certification.
Perform closed loop remediation	When this option is checked, once a certification is completed, all access rights to users in the certification that are revoked are directly de-provisioned using Oracle Identity Manager, for all connected and disconnected applications and resources. When this option is unchecked, then no automatic remediation action is taken.

Note: The global configuration settings apply to the existing certification when modified.

Table 4–15 Global Settings

Name	Description
Enable Interactive Excel	This option, when selected, presents the "Download to Editable Excel" link to the reviewer in the Actions menu during certification sign-off. Clicking this button allows the reviewer to download the entire certification into an editable excel file, which can be completed offline.

4.8.2 Multi-Phased Review and Advanced Delegation

Perhaps the most significant enhancement to certification in this release is the introduction of Collaborative Certification or Multi-Phased review. Collaborative certification has two major dimensions:

- Multi-Phased Review
- Advanced Delegation

4.8.2.1 Multi-Phased Review

Multi-Phased review combines the perspectives of both business-oriented and technical reviewers, so that both types of expertise are utilized. There are three possible phases in a multi-phased review:

- Phase One: Business-review is the required, first phase. The business-reviewer, typically the manager of each user, sees all of the (certifiable) access-privileges of that user. The manager confirms first that the user is a valid holder of privileges, for example, an employee within that enterprise, and then that the user's position within the enterprise justifies the user's access-privileges, that is, role-assignments, accounts and entitlement-assignments.
- Phase Two: Technical-review is an optional, second phase. The technical reviewer is the certifier of each privilege and reviews the members of the privilege.
- Final Review is an optional, final phase. If the certification is configured to enable final review, then the primary reviewer from the first phase can see the decisions that reviewers made in the first two phases and can override those decisions if required.

4.8.2.2 Advanced Delegation

Advanced Delegation allows a certifier to retain overall responsibility while delegating decisions to others (for reasons of bandwidth).

The primary reviewer in Phase One or Phase Two can spread the work to other people. This can be done through delegation or reassignment. The primary reviewer can delegate any set of line-items (any item from page 1 of the certification), to any person that the primary reviewer selects. The primary reviewer can also reassign responsibility for any set of line-items to another person. Reassigned items are removed from the current certification and a new certification is generated with those items. Delegated items are still the responsibility of the primary reviewer.

4.8.3 Understanding How Risk Summaries are Calculated

You can directly assign high, medium, and low risk levels to roles, application instances, and entitlements, as well as to certain predefined risk factors. A risk-aggregation job calculates Risk Summaries for the remaining higher-order data objects that are needed to support the identity certification feature. These objects include every user, user-role assignment, account, and entitlement-assignment in the access catalog. During identity certification, certifiers or reviewers use Risk Summaries to separate high-risk certification items from medium-risk and low-risk items.

This section describes how the system processes risk levels to arrive at Risk Summaries. It also describes the risk-aggregation job, which you can run manually or on a scheduled basis.

Note: In Oracle Identity Manager, roles, application instances, and entitlements (entitlement definitions) are metadata objects, whereas users, accounts, and entitlement-assignments are instance-data objects. Think of metadata objects as "structural" objects that represent and describe your information systems within Oracle Identity Manager, whereas instance-data objects are the individual instances of application data that populate the systems described. For example, consider a customer service application (a resource) that has a predefined role that enables users to create trouble tickets (an entitlement). In this example, a single resource object represents the application and a single entitlement object represents a specific privilege within that application. Now consider there might be thousands of user accounts on this resource, some subset of which has the entitlement-assignment that allows the user to create a trouble ticket. In the access catalog, an account object represents each user account, and an entitlement-assignment object represents each instance of the entitlement assignment. This illustrates the one-to-many relationship that exists between metadata objects and instance data objects. A single resource (metadata object) can have multiple accounts (instance-data objects), and a single entitlement (metadata object) can have multiple assignment instances (instance-data objects). The Oracle Identity Manager solution calculates the risk levels for instance-data objects because it would not be feasible for a human to process risk levels for every user, account, and entitlement-assignments in the access catalog on a recurring basis.

Item Risk refers to the risk levels that you and other administrators can assign to specific roles, application instances, and entitlements in the access catalog. There are other ways that Item Risk can be assigned to metadata objects, but direct assignment is the most common method.

Assigning an Item-Risk level to a metadata object in the UI is straightforward. To do so, you search and open the object in the access catalog and select a High, Medium, or Low risk setting from the details pane below. If you do not directly assign an Item-Risk level to a metadata object in the access catalog, the system assigns a default Item-Risk level for you. Roles, application instances, and entitlements can each have a default value. You can configure a default Item-Risk level using the Risk Mapping page.

Generally speaking, you should reserve high Item-Risk levels for metadata objects that confer highly restricted privileges to users. Note that setting a high Item-Risk level on an object will cause its parent object to also have a high Risk-Summary value. Similarly, setting a medium Item-Risk level on an object will cause its parent object to have at least a medium Risk-Summary value. In order for a higher-order object to have a low Risk-Summary value, all of the objects under it in the system hierarchy would have to have low risk settings.

Risk-Factor Mappings are settings that map risk levels to certain predefined conditions within Oracle Identity Manager. Generally speaking, you should reserve high Risk-Factor levels for conditions in which privileges are being extended to users that may be irregular or dangerous. There are two Risk-Factor categories in Oracle Identity Manager, and each category contains multiple settings. Risk-Factor categories are described as following:

Provisioning Scenarios define the risk levels that should be associated with the method or mechanism used to assign a role, account, or entitlement-assignment to a

user using Oracle Identity Manager. For example, you might configure a risk level of High for objects that are provisioned directly by an administrator, and a risk level of Low for objects that are provisioned based on policies that are tied to roles.

Last Certification Action defines risk level based on the status of the last certification for the account, entitlement-assignment, or user-role assignment under consideration. For example, configure a risk level of Low for any item for which the previous certification decision was to approve, and configure a risk level of Medium for any item for which the previous certification decision was to certify conditionally. Finally, you might configure a value of High for any item for which the previous certification decision was Abstain or Revoke.

The Risk-Aggregation job processes Item-Risk levels and Risk-Factor levels, and calculates Risk Summaries for each higher-order object that supports Identity Certification.

In the first phase of risk aggregation, the Risk-Aggregation job evaluates each individual object's Item-Risk level and its three Risk Factor levels and assigns the highest of the four levels to the object's Risk Summary property. A Risk Summary value is calculated for each individual user object, user-role assignment object, account object, and entitlement-assignment object.

Once Risk Summaries are calculated for every object in the access catalog, the next phase of aggregation begins, in which the Risk Summary of each individual object rolls up to the Risk Summary of the parent object that contains it.

Above the entitlement-assignment level, each data object's Risk Summary value contributes to the Risk Summary of the parent-object that contains it. For example, account objects are one hierarchy level up from entitlement-assignment objects, and User objects are one hierarchy level up from there. So, the Risk Summary of every entitlement-assignment object within an account object contributes to the Risk Summary for that account, and, similarly, the Risk Summary for every account object within the user object contributes to the Risk Summary for that user.

User objects are also one level above user-role assignment objects, so the Risk Summary for every user-role assignment object contributes to the Risk Summary for that user. By default, the risk job is not enabled, and therefore, no risks are evaluated. In order to enable it, you need to go to the scheduler menu, find the risk job and enable it. The Job will be executed at the defined time period.

4.8.4 Creating Certifications

All certification definitions are centrally managed in the Oracle Identity Manager Administrative Console.

To create a new certification definition:

1. Log into Oracle Identity System Administration with administrative rights.
2. Go to **Certifications**, **Certification Definitions**, **Create.**
3. Follow the steps outlined below through the wizard.

The following are the steps outlined in the wizard:

- Certification Type
- Base Selection
- Content Selection
- Configuration

- Reviewers
- Incremental
- Summary

4.8.4.1 Certification Type

Enter the name of the certification, what type of Certification it is, and the Description. Four types of certification options, catered towards different reviewers, exist:

1. **User:** Allows business managers to certify their direct reports and their access rights.
2. **Application Instance:** Allows application instance owners to certify users with accounts in the application instances they own.
3. **Entitlement:** Allows entitlement owners to review the users accessing the entitlements they own.
4. **Role:** Allows Role Owners to certify role memberships and/associated role definitions (that is, access policies).

4.8.4.2 Base Selection

These options change based on the type of Certification that is selected. For User certification, users belonging to Organizations or based on a certain search criteria can be selected. Once the user population is finalized, selection constraints can be applied to the users with varying levels or Risk and Risk Summaries on the users as well as the roles, application instances and entitlements they can access.

4.8.4.3 Content Selection

Once the population is selected, content selection options allow/disallow the inclusion of users with all accounts, Roles with varying levels of risk or selected roles only, application instances with varying levels of risk or selected applications only, and entitlements with varying levels or risk or entitlements outside roles and selected entitlements only. These options control the access rights that are to be presented during the review to reviewers.

4.8.4.4 Configuration

These are configuration settings that pertain to each certification definition and are independent from the global configuration settings explained in Table 4–14, " General Configuration Settings". These are general settings that control the layout and certain actions associated to each certification definition and apply to that certification definition only.

4.8.4.5 Reviewers

This step involves the selection of Reviewers. Based on the certification type, the reviewer selection options change. For the User certification, a User manager, Organization Certifier or a selected user (using search) can be used to designate Reviewers to the certification definition. See Section 4.8.2, "Multi-Phased Review and Advanced Delegation" for information about multi-phased reviewers.

4.8.4.6 Incremental

This step controls whether the certification is of type Incremental. If Enabled is checked, then the certification definition takes into account user access rights that have changed since the previous certification cycle for that same certification definition. If

Show Previous Values is Enabled, it will also show the previously certified user access rights, but they will be automatically certified. An Incremental Date Range can also be specified.

4.8.4.7 Summary

This page summarizes the various configuration options selected, as the administrator navigates the wizard, and is for review purposes. Clicking the Back button can change any configuration action. Clicking create will generate the certification definition, as well as schedule a job for running the definition, and execute that job. This will produce a certification based on the definition immediately for review.

4.8.5 Scheduling Certifications

When the certification definition is created, a job is automatically scheduled and set to run immediately. This will produce the initial certification based on the definition. If you would like to run the definition again at a later time to regenerate the certification, or to setup a scheduled run of the definition, the Scheduler page can be used.

To schedule the certification definition to run at a certain time:

1. Navigate to **System Management, Scheduler**, to search for the certification definition.
2. Select the certification definition. The right hand pane displays the various scheduling options that are available. The schedule options include:
    a. Periodic: to run the certification on a periodic basis.
    b. Cron: allows the administrator to set a cron expression to run the certification at a desired time.
    c. Single: to run the certification once.
    d. No pre-defined schedule: which does not run the certification.
    e. Run Now: which runs the certification definition job immediately.
3. Click **Apply** to apply the changes to the certification definition job scheduler.

4.8.6 Understanding Closed-Loop Remediation and Remediation Tracking

Closed-loop remediation is a feature that allows you to directly revoke roles and entitlements from the Oracle Identity Manager provisioning solution as a result of roles and entitlements revoked during the certification process. The remediation status can be tracked in the remediation-tracking module for auditing purposes.

Refer to the Section 4.8.1, "Identity Certification Configuration" to view how Closed Loop Remediation can be turned on for automated remediation.

The status of remediation of all access rights revoked in completed certifications can be tracked in the Certification Dashboard with the tracking ID that, when clicked, will display the status of remediation of the certification in Oracle Identity Manager (request tracking).

For all disconnected application instances, workflows can be configured in Oracle Identity Manager to route the revoked access rights to a ticketing system or an administrator for manual revocation.

4.8.7 Installing ADFDi Plug-in for Excel-Based Certification Sign-Off

In order for identity certifications to be exported to an Excel file for offline sign-off, the ADF desktop integration plug-in must be installed on the client systems, which have the supported versions of Microsoft Excel. Instructions to download install and configure the plug-in are available here:

DI Runtime Edition Setup Instructions:

http://docs.oracle.com/cd/E26098_01/web.1112/e16180/ap_enduseractions.htm#CIHJABEJ

DI Design-time Edition setup Instructions:

http://docs.oracle.com/cd/E26098_01/web.1112/e16180/inst_conf_dev_env.htm#CHDHJIIG

4.8.8 Pre-Requisites for Identity Certifications

In order to create the certifications to have user accounts and entitlements, the following pre-requisite steps have to be performed for each connector installed in Oracle Identity Manager:

1. Log into Oracle Identity Manager Design Console.
2. Under Development Tools, click **Form Designer**.
3. Click **Search**. This will return the Form Designer table with a list of all available forms.
4. Choose the parent forms for each connector installed in the system. A parent form has the UserID fields to store the account name in the target system. For example, UD_ADUSER, UD_EBS_USER.
5. Choose a form and a new tab, Form Designer opens.
6. Click **Create New Version**. Enter a name, for example "v2" in the popup window.
7. Click **Save** and close the popup window.
8. In the Current version drop down, make sure the newly created version "v2" is selected and click on the Properties tab.
9. Locate the field that uniquely identifies the account in the target system, that is, UserID, UserName, AccountName are typical fields in the predefined connectors.
10. Click **Add Property** and add the 'AccountName = true' property setting.
11. Locate the ITResource field (most connectors will identify this with text ITResourceLookupField as a property) for the target system, click **Add Property**, and add the "ITResource = true" property setting.
12. Save the parent form and click **Make Version Active**.
13. Repeat for each resource.

4.9 Deinstalling Oracle Identity Manager on IBM WebSphere

The procedure to deinstall Oracle Identity Manager on WebSphere is same as deinstalling Oracle Identity Manager on Oracle WebLogic Application Server, which is described in section "Deinstalling Oracle Identity and Access Management" of the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.

5

Managing Access Manager on IBM WebSphere

This chapter contains information regarding the differences when managing Access Manager on IBM WebSphere (as opposed to WebLogic Server).

This chapter contains the following sections:

5.1 Differences Between Access Manager When Deployed on WebLogic Server and IBM WebSphere

The following Access Manager features are only supported when Access Manager is deployed on WebLogic Server.

- OSSO Agent - mod_osso is not supported in non-OHS http servers; for OAM on WebSphere, the partners are expected to be IHS. Migration of OSSO10g customers to OAM-WebSphere is also not supported. Also, OSSO10g was supported only on OC4J; from this, we certify migration only to OAM-WLS.
- IAMSuite Agent - The IAMSuite Agent (DOMAIN Agent) has been marked for deprecation on WebLogic Server. As this is the first release of OAM-WAS, there is no support for the IAMSuite Agent on WAS. Customers should use an IHS WebGate to front end Identity Management components. See the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management* for details.
- OAM NAP Simple Mode - The algorithms required by NAP simple mode are not supported by the IBM JDK.

- NAP Autologin for OAM - OIM Integration - Autologin is supported between OAM-OIM however this must be based out of the TAP front channel integration. NAP based autologin is NOT supported on WebSphere given this is not supported on WebLogic Server.
- IDM Domain Agent - The IDM Domain Agent does not need to be removed or disabled when deploying Oracle Access Management on IBM WebSphere.

Note: For the OAM-OAAM integration, the OAAMAdvancedScheme using TAP is preferred.

5.2 Using Oracle Access Manager WLST Commands on IBM WebSphere

You can run Oracle Access Manager commands from the IBM WebSphere wsadmin command line interface. For details, see Using the Oracle Fusion Middleware wsadmin Commands.

Access Manager commands are documented in the *Web Logic Scripting Tool Command Reference*. Oracle Access Management commands are functionally identical on WebLogic and WebSphere. When running Access Manager wsadmin commands, however, you must prefix the command name with the Access Manager `Oam` category name. For example:

```
Oam.displayOAMMetrics()
```

To connect to any WebSphere server in online mode use the following command:

`./wsadmin.sh -connType SOAP -host` <*HOST_NAME*> `-port` <*SOAP_PORT*> `-user` <*ADMIN_USER*> `-password` <*ADMIN_PASSWORD*>

where:

HOST_NAME, SOAP_PORT, ADMIN_USER, and ADMIN_PASSWORD are the correct values for your environment.

Note that there is not a WebSphere method to get `domainRuntime()`. For this reason you have to pass `domainHome` as an argument when applicable. This is true for both online and offline commands. The `domainHome` for WebSphere Application Server is as follows:

<*WAS_HOME*>`/profiles/`<*PROFILE_NAME*>`/config/cells/`<*CELL_NAME*>

5.3 Increasing the Number of Threads Available to Access Manager

If the number of concurrent requests hitting the Oracle Access Management server is high and all worker threads are currently in a processing state, a `ThreadPoolQueueIsFullException` error may result. The server can accept more requests if you tune the work manager configuration based on the concurrent requests expected.

To do this, follow these steps.

1. Login to the WebSphere Application Server Administrative Console.
2. Choose **Resources > Asynchronous Beans > Work Managers > OAMServerWorkManager**.
3. Increase the maximum number of threads for `OAMServerWorkManager` from 50 to a large value (for example, 500).

5.4 Configuring x509 Authentication

To configure x509 and protect a resource, complete the following steps.

- "Create the Server Certificate and Trust Store"
- "Configure the Stores"
- "Create a User Certificate"
- "Adding the Root CA Certificate to the Store"
- "Protecting a Resource Using the X509 Authentication Scheme"
- "To Access an X509 Protected Resource"

5.4.1 Create the Server Certificate and Trust Store

1. Use a certificate authority to create a signed certificate for the Oracle Access Management server machine in the WebSphere cell. Include the machine name in the certificate details and save the certificate in the `.p12` format.
2. Create the server store as shown in the following sample command:

```
keytool -importkeystore -deststorepass samplepassword -destkeystore server.jks
-srckeystore my-server.p12 - srcstoretype PKCS12 -srcstorepass samplepassword
-alias "Server"
```

> **Note:** Use the keytool utility to create and manage keys and certificates in the JKS keystore format. For command and option notes, refer to the JDK documentation provided with WebSphere Application Server.

3. Change directories to `JAVA_HOME/bin`.
4. Create the trust store by running the following command:

```
keytool -import -alias trust -file scratch/simpleCA/ca.pem -keystore trust.jks
```

5.4.2 Configure the Stores

Configure the WebSphere server instance that needs to be both SSL and client certificate enabled.

Add the Keystore Server Store

1. In the WebSphere administrative console, choose **SSL certificate and key management > Key stores and certificates > New**.

   Complete the form and provide the path to the server store.
2. Click **Personal Certificates** (choose **SSL certificate and key management > Key stores and certificates >** ***server store name*** **> Personal certificates**) and verify that the server certificate is shown. If it is not shown, import the server certificate from the server store.
3. Open the **Signer Certificate** page (choose **SSL certificate and key management > Key stores and certificates >** ***server store name*** **> Signer certificates**) and verify that the Root CA certificate is shown. If it is not shown, add the Root CA certificate.

Add the Trust Store

1. In the WebSphere administrative console, choose **SSL certificate and key management > Keystores and certificates > New**.

   Complete the form and provide the path to the trust store.

2. Open the **Signer Certificate** page (choose **SSL certificate and key management > Key stores and certificates > *trust store name* > Signer certificates**) and verify that the Root CA certificate is shown. If it is not shown, add the Root CA certificate.

3. Choose **SSL certificate and key management > Key stores and certificates > CellDefaultTrustStore > Signer certificates** and add the Root CA certificate.

Create a New SSL Configuration and Adjust Settings

1. Choose **SSL certificate and key management > SSL configurations**.

   Click **New**.

2. Complete the form to create a new configuration for OAM Server. Choose the Trust store name and Keystore name that you added previously.

3. Click **Quality of protection (QoP) settings** (choose **SSL certificate and key management > SSL configuration > *oam server ssl config name* > Quality of protection (QoP) settings**) and select **Required** from the **Client authentication** menu.

   Click **Apply**.

4. Choose **SSL certificate and key management > Manage endpoint security configurations**.

   Expand the Inbound endpoint (**Inbound > DefaultCell(CellDefaultSSLSettings) > nodes > DefaultNode(NodeDefaultSSLSettings) > servers**) and click to edit the oam server instance.

   In the **Specific SSL configuration for this endpoint** section, select **Override inherited values**, and choose the OAM server SSL config name from the **SSL configuration** menu.

   Click **Apply** and repeat for the Outbound endpoint.

5. Synchronize the node.

6. Restart the nodeagent.

7. Restart the Oracle Access Management server.

5.4.3 Create a User Certificate

Use a certificate authority to create a signed user certificate. Include the user name for whom the certificate is requested in the certificate details and save the certificate in the `.p12` format.

Install the certificate in your browser.

5.4.4 Adding the Root CA Certificate to the Store

To enable SSL on WebSphere, add the certificate utility root certificate to the `.oamkeystore` and `amtruststore` file located here:

```
<WAS_HOME>/profiles/Dmgr01/config/cells/DefaultCell01/fmwconfig
```

To Retrieve the .oamkeystore / amtruststore Password

1. From the command line, navigate to the following directory:

   ```
   $WAS_HOME/oracle_common/common/bin/
   ```

2. Run the `wsadmin.sh` command:

   ```
   wsadmin.sh -conntype SOAP -port <SSL_SOAP_PORT> -user <username>
   ```

3. From the wsadmin shell, run the following command:

   ```
   Opss.listCred(map="OAM_STORE", key="jks")
   ```

   The password is displayed.

To add the CA Certificate to the .oamkeystore / amtruststore File

Add the CA certificate to the `.oamkeystore` / `amtrustore` file as shown in the following sample keytool commands.

> **Note:** For keytool command and option notes, refer to the JDK documentation provided with WebSphere Application Server.

```
./keytool -importcert -alias ROOT_CA -file /scratch/CA/ca.pem -keystore
<WAS_HOME>/Dmgr01/config/cells/DefaultCell01/fmwconfig/.oamkeystore -storepass
oru8nd3hhd4t4nrmh6unhv825b -storetype jceks

./keytool -importcert -alias ROOT_CA -file /scratch/CA/ca.pem -keystore
<WAS_HOME>/Dmgr01/config/cells/DefaultCell01/fmwconfig/amtruststore -storepass
oru8nd3hhd4t4nrmh6unhv825b -storetype jks
```

> **Note:** The `-storepass` value in the sample keytool commands is retrieved using the steps in the "To Retrieve the .oamkeystore / amtruststore Password" section.

5.4.5 Protecting a Resource Using the X509 Authentication Scheme

1. In the Oracle Access Management Administration Console, choose **Policy Configuration > Shared Components > Authentication Schemes > X509Scheme**.
2. In the Challenge URL box, change the value to the SSL port of the managed server.

   For example:

   `https://`*<managed server host name>*`:`*<managed server SSL port number>*`/oam/CredCollectServlet/X509`
3. To protect a resource using the X509 authentication scheme, choose **Policy Configuration > Application Domains >** *Domain Name* > **Authentication Policies > Protected Resource Policy**.

   Choose **X509 Scheme** from the **Authentication Scheme** menu.

5.4.6 To Access an X509 Protected Resource

1. Open the resource using the browser that has the installed user certificate.

   The browser will prompt you to select the certificate to use to connect.
2. Choose the valid user certificate and click OK.

The resource is displayed.

5.5 Deploying the RSA SecurID Authentication Plug-in

If deploying the RSA SecurID Authentication Plug-in (`authn_securid`) on WebSphere, note the following requirements:

- Create the following directory structure in WebSphere and place the agent configuration file (`sdconf.rec`) in the `oam` directory:

  Create the following path relative to the Fusion Middleware `config` directory:

  ```
  fmwconfig/../../../servers/oam_server1/oam
  ```

  Or, create the following path relative to the profile home directory:

  ```
  <PROFILE_HOME>/servers/oam_server1/oam
  ```

  Note: See "Configuring Access Manager for RSA SecurID Authentication" in the *Administrator's Guide for Oracle Access Management* for more information.

- Download the following third-party JAR files:
  - `authapi.jar`
  - `cryptoj.jar`

  Add these JAR files to the following directory:

  ```
  fmwconfig/oam/plugin-lib
  ```

  This is a required step to run the custom RSA plug-in.

5.6 Configuring Access Manager Running on WebSphere for Windows Native Authentication

To configure Access Manager running on WebSphere for Windows Native Authentication (WNA), format the path to the keytab file in the `oam-config.xml` file as follows:

`file://`*<path to keytab file>*

In a UNIX environment, specify a path similar to the following:

```
<Setting Name="keytabfile"
    Type="xsd:string">file:///refresh/home/oam.keytab
</Setting>
```

For more information, see "Configuring Access Manager for Windows Native Authentication" in the *Administrator's Guide for Oracle Access Manager*.

5.7 Moving Access Manager From a Test to Production Environment on IBM WebSphere

This section describes how to copy Access Manager from a test environment to a production environment. These same steps can also be used to copy a production environment to a test environment.

This section covers the following topics:

5.7.1 Introduction to Moving Access Manager on IBM WebSphere

You can move Access Manager from a source environment to a target environment.

Moving Access Manager minimizes the amount of work that would otherwise be required to reapply all the customization and configuration changes made in one environment to another. You can install, configure, customize, and validate Access Manager in a source environment. Once the system is stable and performs as desired, you can create the target environment by moving a copy of the components and their configurations from the source environment, instead of redoing all the changes that were incorporated into the source environment.

5.7.2 Limitations and Restrictions

The steps to move Access Manager from one IBM WebSphere environment to another has the following limits and restrictions:

- Use these steps if your policy store resides on a database. These steps do not describe how to move an LDAP-based policy store from test to production.
- Migrating the User Identity Store from one system to another is not supported.
- To move Mobile and Social from a test to a production environment, first complete the steps in this section, then complete the "Moving Mobile and Social From a Test to a Production Environment" steps, which are located in the "Managing Oracle Access Management Mobile and Social on IBM WebSphere" chapter.
- The Test to Production (T2P) commands that Oracle provides for IBM WebSphere are not the same tools that are provided for WebLogic environments. The commands for IBM WebSphere do not support the Full Replication or Golden Template options for moving files between environments.

5.7.3 Overview of Procedures for Moving from a Source to a Target Environment

Moving Access Manager from a test environment to a production environment is a multi-step process:

1. Run the `exportConfig` command on the test (source) system.

   This command does the following:

   - Exports keystores, for example, the oamkeystore and the coherence keystore.
   - Exports OAM Config

- Exports policies, including the password policy
- Exports partners
- Creates an archive file named `wast2p.zip` for the exported data and saves it in a specified directory

Transfer the archive to the production (target) system after the command completes.

2. Export the OPSS Policy domain from the test database and export the OPSS Encryption Key.
3. Run the `importConfig` command on the production (target) system.

   This command does the following:

   - Expands the `wast2p.zip` archive in the production environment.
   - Imports the keystores
   - Imports OAM Config by updating the installation of Access Manager in the production environment
4. Run the `updateConfig` command on the production (target) system.

   This command does the following:

   - Imports policies, including the password policy
   - Imports partners
   - Updates the MultiDataCenter Cluster ID
5. Stop the OracleAdmin Server, the Managed Server, and the Node Agent and import the OPSS Encryption Key.
6. Import the OPSS policy data to the production (target) database.
7. Stop and start the Deployment Manager. Start the Sync Node, the Node agent, and the admin and managed servers.

5.7.4 Prerequisites

Before continuing with the steps in this chapter, verify that you have completed the following requirements.

Install Oracle Access Management

Install Oracle Access Management in the production (target) environment. Ensure that the Oracle Access Management version and build numbers in the test and production environments match, as well as all configuration files.

Ensure the Admin and Managed Servers are Running

The admin server and managed server should be up and running.

5.7.5 Moving Access Manager From Test to Production

Complete the steps in the following order.

1. Run the `exportConfig` command on the test (source) system:

   `Oam.exportConfig('<`*TargetDir*`>')`

   where:

TargetDir is the path to the directory where the archive should be saved.

For example:

```
Oam.exportConfig('scratch/bkup')
```

2. Move the archive created in the previous step to the production environment.
3. Export the OPSS Policy domain from the test database.

   Use the export procedure that is appropriate for your database. For example:

   ```
   ./expdp system/welcome1@orcl DIRECTORY=DATA_PUMP_DIR SCHEMAS=<OPSS_schema
   name>DUMPFILE=export.dmp PARALLEL=2 LOGFILE=export.log
   ```

4. Export the OPSS encryption key using the following wsadmin command from `oracle_common/common/bin`:

   ```
   Opss.exportEncryptionKey('<jpsConfigFilePath>','<keyFilePath>',
   '<keyFilePassword>')
   ```

   where:

   `<jpsConfigFilePath>` is the absolute location of the file in the test environment

   `<keyFilePath>` is the directory in the test environment where you want the `ewallet.p12` file created. Note that the content of this file is encrypted and secured by the `keyFilePassword`.

   `<keyFilePassword>` is the password used to secure the `fileewallet.p12` file. Note that this same password must be used when importing the file.

5. Run the `importConfig` command in the production (target) environment. The Admin and Managed server should be running. Use this command:

   `Oam.importConfig('<`*ZipLocation*`>')`

   where:

   ZipLocation is the path to where the archive file copied in the previous step is located.

   For example:

   ```
   Oam.importConfig('scratch/bkup/wast2p.zip')
   ```

   Note: Due to synchronization issues you may need to restart Deployment Manager in the production environment before performing the import. If the `importConfig` command results in an error or does not produce the expected result, restart the Deployment Manager and run the command again. After import the keystores and OAM configuration should be updated.

6. Run the `updateConfig` command in the production (target) environment. The Admin and Managed server should be running. Use this command:

   `Oam.updateConfig('<`*ZipLocation*`>')`

   where:

   ZipLocation is the path to the directory where the archive copied in the previous step is located

Note: Due to synchronization issues you may need to restart the Admin and Managed Servers in the production environment before performing the update. If the `updateConfig` command results in an error or does not produce the expected result, restart the Admin and Managed Servers and run the command again.

7. Stop the Oracle Admin Server and the Managed Server and stop the Node Agent.
8. Import the OPSS Encryption Key using the following wsadmin command from `oracle_common/common/bin`:

   ```
   Opss.importEncryptionKey('<PROD_jpsConfigFilePath>','<PROD_keyFilePath>',
   '<keyFilePassword>')
   ```

   where:

   `<PROD_jpsConfigFilePath>` is the absolute location of the file in the production environment

   `<PROD_keyFilePath>` is the directory in the production environment where the file `ewallet.p12` is created. Note that the content of this file is encrypted and secured by the `keyFilePassword`.

   `<keyFilePassword>` is the password used to secure the `fileewallet.p12` file.
9. Import the OPSS policy data to the production database.

   Use the import procedure that is appropriate for your database. For example:

   ```
   /impdp system/welcome1@orcl DIRECTORY=DATA_PUMP_DIR DUMPFILE=export.dmp
   PARALLEL=2 LOGFILE=import.log remap_schema=<Test schema name>_OPSS:<Prod schema
   name>_OPSS  remap_tablespace=<Test schema name>_IAS_OPSS:<Prod schema
   name>_IAS_OPSS TABLE_EXISTS_ACTION=REPLACE
   ```

10. Do the following:
    a. Stop and start the Deployment Manager.
    b. Synchronize the node.
    c. Start the node agent.
    d. Start the Admin and Managed Servers

5.8 Installing Access Manager in a High-Availability WebSphere Environment

This section contains information about installing Access Manager in a high-availability WebSphere environment.

The following topics are covered:

- Overview of the Installation Process
- Installation Roadmap
- Configure the Oracle IAM Components on IBM WebSphere on Node 1
- Configure the Oracle IAM Components on IBM WebSphere on Node 2
- Start the Servers

- Next Steps

5.8.1 Overview of the Installation Process

The following steps describe how to install Access Manager on three OAM servers across two nodes. Repeat the steps as needed to install Access Manager on more than three servers.

These steps create the following topology:

Node 1 Machine

- Deployment Manager (Profile: Dmgr01)
- WebSphere server – `OracleAdminServer` (Profile: Custom01)
- WebSphere server - `oam_server1a` (Profile: Custom01)
- WebSphere server - `oam_server1b` (Profile: Custom01)

Node 2 Machine

- WebSphere server - `oam_server2` (Profile: Custom02)

5.8.2 Installation Roadmap

Installing Access Manager in a high-availability IBM WebSphere environment includes the following high-level tasks.

Table 5–1 Installation Flow for Access Manager in a High-Availability IBM WebSphere Environment

No.	Task	Information
1	Review the System Requirements and Certification Information, then install a database that is compatible with Oracle Fusion Middleware.	Refer to Chapter 2, "Installing and Configuring Oracle Identity and Access Management on IBM WebSphere" and complete the following tasks: ■ Task 1: Review the System Requirements and Certification Information ■ Task 2: Obtain the Necessary Software Media or Downloads ■ Task 3: Identify a Database and Install the Required Database Schemas Run the Repository Creation Utility (RCU) to create the OAM and OPSS schemas.
2	Install the IBM WebSphere Software on both the node 1 machine and the node 2 machine.	Refer to Chapter 2, "Installing and Configuring Oracle Identity and Access Management on IBM WebSphere" and complete the following task: ■ Task 4: Install the IBM WebSphere Software If you will also install Oracle Identity Manager, complete the following task: ■ Task 5: Install Oracle SOA Suite

Table 5–1 (Cont.) Installation Flow for Access Manager in a High-Availability IBM WebSphere Environment

No.	Task	Information
3	Install the Oracle Identity and Access Management Suite on both machines in Node 1	Refer to Chapter 2, "Installing and Configuring Oracle Identity and Access Management on IBM WebSphere" and complete the following task: ▪ Task 6: Install Oracle Identity and Access Management Suite If you will also install Oracle Privileged Account Manager, complete the following task: ▪ Task 7: Optional: Enabling TDE in Oracle Privileged Account Manager Data Store
4	Configure the Oracle Identity and Access Management Components in IBM WebSphere on Node 1	Refer to Section 5.8.3, "Configure the Oracle IAM Components on IBM WebSphere on Node 1."
5	Configure the Oracle Identity and Access Management Components in IBM WebSphere on Node 2	Refer to Section 5.8.4, "Configure the Oracle IAM Components on IBM WebSphere on Node 2."
7	Start the servers	Refer to Section 5.8.5, "Start the Servers."
8	Next steps	Refer to Section 5.8.6, "Next Steps."

5.8.3 Configure the Oracle IAM Components on IBM WebSphere on Node 1

To configure Access Manager in a new IBM WebSphere cell, complete the following steps:

1. Start the Oracle Fusion Middleware Configuration Wizard by running the following command from the Oracle Identity and Access Management home:

   ORACLE_HOME`/common/bin/was_config.sh`

2. If necessary, select **Oracle Access Management - 11.1.2.1.0**.
3. On the Select Optional Configuration screen, select the **Application Servers, Clusters and End Points** option, and click Next.
4. On the Configure Application Servers screen, type the following names and click Next:
   - In the **Name** field, type a name for the Oracle Access Management server, for example: *oam_server1a*.
   - In the **Node name** column, select from the list the node agent name for `oam_server1a`, for example: *WebsphereNode1*.
5. On the Configure Clusters screen, do the following:
   a. Click **Add**.
   b. Type a name for the cluster in the **Cluster Name** field, for example: *OAMServerCluster*.
   c. Select the appropriate OAM Server (`oam_server1a`) from the **First cluster member** list.

6. On the Configure Additional Cluster Members screen, complete the following steps only if the second OAM server is also configured on the same node. If the second OAM Server *is not* on the same node, click Next and proceed.
   a. Click **Add**.
   b. In the **Name** field, type a name for the Oracle Access Management server to be added to OAMServerCluster—for example, *oam_server1b*.
   c. Click Next and proceed through the remaining screens.
7. Use the following command to stop the node:

   ```
   $WAS_HOME/profiles/Custom01/bin/stopNode.sh
   ```

8. Validate that the `oam-server-info.properties` file is correct:
   a. Go to the following location and open the file in a text editor:

      ```
      $WAS_HOME/profiles/Dmgr01/config/cells/Cell01/fmwconfig
      ```

   b. Verify that the file contains the correct settings for your environment. The settings should be similar to the following:

      ```
      #-- start of file contents --
      oracle.oam.adminserver=OracleAdminServer
      oracle.oam.runtimeserver=oam_server1a,oam_server1b
      OracleAdminServer= https://oamadminhost.us.example.com:9003
      oam_server1a=https://oamserver1.us.example.com:14101
      oam_server1b=https://oamserver1.us.example.com:15101
      #-- end of file --
      ```

      Note: You can add all of the OAM managed servers in the cluster to this properties file, or you can add them later using the OAM console. In the properties file, the `oracle.oam.runtimeserver` property should list the names of the servers in the cluster.

9. Run the `configureSecurityStoreWas.py` command:

   ```
   $IDM_HOME/common/bin/wsadmin.sh -lang jython -profileName Dmgr01 -f
   $IDM_HOME/common/tools/configureSecurityStoreWas.py -d
   $WAS_HOME/profiles/Dmgr01/config/cells/<cell> -t DB_ORACLE -j cn=jpsroot -m
   create --passcode <OPSS-db-schema-password> --config IAM
   ```

10. Configure the Oracle Internet Directory (OID) store for Oracle Platform Security Services (OPSS):
    a. Run the following command to start the Deployment Manager:

       ```
       $WAS_HOME/profiles/Dmgr01/bin/startManager.sh
       ```

    b. Run the OPSS `wsadmin` command to launch the wsadmin shell.

       `$WAS_HOME/oracle_common/common/bin/wsadmin.sh -conntype SOAP -port <port_number> -user <username> -password <passwd>`

       Note: Use the credentials that you used to set up the WebSphere cell (that is, the wasadmin user name and password). The port details are available in the following file:

       `$WAS_HOME/profiles/Dmgr01/logs/AboutThisProfile.txt`

c. Run the `configureIdentityStore` wsadmin command in the wsadmin shell:

`Opss.configureIdentityStore(propsFileLoc="<location of the oid.properties properties file>")`

A sample `oid.properties` file is provided here:

```
user.search.bases=cn=Users,dc=us,dc=example,dc=com
group.search.bases=cn=Groups,dc=us,dc=example,dc=com
subscriber.name=dc=us,dc=example,dc=com
ldap.host=host06.us.example.com
ldap.port=3333
# admin.id must be the full DN of the user in the LDAP
admin.id=cn=orcladmin,cn=Users,dc=us,dc=example,dc=com
admin.pass=welcome123
user.filter=(&(uid=%v)(objectclass=person))
group.filter=(&(cn=%v)(objectclass=groupofuniquenames))
user.id.map=*:uid
group.id.map=*:cn
group.member.id.map=groupofuniquenames:uniquemember
ssl=false
# primary.admin.id indicates a user who has admin permissions in the LDAP.
# It must be the name of the user, for example, for user "cn=tom", the
# primary.admin.id is "tom"
primary.admin.id=orcladmin
# optional, default to "OID"
idstore.type=OID
# Optional properties for JPS LDAP identity store can also be configured in
# the file.
username.attr=cn
user.object.classes=person
```

In the properties file, ensure that the `primary.admin.id` is set to a user who is part of the *Administrators* group in the specified Oracle Internet Directory instance.

d. Stop the Deployment Manager:

`$WAS_HOME/profiles/Dmgr01/bin/stopManager.sh`

Provide the credentials for the WebSphere cell.

11. Start the Deployment Manager, nodes and servers.

Use the `primary.admin.id` user credentials provided in the previous step.

`$WAS_HOME/profiles/Dmgr01/bin/startManager.sh`

`$WAS_HOME/profiles/Custom01/bin/syncNode.sh <MachineName> <SOAP port>`

`$WAS_HOME/profiles/Custom01/bin/startNode.sh`

`$WAS_HOME/profiles/Custom01/bin/startServer.sh OracleAdminServer`

`$WAS_HOME/profiles/Custom01/bin/startServer.sh oam_server1a`

`$WAS_HOME/profiles/Custom01/bin/startServer.sh oam_server1b`

12. Validate that the configuration is correct from a Web browser by opening `http://oamadminhost:port/oamconsole` using the `primary.admin.id` user credentials provided earlier.

13. Do not stop the Deployment Manager, but stop all of the other processes on the Node1 machine.

```
$WAS_HOME/profiles/Custom01/bin/stopServer.sh oam_server1b

$WAS_HOME/profiles/Custom01/bin/stopServer.sh oam_server1a

$WAS_HOME/profiles/Custom01/bin/stopServer.sh
OracleAdminServer

$WAS_HOME/profiles/Custom01/bin/stopServer.sh

$WAS_HOME/profiles/Custom01/bin/stopNode.sh
```

5.8.4 Configure the Oracle IAM Components on IBM WebSphere on Node 2

1. Run the Oracle Fusion Middleware Configuration Wizard to federate the machine and configure its cell:

   ```
   $IDM_HOME/common/bin/was_config.sh
   ```

2. On the Select Configuration Option screen, select the **Federate Machine and Configure Cell** option.
3. Specify the profile and fnode name information. Enter information about the profile and node names that you want to create for the WebSphere Node 2 Machine.
4. On the Specify Deployment Manager Information screen, enter information about the existing Deployment Manager System.
5. On the Select Optional Configuration screen, select the **Application Servers, Clusters and End Points** option and click Next.
6. Proceed through the wizard and accept the default options until you reach the **Configure Additional Cluster Members** screen.
7. Complete the **Configure Additional Cluster Members** screen as follows:
   a. Click Add.
   b. In the **Name** box, type a name for the second server in the OAMServerCluster—for example, *oam_server2*.
   c. In the **Node name** list, choose the node agent for oam_server2—for example, *WebSphereNode2*.
   d. In the **Cluster name** list, choose the OAMServerCluster.
8. *Optional*. On the Port Configuration screen, edit the port settings for oam_server2 on the Node2 machine so that they match the HTTP and HTTPS (SSL) port settings for oam_server1a on the Node1 machine. (This is an optional step for consistency.) The port specified should match the port setting specified in the `oam-server-info.properties` file.
9. Stop the node on the WebSphere Node 2 machine:

   ```
   $WAS_HOME/profiles/Custom02/bin/stopNode.sh
   ```

10. Add the node 2 OAM server instance to OAM Configuration using one of the following methods.
    - From the OAM console, choose System Configuration, create a new OAM Server instance, and enter the oam_server2 details.

- From the wsadmin command line, use the `createOAMServer` command and enter the oam_server2 details.
- Open the following properties file in a text editor:

  `$WAS_HOME/profiles/Dmgr01/config/cells/Cell02/fmwconfig`

  Add the OAM managed server(s) in the cluster to this properties file. The `oracle.oam.runtimeserver` property should list the names of the server(s) in the cluster. The settings should be similar to the following:

```
#-- start of file contents --
oracle.oam.adminserver=OracleAdminServer
oracle.oam.runtimeserver=oam_server2
OracleAdminServer= https://oamadminhost.us.example.com:9003
oam_server2=https://oamserver2.us.example.com:14101
#-- end of file --
```

5.8.5 Start the Servers

1. Start the servers on the node 1 machine.

   `$WAS_HOME/profiles/Dmgr01/bin/startManager.sh`

   `$WAS_HOME/profiles/Custom01/bin/syncNode.sh` *<MachineName> <SOAP port>*

   `$WAS_HOME/profiles/Custom01/bin/startNode.sh`

   `$WAS_HOME/profiles/Custom01/bin/startServer.sh OracleAdminServer`

   `$WAS_HOME/profiles/Custom01/bin/startServer.sh oam_server1a`

   `$WAS_HOME/profiles/Custom01/bin/startServer.sh oam_server1b`

2. Start the servers on the node 2 machine.

   `$WAS_HOME/profiles/Custom02/bin/syncNode.sh` *<MachineName> <SOAP port>*

   `$WAS_HOME/profiles/Custom02/bin/startNode.sh`

   `$WAS_HOME/profiles/Custom02/bin/startServer.sh oam_server2`

5.8.6 Next Steps

Both nodes are now configured and the OAM manager servers are ready to accept requests.

A load balancing router (LBR) now needs to be configured to route traffic to the managed server configured in both nodes. When the LBR configuration is complete, update the OAM load balancing configuration in oamconsole with the LBR information.

5.9 Managing OAM-Federation on IBM WebSphere

This section describes issues specific to managing OAM-Federation on IBM WebSphere. It contains this topic:

- SSLHandshakeException Error for Google and Yahoo IdP Partners

5.9.1 SSLHandshakeException Error for Google and Yahoo IdP Partners

When you integrate Access Manager with Identity Federation, and configure a Google or Yahoo IdP partner for federated SSO on IBM WebSphere application server through the OpenID protocol, you may see an `SSLHandshakeException` error when you attempt to access the resource.

For a Google partner, the error is as follows:

```
oracle.security.fed.controller.library.LibraryException:
oracle.security.fed.controller.frontend.action.exceptions.ResponseHandlerExcep
tion: oracle.security.fed.util.http.HttpException:
javax.net.ssl.SSLHandshakeException: com.ibm.jsse2.util.j: PKIX path building
failed: java.security.cert.CertPathBuilderException: PKIXCertPathBuilderImpl
could not build a valid CertPath.; internal cause is:
        java.security.cert.CertPathValidatorException: The certificate issued
by OU=XXX Secure Certificate Authority, O=XXX, C=US is not trusted;
...
```

For a Yahoo partner, the error is as follows:

```
[2013-02-15T15:18:58.747-08:00] [oam_server1] [WARNING] [OAM-12001]
[oracle.oam.audit] [tid: WebContainer : 5] [ecid: disabled,0] [APP:
oam_server_11.1.2.0.0] Cannot load audit configuration.
[2013-02-15T15:18:58.749-08:00] [oam_server1] [WARNING] [OAM-12001]
[oracle.oam.audit] [tid: WebContainer : 5] [ecid: disabled,0] [APP:
oam_server_11.1.2.0.0] Cannot load audit configuration.
[2013-02-15T15:18:58.750-08:00] [oam_server1] [WARNING] [OAM-12001]
[oracle.oam.audit] [tid: WebContainer : 5] [ecid: disabled,0] [APP:
oam_server_11.1.2.0.0] Cannot load audit configuration.
[2013-02-15T15:18:59.136-08:00] [oam_server1] [ERROR] [FEDSTS-12078]
[oracle.security.fed.controller.library.api.FedEngineInstance] [tid:
WebContainer : 5] [ecid: disabled,0] [APP: oam_server_11.1.2.0.0] Library
Exception: {0}[[
oracle.security.fed.controller.library.LibraryException:
oracle.security.fed.controller.frontend.action.exceptions.ResponseHandlerExcep
tion: oracle.security.fed.util.http.HttpException:
javax.net.ssl.SSLHandshakeException: com.ibm.jsse2.util.j: PKIX path building
failed: java.security.cert.CertPathBuilderException: PKIXCertPathBuilderImpl
could not build a valid CertPath.; internal cause is:
        java.security.cert.CertPathValidatorException: The certificate issued
by CN=XXX Root, OU="XXX, Inc.", O=XXX Corporation, C=US is not trusted;
...
```

This error is due to missing Yahoo/Google SSL certificates.

Solution

You need to import the Yahoo/Google SSL certificates into the IBM JSSE Trusted keystore.

First obtain the SSL certificates.

1. Using the Firefox browser, go to the https URL that is being accessed.
2. After viewing the page, right click on the page, then view page info, then details, then view certificate, then details tab.
3. Click export, then save.

Next, import the certificates into the keystore using the instructions provided in the following IBM Technote:

```
http://www-01.ibm.com/support/docview.wss?uid=swg21588087
```

Note: When executing the `keytool` command in Step 6 of the Technote:

- The alias is whatever string you want to use to reference that certificate afterwards.
- If you are not sure which `cacerts` to use, import the certificates to all the `cacerts` keystores.

Note: You may need to download Equifax certification from this URL:

`http://www.geotrust.com/resources/root-certificates/index.html`

Under Root Certificates, download Root1 - Equifax Secure Certificate Authority (.pem file).

Import this certificate using the steps described above.

6

Managing Oracle Access Manager Identity Assertion on IBM WebSphere

Oracle Access Manager Identity Assertion Provider for IBM WebSphere Application Server (IBM WebSphere) can be used to provide authentication and single sign-on with Oracle Access Manager 10*g* (10.1.4.3) or 11*g*.

This chapter includes the following topics:

> **Note:** For more information, see "Supported IBM WebSphere Application Server" on page 1-2. Information on using OAM with IBM WebSphere Portal is in Chapter 7, "Integrating Oracle Access Manager Identity Assertion with IBM WebSphere Portal."

6.1 Introduction to OAM Identity Assertion on IBM WebSphere

Oracle Access Manager Identity Assertion Provider is part of Oracle Fusion Middleware. Oracle provides an Identity Assertion Provider for IBM WebSphere that can be used to intercept and validate OAM sessions and generate IBM WebSphere-specific sessions.

IBM WebSphere allows Single Sign On (SSO) with external authenticators by using the Trust Association Interceptor (TAI). TAI interfaces provide mechanisms for external authenticators to perform user authentication and then assert the identity to IBM WebSphere. Oracle Access Manager Identity Assertion Provider for IBM WebSphere

uses the TAI interface to assert the user identity from the OAM session to IBM WebSphere. Upon receiving user identity information from the Identity Assertion Provider, IBM WebSphere queries the existence of the user in the user registry.

Oracle Access Manager Identity Assertion Provider for IBM WebSphere needs a valid OAM session for asserting the user identity to IBM WebSphere. Typically this is achieved by using an IBM HTTP Server (IHS) reverse proxy to front-end IBM WebSphere. OAM WebGate is installed on the IHS proxy and used to authenticate users against Oracle Access Manager. WebGate generates an OAM session token upon successfully authenticating a user. The IHS proxy then forwards this session token to IBM WebSphere. The Identity Assertion Provider intercepts the request and asserts the user identity from the session token for IBM WebSphere.

The Identity Assertion Provider provides identity assertion using either the HTTP Cookie or HTTP Request Headers. Accordingly, the IAP can be configured for Cookie based assertion or header based assertion.

- Cookie-based Assertion: Is based on OAM Session Token (ObSSOCookie). In this configuration, the Identity Assertion Provider checks availability of ObSSOCookie and validates it. On successful validation, user identity in the session cookie is asserted to IBM WebSphere.
- Header-based Assertion: Is based on HTTP Request Header. In this configuration, the Identity Assertion Provider checks availability of a particular (configurable) request header in the request. If available, the user identity within the header is asserted to IBM WebSphere.

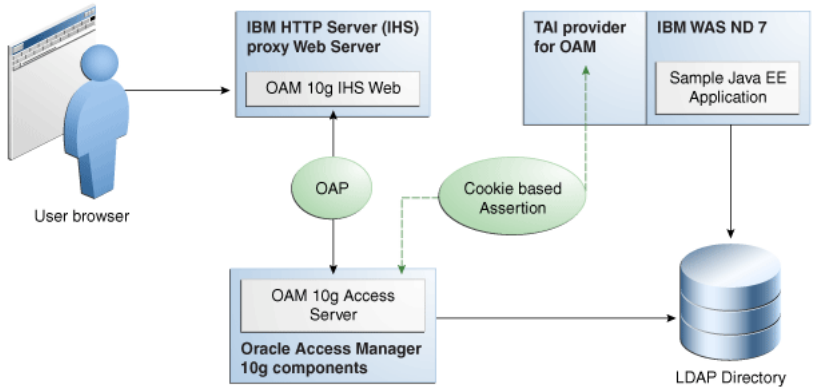
For more information, see the following:

- Scenario 1: Oracle Access Manager 10g (10.1.4.3) with the IAP on IBM WebSphere
- Scenario 2: OAM 11g with the IAP and IBM WebSphere

6.1.1 Scenario 1: Oracle Access Manager 10*g* (10.1.4.3) with the IAP on IBM WebSphere

This scenario describes a Java EE application that relies on Oracle Access Manager 10*g* (10.1.4.3) for authentication and authorization of its users. This application has been deployed on IBM WebSphere and can use the Identity Assertion Provider to provide SSO with Oracle Access Manager 10*g* (10.1.4.3).

Figure 6–1 Components and Process Flow with OAM 10g (10.1.4.3) and the IAP



Process overview: Identity Assertion on IBM WebSphere

1. Browser to IHS Proxy Web Server: User accesses the IBM WebSphere resource using the proxy IHS host and port, which triggers the 10*g* (10.1.4.3) WebGate installed on IHS Web server to authenticate and authorize the user.
2. WebGate to Access Server: WebGate communicates with OAM 10*g* (10.1.4.3) Access Server using Oracle Access Protocol (OAP). Access Server checks the Policy Store to locate any policies protecting the requested resource. WebGate through Access Server collects credential information from the user based on the Authentication Scheme specified and then validates whether the user can be authenticated. On successful authentication, WebGate through Access Server authorizes the user to access the requested resource on the IHS Web server. Additionally, WebGate sets authorization headers in the request as specified in the OAM Policy.
3. Web Server to IBM WebSphere: IHS Web Server acts as a proxy for IBM WebSphere and forwards the request to IBM WebSphere after successful authorization by OAM 10*g* (10.1.4.3) WebGate. IHS Web Server will also forward the HTTP Cookies and Request Headers set in the request to the IBM WebSphere.

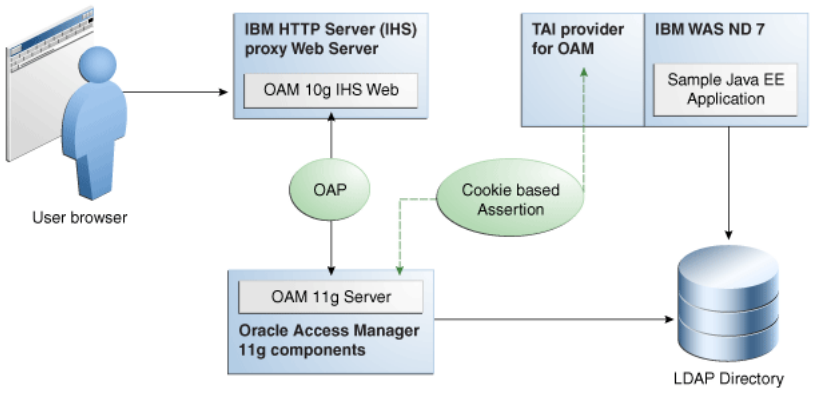
   Requests are intercepted at IBM WebSphere by OAM IAP. The TAI of OAM then validates the Cookie and HTTP Header. OAM IAP communicates with 10*g* (10.1.4.3) Access Server for Cookie-based assertions, to validate the session token and retrieve user information for the session. The TAI asserts this user identity to IBM WebSphere.

   IBM WebSphere checks for the existence of user in the user registry (configured LDAP instance) supplied by the OAM IAP. If the user is found, the assertion is successful. IBM WebSphere does not check for or request user's password in this scenario.
4. SSO Logout: See "Configuring SSO Logout for OAM IAP for IBM WebSphere" on page 6-19.

6.1.2 Scenario 2: OAM 11*g* with the IAP and IBM WebSphere

This scenario describes a Java EE application that relies on Oracle Access Manager 11*g* for authentication and authorization of its users. The Java EE application is deployed on IBM WebSphere to use the OAM IAP for IBM WebSphere for integrating the SSO with Oracle Access Manager 11*g*.

Figure 6–2 Components and Process Flow with OAM 11g and the IAP



Process overview: Identity Assertion with Oracle Access Manager 11*g*

1. Browser to IHS Proxy Web Server: The user accesses the resource (Sample Application on IBM WebSphere) using the proxy IHS host and port, which triggers the OAM 10*g* (10.1.4.3) WebGate installed to authenticate and authorize the user.
2. OAM 10*g* (10.1.4.3) IHS WebGate communicates with OAM 11*g* Server across the Oracle Access Protocol (OAP).

   OAM 11*g* Server checks its policy store to locate policies protecting the resource.

   WebGate and OAM 11*g* Server collect credentials from the user based on the authentication scheme specified in the policy, and the OAM 11*g* Server validates if the user can be authenticated.

   On successful authentication, WebGate and OAM Server authorize the user before access to the requested resource on the IHS Web server is granted. WebGate sets authorization headers in the request as specified in the OAM policy.
3. Web Server to IBM WebSphere: IHS Web Server acts as a proxy for IBM WebSphere and forwards the request to IBM WebSphere after successful authorization by OAM 10*g* (10.1.4.3) WebGate. IHS Web Server also forwards to IBM WebSphere the HTTP Cookies and Request Headers set in the request.

   Requests are intercepted at IBM WebSphere by OAM IAP. The TAI for OAM then validates the Cookie or HTTP Header. OAM IAP communicates with OAM 11*g* Server for Cookie-based assertions, to validate the session token, and retrieve user information for the session. TAI is responsible for asserting this user identity to IBM WebSphere.

   IBM WebSphere checks the existence of the user (supplied by the OAM IAP) in its user registry (configured LDAP instance). If user is found in the user registry, the assertion is successful. IBM WebSphere does not request nor check the user's password in this scenario.
4. SSO Logout: See "Configuring SSO Logout for OAM IAP for IBM WebSphere" on page 6-19.

6.2 Installing Components for the Oracle Access Manager IAP for IBM WebSphere

This section outlines the tasks you must perform to enable OAM Identity Assertion with IBM WebSphere.

The Oracle Access Manager IAP for IBM WebSphere is available as part of Oracle Fusion Middleware suite for IBM WebSphere. The IAP for IBM WebSphere jar is located at:

```
MW_HOME/oracle_common/modules/oracle.oamprovider_11.1.1/
    OAMTrustAssociationInterceptor.jar
```

Oracle Access Manager IAP for IBM WebSphere configuration file is located at:

```
MW_HOME/oracle_common/modules/oracle.oamprovider_11.1.1/
    domain_config/was/oamtai.xml
```

Note: Oracle Access Manager 10*g* (10.1.4.3) components and installation differs from Oracle Access Manager 11*g* components and installation. However, all other component installation tasks are the same.

Task overview: Installing components for IBM WebSphere, OAM, and the IAP

1. Install and set up IBM WebSphere as described in Chapter 2, "Installing and Configuring Oracle Identity and Access Management on IBM WebSphere."
2. IBM HTTP Server 7.x can be used as a reverse proxy in front of IBM WebSphere.

   Note: For IBM HTTP Server 7.x, use IHS22 WebGate package.

3. Oracle Access Manager: Install either:
   - OAM 10*g* (10.1.4.3): As described in the *Oracle Access Manager Installation Guide 10g* and includes:

     10*g* (10.1.4.3) Identity Server
     10*g* (10.1.4.3) Access Server
     10*g* (10.1.4.3) Policy Manager
     10*g* (10.1.4.3) Web Components for OHS 11*g* Web Server: Web Pass, Policy Manager and Web Gate)

   - OAM 11*g*: As described in *Oracle Fusion Middleware Installation Guide for Oracle Identity Management*, which includes:

     Oracle Access Manager 11*g* (11.1.1.3.0)
     Oracle Identity Manager 11*g* (11.1.1.3.0
     Oracle WebLogic Server

4. WebGate: Required whether you use OAM 10*g* (10.1.4.3) or OAM 11*g*, and can be installed after provisioning as described later in this chapter.

6.3 Introduction to the Oracle Access Manager 10*g* (10.1.4.3) Configuration Tool

This section introduces OAMCfgTool (oamcfgtool.jar) is a platform-agnostic configuration tool for use with Oracle Access Manager 10*g* (10.1.4.3). Skip this topic if you have OAM 11*g* deployed.

See Also: *Oracle Fusion Middleware Application Security Guide* for more information on OAMCfgTool

OAMCfgTool is a command-line utility provided to automatically run a series of scripts and set up policies. OAMCfgTool requires a set of parameters as inputs to create the required form-based authentication scheme, policy domain, access policies, and a WebGate profile for the Identity Asserter for single sign-on for IBM WebSphere.

Note: OAMCfgTool requires JRE 1.5 or 1.6. Internationalized login forms for Fusion Middleware applications are supported with the policies protecting those applications.

With OAM 10*g* (10.1.4.3) deployed, if you do not use the OAM Config Tool you must manually create the host-identifier, authentication schemes, and OAM policy manually using the Access System Console, as described in the *Oracle Access Manager Access Administration Guide.*

Example 6–1 a sample template for the configuration file for creating the required artifacts for the OAM IAP for IBM WebSphere. Additional information follows the example.

Example 6–1 Sample URIs_config File for OAMCfgTool and the IAP for IBM WebSphere

```
-- Template-starts --
##################################
#
# OAM-WAS Integration using OAM IAP
#
##################################
protected_uris

##################################
#Resources protected with default authentication scheme
/webcenter/adfAuthentication

##################################
public_uris
##################################
#Public Policy required for Cookie Based Assertion
Cookie Based Assertion
/Authen/SSOToken
-- Template-ends --
```

Example 6–2 illustrates a sample of the command-line syntax for OAMCfgTool when configuring artifacts for OAM 10*g* (10.1.4.3) and the IAP for IBM WebSphere.

Example 6–2 OAMCfgTool Syntax Configures Artifacts for OAM 10g (10.1.4.3) IAP

```
(echo ldappwdjava -jar oamcfgtool.jar
mode=CREATE app_domain=OAMPolicy_for_WAS-IAP
uris_file=/path-to-template-config-file
web_domain=host-id-name
ldap_host=wxyz
ldap_port=6633
ldap_userdn=orcladmin
ldap_base=ldap-base-dn
oam_aaa_host=abcd
oam_aaa_port=7789
oam_aaa_mode=open
log_file=OAMCfg_date.log
log_level=INFO
output_ldif_file=<LDIF_filename>
-noprompt
```

The above sample command produces the following artifacts:

- OAMPolicy_for_WAS-IAP, OAM Policy for protecting IBM WebSphere resources specified under protected_uris and public_uris
- OraDefaultAnonAuthNScheme, Anonymous Authentication Scheme used by OAMPolicy_for_WAS-IAP
- OraDefaultFormAuthNScheme, Form Authentication Scheme used by OAMPolicy_for_WAS-IAP
- Other OAM authentication scheme configuration

For a known resource, the public URI policy needs a Return Attribute in the Authorization Actions for Cookie-based assertion, as shown in Table 6–1. In this case, the return name OAM_REMOTE_USER is not configurable in oamtai.xml.

Table 6–1 Authorization Actions for "Cookie-based Assertion" in Public URI Policy

Type	Name	Return Attribute
HeaderVar	OAM_REMOTE_USER	uid

To enable Header-based assertion, you must set the Return Attribute in Authorization Actions of the Resource (protected_uris) protection policy. With Header-based Assertion, the return name OAM_REMOTE_USER is configurable in the oamtai.xml file and you must ensure that the Header-based Assertion section is uncommented.

Table 6–2 Authorization Actions for "Header Based Assertion" in Protected URI Policy

Type	Name	Return Attribute
HeaderVar	OAM_REMOTE_USER	uid

6.4 Provisioning WebGate and Configuring OAM 10*g* (10.1.4.3) and the IAP for IBM WebSphere

This section provides the steps to obtain the OAMCfgTool, provision the required WebGate, create a form authentication scheme, and create a policy domain and OAM 10*g* (10.1.4.3) policies for the IAP and IBM WebSphere.

> **See Also:** "Introduction to the Oracle Access Manager 10g (10.1.4.3) Configuration Tool" on page 6-5

To acquire OAMCfgTool and configure OAM 10*g* (10.1.4.3) for the IAP for IBM WebSphere:

1. Obtain the OAMCfgTool as follows:
   a. Log in to Oracle Technology Network at:

      `http://www.oracle.com/technology/software/products/middleware/htdocs/111110_fmw.html`

   b. Locate the OAMCfgTool ZIP file with Access Manager Core Components (10.1.4.3.0):

      `oamcfgtool<version>.zip`

   c. Extract and copy oamcfgtool.jar to the computer hosting the IBM WebSphere application to protect.

   d. Confirm that JDK 1.6 (or the latest version) is installed and configured on the host computer.

e. Change to the file system directory containing OAMCfgTool.

2. Provision WebGate, Create the Authentication Scheme, and Policy Domain: Run the following command using values for your environment. For example:

```
(echo ldappwdjava -jar oamcfgtool.jar
mode=CREATE app_domain=OAMPolicy_for_WAS-IAP
uris_file=/path-to-template-config-file
web_domain=host-id-name
ldap_host=wxyz
ldap_port=6633
ldap_userdn=orcladmin
ldap_base=ldap-base-dn
oam_aaa_host=abcd
oam_aaa_port=7789
oam_aaa_mode=open
log_file=OAMCfg_date.log
log_level=INFO
output_ldif_file=<LDIF_filename>
-noprompt
```

3. Review the information provided by the tool. For example, the parameter and values in Step 3 provide the following information:

```
Processed input parameters
Initialized Global Configuration
Successfully completed the Create operation.
 Operation Summary:
     Policy Domain  : OAMPolicy_for_WAS-IAP
     Host Identifier: OAMPolicy_for_WAS-IAP
     Access Gate ID : OAMPolicy_for_WAS-IAP_AG
```

4. Update host identifiers to include possible host-variations.

5. Add following authorization actions to the "Header Based Assertion" Policy.

Type	Name	Return Attribute
HeaderVar	OAM_REMOTE_USER	uid

6. Proceed to "Installing the Required WebGate for the IHS Web Server" on page 6-11.

6.5 Provisioning and Configuring OAM 11*g* for the IAP and IBM WebSphere

This section provides the following topics:

- About Provisioning WebGates and AccessGates with OAM 11g
- Provisioning Agents and Creating OAM 11g Policies for IBM WebSphere

6.5.1 About Provisioning WebGates and AccessGates with OAM 11*g*

This topic introduces OAM 11*g* access clients, known as policy-enforcement agents, and the process that is required to set up the trust mechanism between the agent and Oracle Access Manager 11*g* SSO. The process is known as provisioning (also known as registering an agent).

Only registered policy enforcement agents can communicate with an OAM Server, and process information when a user attempts to access a protected resource. Users

with valid OAM Administrator credentials can register an OAM Agent using the Administration Console.

You can register a WebGate agent before you install it. Required WebGate or AccessGate configuration files are created during registration and stored in the following path:

DOAMIN_NAME/output/$*Agent_NAME*

During registration, you can also create an application domain and default policies. For this reason, registering an agent is also known as "registering a partner application".

During registration, the Agent is presumed to be on the same Web server as the application it is protecting. However, the Agent can be on a proxy Web server and the application can be on a different host.

During Agent registration:

- One key is generated per agent, accessible to the WebGate through a local wallet file on the client host, and to OAM Server through the Java Key Store on the server side.

  The Agent specific key must be accessible to WebGates through a secure local storage on the client machine.
- A key is generated for the partner (application) during registration. (except for 10*g* (10.1.4.3) WebGate agents).
- An OAM application domain is created, named after the Agent, and populated with default authentication and authorization policies. The new application domain uses the same host identifier that was specified for the Agent during registration.

After registration, agent details appear in the OAM Administration Console and are propagated to all Managed Servers in the cluster. If you choose to automatically create policies during agent registration, you can also view and manage the application domain and policies that were registered with the partner application.

Table 6–3 describes each of named text fields where you enter requested information on the Create OAM Agent page.

Table 6–3 Create OAM Agent Pages for OAM 10g (10.1.4.3) and 11g Agents

OAM Agent Element	Description
Agent Name	The identifying name for this WebGate Agent. This is often the name of the computer that is hosting the Web server used by WebGate. **Note**: If the Agent Name exists, an error occurs and registration fails. If the host identifier exists, the unique Agent Base URL is added to the existing host identifier and registration proceeds.
Agent Base URL Optional	The host and port of the computer on which the Web server for the agent is installed. For example, http://*my_ohs_host:port* or https://*my_host:port*. The port number is optional. **Note**: A particular Agent Base URL can be registered once only. There is a one-to-one mapping from the Agent's Base URL to the Web server domain on which the WebGate is installed (as specified with the <hostidentifier> element). However, one domain can have multiple Agent's Base URLs.
Access Client Password	An optional, unique password for this WebGate, which was assigned during WebGate registration. When a registered WebGate connects to an OAM 11*g* Server, the password is used for authentication to prevent unauthorized WebGates from connecting to OAM 11*g* Servers and obtaining policy information.

Table 6–3 (Cont.) Create OAM Agent Pages for OAM 10g (10.1.4.3) and 11g Agents

OAM Agent Element	Description
Security	Level of communication transport security between the Agent and the OAM Server (this must match the level specified for the OAM Server): ■ Open--No transport security ■ Simple--SSL v3/TLS v1.0 secure transport using dynamically generated session keys ■ Cert--SSL v3/TLS v1.0 secure transport using server side x.509 certificates. Choosing this option displays a field where you can enter the Agent Key Password, discussed separately within this table.
Host Identifier	This identifier represents the Web server host.
Auto Create Policies	During agent registration, you can have authentication and authorization policies created automatically. This option is checked (enabled) by default. Default: Enabled **Note**: If you already have a domain and policies registered, you can simply add new resources to it. If you clear this option (no check), no application domain or policies are generated automatically.
Protected Resource (URI) List	URIs for the protected application: /myapp/login, for example. Each URI for the protected application should be specified in a new row of the table for the Protected Resource List. The following resource is protected by default: /** The default matches any sequence of characters within zero or more intermediate levels spanning multiple directories. Add all IBM WebSphere resources to be protected to this list.
Public Resource (URI) List	Each public application should be specified in a new row of the table for the Public Resource List. Add a field and enter URI values for the public applications and resources. Each URI should be specified in a new row of the table for the Public Resource List. Add all IBM WebSphere resources that should not be protected to this list. Note: /Authen/SSOToken is an additional public resource that is used by the Oracle Access Manager Identity Assertion Provider.

See Also: *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service* for more information

6.5.2 Provisioning Agents and Creating OAM 11*g* Policies for IBM WebSphere

This topic describes how to provision agents and create policies for OAM 11*g*.

At least one OAM Server instance must be running in the same mode as the agent. Otherwise, agent registration fails. After provisioning, you can change the communication mode of the OAM Server if needed. Communication between the agent and server continues to work as long as the WebGate mode is at least at the same level as the OAM Server mode (or higher).

To register an agent and create policies for the OAM 11*g* IAP for IBM WebSphere:

1. Log in to the OAM 11*g* Administration Console as usual. For example: http://*host:port*/oamconsole.
2. On the Welcome page, click Add OAM 10*g* (10.1.4.3) Agent in the Agent Configuration panel to open a fresh page:

   Alternatively: From the System Configuration tab, expand the Agents node, the OAM Agents node, and the 10*g* (10.1.4.3) Webgates node, then click the Create command button in the tool bar.

3. On the Create: OAM Agent page, enter required details (those with an *) to register this OAM Agent, as shown in Table 6–3.
4. **Protected Resource List**: In this table, enter individual resource URLs to be protected by this OAM Agent, as shown in Table 6–3.
5. **Public Resource List**: In this table, enter individual resource URLs to be public (not protected), as shown in Table 6–3, including /Authen/SSOToken used by the Oracle Access Manager Identity Assertion Provide.
6. Confirm that the Auto Create Policies box is checked (or clear the box to disable this function).
7. Click Apply to submit the registration (or close the page without applying changes).
8. Check the Confirmation window for the location of generated artifacts and then close the window.
9. Repeat steps in this procedure to register an additional AccessGate and policies for use by WebGate and:
   - Enter a name for this registration.
   - Select the appropriate Security mode.
   - Do not specify a Base URL.
   - Check Auto Create Policies
   - Click Apply
10. Proceed to "Installing the Required WebGate for the IHS Web Server".

6.6 Installing the Required WebGate for the IHS Web Server

After provisioning, you can install the OAM 10*g* (10.1.4.3) WebGate for IHS to operate within either an OAM 10*g* (10.1.4.3) or OAM 11*g* deployment as described here. Ignore any steps that do not apply to your environment.

To download and install the 10*g* (10.1.4.3) WebGate for IHS:

1. Locate and download the WebGate installer as follows:
   a. Go to Oracle Fusion Middleware 11*g*R1 Software Downloads at:

      `http://www.oracle.com/technology/software/products/middleware/htdocs/fmw_11_download.html`

   b. Click **Accept License Agreement**, at the top of the page.
   c. From the **Access Manager WebGates (10.1.4.3.0)** row, click the download link for the desired platform and follow on-screen instructions.
   d. Store the WebGate installer in the same directory with any 10*g* (10.1.4.3) Access System Language Packs you want to install.
2. Launch the WebGate installer for your platform, installation mode, and Web server, and then:
   a. Dismiss the Welcome screen by clicking Next.
   b. Respond with administrator privileges when asked.
   c. Specify the installation directory for the WebGate. For example:

      /OracleAccessManager/WebComponent/

d. **Linux or Solaris**: Specify the location of the GCC runtime libraries on this computer.

e. **Language Pack**—Choose a Default Locale and any other Locales to install, then click Next.

f. Record the installation directory name in the preparation worksheet if you haven't already, then click Next to continue.

The WebGate installation begins, which may take a few seconds.

3. **OAM** 10*g* (10.1.4.3) **Deployment**: Continue installation, as described in the 10*g* (10.1.4.3) *Oracle COREid Access and Identity Installation Guide*, and:

a. Specify the same values when you install the WebGate that were specified when provisioning the WebGate using OAMCfgTool, earlier.

b. Specify any additional requested values to properly finish the installation

c. Copy the files to the WebGate host: *WebGate_install_dir*/access/oblix/config.

d. Restart the WebGate Web server.

e. Proceed to "Preparing the IHS Web Server" on page 6-12.

4. **OAM 11g Deployment**: Cancel the WebGate installer (without finishing) and gather WebGate 10*g* (10.1.4.3) provisioning artifacts (and certificate files, if needed). For example:

a. On the OAM AdminServer host, locate and copy the updated OAM Agent ObAccessClient.xml configuration file (and any certificate artifacts). For example:

DOMAIN_HOME/output/$*Agent_Name*/

ObAccessClient.xml
password.xml (if needed)
aaa_key.pem (your private key generated by openSSL)
aaa_cert.pem (signed certificates in PEM format)

b. On the OAM Agent host, add the artifacts to the WebGate directory path. For example:

WebGate_install_dir/access/oblix/lib/ObAccessClient.xml
WebGate_install_dir/access/oblix/config

c. Restart the WebGate Web server.

d. Run the EditHTTPConf tool to update IHS Server configuration for WebGate.

e. Restart the OAM Server that is hosting the Agent.

f. Proceed to "Preparing the IHS Web Server" on page 6-12.

6.7 Preparing the IHS Web Server

When you have 10*g* (10.1.4.3) IHS2 WebGate (or later), the IHS httpd.conf file includes entries for adding the /oamsso directory to the Web Server root. However, if you have an earlier Oracle Access Manager IHS2 WebGate, you must add the following entries under the WebGate block of the httpd.conf file.

To prepare the IHS Web server:

1. On the computer hosting the WebGate, locate IHS httpd.conf file and confirm the following entries exist (if they do not add them):

```
Alias /oamsso "<webage-install-dir>/access/oamsso"
<LocationMatch "/oamsso/*">
Satisfy All
</LocationMatch>
```

2. Proceed with "Preparing the Login Form for WebGate".

6.8 Preparing the Login Form for WebGate

This section describes how to acquire the proper Oracle Access Manager forms for use with the provisioned and installed 10*g* (10.1.4.3) IHS WebGate. No login forms are used from WebGate

If you have OAM 11*g*, the OAM 11*g* Server instance provides the Login form and you can skip this procedure.

> **Note:** The forms provided with 10*g* (10.1.4.3) WebGates cannot be used with OAM 11*g* Servers.

In an OAM 10*g* (10.1.4.3) deployment, if you have:

- 10*g* (10.1.4.3) IHS2 WebGate (or later), find login.html in *WebGate_install_dir*/access/oamsso/login.html.
- Earlier 10*g* (10.1.4.3) IHS2 WebGate, you must create the directory and place a sample login.html file manually, as described in the following procedure.

To preview the login.html file for 10*g* (10.1.4.3) IHS WebGate:

1. OAM 10*g* (10.1.4.3) with 10*g* (10.1.4.3) IHS2 WebGate (or later), preview login.html in *WebGate_install_dir*/access/oamsso/login.html.
2. OAM 10*g* (10.1.4.3) with 10*g* (10.1.4.2.0) or earlier WebGate for IHS2:
   a. Create an /oamsso subdirectory in the following path: *WebGate_install_dir*/oamsso.
   b. Create and add to the new /oamsso directory a login.html file with the following elements:

```
<!--Sample login Page Code -->
<form name="loginForm" method="post" action="/access/sso">
<b> Username: </b> <input name="userid" type="text" maxLength="80"
size="20" value="">
<b> Password: </b> <input type="password" maxLength="255" size="20"
name="password" autocomplete="off">
<input type="submit" value="Login" name="submit">
</form>
```

3. Proceed to "Configuring IBM WebSphere for OAM SSO and the IAP".

6.9 Configuring IBM WebSphere for OAM SSO and the IAP

This section provides the following topics:

- Configuring a Stand Alone LDAP Registry for OAM in IBM WebSphere

- Adding and Configuring a Virtual Host in IBM WebSphere
- Configuring IHS Reverse Proxy in the IBM WebSphere Console
- Creating the Interceptor Entry in the IBM WebSphere Console
- Configuring the OAM TAI Configuration File

6.9.1 Configuring a Stand Alone LDAP Registry for OAM in IBM WebSphere

This section describes how to configure a stand-alone LDAP registry for OAM within IBM WebSphere.

To configure a stand alone LDAP registry for OAM in IBM WebSphere:

1. Login to your IBM WebSphere console. For example:

   `http://host:port/ibm/console`

2. Go to Security, Global Security.
3. Under User account repository in Available realm definitions, select Standalone Ldap Registry and click Configure.
4. Under General Properties, fill in fields to configure the LDAP directory that is used by OAM:

   Primary administrative user name <OAM admin username>
   Server user identity: keep the default selection
   Type of Ldap Server: <LDAP Directory Type for OAM>
   Host: < host name where LDAP directory resides>
   Port : <LDAP directory bind port>
   Base DN: <LDAP base DN>
   Bind DN: <LDAP bind DN>
   Password: <LDAP password>
   Search timeout: keep the default value (120 seconds)
   Keep default Reuse connection and Ignore case for authorization (checked)

5. Click Apply and OK and save this configuration.
6. On the same page, under Additional Properties, click Advanced Lightweight Directory Access Protocol (LDAP) user registry settings and fill in fields under the General Properties:

   User filter: (&(uid=%v)(objectclass=inetOrgPerson))
   Group filter: (&(cn=%v)(objectclass=ldapsubentry))
   User ID Map: uid
   Group ID Map: cn
   Group Member ID Map: nsRole:nsRole

7. Click Apply and OK and save this configuration.
8. On the same page, under Related Items, click Trusted authentication realms - inbound and confirm that the LDAP entry (host:port) is trusted.
9. Click Test connection to verify the connection configuration.
10. Restart IBM WebSphere.

    If Standalone LDAP Registry is not selected as "Current realm" then under "User account repository" in "Available realm" definitions, select "Standalone Ldap Registry" and click "Set As Current".

11. From now onward, log in to the IBM WebSphere console using OAM LDAP directory login credentials (as registered with IBM WebSphere).

6.9.2 Adding and Configuring a Virtual Host in IBM WebSphere

You must bind your Web applications to virtual hosts (logical name for configuring Web applications to a particular host name). When you request a resource, IBM WebSphere maps the request to an alias of a defined virtual host.

To add and configure a virtual host in IBM WebSphere for the enterprise application:

1. Login to your IBM WebSphere console. For example: http://host:port/ibm/console
2. Go to Environment, Virtual Hosts, and click New
3. Enter the General Properties for your environment, as follows:
   a. Add name: *IHS host name* and click on Ok and then save the changes.
   b. Click the recently created entry *IHS host name*:
4. Under Additional Properties, click Host Aliases, and then click New.
5. Fill in details for General Properties for your environment, as follows:
   a. Host: *Host name where IHS server resides*
   b. Port: *IHS port*
6. Click OK to save the changes and continue with the next steps to configure the virtual host in your deployed enterprise application.
7. Go to Applications, WebSphere Enterprise Applications, and:
   a. Click <enterprise application>.
   b. Under Web Module Properties, click Virtual Hosts.
   c. Select all the Web modules and apply the virtual host that you added.
   d. Click OK, then save the changes.
8. Restart IBM WebSphere where the enterprise application is deployed.
9. Proceed to "Configuring IHS Reverse Proxy in the IBM WebSphere Console".

6.9.3 Configuring IHS Reverse Proxy in the IBM WebSphere Console

This section describes how to configures the IHS server in reverse proxy mode within the IBM WebSphere console.

To configure IHS in reverse proxy mode within IBM WebSphere:

1. Login to your IBM WebSphere console. For example:

   ```
   http://host:port/ibm/console
   ```

2. Go to Server Types, Web Servers.
3. Click New, and provide IHS Web server details.
4. Save changes to see a server entry for IHS.
5. Select the *ServerName* and click Generate Plug-in.
6. Select the *ServerName* and click Propagate Plug-in:

7. Configure the IHS Web server to act as a reverse proxy for IBM WebSphere, as follows:
   a. Locate plugin-cfg.xml in *IHS_install_dir*/Plugins/config/*ServerName*
   b. Remove the following entry:

```
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/*"/>
```

8. Restart the IHS Web server.
9. Proceed to "Creating the Interceptor Entry in the IBM WebSphere Console".

6.9.4 Creating the Interceptor Entry in the IBM WebSphere Console

Tasks are the same whether you are using Oracle Access Manager 10*g* (10.1.4.3) or Oracle Access Manager 11*g*.

At runtime, the IBM WebSphere extension class loader loads classes. The extension class loader class path is specified by the `ws.ext.dirs` system property. Therefore, you must add the IAP for IBM WebSphere OAMTrustAssociationInterceptor.jar file in the IBM WebSphere classpath:

The IAP for IBM WebSphere OAMTrustAssociationInterceptor.jar file is available from the following path:

```
MW_HOME/oracle_common/modules/oracle.oamprovider_11.1.1/
     OAMTrustAssociationInterceptor.jar
```

To add the OAMTrustAssociationInterceptor.jar to the IBM WebSphere classpath:

1. In IBM WebSphere console go to Servers, Server Types, WebSphere Application, Servers, and select the appropriate server.
2. Under the Server Infrastructure section, click Java And Process Management, and then Process Definition.
3. In Additional properties, select Java Virtual Machine, Custom Properties.
4. In the property `ws.ext.dirs`, add the value for `OAMTrustAssociationInterceptor.jar`. For example:

```
MW_HOME/oracle_common/modules/oracle.oamprovider_11.1.1/
OAMTrustAssociationInterceptor.jar
```

5. Confirm that the two values are separated by colon.
6. Create the Interceptor entry for the OAM IAP, as follows:
   a. In the IBM WebSphere console, go to **Security**, **Global Security**, and ensure that "**Enable Application Security**" is checked.
   b. Under the "**Authentication**" section, click "**Web and SIP Security**" tab, and then click the **Trust association** link.
      a. Under **General Properties**, check the "Enable Trust Association".
      b. Under **Additional Properties**, click Interceptors link.
   c. Under **General Properties**, click **Under New**, and provide the Interceptor class name as follows:

```
oracle.security.was.providers.tai.OAMTrustAssociationInterceptorImpl
```

7. Proceed to "Configuring the OAM TAI Configuration File" to configure oamtai.xml as a custom property of Interceptor class path.

6.9.5 Configuring the OAM TAI Configuration File

The oamtai.xml configuration file is used by the OAM Trust Association Interceptor. You must configure the file and modify it for your environment. For details, see:

- About Configuring the OAM TAI Configuration File
- Configuring the OAM TAI Configuration File

6.9.5.1 About Configuring the OAM TAI Configuration File

The oamtai.xml configuration file is available in the following path:

```
MW_HOME/oracle_common/modules/oracle.oamprovider_11.1.1/domain_config_was
/oamtai.xml
```

This file stores the details that are used by the TAI at run time to establish a connection with 10*g* (10.1.4.3) OAM Access Server (or 11*g* OAM Server).

There are two ways to configure the oamtai.xml file:

- Either copy oamtai.xml to *was_profile_dir*`/config/cells/`*cell_name*`/fmwconfig/oamtai.xml`.
- Or perform Step 1 in the following procedure to configure oamtai.xml as a custom property of the Interceptor entry added earlier.

You must modify the oamtai.xml file to establish a connection to the Access Server, using parameters in Table 6–4 and values for your deployment. To enable Header based assertion, ensure that the Header Based Assertion section in oamtai.xml is not commented and use the same `customHeadername` in both oamtai.xml and the OAM policy.

Table 6–4 oamtai.xml Configuration File Parameters

Parameter	Required or Not	Description
hostPort	Required	Hostname and port of the IHS Web server where the resource is hosted. Note: The host:port should be one of the host name variations present in OAM.
resource	Required	The URL to the protected resource. Default = /Authen/SSOToken or the value in the OAM policy if you have updated it.
ip	Optional	IP address of the client computer that needs to access the resource.
operation	Required	Operation requested to access the Authen/SSOToken.
accessGateName	Required	A unique name, without spaces, that identifies the AccessGate to be used while interacting with OAM. With OAMCfgTool the name is derived from the app_domain value, appended with _AG.
AccessGatePassword	Required	A unique password to verify and identify the AccessGate when interacting with OAM. This prevents unauthorized AccessGates from connecting and obtaining policy information. With OAMCfgTool, this is specified with the app_agent_password parameter. This should differ for each WebGate/AccessGate instance.
accessServerHost	Required	OAM Access Server (or OAM 11*g* Server) host name.
accessServerPort	Required	OAM Access Server (or OAM 11*g* Server) port number.
accessServerName	Optional	Name of the OAM Access Server, as identified in the profile (or OAM 11*g* Server registration).

Table 6–4 (Cont.) oamtai.xml Configuration File Parameters

Parameter	Required or Not	Description
transportSecurity	Required	The level of transport security between the 10*g* (10.1.4.3) Access Server and associated WebGates must match. The default value is Open. You can specify a different value with OAMCfgTool oam_aaa_mode value. The following parameters trustStore, keyStore, keyStorePass and globalPass values are required when transport security mode is 'Simple' or 'Cert' ■ trustStore: Specify the absolute path to the trust store. ■ keyStore: Specify the absolute path to the key store ■ keyStorePass: Specify the keystore password, ■ globalPass: Specify the global passphrase value that was defined during IHS WebGate installation and configuration.
debug	Required	Turns OAM debugging on or off. Default: false
minConn	Required	The minimum number of connections that this AccessGate can establish with Access Servers. This number must be the same as or less than the number of Access Servers that are actually associated with the WebGate.
maxConn	Required	The maximum number of connections that this AccessGate can establish with Access Servers. This number must be the same as or greater than the number of Access Servers that are actually associated with the WebGate.
timeOutForConnPool	Required	Connection pool time out period. Specify any value in milliseconds. Default: 30000 (milliseconds)
Anonymous	Required	Configures the anonymous user value. Note: Following two parameters assertionType and customHeaderName are required for Header Based Assertion. Uncomment it if and only if in case of Header based assertion. ■ If user configures the headername here, then the same name will be used to configure as return attribute in OAM policy. And don't change the value of assertion type parameter only uncomment parameter entry ■ If user will not be configuring the header name here, then default header name is "OAM_REMOTE_USER" and same should be configured in OAM policy. Also don't change the value of assertion type parameter only uncomment parameter entry
assertionType	Required	The value should be 'HeaderBasedAssertion', don't change it
customHeaderName	Required	Default value used is " OAM_REMOTE_USER", or according to the OAM Policy if you have updated it. Note: You can provide any value as long as the same value is used in the OAM policy while configuring the Header. Otherwise you must use the default value "OAM_REMOTE_USER" while configuring the policy. In both cases, ensure that the "assertionType" parameter entry in the oamtai.xml file is uncommented.

Note: WebGate timeout should be greater than LTPA timeout. Otherwise, the IAP is not triggered which could cause the WebGate session to time out. If this occurs, a user who logs in with a different userID could get access to the resource because the previously generated LTPA token still exists. LTPA timeout default value is 120 minutes; therefore, the WebGate profile requires a WebGate timeout value greater than 120 minutes.

6.9.5.2 Configuring the OAM TAI Configuration File

The following procedure describes how to configure oamtai.xml for your environment.

Skip Step 1 if oamtai was copied to the following path: *was_profile_dir*/config/cells/*cell_name*/fmwconfig/oamtai.xml.

To configure oamtai.xml as a custom property of the Interceptor:

1. **Custom Interceptor Property**:
   a. In the IBM WebSphere console, go to **Security**, **Global Security**.
   b. Under the "**Authentication**" section, click "**Web and SIP Security tab**"; click the **Trust association** link.
   c. Click the **Trust association** link.
   d. Under **Additional Properties**, click Interceptors link.
   e. Select the Interceptor class name `oracle.security.was.providers.tai.OAMTrustAssociationInterceptorImpl`
   f. Under **Custom Properties**, add a property with the absolute path of oamtai.xml details for the oamtai.xml file:

      Name: *OAMTaiProperty*

      Value: *was_profile_dir*/config/cells/*cell_name*/fmwconfig/oamtai.xml
2. **Modify oamtai.xml**: Use parameters in Table 6–4 with values for your deployment to a establish a connection with the Access Server.
3. **Header Based Assertion**: In the oamtai.xml file, perform the following steps.
   a. Uncomment the "assertionType" entry and retain the value "HeaderBasedAssertion".
   b. Uncomment the "customHeaderName" entry and set the value as desired (Table 6–4).
4. Save the file.
5. **OAM Policy:** Use the same "customHeaderName" value when configuring the OAM policy.
6. Restart IBM WebSphere for changes to take affect.

6.10 Configuring SSO Logout for OAM IAP for IBM WebSphere

This section describes logout with the OAM IAP for IBM WebSphere.

- Configuring Logout for Generic (or Non-ADF) Applications
- Configuring Logout for ADF-Coded Applications

6.10.1 Configuring Logout for Generic (or Non-ADF) Applications

In non-ADF applications, logout is initiated when an application causes the invocation of the logout.html that is configured as the target in the application's logout link.

The logout.html file can be placed at the Web server's doc root, or it can be part of the IBM WebSphere application.

If you are using your own logout.html, you can embed Example 6–3 JavaScript to invoke "delOblixCookie" upon loading the page body. The LTPAToken is deleted by JavaScript; ObSSOCookie is deleted by WebGate.

```
<body onload="delOblixCookie();">
```

> **Note:** If cookie "httponly" property is set to "true" by users for security considerations, javascript cannot delete LTPAToken cookies and SSO logout will not work.

Example 6–3 JavaScript to invoke `delOblixCookie`

```
function delCookie(name,path,domain) {
   var today = new Date();
   var deleteDate = new Date(today.getTime() - 48 * 60 * 60 * 1000); // minus 2
   days
   var cookie = name + "="
            + ((path == null) ? "" : "; path=" + path)
            + ((domain == null) ? "" : "; domain=" + domain)
            + "; expires=" + deleteDate;
   document.cookie = cookie;
}
function delOblixCookie() {
         // set focus to ok button
      var isNetscape = (document.layers);
  if (isNetscape == false || navigator.appVersion.charAt(0) >= 5) {
    for (var i=0; i<document.links.length; i++) {
      if (document.links[i].href == "javascript:top.close()") {
          document.links[i].focus();
          break;
      }
    }
   }
  delCookie('ObTEMC', '/');
  delCookie('ObSSOCookie', '/');
  delCookie('LtpaToken', '/');
  delCookie('LtpaToken2', '/');
  // in case cookieDomain is configured
  // delete same cookie to all of subdomain
  var subdomain;
  var domain = new String(document.domain);
  var index = domain.indexOf(".");
  while (index > 0) {
     subdomain = domain.substring(index, domain.length);
     if (subdomain.indexOf(".", 1) > 0) {
         delCookie('ObTEMC', '/', subdomain);
         delCookie('ObSSOCookie', '/', subdomain);
         delCookie('LtpaToken', '/', subdomain);
         delCookie('LtpaToken2', '/', subdomain);
     }
     domain = subdomain;
     index = domain.indexOf(".", 1);
   }
}
```

To configure logout for generic (non-ADF) applications:

1. Locate the desired logout.html file.

2. Add the JavaScript in Example 6–3 to logout.html to invoke "delOblixCookie" upon loading the page body.
3. In the Oracle Access Manager policy, protect logout.html using the Anonymous Authentication Scheme, as described in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service*.

6.10.2 Configuring Logout for ADF-Coded Applications

In ADF coded Fusion Middleware applications such as Oracle WebCenter Portal application, single sign off is achieved through OPSS. For details, see the following topics:

- Configuring WebGate for Logout
- Configuring OPSS for SSO Logout with Oracle Access Manager
- Configuring oamAuthenProvider.jar in the IBM WebSphere classpath
- Verifying SSO Logout

6.10.2.1 Configuring WebGate for Logout

This topic provides an example (Example 6–4) and procedure that you can use and customize to logout an application protected by OAM 10g with a 10g WebGate

> **Note:** Example 6–4 applies only for an end URI of a single word. For a long URI, you must update the parsing logic accordingly.

To configure WebGate for logout:

1. Create and edit logout.html for the WebGate based on Example 6–4: add and call the function `handleLogout()` for redirecting the logout request to the end URL specified in the logout URL

Example 6–4 Sample logout.html Script

```
<html>
<head>
<script language="javascript" type="text/javascript">

function handleLogout() {

    //get protocol used at the server (http/https)
    var webServerProtocol = window.location.protocol;
    //get server host:port
    var webServerHostPort = window.location.host;
    //get query string present in this URL
    var origQueryString = window.location.search.substring(1);

    //vars to parse the querystring
    var params = new Array();
    var par = new Array();
    var val;

    if (origQueryString != null && origQueryString != "") {

        params = origQueryString.split("&");

        //search for end_url and redirect the user to this
```

```
        for (var i=0; i<params.length; i++) {

        par = params[i].split("=");
        if ("end_url" == par[0]) {
          endUrlVal = par[1];

        //check if val (value of end_url) begins with "/" or "%2F" (is it an URI?)
        if (endUrlVal.substring(0,1) == "/" || endUrlVal.substring(0,1) == "%") {
          if (endUrlVal.substring(0,1) == "%")
            endUrlVal = "/" + endUrlVal.substring(3);

         //modify the end_url value now
           endUrlVal = webServerProtocol + "//" + webServerHostPort + endUrlVal;
         }
    //redirect the user to this URL
    window.location.href = endUrlVal;
         }
       }
   }
}
</script>
</head>
<body onLoad="handleLogout();">
<h3>You have been logged out<h3>

</body>
</html>
```

2. Store your logout.html script to *WebGate_install_dir*/oamsso/logout.html
3. In the httpd.conf file, ensure following entries exist under the WebGate block:

```
Alias /oamsso "<webage-install-dir>/access/oamsso
<LocationMatch "/oamsso/*">
Satisfy All
</LocationMatch>
```

4. Proceed to "Configuring OPSS for SSO Logout with Oracle Access Manager".

6.10.2.2 Configuring OPSS for SSO Logout with Oracle Access Manager

Application configuration for logout depends on whether you have an ADF-coded application integrated with OPSS versus not integrated with OPSS. This topic focuses on ADF-coded applications that are integrated with OPSS.

The following procedure is similar to configuring logout for 10g WebGates, with a specific step for ADF-coded applications, which must send the end_url value to identify where to redirect the user after logout processing. However, with ADF-coded applications, logout occurs when the application causes the following URI to be invoked:

```
/<app context root>/adfAuthentication?logout=true&end_url=<any uri>
```

To configure OPSS for SSO Logout with OAM:

1. Locate and open the jps-config .xml file in the following path:

   was_profile_dir/config/cells/*cell_name*/fmwconfig/jps-config.xml

2. Within `jps-config .xml`, add the following **<propertySet name="props.auth.uri.0">** element and values:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<jpsConfig xmlns="http://xmlns.oracle.com/oracleas/schema/11/jps-config-11_
1.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.oracle.com/oracleas/schema/11/jps-config-11_
1.xsd">
<property value="off" name="oracle.security.jps.jaas.mode"/>
<propertySets>
.
<propertySet name="props.auth.uri.0">
<property value="/oamsso/logout.html" name="logout.url"/>
<property value="${app.context}/adfAuthentication" name="login.url.BASIC"/>
<propertyvalue="${app.context}/adfAuthentication"name="login.url.ANONYMOUS"/>
<property value="${app.context}/adfAuthentication" name="login.url.FORM"/>
</propertySet>
<propertySet name="props.auth.level.0">
<property value="0" name="type-level:ANONYMOUS"/>
<property value="1" name="type-level:BASIC"/>
<property value="2" name="type-level:FORM"/>
.
</propertySets>
```

3. Within jps-config .xml, add the following **<serviceProviders>** element and values:

```
...
</propertySets>
<serviceProviders>
<serviceProvider class="oracle.security.jps.internal.sso.SsoService
Provider" name="sso.provider.0" type="SSO"/>
    </serviceProviders>
```

4. Within jps-config .xml, add the following **<serviceInstances>** element and values:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
...
</serviceProviders>
<serviceInstances>
.
.
<serviceInstance provider="sso.provider.0" name="sso.inst.0">
<property value="oracle.security.jps.wls.internal.sso.WlsToken
Provider" name="token.provider.class"/>
<property value="2" name="default.auth.level"/>
<property value="oracle.security.wls.oam.providers.sso.OAMSSO
ServiceProviderImpl" name="sso.provider.class"/>
<property value="OAMSSOToken" name="token.type"/>
<propertySetRef ref="props.auth.uri.0"/>
<propertySetRef ref="props.auth.level.0"/>
</serviceInstance>
.
.
</serviceInstances>
```

5. Within jpsContexts, add the highlighted <serviceInstanceRef ref="sso.inst.0"/> element and value:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
...
</serviceInstances>
<jpsContexts default="default">
<jpsContext name="default">
<serviceInstanceRef ref="credstore"/>
```

```
<serviceInstanceRef ref="keystore"/>
<serviceInstanceRef ref="policystore.xml"/>
<serviceInstanceRef ref="audit"/>
<serviceInstanceRef ref="idstore.ldap"/>
<serviceInstanceRef ref="sso.inst.0"/>
</jpsContext>
</jpsContexts>
</jpsConfig>
```

6. In the Oracle Access Manager policy, protect `/oamsso/logout.html` with the Anonymous Authentication scheme, as described in theOracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service.
7. Proceed to "Configuring oamAuthenProvider.jar in the IBM WebSphere classpath".

6.10.2.3 Configuring oamAuthenProvider.jar in the IBM WebSphere classpath

To perform logout through OPSS, you must configure oamAuthnProvider.jar in the IBM WebSphere classpath. This is similar to adding the interceptor jar in the IBM WebSphere classpath in "Creating the Interceptor Entry in the IBM WebSphere Console" on page 6-16.

The oamAuthnProvider.jar file is available from the following path:

```
MW_HOME/oracle_common/modules/oracle.oamprovider_11.1.1/
    oamAuthnProvider.jar
```

To add oamAuthnProvider.jar to the IBM WebSphere classpath:

1. In the IBM WebSphere console go to Servers, Server Types, WebSphere Application, Servers, and select the appropriate server.
2. Under the Server Infrastructure section, click Java And Process Management, and then click Process Definition.
3. In Additional properties, select Java Virtual Machine, Custom Properties.
4. In the ws.ext.dirs property, add the value for oamAuthnProvider.jar after the entry for OAMTrustAssociationInterceptor.jar and confirm that the two values are separated by a colon. For example:

   ```
   ws.ext.dir   MW_HOME/oracle_common/modules/oracle.oamprovider_11.1.1/
      OAMTrustAssociationInterceptor.jar:MW_HOME/oracle_common/modules/
      oracle.oamprovider_11.1.1/oamAuthnProvider.jar
   ```

5. Restart IBM WebSphere.
6. Proceed to "Verifying SSO Logout"

6.10.2.4 Verifying SSO Logout

To verify SSO logout:

1. From a browser, enter the URL of the protected resource. For example:

   ```
   http://host:port/<app context root>/adfAuthentication
   ```

2. Confirm that the login page appears and sign in using proper credentials
3. Confirm that the protected resource is served
4. Open a new browser tab or window and access the same resource to confirm that the second attempt does not require another login

5. Logout from one tab using a URL like the following sample:

   ```
   http://host:port/<app context root>/adfAuthentication?logout=true&end_url=<any uri>
   ```

6. Access the resource again to confirm that a login page appears.

6.11 Known Issues

Problem:

Oracle Access Manager Identity Assertion Provider for IBM WebSphere does not support the Simple security mode.

Problem: Inconsistent

Oracle Access Manager Identity Assertion Provider for IBM WebSphere does not generate an LTPA token after successful authentication and valid ObSSOCookie generation.

```
Error
403: AuthenticationFailed
```

And the following error in the trace log:

```
com.ibm.websphere.security.WebTrustAssociationFailedException: Can not assert
user identity as LoggedIn user value is null
```

Solution

Refresh the browser 2 or 3 times. A valid LTPA token is generated.

For the server to communicate with a client in Simple transport security mode, a Master Secret Key is required. Sun JDK has an API that generates the Master Secret Key. However, IBM WebSphere contains the IBM JDK which does not have the API to generate the Master Secret Key.

7

Integrating Oracle Access Manager Identity Assertion with IBM WebSphere Portal

A portal provides a single point of access to enterprise data and applications by presenting a unified and personalized view of that information to employees, customers, and business partners.

This chapter describes how to use the Oracle Access Manager Identity Assertion Provider with IBM WebSphere Portal v7. It includes the following topics:

- Integrating IBM WebSphere Portal with Oracle Access Manager
- Supported Versions and Platforms
- Integrating IBM WebSphere Portal v7.0 with Oracle Access Manager
- Configuring a Stand Alone LDAP Registry for OAM in IBM WebSphere

> **See Also:** Chapter 6, "Managing Oracle Access Manager Identity Assertion on IBM WebSphere" in the *Oracle Fusion Middleware Third-Party Application Server Guide*, which contains much of the information you need to set up IBM WebSphere.

7.1 Integrating IBM WebSphere Portal with Oracle Access Manager

The IBM WebSphere Portal Server runs on top of the IBM WebSphere Application Server (WAS) and uses the WAS security infrastructure to enforce access control. Integrating with the IBM WebSphere Portal provides the following Oracle Access Manager functionality for the portal:

- User and group management
- Password management
- Single sign-on (SSO) to the portal
- Unified logout between Oracle Access Manager, WAS, and the IBM WebSphere Portal

7.2 Supported Versions and Platforms

The same platforms and versions that are supported for Oracle Access Manager and the IBM WebSphere Application Server are supported with IBM WebSphere Portal.

> **Note:** In this chapter, IBM WebSphere Portal Server is abbreviated to IBM WebSphere Portal.

IBM WebSphere Portal v7.0 can be integrated with both:

- Oracle Access Manager 11*g*
- Oracle Access Manager 10*g*

For the latest support information, see:

http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-certification-100350.html

7.3 Integrating IBM WebSphere Portal v7.0 with Oracle Access Manager

Regardless of the Oracle Access Manager release you are integrating with the IBM WebSphere Portal v7.0, a series of installation and configuration steps must be performed as outlined here.

integrating IBM WebSphere Portal with Oracle Access Manager

1. Install IBM WebSphere Application Server and Portal Server as described in Section 9.2, "Installing Components for the Oracle Access Manager IAP for IBM WebSphere" in the *Oracle Fusion Middleware Third-Party Application Server Guide*:

   See Also: See the IBM WebSphere Portal Infocenter documentation for installation details.

2. **Provision Webgate:** Perform steps as described for:
   - **Oracle Access Manager 11g:** Section 9.5, "Provisioning and Configuring OAM 11*g* for the IAP and IBM WebSphere".
   - **Oracle Access Manager 10g:** Section 9.4, "Provisioning WebGate, Configuring OAM 10g (10.1.4.3) and the IAP for IBM WebSphere".
3. **Install Webgate:** Install Webgate as described in Section 9.6, "Installing the Required WebGate for the IHS Web Server".
4. **Prepare Login Form**: Use instructions in Section 9.8, "Preparing the Login Form for WebGate".
5. Configure IBM WebSphere Application Server for OAM SSO and the Portal Server Domain Profile as described in Section 9.9, "Configuring IBM WebSphere for OAM SSO and the IAP".
6. Configure a stand-alone LDAP registry for OAM within IBM WebSphere Portal Server, as described in this chapter: Section 7.4, "Configuring a Stand Alone LDAP Registry for OAM in IBM WebSphere".

7.4 Configuring a Stand Alone LDAP Registry for OAM in IBM WebSphere

This section describes how to configure a stand-alone LDAP registry for Oracle Access Manager within IBM WebSphere Portal Server.

To configure a stand alone LDAP registry for OAM in IBM WebSphere Portal

1. Locate the wp_security_<ldaptype>.properties file in the following path:

   was_portal_profile_dir/ConfigEngine/config/helpers/wp_security_<*ldaptype*>.properties

Here, <*ldaptype*> refers to the directory server type (vendor) in use with Oracle Access Manager. For example, for a Sun One directory server the file name is: wp_security_sunone.properties.

2. Open wp_security_<ldaptype>.properties for editing.
3. Update the following entries with values that reflect your deployment:

```
standalone.ldap.id=<ldap server id>

standalone.ldap.host=host id name
standalone.ldap.port=host port
standalone.ldap.bindDN= <LDAP bind DN>
standalone.ldap.bindPassword= ldappwd
standalone.ldap.serverId=<full DN of ldap admin user>

standalone.ldap.serverPassword=admin user password
standalone.ldap.realm=<realm name>
standalone.ldap.primaryAdminId=<full DN of ldap admin user>
standalone.ldap.primaryAdminPassword= admin user password
standalone.ldap.primaryPortalAdminId= admin user password
standalone.ldap.primaryPortalAdminPassword=oblix

standalone.ldap.primaryPortalAdminGroup=<full DN of admin group>
standalone.ldap.baseDN= <LDAP base DN>

standalone.ldap.et.group.objectClasses=group object class
standalone.ldap.personAccountParent=<ldap base DN>
standalone.ldap.groupParent=<ldap base DN>
```

4. Execute the following command to validate properties:

```
ConfigEngine.sh validate-standalone-ldap -DWasPassword=<admin user passwd>
-DparentProperties =<path to wp_security_<ldaptype>.properties>
```

5. Execute the following command to change the portal-file-based repository to the defined LDAP type.

```
ConfigEngine.sh wp-modify-ldap-security -DWasPassword=<admin user passwd>
-DparentProperties =<path to wp_security_<ldaptype>.properties>
```

6. Upon successful completion of steps 4 and 5, restart the IBM WebSphere Portal and Application Servers.

8

Managing Oracle Adaptive Access Manager on IBM WebSphere

Oracle Adaptive Access Manager is Oracle Identity Management's solution for web access real-time fraud detection and multi-factor online authentication security for the enterprise.

This chapter contains information about managing and integrating Oracle Adaptive Access Manager on IBM WebSphere and explains the particularities of some features on that platform. Only topics that apply specifically to IBM WebSphere are included in this chapter; those that apply to the Oracle WebLogic Server are not described here, but can be found in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*, *Oracle Fusion Middleware Developer's Guide for Oracle Adaptive Access Manager*, and *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.

This chapter contains the following sections:

8.1 Installing and Configuring Oracle Adaptive Access Manager on IBM WebSphere

Prior to configuring Oracle Adaptive Access Manager, you must have installed all the required components, including any dependencies, and configured the environment in preparation of the configuration tasks that follow.

Table 8–1 Installation Tasks

No.	Task	Information
1	Obtain the following software: ■ IBM WebSphere 7.0 and any required Fix Packs for the IBM WebSphere software ■ Oracle Database ■ Oracle Fusion Middleware Repository Creation Utility 11*g* Release 2 (11.1.2.1.0) ■ Oracle Identity and Access Management Suite 11*g* Release 2 (11.1.2.1.0)	For information on obtaining the required software, refer to Task 2: Obtain the Necessary Software Media or Downloads.
2	Create or update OAAM schema in a database using the Repository Creation Utility (RCU).	For information on creating schemas, see Task 3: Identify a Database and Install the Required Database Schemas.
3	Install the IBM WebSphere software.	For information on installing IBM WebSphere software, see Task 4: Install the IBM WebSphere Software.
4	Install Oracle Identity and Access Management Suite.	For instructions on installing Oracle Identity and Access Management on IBM WebSphere, see Chapter 2, "Installing and Configuring Oracle Identity and Access Management on IBM WebSphere" and *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.
5	Configure the Oracle Fusion Middleware components in a new IBM WebSphere cell.	A cell is a group of nodes in a single administrative domain that encompasses the entire management domain. Section 2.8.1, "General Information About Using the Configuration Wizard on IBM WebSphere" describes how to use the Configuration Wizard to configure your Oracle Fusion Middleware products in a simple IBM WebSphere cell. For complete information about using the Oracle Fusion Middleware Configuration Wizard, including information about adding servers and clusters to a cell, refer to the *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server*.
6	Perform Policy Re-association Changes	For information on performing re-association changes, see Task 9: Configure the Database Security Store.
7	Start the servers.	For information on starting the servers, see Starting the Servers.

8.1.1 Starting the Servers

After you finished configuring the Oracle Fusion Middleware software successfully, you can start the IBM WebSphere deployment manager, node, and servers. There are two methods for starting and stopping the servers in your IBM WebSphere cell:

- with Profile Scripts
- with Oracle Enterprise Manager Fusion Middleware Control

The following procedures show how to run the profile scripts and the sequence you must use to start the deployment manager, the node, and the servers in the cell. For more information about starting IBM WebSphere Servers, see Section 2.11, "Task 11: Start the IBM WebSphere Servers."

8.1.1.1 Starting the Deployment Manager

To start the deployment manager, navigate to the following directory in the IBM WebSphere home and enter the command:

```
(UNIX) WAS_HOME/profiles/deployment_mgr_profile_name/bin/startManager.sh
```

For example, on a UNIX operating system:

```
/opt/IBM/WebSphere/AppServer/profiles/Dmgr01/bin/startManager.sh
```

`WAS_HOME` is the path to the `AppServer` directory where IBM WebSphere is installed.

8.1.1.2 Synchronizing Nodes

A node is an administrative grouping of application servers for configuration and operational management within one operating system instance. To synchronize nodes, navigate to the following directory in the IBM WebSphere home and enter the command:

```
(UNIX) WAS_HOME/profiles/profile_name/bin/syncNode.sh
DMGR_HOST DMGR_SOAP_PORT -username admin_user -password admin_password
```

8.1.1.3 Starting the Node

To start the node, navigate to the following directory in the IBM WebSphere home and enter the command:

```
(UNIX) WAS_HOME/profiles/profile_name/bin/startNode.sh
```

For example, to start the node on a UNIX operating system, enter:

```
/opt/IBM/WebSphere/AppServer/profiles/Custom01/bin/startNode.sh
```

8.1.1.4 Starting the OracleAdminServer

To start the OracleAdminServer, navigate to the following directory in the IBM WebSphere home and enter the command:

```
(UNIX) WAS_HOME/profiles/profile_name/bin/startServer.sh OracleAdminServer
```

For example, to start the OracleAdminServer on a UNIX operating system, enter:

```
/opt/IBM/WebSphere/AppServer/profiles/Custom01/bin/startServer.sh
OracleAdminServer
```

After you start the OracleAdminServer, you can start the other servers using the IBM WebSphere Administrative Console or Oracle Enterprise Manager Fusion Middleware Control. For more information, see Section 3.1, "Summary of the Oracle Fusion Middleware Management Tools on IBM WebSphere".

8.1.1.5 Starting the Managed Server Hosting OAAM Administration Server Using Scripts

To start the OAAM Administration Server, navigate to the following directory in the IBM WebSphere home and enter the command:

```
(UNIX) OAAM_PROFILE/bin/startServer.sh oaam_admin_server1
```

The default server name for the OAAM Administration Server is `oaam_admin_server1`.

For example, to start the OAAM Administration Server on a UNIX operating system, enter:

```
/opt/IBM/WebSphere/AppServer/profiles/Custom01/bin/startServer.sh
oaam_admin_server1
```

`OAAM_PROFILE` is the IBM WebSphere profile where the OAAM Administration Server is installed.

For example, on the UNIX operating system, `OAAM_PROFILE` is:

```
WAS_HOME/profiles/profile_name
```

8.1.1.6 Starting the Managed Server Hosting the Oracle Adaptive Access Manager Runtime Server Using Scripts

To start the OAAM Server, navigate to the following directory in the IBM WebSphere home and enter the command:

```
(UNIX) OAAM_PROFILE/bin/startServer.sh oaam_server_server1
```

The default server name for the OAAM runtime server is `oaam_server_server1`.

`OAAM_Profile` is the IBM WebSphere profile where OAAM Server is installed.

For example, to start the OAAM runtime server on a UNIX operating system, enter:

```
/opt/IBM/WebSphere/AppServer/profiles/Custom01/bin/startServer.sh
oaam_server_server1
```

For example, on a UNIX operating system, `OAAM_PROFILE` is:

```
WAS_HOME/profiles/profile_name
```

8.1.2 Stopping the Servers

You can use profile scripts or Oracle Enterprise Manager Fusion Middleware Control to stop the servers in a cell you configured for Oracle Fusion Middleware.

8.1.2.1 Stopping IBM WebSphere Servers with Fusion Middleware Control

You can also stop IBM WebSphere servers from Fusion Middleware Control.

For example, to stop a server from Fusion Middleware Control:

1. Log in to Fusion Middleware Control.

   To locate the port number and URL for Fusion Middleware Control, see Section 3.1.2.2, "Locating the Port Number and URL for Fusion Middleware Control."

2. Navigate to the Server home page.

For more information, see Section 3.1.2.5, "Viewing an IBM WebSphere Server from Fusion Middleware Control".

3. From the **WebSphere Application Server** menu, select **Control**, and then select **Shut down**.

   Fusion Middleware Control displays a confirmation dialog box.

4. Click **Shutdown**.

8.1.2.2 Stopping IBM WebSphere Servers with Profile Scripts

You can also stop IBM WebSphere servers with profile scripts.

To stop the IBM WebSphere servers, navigate to the following directory in the IBM WebSphere home and enter the command:

```
(UNIX) OAAM_PROFILE/bin/stopServer.sh server_name
-username username -password password
```

8.1.2.3 Stopping the OracleAdminServer

To stop the OracleAdminServer, navigate to the following directory in the IBM WebSphere home, and enter the command:

```
(UNIX) WAS_HOME/profiles/profile_name/bin/stopServer.sh OracleAdminServer
```

For example, on a UNIX operating system:

```
/opt/IBM/WebSphere/AppServer/profiles/Custom01/bin/stopServer.sh
OracleAdminServer
```

8.1.2.4 Stopping the Node

To stop the node, navigate to the following directory in the IBM WebSphere home and enter the command:

```
(UNIX) WAS_HOME/profiles/profile_name/bin/stopNode.sh
-username admin_user -password admin_password
```

For example, on a UNIX operating system:

```
/opt/IBM/WebSphere/AppServer/profiles/Custom01/bin/stopNode.sh
-username admin_user -password admin_password
```

8.1.2.5 Stopping the Deployment Manager

To stop the deployment manager, navigate to the following directory in the IBM WebSphere home and enter the command:

```
(UNIX) WAS_HOME/profiles/deployment_mgr_profile_name/bin/stopManager.sh
-username admin_user -password admin_password
```

For example, on a UNIX operating system:

```
/opt/IBM/WebSphere/AppServer/profiles/Dmgr01/bin/stopManager.sh
-username admin_user -password admin_password
```

8.1.3 Creating User with Privileges to Log into the OAAM Administration Console

By default OAAM does not provide a user that has the privileges to log in to the OAAM Administration Console. You must create a user and then grant the necessary groups to the user. Users and groups should be defined in the LDAP directory configured for the IBM WebSphere cell. For details on creating user and groups in IBM WebSphere, see the IBM WebSphere documentation.

Table 8–2 lists the default roles shipped with Oracle Adaptive Access Manager.

Table 8–2 OAAM Roles

Role Name	Roles	Descriptions
OAAMCSRGroup	CSRGroup role	Support Personnel
OAAMCSRManagerGroup	CSRManagerGroup role	Support Personnel
OAAMInvestigatorGroup	Investigator role	Investigators
OAAMInvestigationManagerGroup	InvestigationManager role	Investigators
OAAMRuleAdministratorGroup	RuleAdministratorsGroup role	Security Administrators
OAAMEnvAdminGroup	EnvAdminGroup role	System Administrators
OAAMSOAPServicesGroup	SOAPServicesGroup role	Users are granted this role in order to access the URL: `/oaam_server/services`

8.1.4 Setting Up the CLI Environment for OAAM on IBM WebSphere

The OAAM Command-Line Interface (CLI) scripts enable users to perform various tasks instead of using the OAAM Administration Console.

Setting up the CLI environment involves the following tasks:

1. Set up the CLI work directory
2. Configure `oaam_cli.properties` for CLI script startup (optional).
3. Set environment variables.
4. Set up the Credential Store Framework (CSF) configuration
5. Set up the OAAM database credentials

8.1.4.1 Setting Up the CLI Work Directory

Copy the `CLI` directory to a working directory:

```
cp -r ORACLE_MW_HOME/IDM_HOME/oaam/cli /home/user/IBM/oaam.cli
```

Note: After you installed the Oracle Identity and Access Management suite, an Oracle Home directory for Oracle Identity and Access Management, such as `Oracle_IDM1`, was created under your Middleware Home.

8.1.4.2 Specifying Properties for CLI Script Startup (Optional)

The CLI scripts need the locations of the Middleware Home, IBM WebSphere installation, and IDM home on startup. You can specify these locations in `oaam_cli.properties`, set environment variables, or enter this information at the command line when prompted. If you want to set the locations in the `oaam_cli.properties` file, see Table 8–3.

Table 8–3 Setting Properties in oaam_cli.properties

Property	Environment Variable	Definition
oaam.adminserver.mw.home	`ORACLE_MW_HOME`	Location of your Middleware Home.
oaam.server.was.home	`WAS_HOME`	IBM WebSphere only. Location of your IBM WebSphere installation.
oaam.server.idm.home	`IDM_HOME`	IBM WebSphere only. Location of IDM home within `ORACLE_MW_HOME`

8.1.4.3 Setting Up Environment Variables

To set the Java Home and Middleware Home environment variables, follow these steps:

1. Set `JAVA_HOME` to the JDK in IBM WebSphere (*WAS_HOME*`/java`)

   ```
   set JAVA_HOME=DISK/IBM/WebSphere/Application-Server/java
   ```

2. Set `ORACLE_MW_HOME` to the Middleware directory (complete path to the `Oracle_Common` directory).

   You do not have to set the Middleware Home if you specified it in the `oaam_cli.properties` file.

8.1.4.4 Configuring OAAM Database Details with CSF with MBeans

A credential store is a repository that can hold user name and password combinations, symmetric keys, tickets, or public key certificates. Oracle Platform Security Services includes the Credential Store Framework (CSF), a set of APIs that applications can use to create, read, update, and manage credentials securely. OAAM uses the CSF APIs to access credentials. Credentials are stored in the CSF of the IBM WebSphere Server domain and managed using Oracle Enterprise Manager Fusion Middleware Control.

To be able to access the OAAM encryption keys stored in the CSF, you must configure the OAAM database details with CSF with MBeans. Navigate to the working directory where you copied the `cli` directory and open the file `conf/bharosa_properties/oaam_cli.properties` in a text editor and then set the following properties:

Property Name	Notes about Property Value
oaam.csf.useMBeans	`true` (Keep it as true)
oaam.adminserver.hostname	Hostname where IBM WebSphere Administration Server runs.

Property Name	Notes about Property Value
oaam.adminserver.port	Port number of the IBM WebSphere Administration Server's Deployment Manager's `ORB_LISTENER_ADDRESS`. Usually it is `9100`. Log in to the IBM WebSphere Administrative Console. Navigate to **System Administration** > **Deployment Manager** > **Configuration** > **Additional Properties** > **Ports** > **ORB_LISTENER_ADDRESS**.
oaam.was.client.sasPropFile	Absolute path to the `sas.client.properties` file that is required for IBM WebSphere's remote JMX client: `/home/user/IBM/oaam.cli/sample.sas.client.properties`
oaam.db.url	Specify a valid JDBC URL of the OAAM database.
oaam.adminserver.type	`was`
oaam.db.additional.properties.file	Leave this field blank if there are no additional Oracle TopLink properties. Otherwise, specify the name of the properties file that has additional Oracle TopLink properties. Make sure the file is in the same directory as the `oaam_cli.properties` file.
oaam.db.driver	`oracle.jdbc.driver.OracleDriver` Change this value only if the OAAM schema is in a non-Oracle database.
oaam.db.min.read-connections	`1` Do not change this value unless required.
oaam.db.max.read-connections	`25` Do not change this value unless required.
oaam.db.min.write-connections	`1` Do not change this value unless required.
oaam.db.max.write-connections	`25` Do not change this value unless required.

8.1.4.5 Setting Up OAAM Database Credentials

OAAM CLI needs to retrieve two symmetric keys (generated automatically the first time the administration server is started) and database credential key (OAAM database schema and password. For example: `DEV_OAAM/`*password*) from CSF to connect to the database schema. After successful installation of OAAM, the symmetric keys should already be created, and only OAAM database schema and password credential need to be added to CSF through Fusion Middleware Control.

Steps to verify and add credentials are as follows:

1. Log in to Fusion Middleware Control with administrator access.

To locate the port number and URL for Fusion Middleware Control, see Section 3.1.2.2, "Locating the Port Number and URL for Fusion Middleware Control."

2. In the navigator, expand **WebSphere Cell**, right-click **Cell_WebSphere** and select **Security** > **Credentials**.
3. Expand the map named **oaam**. Two keys `DESede_db_key_alias` and `DESede_config_key_alias` should already exist.
4. Click the **Create Key** button, select **oaam** as map, provide `oaam_db_key` as the key name and provide a OAAM database schema user name and password.
5. Click **OK** to create the key in CSF.

For information on the credential store, see *Oracle Fusion Middleware Application Security Guide*.

8.1.4.6 Running CLI Commands

Once the setup is complete, you can run CLI commands from your CLI working directory. The commands are the same as in the Oracle WebLogic deployment.

A example of running a command is shown as follows:

1. Execute the CLI export command from *CLI_DIR*:

   ```
   sh runImportExport.sh -action export -module properties
   ```

   You will be prompted to select Oracle WebLogic or IBM WebSphere.
2. Select `2` for IBM WebSphere.

   You are prompted for the `WAS_HOME`.

   You are not prompted for `WAS_HOME` if it is already set in the `oaam_cli.properties` file.
3. Set the `WAS_HOME`.

   For example: `/opt/IBM/WebSphere/AppServer`
4. You will be prompted for the `IDM_HOME` directory if `$IDM_HOME` is not set. Give the same ORACLE HOME directory that you gave when installing the Oracle Identity and Access Management Suite. For example: `Oracle_IDM1`. By default, `IDM_HOME` will be set to `Oracle_IDM1`.

   If you specified the `IDM_HOME` in the `oaam_cli.properties` file, you will not be prompted for the `ORACLE_HOME` directory.

8.2 Installing and Configuring Oracle Adaptive Access Manager Offline on IBM WebSphere

To install and configure Oracle Adaptive Access Manager Offline, follow the steps outlined in Section 8.1, "Installing and Configuring Oracle Adaptive Access Manager on IBM WebSphere." You must also perform the following steps on the command line from the `IDM_ORACLE_HOME` directory before running `was_config.sh`:

```
cd IDM_ORACLE_HOME/oaam/oaam_offline/ear
mv oaam_offline_was.ear oaam_offline_was.ear.bak
mkdir temp
cd temp
jar xf ../oaam_offline_was.ear
mkdir war_tmp
```

```
cd war_tmp
jar xf ../oaam_offline.war
cp IDM_ORACLE_HOME/oaam/oaam_libs/was_native_jar/antlr-2.7.6.jar ./WEB-INF/lib
jar -cfm ../oaam_offline.war META-INF/MANIFEST.MF  *
cd ..
rm -rf war_tmp
jar -cfm ../oaam_offline_was.ear META-INF/MANIFEST.MF *
cd ..
rm -rf temp
```

If you do not perform this step, the OAAM Offline environment will not be able to process rules. The offline logs will show the following error:

```
java.lang.NoClassDefFoundError: antlr.TokenStream
```

8.3 Setting Up the Import and Export Feature in CLI

To set up the snapshot import and export feature, copy the `org.eclipse.persistence_1.1.0.0_2-1.jar` file from the `$ORACLE_MW_HOME/modules` directory of an Oracle Identity Management installation on Oracle WebLogic to the `$ORACLE_MW_HOME/modules` directory on IBM WebSphere.

If the `$ORACLE_MW_HOME/modules` directory does not exist on IBM WebSphere, you must create it before copying the file.

If you do not have an Oracle Identity Management deployment in an Oracle WebLogic environment, download the `org.eclipse.persistence_1.1.0.0_2-1.jar` file from "Oracle TopLink Software Downloads" on Oracle Technology Network (OTN) at:

http://www.oracle.com/technetwork/middleware/toplink/downloads/index.html

8.4 Setting Up Reporting and Auditing for OAAM on IBM WebSphere

Set up and enable audit using the database audit store for OAAM on IBM WebSphere. Instructions are provided in this section.

8.4.1 Creating the Audit Schema Using RCU

To switch to a database as the permanent store for your audit records, make sure you have used the Repository Creation Utility (RCU) to create a database store for the audit data.

This section explains how to create the audit schema. Once the database schema is created, you can:

- Create a datasource to point to this schema
- Update the domain configuration to switch the audit data store for audit records

> **Note:** This discussion assumes that RCU and the database is already installed in your environment. For information on using RCU, see the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.

Before You Begin

Before you begin, make sure to collect the details on which database to use, along with the DBA credentials to use. You should be able to log in to the database using `DEV_IAU` or a corresponding user.

Configuring the Database Schema

Follow these steps to configure a schema for the audit data store:

1. Navigate to *RCU_HOME*`/bin` and execute the RCU utility.
2. Choose **Create** at the starting screen. Click **Next**.
3. Enter your database details and click **Next**.
4. Choose the option to create a new prefix.
5. Also, select **Audit Services** from the list of schemas.
6. Click **Next** and accept the tablespace creation.
7. Check for any errors while the schemas are being created.

8.4.2 Starting the IBM WebSphere Administrative Console

Start the deployment manager by entering the command:

`(UNIX)` *`WAS_HOME`*`/profiles/`*`deployment_mgr_profile_name`*`/bin/startManager.sh`

Log in to the IBM WebSphere Administrative Console.

8.4.3 Creating J2C Authentication Data

When configuring the data source, you must first create the J2C authentication data entry. The user ID and password required to access the database are specified in a J2C authentication data entry:

1. Click **Security** > **Global Security** > **Java Authentication and Authorization Service** and select **J2C authentication data**.
2. Click **New** and create the authentication data using the `IAU` schema user name and password.

   For example: `DEV_IAU/`*`password`*.

   You can enter any valid string for the J2C authentication alias.
3. Click **OK**.
4. Stop all application servers from **Servers** > **Server types** > **WebSphere application servers**.
5. Stop the node.

   For example:

   `WAS_HOME``/profiles/`*`Custom01`*`/bin/stopNode.sh`
6. Synchronize the node.

   For example:

   `WAS_HOME``/profiles/`*`Custom01`*`/bin/syncNode.sh `*`host soap-port`*
7. Stop the deployment manager.

For example:

```
WAS_HOME/profiles/Dmgr01/bin/stopManager.sh
```

8. Start the deployment manager.

   For example:

   ```
   WAS_HOME/profiles/Dmgr01/bin/startManager.sh
   ```

9. Start the node.

   For example:

   ```
   WAS_HOME/profiles/Custom01/bin/startNode.sh
   ```

8.4.4 Create Data Sources for Audit Event

As explained in Section 8.4.1, "Creating the Audit Schema Using RCU", after you create a database schema to store audit records in a database, you must set up IBM WebSphere Server audit data sources that points to that schema. To do so:

1. Use the IBM WebSphere Administrative Console to create an IBM WebSphere data source for the OAAM database.

   Log in to your IBM WebSphere Administrative Console:

   ```
   http://host:port/ibm/console
   ```

2. In the left-hand panel of the IBM WebSphere Administrative Console, under the Resource section, expand **JDBC** and click **Datasource**.

3. Select the Cell scope and click **New**.

   You will then be prompted to name the data source for display purposes in the IBM WebSphere Administrative Console as well as provide a JNDI name that the data source will be bound to.

4. Create a data source with JNDI named `jdbc/AuditDB` for each scope of `OracleAdminServer`, `oaam_admin1`, and `oaam_server1`.

   - For the audit data source of `oaam_admin1`, select **OAAM_ADMIN_DB** as the data provider and in **Component-managed authentication alias**, select the J2C authentication identity created in Section 8.4.3, "Creating J2C Authentication Data."
   - For the audit data source of `oaam_server1`, select **OAAM_SERVER_DB** as the data provider and in **Component-managed authentication alias**, select the J2C authentication identity created in Section 8.4.3, "Creating J2C Authentication Data."
   - For the audit data source of `OracleAdminServer`, select **mds-oaam** as the data provider and in **Component-managed authentication alias**, select the J2C authentication identity created in Section 8.4.3, "Creating J2C Authentication Data."

5. Synchronize the nodes before testing the connection of the newly created data sources. Follow the steps in Section 8.1.1.2, "Synchronizing Nodes."

6. Test each connection by selecting the data source and clicking the **Test Connection** button. Ensure the test connection of all three newly created data sources are successful.

8.4.5 Set Audit Repository Using wsadmin Script

To set up the audit repository, follow these steps:

1. Start the Oracle Fusion Middleware wsadmin command-line shell from the common/bin directory of the OAAM Oracle home.

   Use this command syntax:

   ```
   (UNIX) IDM_HOME/common/bin/wsadmin.sh
         -profileName profilename
         -connType SOAP
         -user admin_user
         -password admin_password
   ```

   For example:

   ```
   sh $IDM_HOME/common/bin/wsadmin.sh -profileName Dmgr01 -connType SOAP -user
   wasadmin -password welcome1
   ```

2. Run the wsadmin command to set the audit repository to the database:

   ```
   wsadmin>Audit.setAuditRepository(switchToDB='true',dataSourceName='jdbc/AuditDB
   ',interval='14')
   ```

   The value of `dataSourceName` should be the JNDI name of the data sources that you create in Section 8.4.4, "Create Data Sources for Audit Event."

   If successful, the following information will be displayed:

   ```
   Audit Repository Information updated
   Server has to be restarted
   ```

3. Stop and restart nodes and managers. Start all application servers from **Servers** > **Server types** > **WebSphere application servers**.

8.4.6 Set Audit Policy in Fusion Middleware Control

From Fusion Middleware Control, you can manage the Oracle Fusion Middleware products that you have installed and configured as part of the IBM WebSphere cell.

The Audit Policy Settings page manages audit events. Each component and its events are organized in a tree structure under the Name column. The tree can be expanded to reveal the details of the events available.

1. Log in to Fusion Middleware Control.

   To locate the port number and URL for Fusion Middleware Control, see Section 3.1.2.2, "Locating the Port Number and URL for Fusion Middleware Control."

2. In the Fusion Middleware Control Target Navigation Pane, navigate to **WebSphere Cell** > **Cell_WebSphere**.

3. Right click **Cell_WebSphere** and navigate to **Security** > **Audit Policy Settings**. The Audit Policy Settings page appears.

4. A drop-down list of pre-configured audit levels can be selected. Two predefined levels (**Low**, **Medium**) will automatically pick up a subset of the audit events for all the components. In most cases, the predefined levels are sufficient.

   - **None** - No events are selected for audit.

- **Low** - A small set of events is selected, typically those having the smallest impact on component performance.
- **Medium** - This is a superset of the "Low" set of events. These events may have a higher impact on component performance.
- **Custom** - This level enables you to fine-tune the policy. The table shows the applications running in the domain.

The table consists of these columns:

- **Name** - shows components and applications in the domain.
- **Enable Audit** - shows whether the corresponding event type is being audited. This column is greyed out unless the Custom audit policy is in force.
- **Filter** - shows any filters in effect for the event type.

5. Enable audit events and click **Apply**.
   - To customize the audit policy, use the **Custom** option from the drop-down. This allows you to select all the events or hand-pick the appropriate subset as desired by checking the relevant boxes under the **Enable Audit** column. When you choose the Custom level, an optional filter available for success and failure outcomes of each individual event to further control how they are audited
   - Filters are rule-based expressions that you can define to qualify or filter events for audit. The expressions are based on attributes of the event. For example, a Login type event could specify an initiator as a user filter in which case the event would generate an audit record whenever the specified user logged in.

     A pencil icon indicates that a filter is available for the corresponding event.

     Click the icon to bring up the Edit Filter dialog.
   - Click the **Select Failures Only** button to select only failed events in the policy - for example, a failed authentication. The Enable Audit box is now checked for failed events.
   - **Import/Export** - These buttons enable you to save and re-use a policy configuration. At any time while editing the policy, click **Export** to save the current settings to a file, and **Import** to load the settings from a saved file.
   - Optionally, under **Users to Always Audit**, you can specify a comma-separated list of users to force the audit framework to audit events initiated by these users; auditing occurs regardless of the audit level or filters that have been specified.
   - If you made any policy changes, click **Apply** to save the changes.

     Click **Revert** to discard any policy changes and revert to the existing policy.
6. Stop all relevant application servers, stop node, synchronize node, stop manager, start manager, start node, start application servers/applications.

8.5 Moving OAAM from a Test to a Production Environment

This section describes how to move OAAM from a test environment to a production environment. You can develop and test policies and rules in a test environment, and then eventually roll out the new policies and rules and, optionally, test data to your production environment.

Moving OAAM components minimizes the amount of work that would otherwise be required to reapply all the customizations and configuration changes made in one environment to another. You can install, configure, customize, and validate OAAM in a test environment. Once the system is stable and performs as desired, you can create the production environment by moving a copy of the components and their configurations from the test environment, instead of redoing all the changes that were incorporated into the test environment. If there is an existing production environment, you can move any modifications of the test environment, such as customizations, to the production environment.

To move Oracle Adaptive Access Manager to a new production environment follow the guidelines below. Manually update the production environment for the items pertaining to your deployment.

8.5.1 Exporting the Snapshot from the Test Environment

A snapshot is a backup of the current system configuration. Use the OAAM CLI, export the OAAM snapshot from the test environment. Examples of exporting a snapshot are shown below:

```
runImportExport.sh -action export -module snapshot -snapshotname
"name of snapshot" -description "snapshot description"
```

```
runImportExport.sh -action export -module snapshot -snapshotname
"OAAM Snapshot" -description "OAAM snapshot description"
```

`-snapshotname` , `-description` options are optional. If `snapshotname` is specified then the exported zip file name will be the *value passed for -snapshotname.*`zip`. If `snapshotname` is not specified, the CLI will create a unique filename with a name such as `snapshot`_*unique_value*.

The exported zip file would also contain one `snapshot.properties` file that has the following content:

Property	Description
serverIP	IP of server from where CLI is run
user	Operating system user name
name	Name of snapshot, if specified by `-snapshotname`; if not specified, it will be a system-generated unique name
description	Description of snapshot, if specified by `-description`; if not specified, it will be system generated unique name
serverName	Hostname from where CLI was run

8.5.2 Exporting Individual Configurations

If you are moving some configurations to the production environment and do not need to export the entire snapshot, use the OAAM Administration Console to export the configurations of individual components to a zip file. For information, see the specific chapters in Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager for information on exporting.

Table 8–4 lists the OAAM configurations you can export individually.

Table 8–4 Individual Configurations Export

Configurations	Information
Policies	See "Exporting a Policy" in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.
Rule conditions	See "Exporting a Condition" in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.
Patterns	See "Exporting Patterns" in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.
Configurable actions	See "Exporting Action Templates" in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.
Transaction definitions	See "Exporting Transaction Definitions" in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.
Entities	See "Exporting Entities" in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.
KBA questions	See "Exporting Questions" in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.
KBA validations	See "Exporting Validations" in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.

8.5.3 Backing Up the Production Environment

Before you import the snapshot into a production system, you should be aware that you are about to replace the entire system configuration in the production system. You should create a snapshot of the current production environment before the import since you do not want to lose the current configuration if the import fails or if there are other issues that you had not anticipated. After you have imported the snapshot, you cannot undo the import, but if there is a backup available, you can restore the original configuration into your system immediately should the import fail.

When you create a snapshot, all the configurations for the functional areas are selected, both active and disabled. For example, if there are ten policies within the policy set, and five are active and five are disabled, all policies, their configuration, and status information are included when the snapshot is created.

The status of the items are preserved on backup and restore. For example: disabled items remain disabled when you perform a backup and restore.

You cannot selectively select individual items to include in a snapshot or perform selective restoration. If you only want to include certain configurations in your snapshot, you can export them from their module, and import them back and then create the snapshot.

For snapshots, the metadata is stored with the following items:

Artifact	Comments	Additional Notes
Policy sets	Policy set overrides	No additional notes.
Policies	All policies	Trigger combinations are included.
Rule instances	All rule instances	No additional notes.
Conditions	All rule conditions	No additional notes.

Artifact	Comments	Additional Notes
Groups	Group definitions for all groups whether linked or not	Group members for alerts and actions only will be exported.
Patterns	All patterns	No additional notes.
Transaction definitions	All transaction definitions	No additional notes.
Entities	All entities whether linked or not	No additional notes.
Properties	Only the ones in the database	No additional notes.
Enums	Only the ones in the database	No additional notes.
Configurable actions		No additional notes.
Challenge questions	Includes validations, categories, and configurations (Answer Logic and others)	No additional notes.

Back up the configuration data in the database or file. For file, perform an export using CLI or the Universal Risk Snapshot feature.

8.5.4 Exporting Members of Groups

Snapshots do not include the members of any groups with the exception of actions and alerts. However the groups themselves are included in the snapshot. To back up group members, you must perform an export of groups that is separate from export of the snapshot. These group members must be imported using the **Group** user interface if needed

To export the groups, see "Exporting a Group" in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.

8.5.5 Importing the Snapshot into the Production Environment

Instructions to import a snapshot are provided below.

8.5.5.1 Importing a Snapshot Using Universal Risk Snapshot

To import a snapshot for use in the system using the Universal Risk Snapshot feature, follow the instructions below:

1. Open **System Snapshot** under **Environment** in the Navigation tree.

   The **System Snapshots Search** page is displayed.

2. Click the **Load from File** button.

   A Load and Restore Snapshot dialog appears. You are given the opportunity to back up your current system since importing a snapshot will overwrite what is in the current system.

3. If you want to keep a backup of your current system, select the **Back up the current system now** box, enter the name and notes for the backup, and click **Continue**.

   When the Load and Restore Snapshot dialog appears with a message that the current system has been successfully stored in the database, click **OK**.

   Then, the Load and Restore Snapshot page appears for you to choose a snapshot to load into the server so you can run the basic authentication flows.

4. If you are sure you do not want to back up your current configuration or you are importing the snapshot into an empty system, you can leave the dialog blank and click **Continue**.

   Since you did not choose to back up your system, you are given a warning that you are loading a new snapshot and the details of the metadata may be overwritten. If you decide to take a backup, you can click the **Back** button to take you to the previous page where you can provide details for a backup. If you want to proceed with the import, click **Continue**.

   The Load and Restore Snapshot page appears for you to choose a snapshot to load into the server so you can run the basic authentication flows.

5. Now that you are ready to load the snapshot, click the **Browse** button on the dialog in which you can enter the filename of the snapshot you want to load. A screen appears for you to navigate to the directory where the snapshot file is located. Click **Open**. Then, click the **Load** button to load the snapshot into the system.

   If you are loading the snapshot out of the box for the first time, the snapshot file, `oaam_base_snapshot.zip` is located in the `Oracle_IDM1/oaam/init` directory where the OAAM base content is shipped.

6. Click **OK**.

   Once the snapshot has been loaded, a summary of the snapshot is displayed.

   The Preview tab is available, in which you are given the option to do the following:

   - View the conditions, rules, policies, and so on, in the snapshot.
   - View the actions that are taken on the objects. For example, if you are loading a snapshot with configurable actions and there are no configurable actions in the system, the system disables the configurable actions.
   - Filter the objects to see only the updates, or only the changes, or only the additions, and so on.

     In general, you want to see all that changes in your system when you load the snapshot because it has the potential to invalidate all the content in your system or overwrite your existing metadata.

   The Update button is available so that you can update or change to another snapshot to view what the changes would be as compared to existing system snapshot.

   The items in the snapshot are not effective yet. Unless you click the **Restore** button, the items in the snapshot have not been applied.

7. To apply the snapshot, click **Restore**.

   Once you applied the snapshot, make sure it appears in the System Snapshots page. Perform a search to view all snapshots that loaded into the database. Click any snapshot to view it and click **Restore** to apply changes. You can use this feature to back up the system periodically and it will be stored in the memory of the database or a file or in both.

8.5.5.2 Importing the Snapshot Using CLI

To import the Snapshot using CLI, run the following command:

`runImportExport.sh -action import -module snapshot` *path to a valid snapshot zip file*

8.5.6 Importing Group Members into the Production Environment

Because snapshot imports only copies of action and alert group types, you must import the group members into the production environment. To import the groups into the production environment, see "Importing a Group" in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.

8.5.7 Copying Java Classes into the Production Environment

When the configurable actions are included with a snapshot, you must copy the Java classes to the specified directory after the snapshot creation so that the configurable actions are not broken when they are brought back into a system.

8.5.8 Creating a Custom Shared Library

Shared libraries are files used by multiple applications. The deployed applications use shared library files, so you must define shared libraries for the library files and associate the libraries with specific applications.

8.5.8.1 Copying Customized Files

Copy the following files and configure a custom shared library based on the instructions in this section:

- Any items customized in the OAAM server, such as headers, footers, cascading style sheets (CSS), and JavaScript
- Virtual Authentication Device (VAD) images, which are in a custom jar
- Properties files, resource bundles, and end-user JSP screens

8.5.8.2 Creating the Shared Library

To make library files available to multiple applications, create a shared library by following these steps:

1. Log in to the IBM WebSphere Administrative Console.
2. In the console navigation tree, expand **Environment** and then select **Shared libraries**.
3. In the Shared Libraries page, select **Show scope selection drop-down list with the all scopes option**, and then, select a shared library scope.
4. Above the table, under Preferences, click the **New** button to create a new shared library in the scope as selected in step 3.
5. In the **Name** field in the settings page of a shared library, specify a name for the shared library.
6. In the **Classpath** text box, specify the absolute paths of all the JAR files present in the following directory:

   MW_HOME/Oracle_IDM1/oaam/oaam_libs/*directory_name*.

   These are the paths that the product searches for classes and resources of the shared library.

Note: Each entry should be in a new line, do not use any path separator like ";" or ":".

7. Click **OK** and then **Save**.

8.5.8.3 Adding the Shared Library Reference to OAAM Admin and OAAM Server

To add the Shared Library reference to the application:

1. Log in to the IBM WebSphere Administrative Console.
2. In the console navigation tree, expand **Applications**, and then **Application Types** and click **WebSphere enterprise applications** to open the list of applications.
3. In the Enterprise Applications page, click the application that needs to refer to the shared library.
4. Under Configurations, click **Shared library references**.
5. In the Shared Library Mapping for Modules section, select the checkbox next to the application, and then click the **Reference Shared Libraries** button above the table.
6. Select the shared library from the **Available** list and move it to the **Selected** list.
7. Click **OK** and then **Save**.
8. Repeat steps 2 to 7.
9. Stop all application servers that reference the shared library.
10. Stop the node.
11. Synchronize the node.
12. Stop the deployment manager.
13. Start the deployment manager.
14. Start the node.
15. Start the application servers/applications that reference the shared library.

8.5.9 Recreating KBA and OTP Logic and Policy Overrides

Manually re-create the KBA logic, OTP logic, and policy set overrides using the OAAM Administration Console. For details on recreating these, see the *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.

8.5.10 Validating the Move Was Move Successful

Validate that the move was successful:

1. Log in to OAAM Administration Console.

   `http://oaam_managed_server_host:oaam_admin_managed_server_port/oaam_admin`

2. Navigate to Policies module and check that the rules and groups from the test environment exist in the production environment.
3. Navigate to the KBA module and check that the challenge questions in the test environment exist in the production environment.

4. Test the Web applications that were configured for Oracle Adaptive Access Manager. The user should be redirected to registration and challenge flow.

8.6 Integrating Juniper Networks Secure Access (SA) with OAAM

The integration of Juniper Networks Secure Access (SA) and Oracle Adaptive Access Manager provides enterprises with a remote access control solution with strong multi-factor authentication and advanced real time fraud prevention capabilities to enable secure access to an enterprise's applications.

To integrate Oracle Adaptive Access Manager and Juniper Networks Secure Access (SA) to use Oracle Adaptive Access Manager's Authentication and Forgot Password flows, refer to procedures in this section.

8.6.1 Juniper Networks Secure Access (SA) and OAAM Integration Roadmap

Table 8–5 lists a summary of the high-level tasks for integrating Oracle Adaptive Access Manager and Juniper SA.

Table 8–5 Juniper and OAAM Integration Flow

No.	Task	Information
1	Review prerequisites.	For information, refer to Juniper Integration for OAAM on IBM WebSphere Prerequisite.
2	Configure the authentication provider.	For information, refer to Configuring the Authentication Provider.
3	Configure Oracle Platform Security Services for authentication.	For information, refer to Configuring Oracle Platform Security Services (OPSS) for Integration.
4	Synchronize node and restart servers. 1. Stop the node. 2. Stop the deployment manager. 3. Start the deployment manager. 4. Synchronize the node. 5. Start the node. 6. Restart the OAAM servers.	For information, refer to Synchronizing the Node and Restarting the Servers.
5	Import server properties.	For information, refer to Importing the SAML Configuration-Related Server Properties Using the OAAM Administration Console.
6	Set up Certificate of Trust.	For information, refer to Setting Up Certificate for Signing the Assertion.
7	Modify integration properties.	For information, refer to Modifying Integration Properties Using the OAAM Administration Console.
8	Configure Juniper Networks Secure Access (SA)	For information, refer to Configuring Juniper Networks Secure Access (SA).

8.6.2 Juniper Integration for OAAM on IBM WebSphere Prerequisite

For this integration procedure, you will need to obtain the following software:

- IBM WebSphere 7.0 and any required Fix Packs for the IBM WebSphere software

  To obtain and install the IBM WebSphere software, refer to the IBM WebSphere documentation. For more information, see Section 1.4, "Documentation Resources for Using Oracle Identity and Access Management Suite Products on IBM WebSphere".

- Juniper configured per the Juniper documentation

  For more information on Juniper SA configuration, see the *Juniper Networks Secure Access Administration Guide* available at

  `http://www.juniper.net/techpubs`

- Oracle Adaptive Access Manager configured per *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management* and Chapter 2, "Installing and Configuring Oracle Identity and Access Management on IBM WebSphere."

8.6.3 Configuring the Authentication Provider

In the IBM WebSphere Application Server, a user registry or repository authenticates a user and retrieves information about users and groups to perform security-related functions, including authentication and authorization. The information about users and groups reside within a registry or repository. The IBM WebSphere Application Server makes access control decisions using the user registry or repository.

For information on switching from the IBM WebSphere user registry to use an LDAP user registry, see the IBM WebSphere documentation.

To configure the default identity store in the IBM WebSphere Application Server to point to LDAP, proceed as follows:

1. Prepare LDAP properties file with details of LDAP.

   For information on setting the required properties, see Section 9.1, "IBM WebSphere Identity Stores."

2. Navigate to the following directory:

   ```
   MW_HOME/oracle_common/common/bin
   ```

3. Set `JAVA_HOME` to the IBM JDK (bundled with IBM WebSphere):

   For example (for `csh`):

   ```
   setenv JAVA_HOME /opt/IBM/WebSphere/AppServer/java
   ```

4. Run the `WSADMIN` command to configure the identity store on IBM WebSphere.

   a. Invoke the `WSADMIN` shell by running the `wsadmin.sh` command:

      ```
      ./wsadmin.sh -port SOAP_PORT -user wasadmin_user -password password
      -conntype SOAP
      ```

      For example:

      ```
      ./wsadmin.sh -port port -user wasadmin -password pass -conntype SOAP
      ```

      You must use the same credentials that you provided while setting up the WAS cell.

b. Run the following WSADMIN command from the wsadmin prompt:

```
Opss.configureIdentityStore(propsFileLoc="/tmp/oud.properties") -
oud.properties
```

propsFileLoc specifies the location of the file that contains the property settings for the identity LDAP identity store. This command modifies the configuration file jps-config.xml to include the specifications in the property file.

The OPSS.configIdentityStore() command will change the default identity store in the IBM WebSphere Application Server to point to LDAP; therefore, to access the OAAM Administration Console, you must create the OAAM roles in LDAP and assign them to required users.

Once the OPSS.configIdentityStore() command is run you should see following entry in the jps.config.xml file. The entries may vary from the type of LDAP used and the content of properties file specified to the OPSS.configIdentityStore() command.

```
<serviceInstance name="idstore.ldap.0" provider="idstore.ldap.provider">
            <property name="subscriber.name"
value="dc=oaam,dc=us,dc=oracle,dc=com"/>
            <property name="bootstrap.security.principal.key"
value="bootstrap_idstore"/>
            <property name="idstore.type" value="OPEN_LDAP"/>
            <property name="ldap.url" value="ldap://example.host:1389"/>
            <property name="username.attr" value="uid"/>
            <property name="bootstrap.security.principal.map"
value="BOOTSTRAP_JPS"/>
            <extendedProperty>
                <name>user.search.bases</name>
                <values>
                    <value>ou=users,dc=oaam,dc=us,dc=oracle,dc=com</value>
                </values>
            </extendedProperty>
            <extendedProperty>
                <name>group.search.bases</name>
                <values>
                    <value>ou=groups,dc=oaam,dc=us,dc=oracle,dc=com</value>
                </values>
            </extendedProperty>
            <extendedProperty>
                <name>user.object.classes</name>
                <values>
                    <value>person</value>
                </values>
            </extendedProperty>
        </serviceInstance>
    </serviceInstances>
```

8.6.4 Configuring Oracle Platform Security Services (OPSS) for Integration

Oracle Platform Security Services (OPSS) provides enterprise product development teams, systems integrators (SIs), and independent software vendors (ISVs) with a standards-based, portable, integrated, enterprise-grade security framework for Java Standard Edition (Java SE) and Java Enterprise Edition (Java EE) applications. For information on OPSS, see *Oracle Fusion Middleware Application Security Guide*.

1. On the machine where OAAM is installed, navigate to the `was_profile_dir/config/cells/cell_name/fmwconfig` directory.

   For example:

   ```
   /scratch/xyz/IBM/WebSphere/AppServer/profiles/Dmgr04/config/cells/abc1234567Cell02/fmwconfig
   ```

2. Back up `jps-config.xml`.
3. Open the `jps-config.xml`.
4. Before closing tag `</jpsContexts>` add the following jps context:

   ```
   <!-- This context is used for OAAM Juniper Integration -->
   <jpsContext name="idcontext">
   <serviceInstanceRef ref="user.authentication.loginmodule"/>
   <serviceInstanceRef ref="idstore.ldap"/>
   <serviceInstanceRef ref="credstore"/>
   <serviceInstanceRef ref="keystore"/>
   <serviceInstanceRef ref="policystore.xml"/>
   <serviceInstanceRef ref="audit"/>
   </jpsContext>
   ```

   Change `idstore.ldap` value to the server instance of the ID store that was created after running the OPSS script in Section 8.6.3, "Configuring the Authentication Provider."

   For example, change `idstore.ldap` to `idstore.ldap.0`.
5. Save the file and exit.

> **Note:** Once you save the file, you might want to use an XML editor to check that all the tags are correct. You can also open the file in Internet Explorer to see whether there are any tags missing. If your changes are correct you will be able to open the file successfully in Internet Explorer.

8.6.5 Synchronizing the Node and Restarting the Servers

Stop and start the IBM WebSphere Administration Server, OAAM Admin Server and OAAM Managed Server, since these changes requires a restart.

While stopping the node and deployment manager and synchronizing the node, use the IBM WebSphere administrator credentials.

In the example, `WAS_HOME` is `/opt/IBM/WebSphere/AppServer`.

1. Stop the node.

   For example:

   ```
   WAS_HOME/profiles/Custom01/bin/stopNode.sh -user admin_user -password admin_password
   ```

2. Stop the deployment manager.

   For example:

   ```
   WAS_HOME/profiles/Dmgr01/bin/stopManager.sh -user admin_user -password admin_password
   ```

3. Start the deployment manager.

For example:

```
WAS_HOME/profiles/Dmgr01/bin/startManager.sh
```

4. Synchronize the node.

   For example:

   ```
   WAS_HOME/profiles/Custom01/bin/syncNode.sh localhost 8879 -user admin_user
   -password admin_password
   ```

5. Start the node.

   For example:

   ```
   WAS_HOME/profiles/Custom01/bin/startNode.sh
   ```

6. Restart the OAAM servers.

   For example:

   ```
   WAS_HOME/profiles/Custom01/bin/startServer.sh oaam_admin_server

   WAS_HOME/profiles/Custom01/bin/startServer.sh oaam_server_server
   ```

Note: After restarting the servers, to login to IBM WebSphere Administrative Console, use the User ID and password as provided in the properties file that was given input to `opss.configIdentityStore()` command.

8.6.6 Importing the SAML Configuration-Related Server Properties Using the OAAM Administration Console

Import the SAML configuration-related properties so they are added in the OAAM database.

To import the SAML configuration-related properties, proceed as follows:

1. Log in to the OAAM Administration Console as a administrator with the EnvAdminGroup role. For example:

   ```
   http://oaam_managed_server_host:oaam_admin_managed_server_port/oaam_admin
   ```

2. In the Navigation pane, click **Properties** under the **Environment** node.
3. Click **Import Properties** in the Properties page to import server properties for the integration
4. Browse for `saml_properties.zip` in the *IDM_ORACLE_HOME*`/oaam/init` directory, and click **Open**, and then, click **Import**.

   Once the import is complete it will show you the properties successfully imported.

5. Click **Done** to complete the import.

   This will import the properties needed for the integration. You will modify these properties according to your environment in Section 8.6.8, "Modifying Integration Properties Using the OAAM Administration Console."

8.6.7 Setting Up Certificate for Signing the Assertion

A certificate authority (CA) is a trusted third-party that certifies the identity of third-parties and other entities, such as users, databases, administrators, clients, and servers. The certificate authority verifies the party identity and grants a certificate, signing it with its private key.

To set up the certificate of trust between Juniper SA and OAAM, follow the procedures contained in these sections:

- Creating Private Key for Certificate
- Creating a Certificate Request
- Acting as Your Own Certificate Authority
- Importing the Certificate into Your Keystore

8.6.7.1 Creating Private Key for Certificate

The first step is to create a private key for the certificate. To create this private key, proceed as follows:

1. Change the working directory to the security properties directory:

   ```
   WAS_HOME/java/jre/lib/security
   ```

   where WAS_HOME points to the `AppServer` directory:

   ```
   WAS_HOME=/opt/IBM/WebSphere/AppServer
   ```

2. Create the private key using a key and certificate management utility, called `keytool`. Enter the following command with `cacerts` as the keystore:

   ```
   WAS_HOME/java/jre/bin/keytool -genkey -keyalg rsa -validity 1825 -keysize 2048
   -alias OAAMCert -keystore cacerts -storepass changeit
   ```

3. Enter the details for the certificate.

   An example of the output is shown as follows:

   ```
   What is your first and last name?
     [Unknown]:  ag-oracle-oaam
   What is the name of your organizational unit?
     [Unknown]: Juniper
   What is the name of your organization?
     [Unknown]: Juniper
   What is the name of your City or Locality?
     [Unknown]:  Sunnyvale
   What is the name of your State or Province?
     [Unknown]:  CA
   What is the two-letter country code for this unit?
     [Unknown]:  US
   Is CN=ag-oracle-oaam, OU=Juniper, O=Juniper, L=Sunnyvale, ST=CA, C=US correct?
     [no]:  yes
   ```

   Note: Typically the `CN` of the certificate is the name of the machine.

4. When prompted, enter the keystore password:

   ```
   Enter key password for <OAAMCert>
           (RETURN if same as keystore password):
   ```

```
Re-enter new password:
```

Remember this password as it is needed later for the integration.

8.6.7.2 Creating a Certificate Request

After you create the private key and self-signed certificate, use the `keytool` command to generate a Certificate Signing Request (CSR):

1. Change the working directory to the security properties directory:

   ```
   WAS_HOME/java/jre/lib/security
   ```

   where `WAS_HOME` points to the `AppServer` directory:

   ```
   WAS_HOME=/opt/IBM/WebSphere/AppServer
   ```

2. Run this command to create the certificate request:

   ```
   WAS_HOME/java/jre/bin/keytool -certreq -alias OAAMCert -file server.csr
   -keystore cacerts -storepass changeit
   ```

   In this example you created a certificate request in a file called `server.csr`.

   The file is located in the *WAS_HOME*`/java/jre/lib/security` directory.

3. Copy the `server.csr` file into the `/etc/pki_jungle/myCA` directory.

   ```
   cp WAS_HOME/java/jre/lib/security/server.csr /etc/pki_jungle/myCA/
   ```

8.6.7.3 Submitting the Certificate Signing Request (CSR) to a Certificate Authority

Submit the Certificate Signing Request (CSR) to a certificate authority to obtain a digital certificate. The certificate authority will issue the certificate. You must receive the issued certificate and the root CA Certificate which signed the request.

For testing, you can act as your own certificate authority to sign the certificates. For production scenarios, a certificate from a certificate authority has to be used.

For production scenarios, you can skip Section 8.6.7.4, "Acting as Your Own Certificate Authority" and go to Section 8.6.7.5, "Importing the Certificate into Your Keystore" to import the certificate from an external certificate authority.

8.6.7.4 Acting as Your Own Certificate Authority

For testing purposes, you can act as your own certificate authority to self-sign the certificates. The following sets of instructions walk through setting up to self-sign the certificates. To set this up, proceed with the subsequent example.

8.6.7.4.1 Prerequisites The package `OpenSSL` must be installed in the machine you will use to manage your certificates or create the certificate requests. OpenSSL is an open source implementation of the Secure Sockets Layer (SSL) protocol. OpenSSL implements basic cryptographic functions and provides utility functions.

8.6.7.4.2 Creating the Necessary Directories To create the necessary directories, proceed as follows:

1. Create a directory where all certificate files will be kept. The default directory is `/etc/pki/tls/`. As root, issue the following command to create your own directories:

   ```
   # mkdir -m 0755 /etc/pki_jungle
   ```

2. Then, create the certificate authority's directory by issuing the commands:

```
# mkdir -m 0755 \
/etc/pki_jungle/myCA \
/etc/pki_jungle/myCA/private \
/etc/pki_jungle/myCA/certs \
/etc/pki_jungle/myCA/newcerts \
/etc/pki_jungle/myCA/crl
```

where

- `myCA` is your certificate authority's directory
- `myCA/private` is the directory where your private keys are placed. Be sure that you set restrictive permissions to all your private keys so that they can be read only by root, or the user with whose privileges a server runs. The consequences of a certificate authority private key being stolen would be catastrophic.
- `myCA/certs` directory is where your server certificates will be placed
- `myCA/newcerts` directory is where OpenSSL puts the created certificates in PEM (unencrypted) format and in the form `cert_serial_number.pem` (e.g., 07.pem). OpenSSL needs this directory, so you must create it
- `myCA/crl` is where your certificate revocation list is placed

8.6.7.4.3 Initial OpenSSL Configuration

1. To copy the default OpenSSL configuration file (`openssl.cnf`) from `/etc/pki/tls` to your certificate authority's directory and name it `openssl.my.cnf`, issue the following command as root:

```
# cp /etc/pki/tls/openssl.cnf /etc/pki_jungle/myCA/openssl.my.cnf
```

2. Since this file does not need to be world readable, you can change its attributes by issuing the command:

```
# chmod 0600 /etc/pki_jungle/myCA/openssl.my.cnf
```

3. Create the file that serves as a database for OpenSSL, by issuing the command:

```
# touch /etc/pki_jungle/myCA/index.txt
```

4. Create the file which contains the next certificate's serial number, by issuing the command:

```
# echo '01' > /etc/pki_jungle/myCA/serial
```

   Since certificates have not been created yet, set it to "`01`":

8.6.7.4.4 Creating the CA Certificate and Private Key After completing the initial configuration, you can now generate a self-signed certificate that will be used as your certificate authority's certificate to sign other certificate requests and a private key.

1. Change to your certificate authority's directory.

   As root, issue the OpenSSL command:

```
# cd  /etc/pki_jungle/myCA/
```

   This is where you should issue all OpenSSL commands since it is the location of your OpenSSL's configuration file (`openssl.my.cnf`).

2. Then, create your certificate authority's certificate and private key. As root, issue the following command:

```
# openssl req -config openssl.my.cnf -new -x509 -extensions v3_ca -keyout
private/myca.key -out certs/myca.crt -days 1825
```

This creates a self-signed certificate with the default CA extensions which are valid for five years.

3. When prompted for a passphrase for your certificate authority's private key, set a strong passphrase.

4. When prompted, provide information that will be incorporated into your certificate request. Information for the certificate authority is similar to the example that is shown:

```
Country Name (2 letter code) [GB]:GR
State or Province Name (full name) [Berkshire]:Greece
Locality Name (eg, city) [Newbury]:Thessaloniki
Organization Name (eg, company) [My Company Ltd]:My Network
Organizational Unit Name (eg, section) []:My Certificate Authority
Common Name (eg, your name or your server's hostname)
[]:server.example.com
Email Address []:whatever@server.example.com
```

Two files are created:

`certs/myca.crt`: This is your certificate authority's certificate and can be publicly available and world readable.

`private/myca.key`: This is your certificate authority's private key. Although it is protected with a passphrase, you should restrict access to it so that only root can read it.

5. Although your certificate authority's private key is protected with a passphrase, you should restrict access to it so that only root can read it. To do so, issue the following command:

```
# chmod 0400 /etc/pki_jungle/myCA/private/myca.key
```

8.6.7.4.5 More OpenSSL Configuration (Mandatory)

Modifications to `/etc/pki_jungle/myCA/openssl.my.cnf` are necessary because you use a custom directory for your certificates' management.

1. Open `openssl.my.cnf` in a text editor as root and find the following section (around line 35):

```
_____________________________________________________________
[ CA_default ]
dir                 = ../../CA              # Where everything is kept
certs               = $dir/certs            # Where the issued certs are kept
crl_dir             = $dir/crl              # Where the issued crl are kept
database            = $dir/index.txt        # database index file.
#unique_subject = no                        # Set to 'no' to allow creation
of
                                            # several certificates with same
subject.
new_certs_dir       = $dir/newcerts         # default place for new certs.
certificate         = $dir/cacert.pem       # The CA certificate
serial              = $dir/serial           # The current serial number
#crlnumber          = $dir/crlnumber        # the current crl number must be
```

```
                                                    # commented out to leave a V1 CRL
crl                = $dir/crl.pem                   # The current CRL
private_key        = $dir/private/cakey.pem         # The private key
RANDFILE           = $dir/private/.rand             # private random number file
x509_extensions = usr_cert                          # The extentions to add to the
cert
```

2. Modify the path values to conform to your custom directory and your custom certificate authority key (private key) and certificate and save your changes:

```
 [ CA_default ]
dir     = .                                 # <--CHANGE THIS
certs           = $dir/certs
crl_dir         = $dir/crl
database        = $dir/index.txt
#unique_subject = no
new_certs_dir   = $dir/newcerts
certificate     = $dir/certs/myca.crt       # <--CHANGE THIS
serial          = $dir/serial
#crlnumber      = $dir/crlnumber
crl             = $dir/crl.pem
private_key     = $dir/private/myca.key     # <--CHANGE THIS
RANDFILE        = $dir/private/.rand
x509_extensions = usr_cert
```

8.6.7.4.6 Signing the Certificate Request Now you will sign the certificate request and generate the server's certificate. To do so, proceed as follows:

1. First, copy the server.csr (created in Section 8.6.7.2, "Creating a Certificate Request") to your certificate authority's directory by issuing the following command:

   ```
   # cp WAS_HOME/java/jre/lib/security/server.csr /etc/pki_jungle/myCA/
   ```

2. Change to your certificate authority's directory by issuing the following command:

   ```
   # cd /etc/pki_jungle/myCA/
   ```

3. Then, sign the certificate request by issuing the command:

   ```
   # openssl ca -config openssl.my.cnf -policy policy_anything -out
   certs/server.crt -infiles server.csr
   ```

4. Supply the certificate authority's private key to sign the request. You can check the `openssl.my.cnf` file about what "policy_anything" means. In short, the fields about the `Country`, `State` or `City` are not required to match those of your certificate authority certificate.

   After the steps have been completed, two new files are created:

   - `certs/server.crt.`

     This is the server's certificate, which can be made available publicly.

   - `newcerts/01.pem`

     This is the same certificate, but with the certificate's serial number as a file name. It is not needed.

5. Now, delete the certificate request (`server.csr`) since it is no longer needed.

8.6.7.5 Importing the Certificate into Your Keystore

The SSL VPN must import the public key of this server certificate to decrypt the message sent from OAAM.

You must import the Root CA Certificate followed by the certificate which was issued to you by the certificate authority. The name of the root certificate is `myca.crt` and the name of the issued certificate is `server.crt`.

To import a certificate into a keystore, proceed as follows:

1. Change the working directory to this directory:

   ```
   WAS_HOME/java/jre/lib/security
   ```

   where `WAS_HOME` points to the `AppServer` directory:

   ```
   WAS_HOME=/opt/IBM/WebSphere/AppServer
   ```

2. Copy `/etc/pki_jungle/myCA/certs/myca.crt` and `/etc/pki_jungle/myCA/certs/server.crt` to the *WAS_HOME*`/java/jre/lib/security` directory.

3. Import the root certificate into your keystore using the following `keytool` command:

   ```
   WAS_HOME/java/jre/bin/keytool -importcert -alias rootCA -file myca.crt
   -keystore cacerts -storepass changeit
   ```

   In the preceding syntax:

   - `alias` represents the alias of the Root CA Certificate.
   - `rootCA` -file represents the name of the file that contains the Root CA Certificate.
   - `keystore` represents the name of your keystore.

4. Open `server.crt` in a text editor and remove everything except for content between the `BEGIN CERTIFICATE` and `END CERTIFICATE` tags.

5. Import the issued certificate into your keystore using the following `keytool` command:

   ```
   WAS_HOME/java/jre/bin/keytool -importcert -alias OAAMCert -file server.crt
   -keystore cacerts -storepass changeit
   ```

   In the preceding syntax:

   - `alias` represents the alias of the certificate, which must be the same as the private key alias assigned in Section 8.6.7.1, "Creating Private Key for Certificate."
   - `server.crt` represents the name of the file that contains the certificate.
   - `keystore` represents the name of your keystore.

6. Enter the key password for `<OAAMCert>`.

   The certificate reply was installed in keystore.

   Note: Ensure that the alias is the same as the one you used when creating the request.

8.6.8 Modifying Integration Properties Using the OAAM Administration Console

To define the SAML configuration properties required to establish the integration, proceed as follows:

1. Log in to the OAAM Administration Console.

   `http://oaam_managed_server_host:oaam_admin_managed_server_port/oaam_admin`

2. Double-click **Properties** to open the Properties page.
3. Now type `oracle.saml*` in the Name field and click **Search** to search the integration properties.
4. In the search results, click the property you must modify.
5. In the Properties tab, modify the value for the property and click **Save**.

The properties imported as part of integration that need to be modified are shown in Table 8–6, " SAML Integration Properties".

Table 8–6 SAML Integration Properties

Property	Description
oracle.saml.integration.version	The SAML version used for integration Possible values are `1.1` and `2.0`. The default value is `1.1`. Juniper SA also supports SAML2.0. You must decide the version of SAML to use.
oracle.saml.target.default.url	The target URL (homepage) the user wants to navigate to after successful SAML assertion validation by Juniper SA For example: `https://ag-oracle-oaam.juniperlabs.local/`
oracle.saml.keystore	The full path of the keystore used for storing the certificate required to sign the assertion. In our case it will be `<MW_HOME>/jdk160_18/jre/lib/security/cacerts`
oracle.saml.keystore.password	The password of the keystore
oracle.saml.keystore.certalias	The alias of the certificate used for assertion
oracle.saml.keystore.privatekeypassword	The private key password
oracle.saml.redirect.post.url	The URL where SAML assertion is posted For example: `https://ag-oracle-oaam.juniperlabs.local/dana-na/auth/saml-consumer.cgi`
oracle.saml.set.attributes	Indicates if additional attributes need to be sent to the Juniper SA as part of the assertion Possible values are `false` or `true`. The default value is `false`
oracle.saml.user.attributes	List of attributes required to be appended as part of the assertion The property is only used if `oracle.saml.set.attribute` is set to `true`

Table 8–6 (Cont.) SAML Integration Properties

Property	Description
oracle.saml.attribute.namespace	The name of the namespace used for assertion. The default vale is `JuniperNS`. For SAML1.1 only.
oracle.saml.nameidformat	The `nameid` format used in the SAML assertion The default value is `X509SubjectName`.
oracle.saml.nameidattribute	The `NameID` attribute which identifies the user in the SAML assertion The default value is `distinguishedName`. This must be `distinguishedName` if the `nameid` format is set to `X509SubjectName`.
oracle.saml.issuer.url	The URL of the issuer of SAML This is the machine where the OAAM authentication server is running. For example: `http://abcdefgh.example.com:14300`
oracle.oaam.juniper.intg.jps.context	Optional Context name to be used default is (`idcontext`).

8.6.9 Configuring Juniper Networks Secure Access (SA)

To configure Juniper for this integration, you must:

- Creating SAML 1.1 Authentication Server
- Creating a User Realm for SAML
- Creating a Sign-In Policy

For more information on Juniper SA configuration, see the *Juniper Networks Secure Access Administration Guide* available at

`http://www.juniper.net/techpubs`

8.6.9.1 Creating SAML 1.1 Authentication Server

You must create an Authentication Server in Juniper SA. To do so, proceed as follows:

1. Log in to your Juniper SSL VPN Administrator Console.
2. In the Juniper Administration Console in the left pane, expand the **Authentication** menu, and then click **Auth. Servers**.
3. From the **New** drop-down list, select **SAML Server**, and then click **New Server**.
4. Define the Authentication Server with the values in Table 8–7.

Table 8–7 Create an Authentication Server

Parameter	Details	Value
Server Name	Name of SAMLServer	OAAM SAML 1.1 Enter same value as shown.
SAML Version	SAML version for authentication server	1.1 Enter same value as shown.

Table 8–7 (Cont.) Create an Authentication Server

Parameter	Details	Value
Source Site Inter-Site Transfer Service URL	The entry URL of OAAM server. This is where the user gets redirected to for authentication.	Example: @ `https://ag-oracleoaam.acmegizmo.com:14301/oaam_server/juniperLoginPage.jsp` Specify the host and port according to your environment.
User Name Template	The template used for extracting the value that identifies the authenticated user	`<assertionNameDN.cn>` Enter the same value as shown. You also need to keep '<' and '>' at beginning and end
Allowed Clock Skew (minutes)	Allowed clock skew for assertion.	`30` Enter the same value as shown.
SSO Method	SSO method used for SAML	`Post`
Response Signing Certificate	The certificate used for signing the response.	This is the certificate you obtained from certificate authority. You have imported the same certificate into the Keystore in Section 8.6.7.5, "Importing the Certificate into Your Keystore."

5. Import the server certificate (for example, `server.crt`) created (in Section 8.6.7.4.6, "Signing the Certificate Request").
6. Click **Save Changes** to save the changes.

8.6.9.2 Creating a User Realm for SAML

An authentication realm specifies the conditions that users must meet to sign in. A realm consists of a grouping of authentication resources.

To create a user realm for SAML, proceed as follows:

1. From the Juniper Administration Console, in the left pane, expand the **Users** menu, point to **User Realms**, and then click **New User Realm**.
2. Specify the name as `OAAM SAML 1.1 User Realm`.
3. Select the Authentication Server `OAAM SAML 1.1` that was created in the last step as the authentication server for this user realm.
4. Save the changes.

   Now you should see the newly created user realm.
5. From the Juniper Administration Console, in the left pane, expand the **Users** menu, point to **User Realms**, and then click **OAAM SAML 1.1 User Realm**.
6. From the `OAAM SAML 1.1. User Realm`, click the **Role Mapping** tab to configure one or more role mapping rules.

8.6.9.3 Creating a Sign-In Policy

Create a Sign-In policy which defines the URL on which you need to go on the Juniper SA to get redirected to OAAM for authentication.

1. To create a sign-in policy, in the Juniper Administration Console, expand the **Authentication** menu, point to **Signing In**, and then click **Sign-in Policies**.

2. Click **New URL** and in the **Sign-in URL** field that is displayed, enter `*/OAAM11/` for the URL.
3. For Sign-in Page, select **Default Sign-in Page**.
4. For Authentication Realm, select the **OAAM SAML 1.1 User Realm** that was created earlier. Ensure the **User picks from a list of authentication realms** radio button is selected.
5. Click **Save Changes** and make sure it is enabled.

8.6.10 Verifying the Integration

Once all required components are configured, the next step is to test the Login and Forget Password flows. For information on the Login and Forget Password Flows, see "Authentication and Forgot Password Flows" in *Oracle Fusion Middleware Developer's Guide for Oracle Adaptive Access Manager*. For steps to verify the integration, see "Verify the Integration" in *Oracle Fusion Middleware Developer's Guide for Oracle Adaptive Access Manager*.

8.6.11 Debugging the Integration

To debug the integration on the OAAM end, enable the debug logs. For information on using the log files, see "Debug the Integration" in *Oracle Fusion Middleware Developer's Guide for Oracle Adaptive Access Manager*. Also see Section 3.1.2.8, "Differences When Using Fusion Middleware Control on IBM WebSphere" for the differences in managing Oracle Fusion Middleware on an IBM WebSphere cell as opposed to an Oracle WebLogic Server Domain.

8.6.12 Troubleshooting Common Problems

For information on common problems you might encounter in an Oracle Adaptive Access Manager and Juniper Networks Secure Access (SA) integrated environment, see "Troubleshooting Common Problems" in *Oracle Fusion Middleware Developer's Guide for Oracle Adaptive Access Manager*.

For the issue of concerning invalid SAML assertions, you can refer to the Juniper Knowledge Base article at:

`https://kb.juniper.net/InfoCenter/index?page=content&id=KB21687&cat=MAG_SERIES&actp=LIST`

To synchronize the clock: from the Juniper Administration Console, navigate to **System** > **Status** > **Overview** > **System Date and Time Edit**.

Also see Section 3.1.2.8, "Differences When Using Fusion Middleware Control on IBM WebSphere" for the differences in managing Oracle Fusion Middleware on an IBM WebSphere cell as opposed to an Oracle WebLogic Server Domain.

8.7 Integrating OAAM and Java Message Service Queue for Asynchronous Execution

Messaging is the exchanging of data between applications and clients of different types. OAAM uses JMS (Java Message Service) queues as one of the integration mechanisms. OAAM listens on one or more JMS queues for XML messages. Integration of Java Message Service Queue (JMSQ) and OAAM enables asynchronous execution to monitor and detect unauthorized access to critical and sensitive data housed in systems and databases.

This section includes the configuration and reference information you need to an asynchronous deployment on the IBM WebSphere Application Server.

8.7.1 Installing the Asynchronous Integration Option

This section provides instructions for installing the Asynchronous Integration Option for use with OAAM.

8.7.1.1 Before You Begin

Before you install the Asynchronous Integration option, ensure that the following prerequisites are satisfied:

- Ensure that Oracle Adaptive Access Manager 11*g* is installed and configured. The Asynchronous Integration Option will be installed on top of Oracle Adaptive Access Manager 11*g*.
- Ensure that Oracle Business Intelligence Publisher is installed and configured before proceeding with installation of the report templates. The Asynchronous Integration Option includes various reports as Oracle Business Intelligence Publisher report templates.

8.7.1.2 Extracting the Asynchronous Integration Option Package

The Asynchronous Integration Option does not require an installer. To install Asynchronous Integration Option, you will need an unzip tool to unzip the `osg_integration_kit.zip` file.

1. Create the following directories in the machine where Oracle Adaptive Access Manager 11*g* is installed:
   - osg_install
   - osg_install/osg_integration_kit
2. Extract the contents of the `osg_integration_kit.zip` file to the `osg_install/osg_integration_kit` directory

8.7.1.3 JMS Resources

OAAM uses JMS (Java Message Service) queues as one of the integration mechanisms. OAAM listens on one or more JMS queues for XML messages. For additional information XML message formats, see "XML Schema Example for Message Formats" in *Oracle Fusion Middleware Developer's Guide for Oracle Adaptive Access Manager*.

In an IBM WebSphere deployment, OAAM uses message-driven beans to listen for JMS messages. A message-driven bean is an enterprise bean that allows Java EE applications to process messages asynchronously. The definition of message-driven beans is in the `osg_ejb_was.jar` file. By default, it is configured to look for JMS activation specifications with the `JNDI` name `eis/oaamDefault_Act_Spec` and `eis/oaamHL7_Act_Spec` on IBM WebSphere. OAAM listens for JMS messages sent to JMS queues on IBM WebSphere that are associated with these two JMS activation specifications. Review this configuration and update as necessary for your deployment. This section provides a basic example of configuring JMS resources on IBM WebSphere.

An example configurations for activation of the `eis_oaamDefault_Act_Spec` specification is as follows:

1. Log in to the IBM WebSphere Administrative Console, under **Service Integration** > **Buses**, create a bus named **oaamBus**.

2. In **Members** of oaamBus, click **add** and add **oaam_server1** as a bus member.
3. In **Destinations** of oaamBus, add a new **Queue** named **oaamDefaultDestination**.
4. Navigate to **Resources** > **JMS** > **Queue connection factories** and create a queue connection factory named **oaamQueueConnFactory**. The provider could be **Default Messaging Provider**, and the scope should be **oaam_server1**.
5. Navigate to **Resources** > **JMS** > **Queues** and select the scope of **oaam_server1**, create a queue named **oaamDefaultQueue**, specify **jms/oaamDefaultQueue** as the JNDI name, and specify **Default Messaging Provider** as the provider.
6. Navigate to **Resources** > **JMS** > **Activation specifications**, and create an activation specification named **oaamDefaultActSpec** with following properties:

   Scope: oaam_server1

   Provider: Default Messaging Provider

   JNDI: eis_oaamDefault_Act_Spec

   Destination Type: Queue

   Bus name: oaamBus

   Target Inbound Transport Chain: InboundBasicMessaging

   Provider Endpoints: *host*:*port*:BootStrapBasicMessaging (the port here should be the `SIB_ENDPOINT_ADDRESS` port of `oaam_server` application server)

This section demonstrates how to create a basic JMS resources required by OAAM. Similar artifacts should be created for activation of the `eis/oaamHL7_Act_Spec` specification. Follow the IBM documentation on creating JMS Resources for JMS for your specific requirements. IBM documentation for creating JMS resources including JMS Activation Specification, JMS Queue, and JMS Queue Connection Factory on the IBM WebSphere Administrative Console is available at:

`http://publib.boulder.ibm.com/infocenter/iisinfsv/v8r1/index.jsp?topic=/com.ibm.swg.im.iis.infoservdir.user.doc/topics/t_isd_user_setting_up_jms_in_was.html`

8.7.1.4 Deploy the Asynchronous Integration Extension Files

Most of the asynchronous integration functionality is implemented as an OAAM extension. Follow these steps to deploy the extension files included in the Asynchronous Integration Option package.

1. Log in to the IBM WebSphere Administrative Console.
2. Click **Applications** > **Application Types** > **WebSphere Enterprise Applications**.
3. Select the check box next to the **oaam_server** application and click **Update**.
4. In **Application update options**, select the **Replace, add, or delete multiple files** option and choose **osg_oaam_extension_was.zip** and add it to the application. This will migrate most of the extension files to the OAAM Server application.
5. Make sure that JMS resources mentioned in Section 8.7.1.3, "JMS Resources" are configured properly on IBM WebSphere. OAAM is configured to work with IBM WebSphere JMS Activation Specifications with JNDI name `eis/oaamDefault_Act_Spec` and `eis/oaamHL7_Act_Spec`. JMS queues, JMS queue connection factories, Service Integration Bus also need to be configured on IBM WebSphere properly with target JMS Activation Specifications as described in Section 8.7.1.3, "JMS Resources."

6. Similarly, follow Steps 1-3, then in **Application update options**, select **Replace or add a single module** option and choose **osg_ejb_was.jar** and add it to the application. Make sure that **oaam_server1** is selected and mapped to this module in the **Mapmodule to servers** step.
7. Synchronize the node and restart the OAAM Server application server.

8.7.1.5 Update OAAM Database

OAAM creates database views for the entities and transactions for use in rule conditions and reports. For details on these database views, see *Oracle Fusion Middleware Developer's Guide for Oracle Adaptive Access Manager*.

8.7.2 JMS Integration

With the JMS listener implementation, Oracle Adaptive Access Manager can be configured to listen to JMS queues (or topics) for messages in XML format. JMS message contents looks similar to the OAAM Web Services API calls. Sample JMS message formats are the following:

- VCryptTracker.updateLog
- VCryptTracker.createTransaction
- VCryptTracker.updateEntity
- VCryptRulesEngine.processRules

For an overview of JMS integration in OAAM and for sample XML, see *Oracle Fusion Middleware Developer's Guide for Oracle Adaptive Access Manager*.

8.7.2.1 Configuration

Various aspects of JMS integration can be configured using OAAM properties and user-defined enums.

For a list of the JMS configuration properties used for integration, see the *Oracle Fusion Middleware Developer's Guide for Oracle Adaptive Access Manager*. The following properties do not pertain to the IBM WebSphere deployments:

- oracle.oaam.jms.listeners.default.initial.context.factory
- oracle.oaam.jms.listeners.default.connection.factory
- oracle.oaam.jms.listeners.enum.lsnr_1
- oracle.oaam.jms.listeners.enum. lsnr_1.type
- oracle.oaam.jms.listeners.enum. lsnr_1.url
- oracle.oaam.jms.listeners.enum. lsnr_1.jndiname
- oracle.oaam.jms.listeners.enum.lsnr_1.initial.context.factory
- oracle.oaam.jms.listeners.enum.lsnr_1.connection.factory
- oracle.oaam.jms.listeners.enum. lsnr_1.processor
- oracle.oaam.jms.listeners.enum. lsnr_1.instancecount

For each queue (or topic) to be monitored, one listener must be configured by adding an enum element in user-defined enum `oracle.oaam.jms.listeners.enum`. Any changes to the listener list or properties require the OAAM Server, where the listeners run, to be restarted.

8.7.2.2 Message Structure

OAAM default JMS message processor processes only the messages of type `javax.jms.TextMessage`. Other types of messages would be ignored by this processor. To process other type of messages, you can implement a custom processor by extending either `oracle.oaam.jms.JmsAbstractMessageProcessor` or `oracle.oaam.jms.JmsDefaultMessageProcessor`; develop a standard Message-Driven Bean class, and associate it with a IBM WebSphere JMS Activation Specification in `osg_ejb_was.jar/META-INF/ibm-ejb-jar-bnd.xmi`. After adding the reference to the JMS Activation Specification in `osg_ejb_was.jar`, the jar file needs to be re-deployed to OAAM Server application as described in section Section 8.7.1.4, "Deploy the Asynchronous Integration Extension Files."

Further, the default JMS message processor would processes only if the contexts of the `TextMessage` is a XML string that conforms to the XML schema given at the end of this section.

8.7.3 Database Views for Entities and Transactions

You can define entities and transactions in Oracle Adaptive Access Manager with any number of data fields. In addition, transactions can also be defined to reference entities. Oracle Adaptive Access Manager persists the entity and transaction data in the database.

See *Oracle Fusion Middleware Developer's Guide for Oracle Adaptive Access Manager* for the following topics:

- Generating SQL script file
- Entity view details
- Transaction view details
- Identifiers
- Supported databases

8.7.4 Python Expression

The Asynchronous Integration Option package includes an OAAM condition to execute a given Python expression. The condition can be imported from `osg_install/osg_integration_kit/osg_rule_conditions.zip`.

8.7.4.1 Prerequisite

OAAM needs the `jython.jar` library to run Python expressions. The location of `jython.jar` is at:

```
ORACLE_MW_HOME/ oracle_common/modules/oracle.jrf_11.1.1/jython.jar
```

Follow these steps to add `jython.jar` as a shared library for the OAAM Server application:

1. Log in to IBM WebSphere Administrative Console.
2. Click **Environment** > **Shared libraries** > **New**.
3. Provide the name for this jar and specify the path to `jython.jar`.
4. Apply the changes.
5. Navigate to **Applications** > **Application Types** > **WebSphere Enterprise Applications** > **oaam_server_11.1.2.0.0**.

6. In the details of the oaam server application, under the **Configuration** tab, navigate to the **References** section, click **shared library references**, check **OAAM_Runtime**, and click **Reference Shared libraries**.
7. Move the jython library to the selected list and apply the changes.
8. Synchronize the node and restart the OAAM Server application server.

8.7.4.2 Objects Available in Python

For a list of the objects (variables) accessible from Python expressions, see *Oracle Fusion Middleware Developer's Guide for Oracle Adaptive Access Manager*.

8.7.4.3 Examples

For a list of expressions that can be used in the Python Expression condition, see *Oracle Fusion Middleware Developer's Guide for Oracle Adaptive Access Manager*.

8.8 Setting Up the OAAM Sample Application

An OAAM sample application is available as a form of documentation to familiarize you with OAAM APIs. It is not intended to be used as production code since it only provides basic elements of API usage. Customers implementing a native integration should develop their own application using the OAAM sample application as a reference. The OAAM sample application and related files can be downloaded from My Oracle Support at `https://support.oracle.com`.

8.8.1 Setting Up the Native In-Proc-Based OAAM Sample Application

This section contains instructions to set up the native in-proc-based OAAM sample application.

To set up the native in-proc-based OAAM sample application:

1. Create an `oaam_sample` directory.
2. Extract the `oaam_sample_inproc.zip` file into the `oaam_sample` directory.
3. Open the `web.xml` file in the `oaam_sample/WEB-INF` directory with a text editor and add the following lines as the last section above the `</web-app>` tag:

```
<resource-ref>
       <res-ref-name>jdbc/OAAM_SERVER_DB_DS</res-ref-name>
       <res-type>javax.sql.DataSource</res-type>
       <res-auth>Container</res-auth>
<resource-ref>
```

4. Navigate to the `oaam_sample` directory and run the following command to create an OAAM sample WAR file for the IBM WebSphere Administrative Console:

```
jar cfm ../oaam_sample_inproc_was.war META-INF/MANIFEST.MF .
```

   Running the command generates the `oaam_sample_inproc_was.war` file which you will install in the IBM WebSphere Administrative Console.

5. Log in to the IBM WebSphere Administrative Console and in the console navigation tree, expand **Applications**, and then **Application Types** and click **WebSphere enterprise applications** to open the list of applications.
6. Select the check box next to the **oaam_server_11.1.2.0.0** application and click **Update**.

7. Select **Replace or add a single module** and provide a unique path such as `oaam_sample` and specify the path to the `oaam_sample_inproc_was.war` file that was created in step 4; then click **Next** to proceed to the page, **Step 2**.

   Make sure it is applied to `oaam_server1`. Also make sure `jdbc/OAAM_SERVER_DB_DS` is selected in the Resource Reference page for the module.

   Continue through the rest of the configuration without changing the settings and save the changes.

8. Move `oracle.security.jps.was.deployment.jar` in the `WAS_HOME/plugins` directory and restart the node and deployment manager. Otherwise adding a WAR/JAR file as a module to an application will fail.

9. In the IBM WebSphere Administrative Console, navigate to **Applications** > **Application Types** > **WebSphere enterprise applications** > **oaam_server_11.1.2.0.0** > **Context Root for Web Modules** and specify `oaam_sample` for the OAAM sample web module.

10. Copy the `oaam_native_libs` directory from the `was_native_jar` directory to the directory where `oaam_server` is installed.

11. In the IBM WebSphere Administrative Console, navigate to **Environment** > **Shared Libraries** and select the scope of the cell as `oaam_server1`.

12. Create a new shared library named `oaam_native_libs` and specify the classpath as the path to the directory created in step 10.

13. In the IBM WebSphere Administrative Console, navigate to **Applications** > **Application Types** > **WebSphere enterprise applications** > **oaam_server_11.1.2.0.0** > **Shared library references**, and check the checkbox next to the `oaam_sample` web module.

14. Click **Reference Shared libraries** and move the shared library that was created in step 12 from the **Available** list to the **Selected** list. Then save and synchronize the changes.

15. Deploy `was_native_jars/was_native_web.jar` from the directory that is copied to in step 10. Navigate to **Applications** > **Application Types** > **WebSphere enterprise applications**, and select the check box next to the **oaam_server_11.1.2.0.0** application and click **Update**.

16. Under the Application update options section, select **Replace or add a single module** and in the **Specify the path beginning with the installed application archive file to the module to be replaced or added** field, provide any unique path, such as `oaam_sample_web,` and select the path to the `was_native_web.jar` files and click **Next** to the Step 2 page. Make sure it is applied to `oaam_server1`; Continue with the configuration without changing settings and save and synchronize the changes.

17. Move `oracle.security.jps.was.deployment.jar` in `WAS_HOME/plugins` directory and restart the node/manager.

18. Restart the oaam_server application servers from **Servers** > **Server Types** > **WebSphere application servers**.

19. Log in to the OAAM Administration Console.

20. Import the policies from the `oaam_policies.zip` file.

21. Import the challenge questions from the `oaam_kba_questions_en.zip` file.

22. Import the Transaction definitions from the `Sample_Transaction_Defs.zip` file.
23. Import the Transaction policies from the `Sample_Txn_Models.zip` file.
24. Navigate to the URL:

    `http://host-name:oaam-server-port/oaam_sample`

    You are shown the login page of the OAAM sample application.
25. Enter the username and then password in the next page. You be taken through the registration process and after which you are shown links to the sample transactions.

    Note: The password must be `test` for the initial log in. You must change the password immediately.

26. Move the `oracle.security.jps.was.deployment.jar` back to the WAS_HOME/plugins directory if you moved it out in step 8 and restart the node/manager; otherwise, access control is disabled completely.

8.8.2 Setting Up the Native SOAP-based OAAM Sample Application

This section contains instructions to set up the native SOAP-based OAAM sample application.

To set up the SOAP-based OAAM sample application, proceed as follows:

1. Create an `oaam_sample` directory.
2. Extract the `oaam_sample_soap.zip` file into the `oaam_sample` directory.
3. Use a text editor to edit the `oaam_custom.properties` file under the *customer application deployment*`/WEB-INF/classes` directory:
   - Set the `vcrypt.tracker.soap.url` property.

     `vcrypt.tracker.soap.url=http://host-name:port/oaam_server/services`

     This setting specifies the location of the web services with which the application will communicate.
   - Set `bharosa.image.dirlist` to the absolute directory path where OAAM images are available.
   - Remove the property `bharosa.cipher.encryption.algorithm.enum` related to encryption keys.
4. Navigate to the `oaam_sample` directory and run the command:

   ```
   jar cfm ../oaam_sample_soap_was.war META-INF/MANIFEST.MF .
   ```

   This will generates the `oaam_sample_soap_was.war` which you will install using the IBM WebSphere Administrative Console.
5. In the IBM WebSphere Administrative Console, navigate to **Applications** > **Application Types** > **WebSphere enterprise applications**.
6. From the Enterprise Applications list, select the check box next to the **oaam_server_11.1.2.0.0** application and click **Update**.

7. Under the Application update options section, select **Replace or add a single module** and provide a unique path such as `oaam_sample` and specify the path to the `oaam_sample_soap_was.war` that was created in step 4; then click **Next** to continue to the Step 2 page. Continue through the configuration without changing settings and save the changes.

   Make sure the module is applied to `oaam_server1`.

8. Move `oracle.security.jps.was.deployment.jar` in the `WAS_HOME/plugins` directory and restart the node and deployment manager.

9. In the IBM WebSphere Administrative Console, navigate to **Applications** > **Application Types** > **WebSphere enterprise applications** > **oaam_server_11.1.2.0.0** > **Context Root for Web Modules**, and specify `oaam_sample` for the OAAM sample web module.

10. Copy `oaam_native_libs` directory from the `was_native_jar` directory to the directory where `oaam_server` is installed.

11. In the IBM WebSphere Administrative Console, navigate to **Environment** > **Shared Libraries**, and select the scope of the cell as `oaam_server1`.

12. Create a new shared library named `oaam_native_libs` and specify the classpath as the path to the directory created in step 10.

13. In the IBM WebSphere Administrative Console, navigate to **Applications** > **Application Types** > **WebSphere enterprise applications** > **oaam_server_11.1.2.0.0** > **Shared library references**, and select the check box next to the oaam_sample web module.

14. Click **Reference Shared libraries** and move the shared library that was created in step 12 from the **Available** list to the **Selected** list. Save and synchronize the changes.

15. Deploy `was_native_jars/was_native_web.jar` from the directory that is copied to in step 10. In the console navigation tree, expand **Applications**, and then **Application Types** and click **WebSphere enterprise applications** to open the list of applications.

16. From the Enterprise Applications list, select the check box next to the **oaam_server_11.1.2.0.0** application and click **Update**.

17. Select **Replace or add a single module** and specify any unique path such as `oaam_sample_web` and select the path to the `was_native_web.jar` file and click **Next** to Step 2. Continue through the configuration without changing the settings and then save the configuration.

18. Move `oracle.security.jps.was.deployment.jar` back to the `WAS_HOME/plugins` directory and restart the node/manager.

19. Restart the oaam_server application server from **Servers** > **Server Types** > **WebSphere application servers**.

20. Log in to the OAAM Administration Console.

21. Import the policies into the system from `oaam_policies.zip` file.

22. Import the challenge questions into the system from the `oaam_kba_questions_en.zip` file.

23. Import the Transaction Definitions into the system from the `Sample_Transaction_Defs.zip` file.

24. Import the Transaction policies into the system from the `Sample_Txn_Models.zip` file.
25. Navigate to the URL:

    `http://host-name:oaam-server-port/oaam_sample`

    You are shown the login page of the sample application.

8.9 Installing for a Clustered OAAM Configuration

A clustered environment allows for high availability and scalability. A cluster typically contains Application Servers that span multiple nodes. When creating a cluster, you add the initial cluster member (which must be an existing Application Server). After creating the cluster, you add additional cluster members to it. Each additional cluster member inherits the attributes of the first cluster member.

This section documents the steps to set up OAAM 11.1.2.1.0 in an IBM WebSphere environment in a clustered configuration that has high availability support.

8.9.1 Overview of Clustered Configuration

Perform the steps in this section to install this delivery of Oracle Adaptive Access Manager on IBM WebSphere in a clustered configuration. A cluster is a group of Application Servers that work together to provide scalability and high availability for applications. By creating clusters, you can group servers such that they operate as a single unit for hosting applications and resources. By performing the following steps, you will create a configuration as described in Table 8–8:

Table 8–8 Overview of Clustered Configuration

Deployment Manager Machine	WebSphere Node 2 Machine
WebSphere Deployment Manager	WebSphere Node2
WebSphereNode1	oaam_admin_server2
OracleAdminServer	oaam_server_server2
oaam_admin_server1	
oaam_server_server1	

8.9.2 OAAM Clustered Configuration Roadmap

Table 8–9 lists the tasks for installing and configuring Oracle Adaptive Access Manager on IBM WebSphere in a clustered configuration.

Table 8–9 Installation and Configuration Flow for Oracle Adaptive Access Manager Cluster

No.	Tasks	Description
1.	Install IBM WebSphere.	For more information, see Section 8.9.3, "Task 1: Install IBM WebSphere."
2.	Install and configure the Oracle 11g database	For more information, see Section 8.9.4, "Task 2: Install and Configure the Oracle 11g Database."
3.	Install the Oracle Fusion Middleware Repository Creation Utility	For more information, see Section 8.9.5, "Task 3: Install the Oracle Fusion Middleware Repository Creation Utility."
4.	Create and load the OAAM schema into the database	For more information, see Section 8.9.6, "Task 4: Create and Load the OAAM Schema into the Database."
5.	Install Oracle Adaptive Access Manager	For more information, see Section 8.9.7, "Task 5: Install Oracle Adaptive Access Manager."
6.	Configure IBM WebSphere on the Deployment Manager Machine	For more information, see Section 8.9.8, "Task 6: Configure IBM WebSphere on the Deployment Manager Machine."
7.	Configure Oracle Platform Security Services Security Store	For more information, see Section 8.9.9, "Task 7: Configure Oracle Platform Security Services Security Store."
8.	Start the Deployment Manager	For more information, see Section 8.9.10, "Task 8: Start the Deployment Manager."
9.	Configure IBM WebSphere on IBM WebSphere Node 2 machine	For more information, see Section 8.9.11, "Task 9: Configure IBM WebSphere on IBM WebSphere Node 2 Machine."
10.	Configure an LDAP Server (Optional)	For more information, see Section 8.9.12, "Task 10: Configure an LDAP Server (Optional)."
11.	Set Up Session Persistence in IBM WebSphere	For more information, see Section 8.9.13, "Task 11: Set Up Session Persistence in IBM WebSphere."
12.	Restart the servers	For more information, see Section 8.9.14, "Task 12: Restart the Servers."

8.9.3 Task 1: Install IBM WebSphere

On Deployment Manager Machine and WebSphere Node 2 Machine, install IBM WebSphere Application Server Network Deployment 7.0 with the appropriate fix pack.

To obtain and install the IBM WebSphere software, refer to the IBM WebSphere documentation.

For more information about the Fix Packs available for IBM WebSphere 7.0, refer to the Fix list for IBM WebSphere Application Server V7.0 on the IBM Support Web site.

When you install the IBM WebSphere software, you are prompted for the location where you want to install the software. For the purposes of this documentation, this location is later referred to as the WAS Home, or *WAS_HOME* in examples.

8.9.4 Task 2: Install and Configure the Oracle 11g Database

Install and patch Oracle Database. For information on the databases supported by Oracle Fusion Middleware, see the certification information described in Section 2.1, "Task 1: Review the System Requirements and Certification Information."

Make a note of the database connection information along with the name and passwords for the schemas you create with the Repository Creation Utility. You will need these later when you configure the Oracle Fusion Middleware products.

8.9.5 Task 3: Install the Oracle Fusion Middleware Repository Creation Utility

Install the Oracle Fusion Middleware Repository Creation Utility (RCU). For more information, see "Obtaining and Running Repository Creation Utility" in the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.

8.9.6 Task 4: Create and Load the OAAM Schema into the Database

Create and load the Identity Management - Oracle Adaptive Access Manager schema into the database using the Oracle Fusion Middleware Repository Creation Utility (RCU). For more information, refer to the following documents:

- *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*
- *Oracle Fusion Middleware Repository Creation Utility User's Guide*

8.9.7 Task 5: Install Oracle Adaptive Access Manager

On Deployment Manager Machine and WebSphere Node 2 Machine, install Oracle Adaptive Access Manager. For more information about installing Oracle Adaptive Access Manager, refer to the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.

When you run the Oracle Fusion Middleware installer, you must use the parameter `-DSHOW_APPSERVER_TYPE_SCREEN=true` to let the Oracle Universal Installer prompt for the IBM WebSphere home location.

To start the installer:

```
cd iamsuite/Disk1
./runInstaller -DSHOW_APPSERVER_TYPE_SCREEN=TRUE -jreLoc LOCATION_OF_IBM_JRE
```

Note: Ensure you have the same path for the ORACLE_HOME on both the machines.

8.9.8 Task 6: Configure IBM WebSphere on the Deployment Manager Machine

Before you can use Oracle Fusion Middleware products in an IBM WebSphere environment, you must create and configure the cell in which the Oracle Fusion Middleware applications will run. The Oracle Fusion Middleware Configuration Wizard for IBM WebSphere simplifies the process of creating and expanding a cell for your Oracle Fusion Middleware environment.

On the Deployment Manager Machine, use the Oracle Fusion Middleware Configuration Wizard to create the Oracle Adaptive Access Manager cell. By default, the Configuration Wizard is located at:

```
MW_HOME/Oracle_IDM1/common/bin/was_config.sh
```

For more information, refer to the *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server*.

Table 8–10 provides information about specific Configuration Wizard screens and appropriate information to enter on those screens—the table does not cover self-explanatory, standard screens. For those screens, click **Next** to accept the initial values and continue.

Table 8–10 Information for Specific Configuration Wizard Screens

Screen Name	Input Description
Add Products to Cell	Select **Oracle Adaptive Access Manager Admin Server** and **Oracle Adaptive Access Manager Server Server**.
Select Optional Configuration	At a minimum, you must select the **Application Servers, Clusters and End Points** option—this is a required option.
Configure Application Servers	Perform the following steps: 1. In the Name field, enter a name for the Oracle Adaptive Access Manager server, for example: `oaam_admin_server1`. 2. In the Node Name list, select the Node Agent for oaam_server1. Each node has a Node Agent, which is a service that is used to communicate with the Deployment Manager. For example: **WebSphereNode1**.
Configure Clusters Screen	Perform the following steps: 1. Click **Add** to add a cluster. 2. Enter a name for the cluster in the cluster name field, for example: `OAAMAdminCluster`. 3. Select the appropriate OAAM Admin server from the First cluster member list. This is the Application Server that will be the primary cluster member of the cluster. 4. Click **Add** to add another cluster. 5. Enter a name for the cluster in the cluster name field, for example: `OAAMServerCluster`. 6. Select the appropriate OAAM Server server from the First cluster member list. This is the Application Server that will be the primary cluster member of the cluster.
Configure Additional Cluster Members	Click **Next**, or optionally, add servers to an existing system in the cluster. When you add a new Application Server to a cluster, its attributes are cloned from the primary member of the cluster.

8.9.9 Task 7: Configure Oracle Platform Security Services Security Store

Run the `configureSecurityStoreWas.py` script to configure the Oracle Platform Security Services (OPSS) security store as follows:

```
setenv WSADMIN_CLASSPATH ORACLE_HOME/common/wlst/resources/oes-common.jar
cd IDM_HOME/common/bin/wsadmin.sh
./wsadmin.sh -lang jython -profileName <dmgr-prof-name>
   -f ORACLE_HOME/common/tools/configureSecurityStoreWas.py
   -d <dmgr-cell-home>(WAS_HOME/profiles/<dmgr-profile-name>/
       config/cells/<cell-name>)
```

```
-t DB_ORACLE
-j cn=jpsroot
-m create
--passcode <password>
--config IAM
```

8.9.10 Task 8: Start the Deployment Manager

On the Deployment Manager Machine, start the Deployment Manager as follows:

```
WAS_HOME/bin
./startManager.sh -profileName <dmgr_prof_name>
```

8.9.11 Task 9: Configure IBM WebSphere on IBM WebSphere Node 2 Machine

On the WebSphere Node 2 Machine, launch the Oracle Fusion Middleware Configuration Wizard to federate the machine and configure its cell. By default, the Configuration Wizard is located at:

```
MW_HOME/Oracle_IDM1/common/bin/was_config.sh
```

For more information, refer to the *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server*.

Table 8–11 provides information about specific Configuration Wizard screens and appropriate information to enter on those screens—the table does not cover self-explanatory, standard screens. For those screens, click **Next** to accept the initial values and continue.

Table 8–11 Information for Specific Configuration Wizard Screens

Screen Name	Input Description
Select Configuration Option	Select the **Federate Machine and Configure Cell** option.
Specify Profile and Node Name Information	Enter information about the profile and node names you want to create for the WebSphere Node 2 Machine.
Specify Deployment Manager Information	Enter information about the existing Deployment Manager system.
Select Optional Configuration	Be sure to select the **Application Servers, Clusters and End Points** option—this is a required option.
Configure Additional Cluster Members	Perform the following steps: 1. Click **Add**. 2. In the Name field, enter a name for the second server in the OAAMCluster. For example: `oaam_server2`. 3. In the Node Name list, select the Node Agent for oaam_server2. For example: **WebSphereNode2**. 4. In the Cluster Name list, select the **OAAMServerCluster**.

8.9.12 Task 10: Configure an LDAP Server (Optional)

If you are installing and configuring Oracle Identity and Access Management Suite on IBM WebSphere, you can optionally install and configure a supported LDAP server. Oracle Fusion Middleware components do not support WebSphere's built-in file-based user registry.

8.9.12.1 Installing LDAP Servers

For information about installing a supported LDAP server, see Section 9.1, "IBM WebSphere Identity Stores."

For information on the LDAP servers that are supported by Oracle Fusion Middleware, refer to the certification information on the Oracle Technology Network:

http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-certification-100350.html

8.9.12.2 Create OAAM Administrative Roles and User in LDAP

When using an LDAP store, you must add a number of OAAM roles to the store. In addition to creating these roles, you must create users and assign these users to these roles to facilitate access to the OAAM Admin console.

Use the `ldif` files documented in this section as a reference to create users and groups in LDAP. You will have different values for the LDAP attributes. For example:

```
dn: cn=an_oaam_user, cn = Users, dc=us/in/etc, dc=companyname,
dc =com/org/etc
```

oaam_user.ldif

```
dn:  cn=oaamadmin,cn=Users,dc=us,dc=oracle,dc=com
cn:  oaamadmin
sn:  oaamadmin
description:  oaamadmin
uid: oaamadmin
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetorgperson
objectclass: orcluser
objectclass: orcluserV2
userpassword: mypasswd
```

oaam_group.ldif

```
dn: cn=OAAMCSRGroup,cn=Groups,dc=us,dc=oracle,dc=com
cn: OAAMCSRGroup
displayname: OAAMCSRGroup
description: OAAMCSRGroup
uniquemember: cn=oaamadmin,cn=Users,dc=us,dc=oracle,dc=com
objectclass: top
objectclass: groupofuniquenames
objectclass: orclgroup

dn: cn=OAAMCSRManagerGroup,cn=Groups,dc=us,dc=oracle,dc=com
cn: OAAMCSRManagerGroup
displayname: OAAMCSRManagerGroup
description: OAAMCSRManagerGroup
uniquemember: cn=oaamadmin,cn=Users,dc=us,dc=oracle,dc=com
objectclass: top
objectclass: groupofuniquenames
objectclass: orclgroup

dn: cn=OAAMEnvAdminGroup,cn=Groups,dc=us,dc=oracle,dc=com
cn: OAAMEnvAdminGroup
displayname: OAAMEnvAdminGroup
description: OAAMEnvAdminGroup
```

```
uniquemember: cn=oaamadmin,cn=Users,dc=us,dc=oracle,dc=com
objectclass: top
objectclass: groupofuniquenames
objectclass: orclgroup

dn: cn=OAAMInvestigationManagerGroup,cn=Groups,dc=us,dc=oracle,dc=com
cn: OAAMInvestigationManagerGroup
displayname: OAAMInvestigationManagerGroup
description: OAAMInvestigationManagerGroup
uniquemember: cn=oaamadmin,cn=Users,dc=us,dc=oracle,dc=com
objectclass: top
objectclass: groupofuniquenames
objectclass: orclgroup

dn: cn=OAAMInvestigatorGroup,cn=Groups,dc=us,dc=oracle,dc=com
cn: OAAMInvestigatorGroup
displayname: OAAMInvestigatorGroup
description: OAAMInvestigatorGroup
uniquemember: cn=oaamadmin,cn=Users,dc=us,dc=oracle,dc=com
objectclass: top
objectclass: groupofuniquenames
objectclass: orclgroup

dn: cn=OAAMRuleAdministratorGroup,cn=Groups,dc=us,dc=oracle,dc=com
cn: OAAMRuleAdministratorGroup
displayname: OAAMRuleAdministratorGroup
description: OAAMRuleAdministratorGroup
uniquemember: cn=oaamadmin,cn=Users,dc=us,dc=oracle,dc=com
objectclass: top
objectclass: groupofuniquenames
objectclass: orclgroup

dn: cn=OAAMSOAPServicesGroup,cn=Groups,dc=us,dc=oracle,dc=com
cn: OAAMSOAPServicesGroup
displayname: OAAMSOAPServicesGroup
description: OAAMSOAPServicesGroup
uniquemember: cn=oaamadmin,cn=Users,dc=us,dc=oracle,dc=com
objectclass: top
objectclass: groupofuniquenames
objectclass: orclgroup
```

Instructions

Load the user and group into LDAP issuing the following commands from the LDAP server:

```
ldapadd -h myoid.myompany.com -p 389 -D cn="orcladmin" -w mypasswd -c -v \
  -f oaam_user.ldif
ldapadd -h myoid.mycompany.com -p 389 -D cn="orcladmin" -w mypasswd -c -v \
  -f oaam_group.ldif
```

8.9.13 Task 11: Set Up Session Persistence in IBM WebSphere

The configuration for session persistence in a clustered environment is different for WebSphere Application Server than for Oracle WebLogic Server, in which the element `<persistent-store-type>replicated_if_clustered</persistent-store-type>` can be set in the servlet to provide session persistence. In WebSphere Application Server, the web container's session management property for session persistence is `sessionPersistenceMode`, which is stored in the

WAS_HOME/profiles/profile_name/config/cells/cell_name/nodes/node_name/servers/server_name/server.xml file.

You can display the current sessionPersistenceMode value in the IBM WebSphere Administrative Console.

You can set the session persistence mode while creating a cluster through the IBM WebSphere Administrative Console. If you use the Fusion Middleware Configuration Wizard to create a cluster, after the cluster is created, you must configure the session persistence mode manually, through the IBM WebSphere Administrative Console, and then restart the application servers.

To display the session persistence mode in the IBM WebSphere Administrative Console:

1. Log in to the IBM WebSphere Administrative Console as an administrator.
2. Expand **Servers** and **Server Types** on the left, and click **WebSphere application servers**.
3. Click an OAAM application server on the Application Servers page.
4. Expand **Web Container Settings,** and click **Web container**, under Container Settings on the Configuration page for the application server.
5. Click **Session management**, under Additional Properties on the Web container page.
6. Click **Distributed environment settings**, under Additional Properties on the Session management page.
7. The Configuration page for Distributed environment settings displays the current sessionPersistenceMode value under General Properties, as the selected value for Distributed sessions.

   The value of sessionPersistenceMode can be None, Database, or Data_Replication (displayed as the Memory-to-memory replication mode on the console.

To set the session persistence mode for a cluster:

- If you use the IBM WebSphere Administrative Console to create a cluster, you must select the option **Configure HTTP session memory-to-memory replication**.

  When this option is selected, a replication domain is created automatically, and the sessionPersistenceMode property is automatically set to the Data_Replication mode for each member of the new cluster. The created Replication domain, which has the same name as the cluster name, is automatically set for the **Replication Domain** property.

- If you use the Fusion Middleware Configuration Wizard to create a cluster, the Configuration Wizard does not present any options for cluster creation, and it creates the new cluster without creating a replication domain.

  The sessionPersistenceMode property is not set to the Data_Replication mode for each cluster member. In this case, you must configure sessionPersistenceMode manually, as the following procedure describes.

To configure the session persistence mode manually:

1. Log in to the IBM WebSphere Administrative Console as an administrator.
2. Create a replication domain (if there is none, or if you want to create a new replication domain for a cluster you created through the Fusion Middleware Configuration Wizard):

a. Expand **Environment** on the left of the console, and click **Replication domains**.

b. Click the **New** button on the Replication domains page.

c. Enter the domain name, and select the option **Entire Domain** under Number of Replicas.

d. Click the **Apply** or **OK** button, and then **Save**.

3. Set the `sessionPersitenceMode` property value for every server member in the cluster:

a. Expand **Servers** and **Server Types** on the left, and click **WebSphere application servers**.

b. Click an OAAM application server on the Application Servers page.

c. Expand **Web Container Settings**, and click **Web container**, under Container Settings on the Configuration page for the application server.

d. Click **Session management**, under Additional Properties on the Web container page.

e. Click **Distributed environment settings**, under Additional Properties on the Session management page.

f. Select the **Memory-to-memory replication** option, under Distributed sessions on the Configuration page for Distributed environment settings (to set `sessionPersistenceMode` to `Data_Replication`.

g. On the Configuration page for Memory-to-memory replication, select the replication domain you created for this cluster.

h. Select the proper Replication mode for your configuration.

i. Click the **Apply** or **OK** button, and then **Save**.

4. Restart the OAAM application servers.

8.9.14 Task 12: Restart the Servers

To restart the servers, follow these steps:

1. Stop the node and Deployment Manager on the Deployment Manager machine. Execute the following from $WAS_HOME/bin:

```
./stopNode.sh -profileName <server_prof_name> -username <wasadmin username>
-password <password>
./stopManager.sh -profileName <dmgr_prof_name> -username <wasadmin username>
-password <password>
```

2. Stop the node on WebSphere Node 2 machine. Execute:

```
./stopNode.sh -profileName <server_prof_name> -username <wasadmin username>
-password <password>
```

3. Start the Deployment Manager, node, and servers on Deployment Manager machine. Execute:

```
./startManager.sh -profileName <dmgr_prof_name>
./startNode.sh -profileName <server_prof_name>
./startServer.sh OracleAdminServer -profileName <server_prof_name>
./startServer.sh <oaam_server_name> -profileName <server_prof_name>
```

4. Start the node and opam server on WebSphere Node 2 machine:

```
./startNode.sh -profileName <server_prof_name>
./startServer.sh <oaam_server_name> -profileName <server_prof_name>
```

9

Managing Oracle Fusion Middleware Security on IBM WebSphere

This chapter contains information about managing Oracle Fusion Middleware security on IBM WebSphere, and it explains the particularities of some Oracle Platform Security Services (OPSS) features on that platform.

OPSS is a security platform that can be used to secure applications deployed in any of the supported platforms or in standalone applications.

Only topics that apply specifically to IBM WebSphere are included in this chapter; those that apply uniformly to all platforms are not described here, but can be found in *Oracle Fusion Middleware Application Security Guide*.

This chapter contains the following sections:

9.1 IBM WebSphere Identity Stores

On IBM WebSphere, OPSS supports LDAP-based registries only; in particular, it does not support WebSphere's built-in file-based user registry.

For information about the list of LDAP authenticators supported for Oracle Fusion Middleware, visit
`http://www.oracle.com/technology/software/products/ias/files/fusion_certification.html`

For the special configuration required for the Open LDAP 2.2, see *Oracle Fusion Middleware Application Security Guide*.

The configuration and seeding of a repository is explained in the following sections:

- Configuring a Registry
- Seeding a Registry

9.1.1 Configuring a Registry

The configuration of an LDAP registry on IBM WebSphere is accomplished with the command `configureIdentityStore`, an online administration command with the following syntax:

```
wsadmin> Opss.configureIdentityStore(propsFileLoc="fileLocation")
```

`propsFileLoc` specifies the location of the file that contains the property settings for the identity LDAP identity store. This command modifies the configuration file `jps-config.xml` to include the specifications in the property file.

After running Opss.configurIdentityStore, the server must be restarted.

The following properties are required and must be specified in property settings file:

- ldap.host
- ldap.port
- admin.id
- admin.pass
- idstore.type
- user.search.bases
- user.id.map
- group.id.map
- group.member.id.map
- group.search.bases
- primary.admin.id

The following list includes optional properties specific to a IBM WebSphere registry:

- group.filter
- user.filter

The following sample illustrates the property settings for an Oracle Directory Server Enterprise Edition identity store:

```
user.search.bases=cn=Users,dc=us,dc=oracle,dc=com
group.search.bases=cn=Groups,dc=us,dc=oracle,dc=com
subscriber.name=dc=us,dc=oracle,dc=com
ldap.host=stamw10.examplehost.exampledomain.com
ldap.port=3060
# admin.id must be the full DN of the user in the LDAP
admin.id=cn=orcladmin
admin.pass=welcome1
user.filter=(&(uid=%v)(objectclass=person))
group.filter=(&(cn=%v)(objectclass=groupofuniquenames))
user.id.map=:uid
```

```
group.id.map=:cn
group.member.id.map=groupofuniquenames:uniquemember
ssl=false
# primary.admin.id indicates the user you want to be the primary
# administrative user on WebSphere. It should be a user under user.search.bases.
# later you need to use this user's user name and password to manage or
# start/stop the server.
primary.admin.id=orcladmin
# optional, default to "OID"
idstore.type=IPLANET
# other, optional identity store properties can be configured in this file.
username.attr=cn
```

The list of valid identity store types is the following:

- OID
- IPLANET
- OVD
- ACTIVE_DIRECTORY
- OPEN_LDAP

9.1.2 Seeding a Registry

Some Oracle Fusion Middleware components require that certain users and groups be present in the IBM WebSphere identity store. To ensure that this requirement is met, use any tools to seed the required data; in particular, you can use an LDIF file and the LDAP utility `bulkload` to load users and groups into the identity store. Here is a sample LDIF file:

```
dn: cn=OracleSystemUser,dc=com
userPassword: welcome1
sn: OracleSystemUser
cn: OracleSystemUser
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top

dn: cn=OracleSystemGroup,dc=com
cn: OracleSystemGroup
objectclass: groupOfUniqueNames

dn: cn=Administrators,dc=com
cn: Administrators
objectclass: groupOfUniqueNames

dn: cn=SystemMDBRole,dc=com
cn: SystemMDBRole
objectclass: groupOfUniqueNames
uniquemember: cn=OracleSystemUser,dc=com
```

9.2 Configuring the Trust Association Interceptor

HTTP clients can pass identity information to WebSphere Application Server using the Trust Association Interceptor (TAI). OPSS uses TAI as the asserter that intercepts calls coming into WebSphere cells to support identity propagation across containers and cells.

To configure TAI, proceed as follows:

1. Login to the IBM WebSphere Administrative Console.
2. Select **Security > Click Global Security**.
3. In the opened page, navigate to **Authentication**.
4. Expand **Web** and **SIP** security, and click **Trust Association**.
5. Check the box **Enable Trust Association** and save your changes.
6. Return to the Trust Association page and click **Additional Properties > Interceptors**.
7. Click **New**.
8. In the **Interceptor Class Name** box, enter the following string:

   ```
   oracle.security.jps.was.providers.trust.TrustServiceAsserterTAI
   ```

   This class is packaged in the JAR file `jps-was.jar`.
9. Save your changes.

9.3 Migrating Policies at Deployment

The migration of application policies at deployment is controlled by several parameters configured in the file `META-INF/opss-application.xml`. For an example of this file, see Sample opss-application File. To reassociate the policy store after deployment, see Reassociating Policies with reassociateSecurityStore.

The supported parameters, including configuration examples, are explained in the following sections:

- jps.policystore.migration
- jps.policystore.applicationid
- jps.policystore.removal

Note that the following parameters are not supported on IBM WebSphere:

```
JpsApplicationLifecycleListener
Jps.apppolicy.idstoreartifact.migration
Jps.policystore.migration.validate.principal
```

9.3.1 jps.policystore.migration

This parameter specifies whether the migration should take place, and, when it does, whether it should merge with or overwrite matching policies present in the target store.

On IBM WebSphere, it is configured as illustrated in the following fragment:

```
<service type="POLICY_STORE">
 <property name="jps.policystore.applicationid" value="stripeid" />
 <property name="jps.policystore.migration" value="overwrite" />
 <property name="jps.policystore.removal" value="off" />
</service>
```

For more details about this parameter, see *Oracle Fusion Middleware Application Security Guide*.

9.3.2 jps.policystore.applicationid

This parameter specifies the target stripe into which policies are migrated.

On IBM WebSphere, it is configured as illustrated in the following fragment:

```
<service type="POLICY_STORE">
 <property name="jps.policystore.applicationid" value="stripeid" />
 <property name="jps.policystore.migration" value="overwrite" />
 <property name="jps.policystore.removal" value="off" />
</service>
```

For more details about this parameter, see *Oracle Fusion Middleware Application Security Guide*.

9.3.3 jps.policystore.removal

This parameter specifies whether the removal of policies at undeployment should *not* take place.

On IBM WebSphere, it is configured as illustrated in the following fragment:

```
<service type="POLICY_STORE">
 <property name="jps.policystore.applicationid" value="stripeid" />
 <property name="jps.policystore.migration" value="overwrite" />
 <property name="jps.policystore.removal" value="off" />
</service>
```

For more details about this parameter, see *Oracle Fusion Middleware Application Security Guide*.

9.4 Migrating Credentials at Deployment

The migration of application credentials at deployment is controlled by a parameter configured in the file `META-INF/opss-application.xml`. For an example of this file, see Sample opss-application File.

The supported parameter, including a configuration example, are explained in the following section:

- jps.credstore.migration

Note that the following parameter is not supported on IBM WebSphere:

`jps.ApplicationLifecycleListener`

9.4.1 jps.credstore.migration

This parameter specifies whether the migration should take place, and, when it does, whether it should merge with or overwrite matching credentials present in the target store.

On IBM WebSphere, it is configured as illustrated in the following fragment:

```
<service type="CREDENTIAL_STORE">
 <property name="jps.credstore.migration" value="overwrite" />
</service>
```

Setting `jps.credstore.migration` to `overwrite` requires that the system property `jps.app.credential.overwrite.allowed` be set to `true`.

For more details about this parameter, see *Oracle Fusion Middleware Application Security Guide*.

9.5 Reassociating Policies with reassociateSecurityStore

For complete details about the scrip `reassociateSecurityStore` to reassociate the policy store, see *Oracle Fusion Middleware Application Security Guide*. Since this script is likely to run for some time, to avoid exceptions, one may need to reset the default connection to the server timeout to an appropriate larger value.

To reassociate a security store for Oracle Entitlements Server, use the command `configureSecurityStoreWas` as explained in Section 10.3.3.

9.6 Deployment Mode

On IBM WebSphere, deployment is supported *only* in online mode; no offline deployment is supported.

9.7 Configuring the JpsFilter and the JpsInterceptor

On IBM WebSphere, both the JpsFilter and the JpsInterceptor must be manually configured.

For the properties supported and configuration examples, see *Oracle Fusion Middleware Application Security Guide*.

9.8 Using System Variables in Code Source URLs

The system variables `oracle.deployed.app.dir` and `oracle.deployed.app.ext` can be used to specify a URL independent of the platform. For a configuration example using these variables, see *Oracle Fusion Middleware Application Security Guide*.

9.9 Sample opss-application File

The following sample illustrates the contents of the `opss-application.xml` file.

```
<?xml version="1.0" encoding="UTF-8" standalone='yes'?>
<opss-application
xmlns="http://xmlns.oracle.com/oracleas/schema/11/opss-application-11_1.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.oracle.com/oracleas/schema/11/opss-application-11
_1.xsd" schema-major-version="11" schema-minor-version="1">
  <services>
    <service type="POLICY_STORE">
      <property name="jps.policystore.applicationid" value="stripeid" />
      <property name="jps.policystore.migration" value="MERGE" />
    </service>
    <service type="CREDENTIAL_STORE">
      <property name="jps.credstore.migration" value="MERGE" />
    </service>
  </services>
</opss-application>
```

9.10 About the File web.xml

The element <auth-method> in a web.xml file is WebLogic-specific and not supported on IBM WebSphere; if found, it must be replaced with the equivalent functionality supported for IBM WebSphere's web.xml files.

9.11 Executing Common Audit Framework wsadmin Commands

To run audit commands provided by Oracle Fusion Middleware's Common Audit Framework, you need to do the following:

1. Start the Oracle Fusion Middleware wsadmin command-line shell.
2. Prefix the audit commands with the keyword Audit. For example:

```
wsadmin> Audit.getAuditPolicy()
wsadmin> Audit.setAuditPolicy()

wsadmin> Audit.setAuditRepository
(['switchToDB'],['dataSourceName'],['interval'])
(see Section 9.13, "Creating a Data Source" for a related topic)
```

For details about the audit commands, see the *Oracle Fusion Middleware Application Security Guide*.

9.12 Configuring Audit of Federation Events

This section contains these topics:

- Enabling Auditing of Federation Events
- Moving Oracle Identity Federation Audit Records to Database

9.12.1 Enabling Auditing of Federation Events

To enable auditing of identity federation events, ensure that the filter preset is set to "High". At the default level ("Low"), federation events do not generate audit records.

9.12.2 Moving Oracle Identity Federation Audit Records to Database

Problem

When you configure the audit service to move audit records to the database, the Oracle Identity Federation busstop file at: %DOMAIN_HOME%/servers/%INSTANCE_NAME%/logs/auditlogs/OIF is updated. However these audit records are not populated in the database.

Resolution

To resolve this, enter the wsadmin scripting environment and run the following command:

```
wsadmin>sts_commands.putStringProperty("/notifierconfig/CommonAuditListenerConfig/
auditbusstop","%DOMAIN_HOME%/logs/%INSTANCE_NAME%/auditlogs")
```

This action should result in a "Command was successful." message.

9.13 Creating a Data Source

To create a JDBC data source in a WebSphere cell, proceed as follows:

1. Log in to the WebSphere Console and navigate to **Resources, JDBC, DataSources**.
2. Select the appropriate **Scope** from the pull-down list.
3. Click the button **New** to display the Create a data source page, and go through the steps listed on the left panel.
4. In step 1, enter a **Data Source Name** and a **JNDI Name**; note that the Scope box is read-only and contains the scope selected earlier on. Click **Next** to go to the next step.
5. In step 2.1, set the **Database Type** to **Oracle**, **Implementation Type** to **Connection Pool Data Source**, and enter a **Name** for the provider. Click **Next** to go to the next step.
6. In step 2.2, ensure that the path designated by the variable **ORACLE_JDBC_DRIVER_PATH** is correctly set. Click **Next** to go to the next step.
7. In step 3, set JDBC **URL** to the appropriate value; a sample value is `jdbc:oracle:thin:@xyz12345.example.com:4321:orcl`. Click **Next** to go to the next step.
8. In step 4, click the link **Global J2C Authentication Alias** to display the page Data Sources > JAAS - J2C Authentication Data.
9. In that page, click **New** to display the New page.
10. In the New page, enter an **Alias**, and set **User ID** and **Password** to the user name and password of the data base user. Click **OK** to go back to the JAAS-J2C Authorization page.
11. In that page, if necessary, expand the Message box and click **Save.**
12. Use the Previous button on your browser to go back to the page in step 4 above. To be able to see the authentication alias you entered, refresh the page by clicking the Previous and Next buttons on your browser.
13. Set **Component-Managed Authentication Alias** and **Container-Managed Authentication Alias** to the authentication alias you entered (which should now show on the pull-down lists), and **Mapping-Configuration Alias** to **DefaultPrincipalMapping**. Click **Next**.
14. Click **Finish** and then **Save**, to save the specified data source.
15. To validate the newly created data source, navigate to the **DataSource** page and click **Test Connection**.

Note: Some of the steps in the preceding procedure can be accomplished in pages not referenced in the procedure; examples of these pages are the **Creating a JDBC Provider** and **Creating J2C Authentication Data** pages.

10

Managing Oracle Entitlements Server on IBM WebSphere

This chapter contains information about installing and managing Oracle Entitlements Server on IBM WebSphere.

This chapter contains the following sections:

10.1 Overview of Oracle Entitlements Server Installation on IBM WebSphere

Oracle Entitlements Server is a fine-grained authorization and entitlement management solution that can be used to precisely control the protection of application resources. It simplifies and centralizes security for enterprise applications and SOA by providing comprehensive, reusable, and fully auditable authorization policies and a simple, easy-to-use administration model. For more information, see "Introducing Oracle Entitlements Server" in *Oracle Fusion Middleware Administrator's Guide for Oracle Entitlements Server*.

Oracle Entitlements Server 11*g* Release 2 (11.1.2.1.0) includes two distinct components:

- **Oracle Entitlements Server Administration Server**: this component is included in the Oracle Identity and Access Management 11*g* Release 2 (11.1.2.1.0) installation.
- **Oracle Entitlements Server Client (Security Module)**: this component has its own installer and is not included in the Oracle Identity and Access Management 11*g* Release 2 (11.1.2.1.0) installation and does not require IBM WebSphere.

10.2 Installation and Configuration Roadmap for Oracle Entitlements Server on IBM WebSphere

Table 10–1 lists the tasks for installing and configuring Oracle Entitlements Server.

Table 10–1 Installation and Configuration Flow for Oracle Entitlements Server

No.	Task	Description
1	Review the system requirements and certification documents to ensure that your environment meets the minimum installation requirements for the components you are installing.	For more information, see Section 2.1, "Task 1: Review the System Requirements and Certification Information."
2	Obtain the necessary software.	For more information, see Section 2.2, "Task 2: Obtain the Necessary Software Media or Downloads."
3	Install Oracle Database for the Oracle Entitlements Server policy store and create and load the appropriate schemas for Oracle Entitlements Server.	When using Oracle Database for Oracle Entitlements Server, you must create schemas for Oracle Entitlements Server using Oracle Fusion Middleware Repository Creation Utility. For more information, see Section 2.3, "Task 3: Identify a Database and Install the Required Database Schemas."
4	Install IBM WebSphere.	For more information, see Section 2.4, "Task 4: Install the IBM WebSphere Software."
5	Review the special instructions for installing Oracle Fusion Middleware with IBM WebSphere.	For more information, see Section 2.6.1, "Special Instructions When Installing Oracle Identity and Access Management with IBM WebSphere".
6	Install the Oracle Identity and Access Management software, including Oracle Entitlements Server.	Oracle Entitlements Server is included in the Oracle Identity and Access Management Suite. You can use the Oracle Identity and Access Management 11*g* Installer to install Oracle Identity and Access Management Suite. For Oracle Entitlements Server, select the **AS Common Schemas - Oracle Platform Security Services** schema. By default, the **AS Common Schemas - Metadata Services** schema is also selected. For more information, see Section 2.6, "Task 6: Install Oracle Identity and Access Management Suite".
7	Configure Oracle Entitlements Server Administration Server using the Oracle Fusion Middleware Configuration Wizard.	For more information, see Section 10.3, "Configuring Oracle Entitlements Server Administration Server".
8	Install the Oracle Entitlements Server Client Software.	For more information, see Section 10.4, "Installing Oracle Entitlements Server Client".
9	Configure the Oracle Entitlements Server Client.	For more information, see Section 10.5, "Configuring IBM WebSphere Security Module".
10	Get stared with Oracle Entitlements Server.	For more information, see Section 10.8, "Getting Started with Oracle Entitlements Server After Installation".

10.3 Configuring Oracle Entitlements Server Administration Server

This section discusses the following topics:

- Prerequisites
- Configuring Oracle Entitlements Server in a New IBM WebSphere Cell

- Configuring Security Store for Oracle Entitlements Server Administration Server
- Configuring the Identity Store
- Starting the Administration Server
- Verifying Oracle Entitlements Server Administration Server Configuration

10.3.1 Prerequisites

Oracle Entitlements Server Administration Server must already be installed, as described in Section 10.2, "Installation and Configuration Roadmap for Oracle Entitlements Server on IBM WebSphere."

10.3.2 Configuring Oracle Entitlements Server in a New IBM WebSphere Cell

A special version of the Oracle Fusion Middleware Configuration Wizard is used to configure Oracle Entitlements Server Administration Server in an IBM WebSphere environment. For more information about using this wizard, see Section 2.8, "Task 8: Configure Your Oracle Identity and Access Management Components in a New IBM WebSphere Cell".

For complete information about configuring the Administration Server, see "Configuring Oracle Entitlements Server in a New WebLogic Domain" in *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*. When configuring the Administration Server in an IBM WebSphere environment, note the following applies:

- Select template Oracle Entitlements Server **for Admin Server - 11.1.1.0 [Oracle_IDM1]**. The following options are also selected by default: **Oracle Platform Security Service 11.1.1.0**, **Oracle JRF 11.1.1.0**.
- In JDBC configuration screen, set the correct **service_name**, **hostname, user_name**, **password & port** for the OPSS data-source.
- When the IBM WebSphere cell is created and the Oracle Entitlements Server template is selected, SSL is enabled by default. The DemoTrust certificate is used as the trust certificate, and a certificate signed by DemoTrust is used as the keystore certificate. In a production environment, change the SSL configuration to use a custom certificate.

10.3.3 Configuring Security Store for Oracle Entitlements Server Administration Server

You must run the `configureSecurityStoreWas.py` script to configure the security store for Oracle Entitlements Server Administration Server. For more information about `configureSecurityStoreWas.py` script, see Section 2.9, "Task 9: Configure the Database Security Store". However, when using the script to configure Oracle Entitlements Server, you do not use `--config IAM` option.

The `configureSecurityStoreWas.py` script is located in the `<IAM_HOME>\common\tools` directory. For example, to obtain help use the `--help` option as follows:

```
<IAM_HOME>\common\bin\wsadmin.sh
<IAM_HOME>\common\tools\configureSecurityStoreWas.py --help
```

Configure the security store for Oracle Entitlements Server Administration Server as follows:

- On UNIX:

```
./wsadmin.sh -lang jython -profileName DEPLOYMENT_MANAGER_PROFILE_NAME -f
```

```
IAM_HOME/common/tools/configureSecurityStoreWas.py -d
PATH_TO_DEPLOYMENT_MANAGER_CELL_DIRECTORY -t DB_ORACLE -j cn=jpsroot -m create
--passcode OPSS_SCHEMA_PASSWORD --wasadmin WAS_ADMIN_USERNAME
```

For example:

```
./wsadmin.sh -lang jython -profileName Dmgr01 -f
IAM_HOME/common/tools/configureSecurityStoreWas.py -d
IAM_HOME/was/install/was6076/profiles/Dmgr01/config/cells/DefaultCell01 -t
DB_ORACLE -j cn=jpsroot -m create --passcode opsschemapassword --wasadmin
WAS_ADMIN_USERNAME
```

Note: If the password is not specified at the command line, `configureSecurityStoreWas.py` script will prompt for the datasource password and accept user input.

Table 10–2 lists the valid `configureSecurityStoreWas` parameters in an IBM WebSphere environment.

Table 10–2 configureSecurityStoreWas Parameters

Parameter	Description
`--config`	The configuration mode of the domain. For example: `IAM`. It is optional, default value is `None`. ■ If `--config` is specified, OES Admin Server will be configured in *mixed mode*, meaning it can only distribute policies to Security Modules in non-controlled mode and controlled pull mode. For example: If the OES Administration Server is deployed in the domain where other Oracle Identity and Access Management components (OIM, OAM, OAAM, OPAM, or OIN) are deployed, then the domain is configured in mixed mode. In this case, the OES Administration Server is used for managing the Oracle Identity and Access Management policies only. It should not be used to manage the policies for any other applications protected by OES Security Modules. ■ If `--config` is not specified, OES Admin Server will be configured in *non-controlled mode*, meaning it can distribute policies to Security Modules in controlled push mode. For example: If you want to use OES Administration Server to manage custom applications which are protected by OES Security Modules, then the OES Administration Server must be deployed in a domain with non-controlled distribution mode.
`--datasource`	The data source of security store configured in domain. Default value is `opss-DBDS`.
`-d domaindir`	Location of the OES Administration Server Domain.
`--farmname`	The security store farm name. It is optional, default value is cn=IAM.
`fixjse`	Use `fixjse` to update the domain's Database Security Store credentials used for access by JSE tools.
`--help`	Prints usage message.

Table 10–2 (Cont.) configureSecurityStoreWas Parameters

Parameter	Description
join	Use join if you want to use an existing database security store for the domain.
-j jpsroot	The distinguished name of jpsroot. It is optional, default value is cn=jpsroot.
--keyfilepath	The directory containing the encryption key file ewallet.p12. If -m join is specified, this option is mandatory.
--keyfilepassword	The password used when the domain's key file was generated. If keyfilepassword is not supplied on the command line, the user is prompted for the password. If -m join is specified, this option is mandatory.
-m mode create	Valid values are create, fixjse, join, validate. For example, use create if you want to create a new database security store.
--passcode	The OPSS schema password.
-t servertype	The policy store type. For example: DB_ORACLE, DB_DERBY, or OID. It is optional, default value is DB_ORACLE.
--username	The user name of the OPSS schema. If -m fixjse is specified, this option is mandatory.
validate	Use validate to verify whether the Security Store has been configured correctly. This command validates diagnostics data created during initial creation of the Security Store.
validatefix	Use validatefix to fix diagnostics data present in the Security Store.
--wasadmin	The admin user name of the WAS domain. Optional parameter used when -m create is specified. When passed, this user will be granted the SystemAdmin and APMAdmin role.

Configuring the Database Security Store Using the Join Option

When using -m join option to configure the database, you must first export the domain encryption key from a domain in the same logical Oracle Identity and Access Management deployment already configured to work with the database security store, and then run the configureSecurityStoreWas.py script. This requires you run exportEncryptionKey on the first domain, and then use the ewallet.p12 file generated as a parameter for -keyfile. The password (in exportEncryptionKey) is required as the parameter for -keyfilepassword. For more information and examples, see "Configuring the Database Security Store" in *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.

10.3.4 Configuring the Identity Store

You must configure the identity store for Oracle Platform Security Services (OPSS). For more information, see Section 2.10, "Task 10: Configure the Identity Store".

10.3.5 Starting the Administration Server

Start the Administration Server (OracleAdminServer) by running the following command on the command line:

- **UNIX**

```
profiles/profile_name/bin/startServer.sh OracleAdminServer
```

For more information, see Section 2.11, "Task 11: Start the IBM WebSphere Servers."

10.3.6 Verifying Oracle Entitlements Server Administration Server Configuration

To verify that your Oracle Entitlements Server Administration Server configuration was successful, use the following URL to log in to the Oracle Entitlements Server Administration Console:

```
http://hostname:port/apm/
```

where `hostname` is the DNS name or IP address of the Administration Server and `port` is the address of the port on which the Administration Server listens for requests.

For more information, see "Accessing the Administration Console" in *Oracle Fusion Middleware Administrator's Guide for Oracle Entitlements Server*.

10.4 Installing Oracle Entitlements Server Client

This section discusses the following topics:

- Prerequisites
- Obtaining Oracle Entitlements Server Client Software
- Installing Oracle Entitlements Server Client
- Applying a Patch

10.4.1 Prerequisites

Before installing the Oracle Entitlements Server Client software, you must install and configure Oracle Entitlements Server Administration Server as described in Section 10.2, "Installation and Configuration Roadmap for Oracle Entitlements Server on IBM WebSphere".

10.4.2 Obtaining Oracle Entitlements Server Client Software

For information about where to download the Oracle Entitlements Server Client software, refer to the *Oracle Fusion Middleware Download, Installation, and Configuration Readme* Files on the Oracle Technology Network (OTN): `http://download.oracle.com/docs/cd/E23104_01/download_readme.htm`.

10.4.3 Installing Oracle Entitlements Server Client

To install Oracle Entitlements Server Client 11*g* Release 2 (11.1.2.1.0) on IBM WebSphere, follow the instructions for installing on Oracle WebLogic Server. For information, see "Installing Oracle Entitlements Server Client" in *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.

10.4.4 Applying a Patch

You may need to apply a patch if you have installed Oracle Entitlements Server Client software in a separate Middleware Home than the Oracle Entitlements Server Administration Server. For information, see "Applying a Patch Using OPatch" in *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.

10.5 Configuring IBM WebSphere Security Module

You can configure the WebSphere Security Module in a JRF environment or in a non-JRF environment. Depending on the option you select complete one of the following:

- Configuring WebSphere Security Module in a Non-JRF Environment
- Configuring WebSphere Security Module in a JRF Environment

For information about using a Security Sockets Layer (SSL) connection and configuring third party digital certificate with a WebSphere Security Module, see "Using Default or Third Party Digital Certificates" in *Oracle Fusion Middleware Administrator's Guide for Oracle Entitlements Server*.

10.5.1 Configuring WebSphere Security Module in a Non-JRF Environment

To configure the WebSphere Security Module in a non-JRF environment, complete the following steps:

1. Create a new application server using the IBM WebSphere console and name it `OesServer`.
2. Start the Oracle Entitlements Server (`OesServer`) you created for IBM WebSphere.
3. Open the `smconfig.prp` file in a text editor and specify the pd client port and the pd app client context. The pd client port number is the SSL port number of the IBM WebSphere application server and pd app client context is the location where the was-client.jar is deployed. For example:

   ```
   oracle.security.jps.pd.was.client.appcontext=pd-client
   oracle.security.jps.pd.clientPort=8002
   ```

4. Run the `config.sh` command as follows:

   ```
   $OES_CLIENT_HOME/oessm/bin/config.sh -smType was -smConfigId mySM_WAS
   -serverLocation WAS_HOME
   ```

   `WAS_HOME` is the location of the IBM WebSphere Application Server.

   For any distribution mode you choose, you must specify the IBM WebSphere server user name and password, when prompted.

   In controlled push mode, you will be prompted for Oracle Entitlements Server Administration Server user name, Oracle Entitlements Server Administration Server password, and a new key store password for enrollment.

   In non-controlled and controlled-pull modes, you will be prompted for Oracle Entitlements Server schema user name and password.

   Table 10–3 describes the parameters you specify on the command line.

Table 10–3 IBM WebSphere Security Module Parameter

Parameter	Description
`smType`	Type of security module instance you want to create. For example, `was`.
`smConfigId`	Name of the security module instance. For example, `mySM_WAS`.
`serverLocation`	Location of the IBM WebSphere Server.

Note: Controlled-push distribution is the default distribution mode for IBM WebSphere Security Module a non-JRF environment.

5. Configure SSL for the IBM WebSphere application server as follows:

   a. Import the Oracle WebLogic Server demo trust certificate into IBM WebSphere node default trust keystore and cell default trust keystore by using keytool to export WLS demo trust certificate from WLS demo trust keystore file, or `OES trust.jks` file into a `.der`, as shown in the following example:

   ```
   keytool -exportcert -keystore $OES_CLIENT_HOME/oessm/enroll/DemoTrust.jks
   -alias wlscertgencab -file ~/was.der
   ```

   b. Import the `was.der` file into WAS node default trust keystore and cell default trust keystore as follows:

      – You may find the import in IBM WebSphere Administration Server console:

        security > SSL certificate and key management > Key stores and certificates > <*NodeDefaultTrustStore*> <*CellDefaultTrustStore*> (select one name) **> Signer certificates**.

      – Click **Add**.

      – Enter an alias. For example, **WAS**.

      – Choose the `.der` file that you exported earlier, and select data type as **DER**.

   c. Import the issued private key into the IBM WebSphere node default keystore as follows:

      – You may find the private key to import in IBM WebSphere Administration Server console:

        security > SSL certificate and key management > Key stores and certificates > NodeDefaultKeyStore > Personal certificates.

      – Click **Import**.

      – Select Keystore and enter the path to the keystore file (located in `OES_CLIENT_HOME/oes_sm_instances/mySM_WAS/security/identity.jks`)

      – Select **JKS** as type and enter the password you used to create the keystore file.

      – The certificate alias name is the same name as the hostname.

   Note: You must import demo trust certificate into two trust stores for the WAS ND edition. For the private key, you must import one keystore.

   d. Enable Inbound SSL for the server running IBM WebSphere Security Module as follows:

      – In the IBM WebSphere administration console, go to **Security > SSL certificate and key management > Manage endpoint security configurations**.

- Expand inbound tree to **get:Inbound > DefaultCell(CellDefaultSSLSettings) > nodes > DefaultCellFederatedNode > servers > <***server name running IBM WebSphere Security Module***>** and select the server.
- In the General Properties page, select **Override inherited values**.
- From the **SSL configuration** list, select **NodeDefaultSSLSettings**.
- Click **Update certificate alias list** button and then choose the private key alias you (customer) wanted to use for SSL in the **Certificate alias in key store** list.
- Click **Apply**.

e. Enable Outbound SSL for the server running IBM WebSphere Security Module, follows:

- In the IBM WebSphere administration console, go to **Security > SSL certificate and key management > Manage endpoint security configurations**.
- Expand outbound tree to **get:Outbound > DefaultCell(CellDefaultSSLSettings) > nodes > DefaultCellFederatedNode > servers > <***server name running IBM WebSphere Security Module***>** and select the server.
- In the General Properties page, select **Override inherited values**.
- From the **SSL configuration** list, select **NodeDefaultSSLSettings**.
- Click **Update certificate alias list** and choose the new imported private key alias in the **Certificate alias in key store** list.
- Click **Apply**.

10.5.2 Configuring WebSphere Security Module in a JRF Environment

To configure WebSphere Security Module in a JRF environment, complete the following steps:

1. Configure IBM WebSphere Application Server, as described in Chapter 2 and *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server*.

   Note: In the **Add Products to Cell** screen, ensure that you select **Oracle JRF for WebSphere - 11.1.1.0 [oracle_common]**

2. Run the `config.sh` command as follows:

   ```
   $OES_CLIENT_HOME/oessm/bin/config.sh -smType was -smConfigId mySM_WAS
   -serverLocation WAS_HOME -profileName dmgr_profile_name
   ```

   `WAS_HOME` is the location of the IBM WebSphere Application Server.

   For any distribution mode you choose, you must specify the IBM WebSphere server user name and password, when prompted.

   In controlled push mode, you will be prompted for Oracle Entitlements Server Administration Server user name, Oracle Entitlements Server Administration Server password, and a new key store password for enrollment.

In non-controlled and controlled-pull modes, you will be prompted for Oracle Entitlements Server schema user name and password.

Table 10–4 describes the parameters you specify on the command line.

Table 10–4 IBM WebSphere Security Module Parameter

Parameter	Description
`smType`	Type of security module instance you want to create. For example, `was`.
`smConfigId`	Name of the security module instance. For example, `mySM_WAS`.
`serverLocation`	Location of the IBM WebSphere Server.
`host`	Specify the WebSphere host name.
`port`	Specify the WebSphere Node Manager port. For example: `8882`
`user`	Specify the WebSphere username. For example: `websphere`
`password`	Specify the WebSphere password.

Note: Non-controlled distribution is the default distribution mode for IBM WebSphere Security Module in a JRF environment.

After configuring WebSphere Security Module in a JRF environment, you must set up a connection to Oracle Database.

Setting Up Connection to Oracle Database

For setting up connection to Oracle Database, complete the following steps:

1. Create a JDBC Data Source using the WebLogic Server Administration Console. For more information, see "Create JDBC generic data sources" in *Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help*.
2. Open the `jps-config.xml` file.

   File is located in *<OES_DOMAIN_HOME>*`/config/oeswlssmconfig` directory (on UNIX).
3. Locate `pdp.service` and replace the existing `jdbc.url` property with the following property:

   ```
   <property value="jdbc/APMDBDS" name="datasource.jndi.name"/>
   ```

   Note: *jdbc/APMDBDS* is the name of the JDBC datasource used for the Oracle Entitlements Server.
4. Delete the following properties:
   - `jdbc.driver`
   - `jdbc.url`
   - `bootstrap.security.principal.key`
   - `bootstrap.security.principal.map`

5. Save the `jps-config.xml` file.

10.6 Using Oracle Entitlements Server wsadmin Commands on IBM WebSphere

You can run Oracle Entitlements Server commands from the IBM WebSphere wsadmin command line interface. For more information, see Section 3.1.3, "Using the Oracle Fusion Middleware wsadmin Commands".

Oracle Entitlements Server commands are documented in *Oracle Fusion Middleware Administrator's Guide for Oracle Entitlements Server*. Oracle Entitlements Server commands are functionally identical on WebLogic Server and IBM WebSphere Server.

10.7 Configuring Security Modules for Other Application Servers

In addition to IBM WebSphere Security Module, Oracle Entitlements Server includes additional Security Modules for other application servers. For more information about configuring them, see "Configuring Security Modules" in *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.

10.8 Getting Started with Oracle Entitlements Server After Installation

After installing Oracle Entitlements Server, refer to the following documents:

- *Oracle Fusion Middleware Administrator's Guide for Oracle Entitlements Server*
- *Oracle Fusion Middleware Developer's Guide for Oracle Entitlements Server*

Note that the information provided in these two documents is generally specific to using Oracle Entitlements Server on Oracle WebLogic Server.

10.9 Configuring High Availability for Oracle Entitlements Server

This section describes how to configure high availability for Oracle Entitlements Server on IBM WebSphere. In this release, the Oracle Entitlements Server Administration Server can be configured as a cluster in an active-active configuration.

This section discusses the following topics:

- Overview of a Cluster Configuration
- Horizontal Cluster Topology
- High Availability Installation and Configuration Roadmap
- Configuring a Two Node Horizontal Cluster

> **See Also:** Section 3.4, "Configuring Oracle Fusion Middleware High Availability on IBM WebSphere"

10.9.1 Overview of a Cluster Configuration

A *cluster* is a set of processes running on a single or multiple computers that share the same workload. After system components have been clustered, they are viewed functionally as a single entity from the perspective of a client for runtime processing and manageability. There is a close correlation between clustering and redundancy. A cluster provides redundancy for a system. With a load-balancing mechanism in place,

the instances are redundant. If any of the instances fail, requests to the failed instance can be sent to the surviving instances. If failover occurs during a transaction in a clustered environment, the session data is retained as long as there is at least one surviving instance available in the cluster.

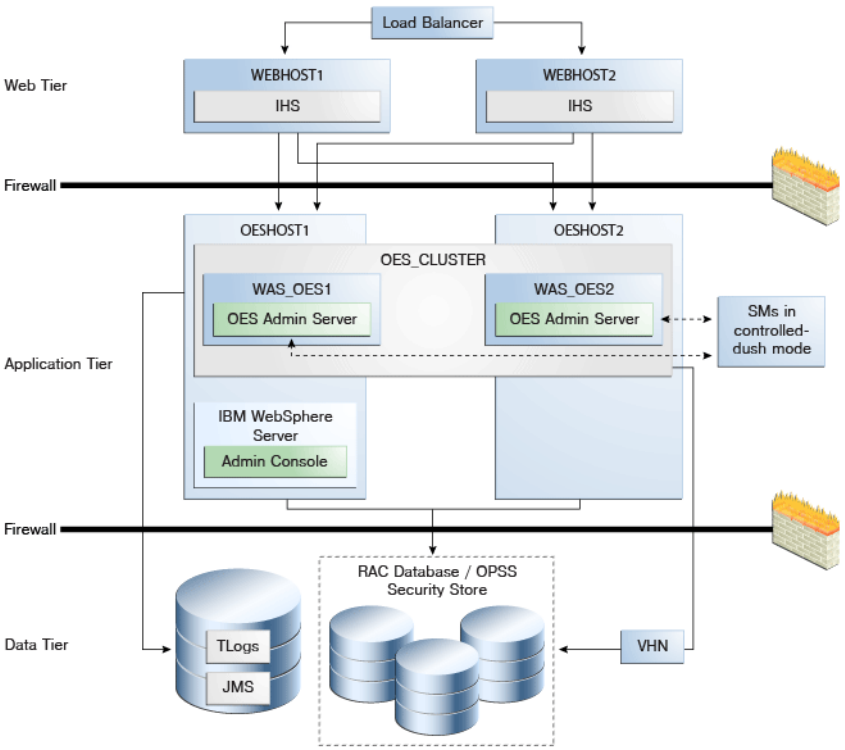
> **See Also:** "Oracle Entitlements Server High Availability Concepts" in *Oracle Fusion Middleware High Availability Guide*.

In this release, the Oracle Entitlements Server Administration Server can be configured in a two-node cluster. Table 10–5 summarizes the components in a typical two-node cluster.

Table 10–5 Overview of Horizontal (2-Node) Clustered Configuration

Deployment Manager Machine	WebSphere Node 2 Machine
1. WebSphere Deployment Manager (DMGR)	WebSphere Node 2
2. WebSphere Node 1	OES_SERVER_2
3. OracleAdminServer	
4. OES_SERVER_1	

Figure 10–1 Oracle Entitlements Server on IBM WebSphere in Controlled-push Mode



The components illustrated in Figure 10–1 are detailed in Table 10–5. On OESHOST1, you see the following installations:

- An Oracle Entitlements Server instance is installed in the WAS_OES1 Managed Server and a APM instance is installed in the WAS_OES1 Managed Server.
- The Oracle RAC database is configured in a JDBC multi data source to protect the instance from Oracle RAC node failure.
- An IBM WebSphere Server Administration Server is installed. Under normal operations, this is the active Administration Server.

On OESHOST2, you see the following installations:

- An Oracle Entitlements Server instance is installed in the WAS_OES2 Managed Server and an APM instance is installed in the WAS_OES2 Managed Server.
- The Oracle RAC database is configured in a JDBC multi data source to protect the instance from Oracle RAC node failure.
- The instances in the WAS_OES1 and WAS_OES2 Managed Servers on OESHOST1 and OESHOST2 are configured as the OES_CLUSTER cluster.

10.9.2 Horizontal Cluster Topology

A typical horizontal cluster topology consists of two nodes. The following is an example configuration.

Deployment Manager (DMGR Host), Node 1

1. Create an IBM WebSphere cell and Oracle-standard local nodes.

   Includes cell, management profiles (contains node with DMGR) and managed profile (contains custom node used to house local servers such as `OracleAdminServer`).

2. If the local node will be used to house layered product or component managed servers, select the product or component template.
3. Create a cluster using the same server used in step 2.
4. Add cluster members as necessary to scale-out.

Remote Host, Node 2

Assumes a cluster has already been created in the cell.

1. Create custom profile and federate node to cell.
2. Add cluster member to existing cluster.

   Select node associated with host.

3. Repeat for each additional remote host.

10.9.3 High Availability Installation and Configuration Roadmap

This section describes the process to configure a typical two-node horizontal cluster for Oracle Entitlements Server. For complete information about installing and configuring Oracle Identity and Access Management on IBM WebSphere, see Chapter 2. For more information about configuring the Administration Server on IBM WebSphere, see Section 10.3, "Configuring Oracle Entitlements Server Administration Server".

Table 10–6 lists the key tasks for installing and configuring high availability for Oracle Entitlements Server.

Table 10–6 Key Tasks: High Availability Installation and Configuration Flow for Oracle Entitlements Server

No.	Task	Description
1	Review the information about how to install Oracle Identity and Access Management on IBM WebSphere and how to configure a high availability environment on IBM WebSphere.	For information about installing Oracle Identity and Access Management on IBM WebSphere, see Chapter 2. For information about configuring a highly available environment on IBM WebSphere, see Section 3.4, "Configuring Oracle Fusion Middleware High Availability on IBM WebSphere".
2	Review the information about configuring high availability for Oracle Entitlements Server.	For more information, see Section 10.9.1, "Overview of a Cluster Configuration" and Section 10.9.2, "Horizontal Cluster Topology".
3	Review the System Requirements and Certification Information.	For more information, see Section 2.1, "Task 1: Review the System Requirements and Certification Information".
4	Install Oracle Database for the Oracle Entitlements Server policy store and create and load the appropriate schemas for Oracle Entitlements Server.	When using Oracle Database for Oracle Entitlements Server, you must create schemas for Oracle Entitlements Server using Oracle Fusion Middleware Repository Creation Utility. For more information, see Section 2.3, "Task 3: Identify a Database and Install the Required Database Schemas".
5	Install IBM WebSphere on OESHOST1 and OESHOST2.	For more information, see Section 2.4, "Task 4: Install the IBM WebSphere Software".
6	Install Oracle Entitlements Server Administration software on OESHOST1 and OESHOST2.	Oracle Entitlements Server is included in the Oracle Identity and Access Management Suite. You can use the Oracle Identity and Access Management 11*g* Installer to install Oracle Identity and Access Management Suite. For Oracle Entitlements Server, select the AS Common Schemas - Oracle Platform Security Services schema. By default, the AS Common Schemas - Metadata Services schema is also selected. For more information, see Section 2.6, "Task 6: Install Oracle Identity and Access Management Suite".
7	Configure the Deployment Manager Machine on the primary host (IBM WebSphere Node 1).	The database security store and identity store are also configured. For more information, see Section 10.9.4.1, "Configure the Deployment Manager Machine (Primary Host, IBM WebSphere Node 1)".
8	Configure the Remote Machine on the secondary host (IBM WebSphere Node 2).	For more information, see Section 10.9.4.2, "Configure the Remote Machine (Secondary Host, IBM WebSphere Node 2)".
9	Restart the Node Agents and IBM WebSphere Servers.	For more information, see Section 10.9.4.3, "Restart the Servers".

10.9.4 Configuring a Two Node Horizontal Cluster

To configure high availability for the Oracle Entitlements Server Administration Server user interface, you must create an IBM WebSphere Server cluster. To configure

a high availability database for the Administration Server user interface, you use multi source data source, Oracle RAC, and other typical elements.

This section discusses the following topics:

- Configure the Deployment Manager Machine (Primary Host, IBM WebSphere Node 1)
- Configure the Remote Machine (Secondary Host, IBM WebSphere Node 2)
- Restart the Servers

10.9.4.1 Configure the Deployment Manager Machine (Primary Host, IBM WebSphere Node 1)

Use the Oracle Fusion Middleware Configuration Wizard to configure Oracle Entitlements Server in a new IBM WebSphere cell. For complete information about using the Oracle Fusion Middleware Configuration Wizard, including information about adding servers and clusters to a cell, see *Oracle Fusion Middleware Configuration Guide for IBM WebSphere Application Server*.

To configure Oracle Entitlements Server in a new IBM WebSphere cell, complete the following steps:

1. Create an IBM WebSphere cell for OES Admin Server by starting the Oracle Fusion Middleware Configuration Wizard. Run the following command from the Oracle Identity and Access Management home:

   (UNIX) *ORACLE_HOME*/common/bin/was_config.sh

   The Welcome screen of the Oracle Fusion Middleware Configuration Wizard appears. Provide the information required to create the cell.

2. In **Add Products to Cell** screen, select **Oracle Entitlements Server for Manage Server - 11.1.1.0**. The following options are also selected by default: **Oracle Platform Security Service 11.1.1.0**, **Oracle JRF 11.1.1.0**.

3. In **Configure JDBC Components Schema** screen, configure JDBC properties for all of the schemas. Set the correct **service_name**, **hostname**, **user_name**, **password**, and **port** for the OPSS data-source.

   For the RAC Database, check **Configure selected component schemas as RAC data source schemas in the next panel**, and click **Next** to go to RAC data source configuration panel.

4. For the RAC data source configuration panel, enter the following information:
   - In the **Service Name** field, enter the Oracle RAC Database service name.
   - In the **User Name** field, enter the OPSS schemas' user name.
   - In the **Password** field, enter the OPSS schemas' password.
   - Provide/add the two RAC nodes' hostnames and the ports.

5. In **Test JDBC - Component Schema** screen, click **Select All** then **Test Connections**.

6. In **Select Optional Configuration** screen, select **Administration Server and Managed Servers, Clusters and Machines**.

7. In **Configure Application Servers** screen, one managed server named **oes_server1** is created automatically. You can rename **oes_server1** and modify the attributes if needed. For the second OES_SERVER, click **Add** and provide a name: **OracleAdminServer**.

8. In the **Configure Clusters** screen, click **Add** to create a cluster. Enter the **Cluster Names**, for example, ***OES_Server_Cluster_1***. Next select ***oes_server1*** as **First cluster member**.
9. Skip the **Configure Additional Cluster Members** and **Configure End Points** screens by clicking **Next**.
10. In **Target Deployments to Cluster or Servers** screen, from left pane select the following Targets individually, then in right pane select the following Applications for each Target:
    - **OES_ServerCluster_1**: oracle.security.apm, DMS Application_11.1.1.3.0, oracle.oes.admin.enroll_11.1.1.3.0, oracle.oes.admin.pd.ssl_11.1.1.3.0, wsil-nonwls
    - **DMGR**: Dmgr DMS Application_11.1.1.1.0
    - **OracleAdminServer**: DMS Application_11.1.1.0, FMW Welcome Page Application_11.1.0.0.0, wsil-nonwls
11. In **Target Services to Clusters or Servers** screen, from left pane select the following **Targets** individually, then select the following libraries from Services list:
    - **OES_ServerCluster_1**: all libraries in Service list
    - **OracleAdminServer**: all libraries in Service list
12. In **Target Custom Services to Servers** screen, accept the default services.
13. In the **Configuration Summary** screen, click **Create**.

    The **Configuring Cell** screen appears and displays progress.
14. After the cell is created, stop the node agent process. See Section 10.9.4.3, "Restart the Servers".
15. Configure the database security store by running the `configureSecurityStoreWas.py` script. See Section 10.3.3, "Configuring Security Store for Oracle Entitlements Server Administration Server".
16. Configure the identity store for Oracle Platform Security Services (OPSS). See Section 2.10, "Task 10: Configure the Identity Store".
17. Start the server manager (Dmgr02). See Section 10.9.4.3, "Restart the Servers".
18. Start the node agent, sync node to custom profile, and start servers (Custom02). See Section 10.9.4.3, "Restart the Servers".

10.9.4.2 Configure the Remote Machine (Secondary Host, IBM WebSphere Node 2)

To configure Oracle Entitlements Server in a new IBM WebSphere cell, complete the following steps:

1. Start the Oracle Fusion Middleware Configuration Wizard to federate the machine and configure its cell. Run the command from the Oracle Identity and Access Management home:

   (UNIX) *ORACLE_HOME*/common/bin/was_config.sh
2. In the **Select Configuration Option** screen, select the **Federate Machine** and **Configure Cell** options.
3. In the **Specify Profile and Node Name Information** screen, specify the Application Server Profile Name and Application Server Node Name.

4. In the **Specify Deployment Manager Information** screen, enter information about the existing Deployment Manager System to federate machine.
5. Skip the following wizard screens by clicking **Next**: **Add Products to Cell**, **Configure JDBC Component Schema**, **Test JDBC Component Schema**.
6. In **Select Optional Configuration** screen, select **Application Servers, Clusters and End Points** and **Deployments and Services**.
7. Skip the following wizard screens by clicking **Next**: **Configure Application Servers**, **Configure Clusters**.
8. In **Configure Additional Cluster Members** screen, complete the fields to add the remote cluster member to the cluster created on the primary host. For example:
   a. Click **Add**.
   b. **Name**: provide the name for the cluster's remote (secondary) host, such as ***oes_server2***.
   c. **Node Name**: select node agent for ***oes_server2*** from list, such as ***WebSphereNode2***.
   d. **Cluster Name**: select ***OES_ServerCluster_1*** from list.
9. Skip the following wizard screens (accept the default values) by clicking **Next**: **Configure End Points**, **Target Deployments to Clusters or Servers**, **Target Services to Clusters or Servers**, **Target Custom Services to Servers**.
10. In **Configuration Summary** screen, click **Extend**.

    The **Configuring Cell** screen appears and displays progress.
11. Stop the node agent process on the secondary host. See Section 10.9.4.3, "Restart the Servers".
12. Start the node agent, sync node to custom profile, and start servers (Custom02). See Section 10.9.4.3, "Restart the Servers"

10.9.4.3 Restart the Servers

To restart the servers:

1. Stop the node and Deployment Manager on Deployment Manager machine. Execute the following from $WAS_HOME/bin:

```
./stopNode.sh -profileName <server_prof_name> -username <wasadmin username>
   -password <password>
./stopManager.sh -profileName <dmgr_prof_name> -username <wasadmin username>
   -password <password>
```

2. Stop the node on IBM WebSphere Node 2 machine. Execute:

```
./stopNode.sh -profileName <server_prof_name> -username <wasadmin username>
   -password <password>
```

3. Start the Deployment Manager, node, and servers on Deployment Manager machine. Execute:

```
./startManager.sh -profileName <dmgr_prof_name>
./startNode.sh -profileName <server_prof_name>
./startServer.sh OracleAdminServer -profileName <server_prof_name>
./startServer.sh <server_name> -profileName <server_prof_name>
```

4. Start the node and OES server on IBM WebSphere Node 2 machine:

```
./startNode.sh -profileName <server_prof_name>
./startServer.sh <server_name> -profileName <server_prof_name>
```

11

Managing Oracle Privileged Account Manager on IBM WebSphere

Most of the conceptual and procedural information contained in the *Oracle Fusion Middleware Administrator's Guide for Oracle Privileged Account Manager* applies to both WebLogic and WebSphere environments.

This chapter provides information that is specific to using Oracle Privileged Account Manager on IBM WebSphere.

The topics include:

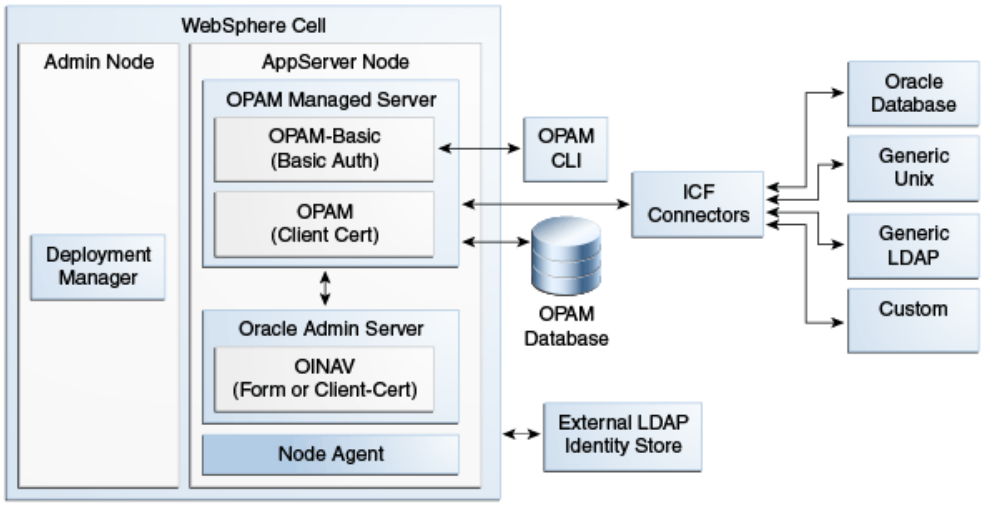
- Differences in How Oracle Privileged Account Manager is Deployed in Oracle Fusion Middleware
- Differences in Getting Started with Administering Oracle Privileged Account Manager
- Differences in Oracle Privileged Account Manager Authorization
- Differences in Adding and Managing an Oracle Privileged Account Manager Server on IBM WebSphere
- Differences in Managing Oracle Privileged Account Manager Auditing and Logging
- Differences in Performing Advanced Configuration Tasks for Oracle Privileged Account Manager on IBM WebSphere
- Differences When Integrating with Oracle Identity Manager
- Differences When Using the Oracle Privileged Account Manager Command Line Tool and REST Interfaces on IBM WebSphere
- Configuring Oracle Privileged Account Manager for High Availability in a Clustered Environment
- Limitations and Known Issues When Using Oracle Privileged Account Manager on IBM WebSphere

11.1 Differences in How Oracle Privileged Account Manager is Deployed in Oracle Fusion Middleware

This section describes the differences in how Oracle Privileged Account Manager on IBM WebSphere is deployed within Oracle Fusion Middleware.

Figure 11–1 illustrates a WebSphere cell configuration:

Figure 11–1 Oracle Privileged Account Manager on IBM WebSphere Deployed Within Oracle Fusion Middleware



As you examine this figure, note that the cell configuration contains two profiles:

- **Deployment Manager profile**: This profile contains an **Admin Node** in which a Deployment Manager server is running.
- **AppServer profile**: This profile contains an **AppServer Node** in which the following servers are running:
  - **OracleAdminServer**: The Oracle Identity Navigator application, which hosts the Oracle Privileged Account Manager Console, is deployed on this server. The chosen authorization mode is either form or client-cert, as required.

    Because IBM WebSphere does not have an embedded LDAP server, you must configure an external LDAP server to serve as an identity store for users, groups, and so forth.
  - **Oracle Privileged Account Manager Managed Server**: Two Oracle Privileged Account Manager applications are deployed on this server. One application uses a basic authorization-mode, which is required for the Oracle Privileged Account Manager command line tool. The other application uses a client-cert authorization mode, which the Oracle Privileged Account Manager Console uses to talk to the Oracle Privileged Account Manager server.

    This server is similar to a WebLogic Managed Server where the data source is targeted for the Oracle Privileged Account Manager database store and where the ICF connectors are configured.

11.2 Differences in Getting Started with Administering Oracle Privileged Account Manager

This section contains information about starting to administer Oracle Privileged Account Manager in an IBM WebSphere environment.

The topics include

- Default Ports
- Starting Oracle Privileged Account Manager on IBM WebSphere

11.2.1 Default Ports

After installing 11*g* Release 2 on IBM WebSphere, Oracle recommends that you become familiar with the following default ports for Oracle Privileged Account Manager in this release:

Table 11–1 Default Ports

Port Type	Default Port	Description
Oracle Privileged Account Manager	18102	Default SSL-enabled port for the Oracle Privileged Account Manager application server (`opam_server1`). In a shiphome (such as an out-of-the-box environment) there are two WebSphere servers relevant to Oracle Privileged Account Manager: ■ The OracleAdminServer in the AppServer node runs Oracle Identity Navigator and the Oracle Privileged Account Manager Console. ■ An additional server in the AppServer node that runs the Oracle Privileged Account Manager application server (`opam_server1`).
OracleAdminServer	9002	Default non-SSL port for the OracleAdminServer application server (where Oracle Identity Navigator and the Oracle Privileged Account Manager Console are deployed).
OracleAdminServer responds to SSL	9003	Default SSL-enabled port for the OracleAdminServer application server (where Oracle Identity Navigator and the Oracle Privileged Account Manager Console are deployed).

11.2.2 Starting Oracle Privileged Account Manager on IBM WebSphere

This section provides information about tasks you must perform before starting the Oracle Privileged Account Manager Console on IBM WebSphere.

The topics include

- Before You Begin
- Configuring Oracle Privileged Account Manager on IBM WebSphere
- Setting Up Non-TDE Mode

11.2.2.1 Before You Begin

Before starting Oracle Privileged Account Manager, perform the following step:

Seeding the Identity Store for Oracle Privileged Account Manager

Seeding the identity store is a required task. For more information about seeding the identity store with the necessary Oracle Privileged Account Manager users and groups, see "Preparing the Identity Store" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Privileged Account Manager*.

11.2.2.2 Configuring Oracle Privileged Account Manager on IBM WebSphere

To configure Oracle Privileged Account Manager on IBM WebSphere, perform the following steps from the machine where the Deployment Manager is running:

1. From a command window, set the following:

   On **UNIX**:

   ```
   setenv WAS_HOME
   ```

```
setenv ORACLE_HOME
setenv DMGR_CELL_HOME
```

Note: When setting DMGR_CELL_HOME, use a value that is similar to the following:

$WAS_HOME/profiles/<*DMGR Profile Name*>/config/cells/<*Cell Name*>

2. Go to the *$ORACLE_HOME*/opam/bin directory and run the following script:

   On **UNIX**:

   ```
   opam-was_config.sh
   ```

   Provide the following information when prompted:

   - Deployment Manager Hostname
   - Deployment Manager SOAP Connector Port
   - Deployment Manager Bootstrap Address Port
   - WebSphere Admin Username
   - WebSphere Admin Password

   Note: The port values are located in this file:

   $WAS_HOME/profiles/<*Dmgr profile*>/properties/portdef.props

3. After running the script, you must:
   1. Stop the Deployment Manager
   2. Start the Servers

Stop the Deployment Manager

Stop the Deployment Manager by navigating to the following directory in the IBM WebSphere home and entering the following command:

On **UNIX**:

profiles/*dmgr_profileName*/bin/stopManager.sh

For example, on a UNIX operating system:

/disk01/IBM/WebSphere/AppServer/profiles/Dmgr01/bin/stopManager.sh

Note: If you are running the stopManager.sh (or stopManager.bat) command from the *WAS_HOME*/bin directory, then you must specify the -profileName parameter. For example, on a UNIX operating system:

WAS_HOME/bin/stopManager.sh -profileName *dmgr_profileName*

Start the Servers

After stopping the node and IBM WebSphere Deployment Manager, you can start the Deployment Manager, node, and servers as described in Section 2.11, "Task 11: Start the IBM WebSphere Servers."

> **Note:** when you perform the final step to start any additional servers, be sure to use **opam_server1** as the Oracle Privileged Account Manager server name.

After starting the servers:

- If you enabled Transparent Data Encryption (TDE) mode as described in Section 2.7, "Task 7: Optional: Enabling TDE in Oracle Privileged Account Manager Data Store (For Oracle Privileged Account Manager Users Only)," then you have finished installing and configuring Oracle Privileged Account Manager on IBM WebSphere. No further steps are required. You can now verify the Oracle Privileged Account Manager functionality.
- If you decided *not* to enable TDE mode, then you must complete steps to set up non-TDE mode. Continue to Section 11.2.2.3, "Setting Up Non-TDE Mode" for instructions.

11.2.2.3 Setting Up Non-TDE Mode

> **Note:** Oracle Privileged Account Manager can operate with Oracle Database TDE (Transparent Data Encryption) mode. Oracle strongly recommends to enable the TDE mode for enhanced security.

If you want to disable TDE mode, you must set the flag `tdemode` to `false`.

> **Note:** The steps described in this section are required only if you chose to skip Section 2.7, "Task 7: Optional: Enabling TDE in Oracle Privileged Account Manager Data Store (For Oracle Privileged Account Manager Users Only)".

Complete the following steps to disable TDE mode:

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 modifyglobalconfig
   -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-basic
   ```

Note: You can enable or disable TDE mode at any point after installing and configuring Oracle Privileged Account Manager. For more information about changing the TDE mode at a later time, refer to the "Securing Data On Disk" topic in the *Oracle Fusion Middleware Administrator's Guide for Oracle Privileged Account Manager*.

When the scripts are finished running, you will be finished installing and configuring Oracle Privileged Account Manager on IBM WebSphere. No further steps are required, and you can verify the Oracle Privileged Account Manager functionality.

11.3 Differences in Oracle Privileged Account Manager Authorization

This section contains information about understanding Oracle Privileged Account Manager authorization on IBM WebSphere.

11.3.1 Administration Role Types

Most of the information in the "Administration Role Types" section of the *Oracle Fusion Middleware Administrator's Guide for Oracle Privileged Account Manager* is applicable for both WebLogic and WebSphere environments. However, the following information is specific to understanding the bootstrap user in the WebSphere environment.

After installation, the default administrator is the `wasadmin` user (also known as the *bootstrap* user) who is a member of the Administrators group. You must use the `wasadmin` user to create and assign users to the Oracle Privileged Account Manager Admin Roles described in Table 2-1 in the *Oracle Fusion Middleware Administrator's Guide for Oracle Privileged Account Manager*. 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, you can use the `wasadmin` user, as the bootstrap user, to map the users from the domain identity store to the Oracle Privileged Account Manager Common Admin Roles detailed 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 `wasadmin` 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 defined in your domain identity store.

11.4 Differences in Adding and Managing an Oracle Privileged Account Manager Server on IBM WebSphere

The "Adding and Managing an Oracle Privileged Account Manager Server" chapter of the *Oracle Fusion Middleware Administrator's Guide for Oracle Privileged Account Manager* advises that you review the Oracle publications listed in Table 4-1 Reference Publications before you start configuring the Oracle Privileged Account Manager server.

If you are adding and managing an Oracle Privileged Account Manager server on IBM WebSphere, review Section 9.1, "IBM WebSphere Identity Stores" for information about the following topics:

- IBM WebSphere concepts and terminology
- Creating a default authenticator in Oracle WebLogic Server
- Configuring the OVD authenticator in Oracle WebLogic Server

11.5 Differences in Managing Oracle Privileged Account Manager Auditing and Logging

This section provides information that is specific to configuring Oracle Privileged Account Manager auditing and logging on IBM WebSphere.

The topics include:

- Configuring Auditing for Oracle Privileged Account Manager
- Configuring Basic Logging for Oracle Privileged Account Manager

11.5.1 Configuring Auditing for Oracle Privileged Account Manager

The procedures for configuring file-based auditing or database-based auditing on an IBM WebSphere server are essentially the same as described in "Configuring Auditing in Oracle Privileged Account Manager" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Privileged Account Manager*; except for the following:

- For both file-based auditing and database-based auditing, when instructed to launch the application server shell, you must launch WSAdmin rather than WLST.
- WebSphere executes commands beginning with **`Audit.`** When performing any of the steps that use a WLST audit command (`getAuditPolicy`, `setAuditPolicy`, `getAuditRepository`, or `setAuditRepository`), you must ensure **`Audit.`** precedes the command name.

  For example, `Audit.getAuditPolicy()` on WebSphere is equivalent to `getAuditPolicy()` on WebLogic.

> **Note:** Refer to Section 8.4, "Setting Up Reporting and Auditing for OAAM on IBM WebSphere" for more information about executing these steps. The steps for Oracle Privileged Account Manager are analogous.

11.5.2 Configuring Basic Logging for Oracle Privileged Account Manager

The procedures for configuring Oracle Privileged Account Manager logging on an IBM WebSphere server is essentially the same as described in "Configuring Basic Logging" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Privileged Account Manager*; but with the following caveats:

- Instead of invoking WLST to run the commands, you must first enter the WSAdmin shell. For more information, visit the following website:

  `http://pic.dhe.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=%2Fcom.ibm.websphere.nd.doc%2Finfo%2Fae%2Fae%2Frxml_commandline.html`

However, instead of invoking the default IBM version of WSAdmin, you must invoke the Oracle version of WSAdmin to obtain support for Oracle commands. You can invoke the Oracle version of WSAdmin from the following location:

```
IAM_HOME/common/bin
```

- To invoke the WLST commands, you must precede each command with **`OracleODL`**. For example,
  - To use the `getLogLevel` command in WLST, you must run

    ```
    getLogLevel(logger="oracle.idm.opam")
    ```

  - To use the command on WebSphere, you must run

    ```
    OracleODL.getLogLevel(logger="oracle.idm.opam")
    ```

- The log messages generated by Oracle Privileged Account Manager's logger (such as, `oracle.idm.opam`) are stored in the following location:

  ```
  $WAS_HOME/profiles/[ProfileName]/[ServerName]/logs/[AppServerName]/
  [AppServerName]-diagnostic.log
  ```

11.6 Differences in Performing Advanced Configuration Tasks for Oracle Privileged Account Manager on IBM WebSphere

This section describes the differences in performing the following advanced configuration tasks for Oracle Privileged Account Manager on IBM WebSphere:

- Differences When Configuring Oracle Privileged Account Manager to Communicate with Target Systems Over SSL
- Differences When Securing Data On Disk

11.6.1 Differences When Configuring Oracle Privileged Account Manager to Communicate with Target Systems Over SSL

To communicate securely over SSL with a target system, the IBM WebSphere 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 IBM WebSphere container in which it runs. To have the IBM WebSphere 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 IBM WebSphere 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 IBM WebSphere 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. To import the certificate into the IBM WebSphere Cell's truststore,
   a. Log in to the IBM WebSphere Console.
   b. Select **Security** > **SSL certificate and key management** > **Key stores and certificates** > **CellDefaultTrustStore** > **Signer certificates** > **Add**.
   c. From the Add screen, enter your hostname into the **Alias** field.
   d. Specify the **Data Type**, as follows:

If the exported certificate is in:	Then select:
BASE64-encoded format	Base 64 encoded ASCII Data
Binary format	Binary DER data

   e. Locate the certificate file to be imported on the local file system where IBM WebSphere is running. Enter the full path and file name into the **File Name** field.
4. Import the file, and then verify that it imported correctly.

Note: For more information about managing Oracle Fusion Middleware security on IBM WebSphere, refer to Chapter 11, "Managing Oracle Privileged Account Manager on IBM WebSphere."

For more general topics and concepts, refer to the *Oracle Fusion Middleware Application Security Guide*.

11.6.2 Differences When Securing Data On Disk

After initial installation, the procedures for enabling or disabling Oracle Database Transparent Data Encryption (TDE) mode for Oracle Privileged Account Manager on IBM WebSphere are essentially the same as described in "Securing Data on Disk" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Privileged Account Manager*.

The only difference is that for both the "Enabling TDE Mode" and "Disabling TDE Mode" sections, the ***OPAM_Server_Url*** must be in the following form:

https://*OPAM_Managed_Server_Hostname*:*OPAM_Managed_Server_SSL_port*/**opam-basic**

11.7 Differences When Integrating with Oracle Identity Manager

If you are configuring Oracle Privileged Account Manager for integration with Oracle Identity Management, the procedures for retrieving and importing the CA Certificate are slightly different than described in "Adding the CA Certificate" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Privileged Account Manager*.

Difference When Retrieving the CA Certificate

In the first step, when you are directed to connect to the Oracle Privileged Account Manager server web service, you must connect to

https://*opamhost*:*opamSSLport*/**opam-basic**

Differences When Importing the CA Certificate

Use these steps to import the CA certificate to an IBM WebSphere truststore:

1. Log in to the IBM WebSphere Console.
2. Select **Security** > **SSL certificate and key management** > **Key stores and certificates** > **CellDefaultTrustStore** > **Signer certificates** > **Add**.
3. From the Add screen, enter the hostname of the Oracle Privileged Account Manager server into the **Alias** field.
4. Select the **Base 64 encoded ASCII Data** data type because the Oracle Privileged Account Manager server CA certificate (`.pem`) file was exported in BASE64-encoded format.
5. Locate the Oracle Privileged Account Manager server CA certificate (.pem) file on the local file system where IBM WebSphere is running. Enter the full path and file name into the **File Name** field.
6. Save the `.pem` file to `master configuration`.

11.8 Differences When Using the Oracle Privileged Account Manager Command Line Tool and REST Interfaces on IBM WebSphere

When using the Oracle Privileged Account Manager command line tool or REST interfaces on IBM WebSphere, you must be aware of the following differences:

- The target URL for Oracle Privileged Account Manager is,

  `https://`*opamhost*`:`*opamSSLport*`/opam-basic`

  This difference *only* applies to the command line tool and REST interfaces. In the Oracle Privileged Account Manager Console when you add an Oracle Privileged Account Manager server, you use the same URL for both IBM WebSphere and WebLogic.

- The default port for the OracleAdminServer (where the Oracle Privileged Account Manager Console runs) is 9002. The Oracle Privileged Account Manager Managed Server port (18102) is the same on both IBM WebSphere sand WebLogic.

11.9 Configuring Oracle Privileged Account Manager for High Availability in a Clustered Environment

This section describes how to install and configure Oracle Privileged Account Manager on IBM WebSphere in a clustered configuration with High Availability support.

> **Note:** This information is specific to Oracle Privileged Account Manager on IBM WebSphere, and is provided to supplement the instructions provided in "Oracle Privileged Account Manager High Availability" in the *Oracle Fusion Middleware High Availability Guide*.

Topics in this section include:

- Section 11.9.1, "Overview of a Clustered Configuration"

- Section 11.9.2, "Installing Oracle Privileged Account Manager for a Clustered Configuration"

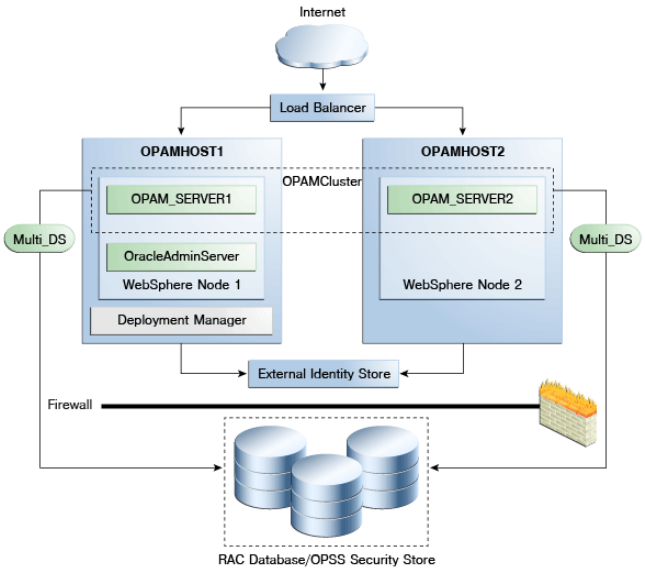
11.9.1 Overview of a Clustered Configuration

To set up Oracle Privileged Account Manager in a clustered configuration with high availability support, you must configure two machines as follows:

- Deployment Manager machine:
  - IBM WebSphere Deployment Manager
  - IBM WebSphere Node 1
  - OracleAdminServer
  - `opam_server1`
- IBM WebSphere Node 2 machine:
  - IBM WebSphere Node 2
  - `opam_server2`

Refer to the following figure.

Figure 11–2 Oracle Privileged Account Manager Clustered Configuration with HA Support



11.9.2 Installing Oracle Privileged Account Manager for a Clustered Configuration

To install and configure Oracle Privileged Account Manager for maximum high availability, perform the following tasks:

1. Identify a Database and Install the Required Database Schema
2. Install IBM WebSphere
3. Install the Oracle Identity and Access Management Suite
4. Configure IBM WebSphere on the Deployment Manager Machine

5. (Optional) Set Up TDE Mode
6. Configure the Oracle Platform Security Services Security Store
7. Start the Deployment Manager
8. Configure IBM WebSphere on the IBM WebSphere Node 2 Machine
9. Configure the External LDAP Server
10. Configure Oracle Privileged Account Manager
11. Restart the Servers

11.9.2.1 Identify a Database and Install the Required Database Schema

You must install a database and load the Oracle Privileged Account Manager schema into that database.

For more information, refer to Section 2.3, "Task 3: Identify a Database and Install the Required Database Schemas."

11.9.2.2 Install IBM WebSphere

Install the IBM WebSphere Application Server software, including the latest Fix Pack, on both the Deployment Manager machine and on the IBM WebSphere Node 2 machine.

For instructions, refer to Section 2.4, "Task 4: Install the IBM WebSphere Software."

11.9.2.3 Install the Oracle Identity and Access Management Suite

You must install the Oracle Identity and Access Management Suite on both the Deployment Manager machine and on the IBM WebSphere Node 2 machine.

For instructions, refer to the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.

Special Instructions

When installing Oracle Fusion Middleware products on IBM WebSphere, the following special instructions apply:

- When you run the Oracle Fusion Middleware installer, you must use the `-DSHOW_APPSERVER_TYPE_SCREEN=true` parameter to let the Oracle Universal Installer prompt for the IBM WebSphere home location.

  For example,

  ```
  diskname/iamsuite/Disk1/runInstaller -jreLoc
    diskname/IBM/WebSphere/AppServer/java/jre -DSHOW_APPSERVER_TYPE_SCREEN=true
  ```

- When you are prompted to specify a JRE/JDK location, you can specify the following directory in the IBM WebSphere home:

  On UNIX: *WAS_HOME*/java

  For example, if you are using the default location for a typical IBM WebSphere Application Server directory on a UNIX operating system:

  ```
  diskname/IBM/WebSphere/AppServer/java
  ```

- When you are prompted to provide a Middleware home, note that you can enter a new Middleware home directory path.

When you install Oracle Fusion Middleware products on Oracle WebLogic Server, you create the Middleware home. This is because Oracle WebLogic Server is included in the Middleware home.

In contrast, when you install Oracle Fusion Middleware on IBM WebSphere, you create the Middleware home when you install the Oracle Fusion Middleware software. This is because the IBM WebSphere software is not installed inside the Middleware home. It is installed in a separate directory structure.

- When you select IBM WebSphere as your application server and you are prompted for the Application Server location, enter the path to the IBM WebSphere Application Server directory that you created in Section 2.4, "Task 4: Install the IBM WebSphere Software."

  For example,

  diskname/IBM/WebSphere/AppServer/

11.9.2.4 Configure IBM WebSphere on the Deployment Manager Machine

On the Deployment Manager machine, use the Oracle Fusion Middleware Configuration Wizard to create the Oracle Privileged Account Manager cell. By default, the Configuration Wizard is located at

MW_HOME/Oracle_IDM1/common/bin/was_config.sh

Select **Oracle Privileged Account Manager (Form auth-mode OINAV)** or **Oracle Privileged Account Manager (Client-cert auth-mode OINAV)**, depending on the auth-mode required for Oracle Identity Navigator.

Table 11–2 provides information about specific Configuration Wizard screens and the appropriate information to enter on those screens—it does not cover self-explanatory, standard screens.

Table 11–2 Information for Specific Configuration Wizard Screens

Screen Name	Input Description
Select Optional Configuration	At a minimum, you must select the **Application Servers, Clusters and End Points** option—this is a required option.
Configure Application Servers	Perform the following steps: 1. In the **Name** field, enter a name for the Oracle Privileged Account Manager server. For example: opam_server1. 2. In the **Node Name** list, select the Node Agent for opam_server1. For example: **WebSphereNode1**.
Configure Clusters Screen	Perform the following steps: 1. Click **Add** to add a cluster. 2. Enter a name for the cluster in the **cluster name** field. For example: OPAMCluster. 3. Select the appropriate Oracle Privileged Account Manager server from the **First cluster member** list.
Configure Additional Cluster Members	Click **Next** or, optionally, add servers to an existing system in the cluster.

11.9.2.5 (*Optional*) Set Up TDE Mode

Oracle Privileged Account Manager can operate with Oracle Database TDE (Transparent Data Encryption) mode. You can choose to enable or disable TDE mode;

however, Oracle strongly recommends that you enable the TDE mode for enhanced security.

This section includes the following topics:

- Section 11.9.2.5.1, "Enabling TDE in the Database"
- Section 11.9.2.5.2, "Enabling Encryption in the Oracle Privileged Account Manager Schema"

11.9.2.5.1 Enabling TDE in the Database To enable TDE (Transparent Data Encryption) in the database for Oracle Privileged Account Manager, refer to "Enabling Transparent Data Encryption" in the *Oracle Database Advanced Security Administrator's Guide*.

For more information about using TDE, refer to "Securing Stored Data Using Transparent Data Encryption" in the *Oracle Database Advanced Security Administrator's Guide*.

11.9.2.5.2 Enabling Encryption in the Oracle Privileged Account Manager Schema To enable encryption in the Oracle Privileged Account Manager schema, run the `opamxencrypt.sql` script with the Oracle Privileged Account Manager schema user, using sqlplus or any other client.

```
IAM_HOME/opam/sql/opamxencrypt.sql
```

For example,

```
sqlplus DEV_OPAM/welcome1 @IAM_HOME/opam/sql/opamxencrypt.sql
```

11.9.2.6 Configure the Oracle Platform Security Services Security Store

> **Note:** You must execute this task from the machine where you are running the Deployment Manager.

To configure the Oracle Platform Security Services (OPSS) Database Security Store, follow the instructions in Section 2.9, "Task 9: Configure the Database Security Store."

11.9.2.7 Start the Deployment Manager

To start the Deployment Manager, go to the *WAS_HOME*/bin location and execute the following command:

```
./startManager.sh -profileName <dmgr_prof_name>
```

11.9.2.8 Configure IBM WebSphere on the IBM WebSphere Node 2 Machine

On WebSphere Node 2 machine, launch the Oracle Fusion Middleware Configuration Wizard to federate the machine and configure its cell. By default, the Configuration Wizard is located at

```
MW_HOME/Oracle_IDM1/common/bin/was_config.sh
```

Table 11–3 provides information about specific Configuration Wizard screens and the appropriate information to enter on those screens—it does not cover self-explanatory, standard screens.

Table 11–3 Information for Specific Configuration Wizard Screens

Screen Name	Input Description
Select Configuration Option	Select the **Federate Machine and Configure Cell** option.
Specify Profile and Node Name Information	Enter information about the profile and node names you want to create for the WebSphere Node 2 Machine.
Specify Deployment Manager Information	Enter information about the existing Deployment Manager system.
Select Optional Configuration	Be sure to select the **Application Servers, Clusters and End Points** option—this is a required option.
Configure Additional Cluster Members	Perform the following steps: 1. Click **Add** to add a cluster. 2. In the **Name** field, enter a name for the second server in the OPAMCluster. For example: `opam_server2`. 3. Select a Node Agent for opam_server2 from the **Node Name** list. For example: **WebSphereNode2**. 4. Select the **OPAMCluster** from the **Cluster Name** list.

11.9.2.9 Configure the External LDAP Server

On IBM WebSphere, OPSS supports only LDAP-based registries. OPSS does not support IBM WebSphere's built-in file-based user registry.

To configure the Oracle Internet Directory store for OPSS:

1. Enter the following command:

   ```
   cd <oracle_common>/common/bin
   ```

2. Run the `wsadmin` command using the same credentials that you provided when you set up the IBM WebSphere cell.

   ```
   ./wsadmin.sh -conntype SOAP -port <port_number> -user <username>
     -password <passwd>
   ```

   The port details are available in this file:

   ```
   $WAS_HOME/profiles/Dmgr01/logs/AboutThisProfile.txt
   ```

3. Enter the following command:

   ```
   Opss.configureIdentityStore(propsFileLoc="<location of properties file>")
   ```

   Use the following sample properties file for reference:

   ```
   user.search.bases=cn=Users,dc=myhost,dc=mycompany,dc=com
   group.search.bases=cn=Groups,dc=myhost,dc=mycompany,dc=com
   subscriber.name=dc=myhost,dc=mycompany,dc=com
   ldap.host=ldaphost.mycompany.com
   ldap.port=3333
   # admin.id must be the full DN of the user in the LDAP
   admin.id=cn=orcladmin,cn=Users,dc=myhost,dc=mycompany,dc=com
   admin.pass=welcome1
   user.filter=(&(uid=%v)(objectclass=person))
   group.filter=(&(cn=%v)(objectclass=groupofuniquenames))
   user.id.map=*:uid
   group.id.map=*:cn
   group.member.id.map=groupofuniquenames:uniquemember
   ssl=false
   ```

```
# primary.admin.id indicates a user who has admin permissions in the LDAP, must
be the name of the user, for example, for user "cn=tom", the primary.admin.id
is "tom"
primary.admin.id=orcladmin
# optional, default to "OID"
idstore.type=OID
# Optional properties for JPS LDAP identity store can also be configured in the
file.
username.attr=cn
user.object.classes=person
```

Note: After completing preceding steps, you must seed the identity store with the necessary Oracle Privileged Account Manager users and groups.

For instructions, refer to "Preparing the Identity Store" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Privileged Account Manager*

11.9.2.10 Configure Oracle Privileged Account Manager

You are now ready to configure Oracle Privileged Account Manager. You must perform this task from the Deployment Manager machine.

For instructions, refer to Section 11.2.2.2, "Configuring Oracle Privileged Account Manager on IBM WebSphere."

11.9.2.11 Restart the Servers

To restart the servers:

1. Stop the Deployment Manager on the Deployment Manager machine. Execute the following from `$WAS_HOME/bin`:

   ```
   ./stopManager.sh -profileName <dmgr_prof_name> -username <username>
      -password <password>
   ```

2. Stop the node on WebSphere Node 2 machine. Execute:

   ```
   ./stopNode.sh -profileName <server_prof_name> -username <username>
      -password <password>
   ```

3. Start the Deployment Manager, node, and servers on Deployment Manager machine. Execute:

   ```
   ./startManager.sh -profileName <dmgr_prof_name>
   ./syncNode.sh <dmgr_host_name> <SOAP connector port> -username <username>
      -password <password>
   ./startNode.sh -profileName <server_prof_name>
   ./startServer.sh OracleAdminServer -profileName <server_prof_name>
   ./startServer.sh <opam_server_name> -profileName <server_prof_name>
   ```

4. Start the node and Oracle Privileged Account Manager server on WebSphere Node 2 machine:

   ```
   ./syncNode.sh <dmgr_host_name> <SOAP connector port> -username <username>
      -password <password>
   ./startNode.sh -profileName <server_prof_name>
   ./startServer.sh <opam_server_name> -profileName <server_prof_name>
   ```

After starting the servers:

- If you enabled Transparent Data Encryption (TDE) mode as described in Section 11.9.2.5, "(Optional) Set Up TDE Mode," then you have finished installing and configuring Oracle Privileged Account Manager on IBM WebSphere. No further steps are required. You can now verify the Oracle Privileged Account Manager functionality.
- If you decided *not* to enable TDE mode, then you must complete steps to set up non-TDE mode on both of the nodes. Refer to Section 11.2.2.3, "Setting Up Non-TDE Mode" for instructions.

11.10 Limitations and Known Issues When Using Oracle Privileged Account Manager on IBM WebSphere

This section describes any limitations or known issues for this delivery of Oracle Privileged Account Manager on IBM WebSphere.

11.10.1 Limitations

There are no limitations for this release of Oracle Privileged Account Manager on IBM WebSphere:

11.10.2 Known Issues

This section describes any known issues for using Oracle Privileged Account Manager on IBM WebSphere.

Table 11–4 Known Issues for Oracle Privileged Account Manager on IBM WebSphere

Bug #	Issue
16074104	When Oracle Privileged Account Manager is running on IBM WebSphere, you cannot add CSF mappings corresponding to a Oracle WebLogic Server domain. Similarly, when Oracle Privileged Account Manager is running on Oracle WebLogic Server, you cannot add CSF mappings corresponding to a IBM WebSphere cell.

12

Managing Oracle Identity Navigator on IBM WebSphere

This chapter contains information about managing Oracle Identity Navigator on IBM WebSphere. This chapter contains the following sections:

12.1 Differences in Managing Oracle Identity Navigator on IBM WebSphere

This section describes the differences when managing Oracle Identity Navigator on IBM WebSphere.

12.1.1 Configuring a Proxy to Access News Feeds

You may need to specify a proxy so that Oracle Identity Navigator can access Oracle news feeds from inside your firewall. The Identity Management Discussion Forums uses SSL.

To configure the properties:

1. In the WebSphere administration console, navigate to Servers > Server Types > Application servers > OracleAdminServer > Java and Process Management > Process definition > Java Virtual Machine > Custom properties.
2. Add the following:

```
-http.proxyHost=proxy_server_host
-http.proxyPort=proxy_server_port
-http.nonProxyHosts=non_proxy_hosts
-https.proxyHost=ssl_proxy_server_host
-https.proxyPort=ssl_proxy_server_port
```

Next, import the forum certificate in to the Cell's trust store.

To import the forum certificate:

1. Using Internet Explorer or Chrome browser, export the certificate at `https://forums.oracle.com/forums/rss/rssthreads.jspa?forumID=47`.
2. Log into the IBM WebSphere Administrative Console.

3. Navigate to Security > SSL certificate and key management > Key stores and certificates > CellDefaultTrustStore > Signer certificates > Add.
4. Set Alias to forums.oracle.com_cert.
5. Set File name to the exported certificate obtained in Step 1. It should point to a file on the local file system on which IBM WebSphere is running.
6. Set Data type to same file format as certificate.
   - Base64 encoded certificate: select Base64-encoded ASCII data.
   - DER encoded binary format certificate: select Binary DER data

You do not need to restart the server.

12.1.2 Configuring Single Sign-On

For information about configuring single sign-on in an IBM WebSphere environment, see Chapter 6, "Managing Oracle Access Manager Identity Assertion on IBM WebSphere."

12.2 Limitations When Using Oracle Identity Navigator on IBM WebSphere

The following functionality is not supported in this release:

- Product Discovery. The Discover Products option under Product Registration in the Administration tab is not available.

13

Managing Oracle Access Management Mobile and Social on IBM WebSphere

This chapter contains information about managing Oracle Access Management Mobile and Social on IBM WebSphere.

This chapter contains the following sections:

13.1 Using Mobile and Social WLST Commands on IBM WebSphere

You can run Oracle Access Management commands from the IBM WebSphere wsadmin command line interface. For details, see Using the Oracle Fusion Middleware wsadmin Commands.

Oracle Access Management commands are documented in the *Web Logic Scripting Tool Command Reference*. Oracle Access Management commands are functionally identical on WebLogic and WebSphere. When running Mobile and Social wsadmin commands, however, you must prefix the command name with the Mobile and Social `idaas_commands` category name. For example:

```
idaas_commands.createServiceProvider(...)
```

13.2 Configuring Mobile Services for Oracle Adaptive Access Manager

Most topics in the *Administrator's Guide for Oracle Access Management* apply to both WebSphere and WebLogic environments. In the "Configuring Mobile Services" chapter, when referring to the "Configuring Mobile Services for Oracle Adaptive Access Manager" section, use the following modified steps instead of the steps documented in the "Configuring the WebLogic Administration Domain" section.

13.2.1 Creating an Administrator for OAAM Administration

Add users and groups from the WebSphere administration console. To do so, click **Users and Groups** > **Manage Users/Manage Groups**. Refer to the WebSphere documentation for more information.

13.2.2 Adding Oracle Access Management Server as Target of OAAM Data Source

Create a new data source in WebSphere using the WebSphere administration console using the same name and values of the `OAAM_SERVER_DS` datasource defined in the `oaam_server` scope. Create this new DS in the scope of the managed server where `oam_server` is installed.

> **Note:** To extend an OAM domain for OAAM, run `was_config.sh` on top of the OAM install and choose the option to use the existing WebSphere Application Server profile.

13.3 Supporting Internet Identity Services on IBM WebSphere

Complete the item in this section to configure IBM WebSphere to support Internet Identity Services.

This section includes the following items:

- Adding CA Certificates to the IBM Trust Store
- Configuration Requirements for Apps Protected by Access Manager

13.3.1 Adding CA Certificates to the IBM Trust Store

Follow these steps to configure WebSphere to provide proper SSL support for Internet Identity Services in Mobile and Social.

Import the default IBM certificates from the trust keystore `trust.p12` into the JDK `cacerts` keystore. This will ensure that both the relying party (that is, the Internet Identity Service Provider) and the Oracle Access Management console can use SSL properly.

1. In Mozilla Firefox open the following URL using the correct values for the machine where the instance is installed:

   `http://<host name>:<port>/ibm/console`

   The browser presents a security page and prompts you to trust the certificate.

2. View the certificate and export it to a file using the `.der` format.

   If necessary, copy the `.der` file to the server where WebSphere is deployed.

3. In the WebSphere Application Server Administrative Console, choose **Security > SSL certificate and key management**.

4. For both the Cell and Node levels where OAM is deployed, change the trust store file name setting from `trust.p12` to the `cacerts` file that ships with the default IBM WebSphere JDK. Typically this file is located here:

   `<WAS_HOME>/java/jre/lib/security/cacerts`

   Save your changes.

5. Click **Signer Certificates** to see all the signer certificates in the `cacerts` file.

6. Click **Add**, type an alias name, and type the path to the `.der` file you exported in step 2.

   Save your settings.

7. Hard-restart both the OAM managed server and the server hosting the Oracle Access Management Console.

13.3.2 Configuration Requirements for Apps Protected by Access Manager

If your apps are protected by Access Manager and use Internet Identity Services to provide users with additional log-in and registration options, you must configure user LDAP so that local log-in works properly. Because WebSphere does not include an embedded LDAP server like WebLogic, the LDAP user repository must be configured manually.

Configure your environment as follows:

- Add to the LDAP repository the wasadmin user that is used to log in to the Administration Console.
- Ensure that the following uid attributes are the same:
  - The uid attribute in the app using Internet Identity Services. See the "Editing or Deleting an Application Profile" section in the *Administrator's Guide for Oracle Access Management* for details. The *OAMApplication* Application Profile that is included with Mobile and Social is preconfigured to work with Access Manager and requires only minor configuration changes to get working in your environment.
  - The uid attribute used for Access Manager log-in. See the "Editing or Deleting an Application Profile" section in the *Administrator's Guide for Oracle Access Management* for details.
  - The uid attribute for (Mobile and Social) User Profile Services. See the "Editing or Deleting a User Profile Service Provider" section in the *Administrator's Guide for Oracle Access Management* for details.
- If your app is directly integrated with Mobile and Social, and if Internet Identity Services and User Profile Services both point to a user repository other than the Access Manager user repository, both configurations should have the same uid attribute.

> **Note:** For all configurations, do not use the same attribute for the account linking attribute and the uid attribute. The account linking attribute and the uid attribute must be separate.

13.4 Moving Mobile and Social From a Test to a Production Environment

When moving Mobile and Social from a test environment to a production environment, complete the following configuration steps on each production machine after running the Test-to-Production scripts.

> **Note:** Before performing these steps, complete the "Moving Access Manager From a Test to Production Environment on IBM WebSphere" steps located in the "Managing Access Manager on IBM WebSphere" chapter.

1. Launch the Oracle Access Management Console.
2. On the **Policy Configuration** tab, choose **Shared Components** > **Authentication Schemes** > **OIC Scheme** and click Open.

   The Authentication Schemes configuration page opens.

Update the **Challenge Redirect URL** value to point to the production machine, not the test machine, then click **Apply**.

For example: `https://`*`production_machine`*`:`*`port`*`/oic_rp/login.jsp`

3. Update the Mobile and Social credential store framework (CSF) entry to point from the test machine to the production machine. To do this, run the following WLST command:

```
createCred(map="OIC_MAP", key=" https://<production machine host>:<production
machine port>/oam/server/dap/cred_submit ", user="="<description>", password="
DCC5332B4069BAB4E016C390432627ED", desc="<description>");
```

For `password`, use the value from `oam-config.xml`, which is located in the *domain home*`/config/fmwconfig` directory on the production machine. Use the value from the `RPPartner` entry, `TapCipherKey` attribute.

4. In the Oracle Access Management Console, do the following:
   a. Select the **System Configuration** tab.
   b. Choose **Mobile and Social** > **Internet Identity Services**.
   c. In the **Application Profiles** section, select **OAMApplicaton** and click **Edit**. (If using an application profile name other than OAMApplication, edit that instead.)
   d. Update the **Registration URL** field host name and port to point to the production machine.

      Click **Apply**.

14

Integrating Access Manager, Oracle Adaptive Access Manager, and Oracle Identity Manager on IBM WebSphere

This chapter documents how to integrate Access Manager, Oracle Adaptive Access Manager, and Oracle Identity Manager on IBM WebSphere. The following integrations are covered.

- Integrating Access Manager and Oracle Identity Manager on IBM WebSphere
- Integrating Access Manager and OAAM on IBM WebSphere
- Integrating Access Manager, OAAM, and OIM on IBM WebSphere

14.1 Integrating Access Manager and Oracle Identity Manager on IBM WebSphere

This section contains information about integrating Access Manager with Oracle Identity Manager using the IBM HTTP Server (IHS) WebGate 11*g* on IBM WebSphere.

This section includes the following topics:

- Integration Roadmap
- Configure Additional IHS Web Server Reverse Proxies (Optional)
- Configuring the Identity Store
- Restart the OIM Servers
- Copy the OAM 11g SSO Agent Artifacts
- Configure the Web Server to Route Requests to OIM on WebSphere

14.1.1 Integration Roadmap

The process of integrating Access Manager with Oracle Identity Manager on IBM WebSphere includes the following high-level tasks.

Table 14–1 Integration Flow for Access Manager and Oracle Identity Manager of IBM WebSphere

No.	Task	Information
1	Install Oracle Identity Manager on a WebSphere server.	For information, see Installing and Configuring Oracle Identity and Access Management on IBM WebSphere.

Table 14–1 (Cont.) Integration Flow for Access Manager and Oracle Identity Manager of IBM WebSphere

No.	Task	Information
2	Enable LDAP synchronization for Oracle Identity Management.	For information, see: ■ "Configuring OIM Server", ■ "Completing the Prerequisites for Enabling LDAP Synchronization", and ■ "Creating Adapters in Oracle Virtual Directory" in *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.
3	Install Oracle Access Management on a WebSphere server.	For information, see Installing and Configuring Oracle Identity and Access Management on IBM WebSphere. If necessary, Oracle Access Management can be installed in the same WebSphere cell as Oracle Identity Manager.
4	Install IBM HTTP Server (IHS) 7.0	Download IBM HTTP Server version 7.0.0.0 from the IBM web site. Install the IHS 7 software using the default values.
5	Install and configure the IHS 11*g* Webgate for OAM.	For information, see "Installing and Configuring IHS 11*g* Webgate for OAM" in *Installing Webgates for Oracle Access Management*. Install the IHS 11*g* Webgate on the IBM HTTP Server (IHS) host.
6	Install WebServer Plug-in for WebSphere Application Server 7.0 on each WebSphere Application Server machine.	Download WebServer Plug-in for WebSphere Application Server 7.0 from the IBM web site. Install the software using the default values.
7	Add an IHS Web server reverse proxy.	For information, see Configure Additional IHS Web Server Reverse Proxies (Optional).
8	Configure the Identity Store.	For information, see Configuring the Identity Store.
9	Configure the OAM TAI Configuration File	For information, see Configuring the OAM TAI Configuration File
10	Restart all the servers in the OIM WebSphere cell.	For information, see Restart the OIM Servers.
11	Copy the OAM 11*g* SSO agent artifacts.	For information, see Copy the OAM 11g SSO Agent Artifacts.
12	Configure IHS to route requests to Oracle Identity Manager	For information, see Configure the Web Server to Route Requests to OIM on WebSphere.

14.1.2 Configure Additional IHS Web Server Reverse Proxies (Optional)

Complete the steps in this section to configure additional IHS Web servers to use as reverse proxies.

1. Start the IBM HTTP Server using the `adminctl` command as root user.

```
bin/adminctl start
```

2. Open the WebSphere administrative console.

   Refer to the IBM documentation for IBM HTTP Server for details.

3. Choose **Servers** > **Server Types** > **Web servers**.

   Click **New** to add a new Web server to WebSphere.

4. Specify the Web server name, type, host name, and platform and click Next.
5. Select the template name on the "Select a Web server template" screen.
6. Type the properties for the IBM HTTP Server and the IBM HTTP Server Administration Server, and click Next.
7. Confirm the new Web server properties and click Finish.
8. Select the newly created Web server and click **Generate Plug-in**.

   Next, click **Propagate Plug-in**.

9. Click **Save** to save the configuration to the master configuration.
10. Start the new Web server.

14.1.3 Configuring the Identity Store

The Identity Store must be configured so that it can be used by Access Manager, Oracle Identity Management, and WebSphere. It must be seeded with the required users and groups.

Use `idmConfigTool` to configure the Identity Store.

> **Note:** Refer to the "Using the idmConfigTool Command" chapter in the *Integration Guide for Oracle Management Suite* for information about `idmConfigTool`. The following steps assume that you understand the conceptual information provided in that chapter.

14.1.3.1 Set the Environment Variables for idmConfigTool

Set the following environment variables before running `idmConfigTool`.

If Oracle Identity Manager and Oracle Access Management are installed on different WebSphere cells, set the environment variable for each cell.

Variable	Set to
`MW_HOME`	Set the value to the full path of the installation's Middleware home.
`ORACLE_HOME`	Set to the full path of the Oracle home. For IDM integrations, set to `Oracle_IDM1`
`APPSERVER_TYPE`	Set to `was`
`WAS_HOME`	Set the value to the full path of the WebSphere application server home directory. For example: `/WASSH/WebSphere/AppServer`

Variable	Set to
JAVA_HOME	Set the value to the full path of the IBM JDK. For example: /WASSH/WebSphere/AppServer/java **Important:** Do not use a JDK other than the IBM JDK.
WAS_DMGR_PROFILE_HOME	Set to the deployment manager profile home directory. The deployment manager deploys applications to a cell of application servers, which it manages. A profile defines the runtime environment. The profile includes all of the configurable files that the server processes in the runtime environment. Set to an absolute path, for example: /WASSH/WebSphere/AppServer/profiles/Dmgr01

14.1.3.2 Run idmConfigTool

At a command prompt run the following idmConfigTool commands in this order.

For information about command syntax and the use of properties files, see the "Syntax and Usage" section of the "Using the idmConfigTool Command" chapter.

1. Stop the Oracle Identity Manager, SOA, and OracleAdminServer servers, as well as the NodeAgent in the OIM WebSphere cell.

   For instructions, refer to the WebSphere Application Server documentation.

2. Locate the preConfigIDStore.props property file in the idmCfgToolProps directory. See the "Using the idmConfigTool Command" chapter for an example preConfigIDStore.props properties file.

   Run the preConfigIDStore command from the OAM WAS cell as follows:

   ```
   ./idmConfigTool.sh -preConfigIDStore
   input_file=/idmCfgToolProps/preConfigIDStore.props
   ```

3. Create the wasadmin user by running the prepareIDStore mode=WAS command.

   The properties file for this command is similar to the properties file for the prepareIDStore=WLS command, except the IDSTORE_WASADMINUSER property is specified.

   Here is a sample properties file.

   ```
   IDSTORE_HOST : xyz1234.us.example.com
   IDSTORE_PORT : 3060
   IDSTORE_BINDDN : cn=orcladmin
   IDSTORE_USERNAMEATTRIBUTE: cn
   IDSTORE_WASADMINUSER: wasadmin
   IDSTORE_USERSEARCHBASE: cn=Users,dc=us,dc=example,dc=com
   IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=us,dc=example,dc=com
   IDSTORE_SEARCHBASE: dc=us,dc=example,dc=com
   ```

   Create the properties file and run the command from the OIM WAS cell. In this example the properties file is named prepareIDStore.props:

   ```
   ./idmConfigTool.sh -prepareIDStore mode=WAS
   input_file=/idmCfgToolProps/prepareIDStore.props
   ```

4. Run the prepareIDStore mode=OAM command from the OAM WAS cell.

   ```
   ./idmConfigTool.sh -prepareIDStore mode=OAM
   ```

```
input_file=/scratch/idmCfgToolProps/prepareIDStore.props.oam.template
```

5. Prepare to run the `prepareIDStore mode=OIM` command from the OIM WAS cell by creating a properties file.

   Here is a sample properties file.

```
IDSTORE_HOST: xyz5678.us.example.com
IDSTORE_PORT: 3060
IDSTORE_BINDDN: cn=orcladmin
IDSTORE_USERNAMEATTRIBUTE: cn
IDSTORE_LOGINATTRIBUTE: uid
IDSTORE_USERSEARCHBASE: cn=Users,dc=us,dc=example,dc=com
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=us,dc=example,dc=com
IDSTORE_SEARCHBASE: dc=us,dc=example,dc=com
IDSTORE_SYSTEMIDBASE: cn=systemids,dc=us,dc=example,dc=com
IDSTORE_OIMADMINUSER: oimLDAP
IDSTORE_OIMADMINGROUP: OIMAdministrators
OIM_DB_URL: jdbc:oracle:thin:@xyz5678.us.example.com:5522:wasdb1
OIM_DB_SCHEMA_USERNAME: dev_oim
OIM_WAS_CELL_CONFIG_DIR:
/wassh/WebSphere/AppServer/profiles/Dmgr04/config/cells/xyz5678Cell04/fmwconfig
```

   The following `prepareIDStore mode=OIM` command properties are specific to WebSphere:

Parameter	Description
`OIM_DB_URL`	The OIM DB connection URL.
`OIM_DB_SCHEMA_USERNAME`	The OIM DB schema User.
`OIM_WAS_CELL_CONFIG_DIR`	Location of the `fmwconfig` directory within the OIM cell.

   Create and save the properties file.

   Run the command from the OIM WAS cell. In this example the properties file is named `prepareIDStore.props`.oim.template:

```
./idmConfigTool.sh -prepareIDStore mode=OIM
input_file=/idmCfgToolProps/prepareIDStore.props.oim.template
```

6. Prepare to run the `configOAM` command from the OAM WAS cell by creating a properties file.

   Here is a sample properties file.

```
WLSHOST: abc1234.us.example.com
WLSPORT: 9810
WLSADMIN: orcladmin
WLSPASSWD: welcome1
IDSTORE_HOST: xyz5678.us.example.com
IDSTORE_PORT: 3060
IDSTORE_DIRECTORYTYPE:OID
IDSTORE_BINDDN: cn=orcladmin
IDSTORE_USERNAMEATTRIBUTE: cn
IDSTORE_LOGINATTRIBUTE: uid
IDSTORE_SYSTEMIDBASE: cn=systemids,dc=us,dc=example,dc=com
IDSTORE_USERSEARCHBASE: cn=Users,dc=us,dc=example,dc=com
IDSTORE_SEARCHBASE: dc=us,dc=example,dc=com
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=us,dc=example,dc=com
```

```
IDSTORE_OAMSOFTWAREUSER: oimLDAP
IDSTORE_OAMADMINUSER: oamadmin
PRIMARY_OAM_SERVERS: abc1234.us.example.com:5575
WEBGATE_TYPE: ohsWebgate11g
ACCESS_GATE_ID: oimwebgate
OAM11G_IDM_DOMAIN_OHS_HOST:abc1234.us.example.com
OAM11G_IDM_DOMAIN_OHS_PORT:7777
OAM11G_IDM_DOMAIN_OHS_PROTOCOL:http
OAM11G_WG_DENY_ON_NOT_PROTECTED: false
OAM_TRANSFER_MODE: open
OAM11G_OAM_SERVER_TRANSFER_MODE:open
OAM11G_IDM_DOMAIN_LOGOUT_URLS:
/console/jsp/common/logout.jsp,/em/targetauth/emaslogout.jsp
COOKIE_DOMAIN: .us.example.com
OAM11G_IDSTORE_ROLE_SECURITY_ADMIN: OAMAdministrators
OAM11G_SSO_ONLY_FLAG: false
OAM11G_OIM_INTEGRATION_REQ: true
OAM11G_IMPERSONATION_FLAG:true
OAM11G_SERVER_LBR_HOST:abc1234.us.example.com
OAM11G_SERVER_LBR_PORT:14100
OAM11G_SERVER_LBR_PROTOCOL:http
COOKIE_EXPIRY_INTERVAL: 120
OAM11G_OIM_OHS_URL:http://tx401alu.us.example.com:7777
OAM11G_SERVER_LOGIN_ATTRIBUTE: uid
```

The following table describes the `configOAM` command properties as they apply to WebSphere:

Parameter	Description
`WLSHOST`	The WebSphere Application Server host.
`WLSPORT`	The WebSphere Application Server bootstrap port.
`WLSADMIN`	Login ID for the OAM WebSphere Admin console.
`WLSPASSWD`	(Optional) Login password for the OAM console/WebSphere Admin console. To ensure security, do not save the password in the properties file.

Create and save the properties file.

Run the command **from the OAM WAS cell**. In this example the properties file is named `oamcfg.props`:

```
./idmConfigTool.sh -configOAM input_file=/oamcfg.props
```

7. Update the virtual host configuration.

   In the WebSphere console on the OAM host, choose **Environment > Virtual Hosts > *default_host* > Host Aliases**.

   Add the new IBM HTTP Server host and port, then restart the Oracle Access Management (OAM) server.

8. Prepare to run the `idmConfigTool.sh -configOIM` command by creating a properties file.

   Here is a sample properties file.

```
LOGINURI: /${app.context}/adfAuthentication
LOGOUTURI: /oamsso/logout.html
AUTOLOGINURI: None
ACCESS_SERVER_HOST: abc1234.us.example.com
ACCESS_SERVER_PORT: 5575
ACCESS_GATE_ID: oimwebgate
COOKIE_DOMAIN: .us.example.com
COOKIE_EXPIRY_INTERVAL: 120
OAM_TRANSFER_MODE: OPEN
WEBGATE_TYPE: ohsWebgate10g
SSO_ENABLED_FLAG: true
IDSTORE_PORT: 3060
IDSTORE_HOST: xyz5678.us.example.com
IDSTORE_DIRECTORYTYPE: OID
IDSTORE_ADMIN_USER: cn=orcladmin
IDSTORE_USERSEARCHBASE: cn=Users,dc=us,dc=example,dc=com
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=us,dc=example,dc=com
MDS_DB_URL: jdbc:oracle:thin:@xyz5678.us.example.com:5522:wasdb1
MDS_DB_SCHEMA_USERNAME: dev_mds
WLSHOST: xyz5678.us.example.com
WLSPORT: 9809
WLSADMIN: wasadmin
DOMAIN_NAME: IDMDomain
OIM_MANAGED_SERVER_NAME: oim_server1
DOMAIN_LOCATION: /IDMPS2/user_projects/domains/IDMDomain
IDSTORE_LOGINATTRIBUTE: uid
IDSTORE_SEARCHBASE: dc=us,dc=example,dc=com
OIM_WEB_SERVER_HOST: tx401alu.us.example.com
OIM_WEB_SERVER_PORT: 7777
OAM11G_WLS_ADMIN_HOST: abc1234.us.example.com
OAM11G_WLS_ADMIN_PORT: 9810
OAM11G_WLS_ADMIN_USER: wasadmin
```

The following notes apply to this properties file:

- The `ACCESS_SERVER_PORT` must be the Access Manager NAP port.
- If your OAM Servers are configured to accept requests using the simple mode, set `OAM_TRANSFER_MODE` to `SIMPLE`. Otherwise set `OAM_TRANSFER_MODE` to `OPEN`.
- Set `WEBGATE_TYPE` to `ohsWebgate11g` if Webgate version 11 is used, or `ohsWebgate10g` if Webgate version 10 is used.
- Set `IDSTORE_PORT` to your Oracle Internet Directory port if you are using Oracle Internet Directory as your Identity Store. If not, set it to your Oracle Virtual Directory port.
- Set `IDSTORE_HOST` to your Oracle Internet Directory host or load balancer name if you are using Oracle Internet Directory as your Identity Store. If not, set it to your Oracle Virtual Directory host or load balancer name.
- Set `IDSTORE_DIRECTORYTYPE` to `OVD` if you are using Oracle Virtual Directory server to connect to either a non-OID directory or Oracle Internet Directory. Set it to `OID` if your Identity Store is in Oracle Internet Directory and you are accessing it directly rather than through Oracle Virtual Directory.
- `MDS_DB_URL` in this case represents a single instance database. The string following the '`@`' symbol must have the correct values for your environment. SID must be the actual SID, *not* a service name. If you are using a single instance database, then set MDS_URL to: `jdbc:oracle:thin:@DBHOST:1521:SID`.

- The value of IDSTORE_ADMIN_USER must contain the complete LDAP DN of the user. The entry should be similar to "cn=oamadmin,cn=Users,dc=myhost,dc=mycompany,dc=com" instead of just "oamadmin".
- Set WLSPORT to the WebSphere bootstrap port for Oracle Identity Manager (OIM).
- Set WLSADMIN to the primary administrative user name configured in the OIM WebSphere cell. This is wasadmin by default. If the configOIM command is being re-run, provide the current primary administrative user name configured in the OIM WebSphere cell.
- Set DOMAIN_NAME to the WebSphere cell name.
- Set DOMAIN_LOCATION to the cell home. For example:

  <*WAS_HOME*>/profiles/Dmgr01/config/cells/<*host name*>
- Set OAM11G_WLS_ADMIN_PORT to the WebSphere bootstrap port of Oracle Access Management (OAM).
- Set OAM11G_WLS_ADMIN_USER to the primary administrative user name configured in the OAM WebSphere cell.

Note: If Oracle Identity Manager (OIM) and Oracle Access Management (OAM) are configured in two different WebSphere cells, you must specify the following properties:

- OAM11G_WLS_ADMIN_HOST
- OAM11G_WLS_ADMIN_PORT
- OAM11G_WLS_ADMIN_USER

If OIM and OAM are part of the same WebSphere cell, you do not have to specify these properties.

The following configOIM command properties are specific to WebSphere:

Parameter	Description
IDSTORE_SEARCHBASE	The ID store search base.
OIM_WEB_SERVER_HOST	The IBM HTTP Server (IHS) host or Oracle HTTP Server (OHS) host.
OIM_WEB_SERVER_PORT	The IBM HTTP Server (IHS) port or Oracle HTTP Server (OHS) port.

Create and save the properties file.

Run the command from the OIM WAS cell. In this example the properties file is named oimcfg.props:

```
./idmConfigTool.sh -configOIM input_file=/oimcfg.props
```

When prompted for WLS passwords for OIM/OAM, provide the corresponding WebSphere dmgr admin user passwords:

- When prompted for WLSPASSWD, enter the OIM WebSphere admin password, for example: welcome1

- When prompted for `OAM11G_WLS_ADMIN_PASSWD`, enter the OAM WebSphere admin password, for example: `welcome1`

14.1.3.3 Configure the OAM TAI Configuration File

Configure `oamtai.xml` for your environment. See Section 6.9.5.2, "Configuring the OAM TAI Configuration File" for details.

14.1.4 Restart the OIM Servers

Restart all of the servers in the OIM WAS cell. Restart OIM Dmgr and sync node, and then start nodeagent and the other servers in the following order.

Note - In the following sample commands, change the host name, port, user name, and password as appropriate.

Stop and Start OIM Dmgr

```
/WASSH/WebSphere/AppServer/profiles/Dmgr01/bin/stopManager.sh -username wasadmin
-password welcome1

/WASSH/WebSphere/AppServer/profiles/Dmgr01/bin/startManager.sh
```

Restart Sync Node

In the following sample command, replace the values described below with values for your environment:

```
/WASSH/WebSphere/AppServer/profiles/Custom01/bin/syncNode.sh
deploymgrHost.us.example.com 8881 -username wasadmin -password welcome1
```

- *deploymgrHost.us.example.com* - Provide the hostname of the OIM cell's Deployment manager.
- *8881* - Provide the SOAP port (SOAP_CONNECTOR_ADDRESS) of the OIM cell's Deployment manager SOAP_CONNECTOR_ADDRESS.
- *wasadmin* - Provide the "primary administrative user name" configured in the OIM WebSphere cell.
- *password* - Provide the password of the primary administrative user.

Start the Node Agent and Servers

Start the servers as follows:

```
/WASSH/WebSphere/AppServer/profiles/Custom01/bin/startNode.sh

/WASSH/WebSphere/AppServer/profiles/Custom01/bin/startServer.sh soa_server1

/WASSH/WebSphere/AppServer/profiles/Custom01/bin/startServer.sh oim_server1

/WASSH/WebSphere/AppServer/profiles/Custom01/bin/startServer.sh OracleAdminServer
```

14.1.5 Copy the OAM 11*g* SSO Agent Artifacts

Complete the following steps.

1. Copy the OAM 11*g* SSO Agent artifacts created after running `configOAM` to the IHS webgate configuration location.

   For example, the following OAM 11*g* SSO Agent artifacts are created after running `configOAM`:

<*WAS_HOME*>/profiles/Custom01/output/<*webgate name*>/

These artifacts should be copied to the following location:

<*WebGate Instance Directory*>/webgate/config

2. Restart IBM HTTP Server (IHS) or Oracle HTTP Server (OHS).

14.1.6 Configure the Web Server to Route Requests to OIM on WebSphere

1. Open the IBM HTTP Server (IHS) or Oracle HTTP Server (OHS) `httpd.conf` file and locate the `WebSpherePluginConfig` entry, for example:

```
WebSpherePluginConfig /scratch/mw/was-plugin/config/ohsSLC/plugin-cfg.xml
```

2. Open the following file:

```
/mw/was-plugin/config/ohsSLC/plugin-cfg.xml
```

Edit the file as follows:

a. Locate the `UriGroup` element and add the following entries:

```
<Uri Name="/identity/">
        <Uri Name="/sysadmin/">
        <Uri Name="/oim/">
```

b. Locate the `VirtualHostGroup` element and make sure the "default_host" `VirtualHostGroup` has an entry as follows:

```
<VirtualHost Name=":XXXX">
```

where XXXX is the HTTP Server port, for example: 7777.

c. Locate the `ServerCluster` element and make sure that the `Transport` elements have the correct OIM and OAM WebSphere HOST and PORT properties configured:

```
<Transport Hostname="sdf1234.us.example.com" Port="14000" Protocol="http"/>
<TransportHostname="sdf1234.us.example.com"Port="14001"Protocol="https">
   <Property name="keyring"
value="/scratch/aime1/ihs-webgate/Plugins/etc/plugin-key.kdb"/>
   <Property name="stashfile"
value="/scratch/aime1/ihs-webgate/Plugins/etc/plugin-key.sth"/>
</Transport>

<Transport Hostname="jkl555.us.example.com" Port="14100" Protocol="http"/>
<Transport Hostname="jkl555.us.example.com" Port="14101" Protocol="https">
  <Property name="keyring"
value="/scratch/aime1/ihs-webgate/Plugins/etc/plugin-key.kdb"/>
  <Property name="stashfile"
value="/scratch/aime1/ihs-webgate/Plugins/etc/plugin-key.sth"/>
 </Transport>
```

where:

`"sdf1234.us.example.com"` is the OIM host

`"14000"` is the OIM HTTP port

`"14001"` is the OIM HTTPS port

`"jkl555.us.example.com"` is the OAM host

`"14100"` is the OAM HTTP port

`"14101"` is the OAM HTTPS port

3. Restart IBM HTTP Server (IHS) or Oracle HTTP Server (OHS).

14.2 Integrating Access Manager and OAAM on IBM WebSphere

For an overview of Access Manager and OAAM integration, see the "Integrating Oracle Adaptive Access Manager with Access Manager" and "Integrating Access Manager, OAAM, and OIM" chapters in *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*. Differences between setup in IBM WebSphere and WebLogic are noted in this section.

14.2.1 Configuring OAAM Basic Integration with Access Manager

OAAM Basic integration with Access Manager is a native integration. Access Manager is integrated with Oracle Adaptive Access Manager through the shared libraries, which provide the rules engine and the runtime functionality of Oracle Adaptive Access Manager. The OAAM Server is not needed in this deployment since the OAAM runtime functionality is available through the libraries.

14.2.1.1 Prerequisites for OAAM Basic Integration with Access Manager

Prior to configuring Access Manager with Oracle Adaptive Access Manager, you must have installed all the required components, including any dependencies, and configured the environment in preparation of the integration tasks. For information on the required components that must be installed and configured before the integration tasks are performed, see "Prerequisites for OAAM Basic Integration with Access Manager" in *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.

Instead of WebLogic, you will install and configure Oracle Fusion Middleware with IBM WebSphere. You must first install (but not configure) IBM WebSphere and apply the latest Fix Pack for IBM WebSphere.

14.2.1.2 Protecting Resource in Authentication Policy with OAAMBasic Scheme

The IDMDomainAgent is not used on IBM WebSphere. You must register a new WebGate agent. For information on managing an authentication scheme, see the "Managing Authentication and Shared Policy Components" chapter in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service*. For general information with links to more detailed sections on registering an agent, see the "Introduction to Agents and Registration" chapter in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service*.

14.2.1.3 Creating User with Privileges to Log into the OAAM Administration Console

By default there is not a user that has the correct privileges to log in to the OAAM Administration Console. You must create a user that has the correct privileges to log in to the OAAM Administration Console and then grant the necessary groups to the user. For details on creating user and groups in IBM WebSphere, see the IBM WebSphere documentation. Users and groups should be defined in LDAP directory configured for the IBM WebSphere cell. An example of an OAAM user is `oaamadmin`. For information on the OAAM roles, see Section 8.1.3, "Creating User with Privileges to Log into the OAAM Administration Console."

14.2.1.4 Modifying oam-config.xml

Locate and modify the `oam-config.xml` file manually. The `oam-config.xml` file contains all Access Manager-related system configuration data and is located in the *was_profile_dir*`/config/cells/`*cell_name*`/fmwconfig` directory. For example,

```
/scratch/xyz/IBM/WebSphere/AppServer/profiles/Dmgr04/config/cells/adc2170813Cell02
/fmwconfig
```

Set the `OAAMEnabled` property to `true` as shown in the following example:

```
<Setting Name="OAAM" Type="htf:map">
<Setting Name="OAAMEnabled" Type="xsd:boolean">true</Setting>
<Setting Name="passwordPage" Type="xsd:string">/pages/oaam/password.jsp</Setting>
<Setting Name="challengePage"
Type="xsd:string">/pages/oaam/challenge.jsp</Setting>
<Setting Name="registerImagePhrasePage"
Type="xsd:string">/pages/oaam/registerImagePhrase.jsp</Setting>
<Setting Name="registerQuestionsPage"
Type="xsd:string">/pages/oaam/registerQuestions.jsp</Setting>
```

14.2.1.5 Starting the OAAM Admin Server

Start the OAAM Admin Server to register the newly created managed servers with the domain.

1. Open a command prompt and change to the following bin directory:

   For example:

   WAS_HOME`/profiles/`*Custom01*`/bin`

2. Enter the following command:

   `./startServer.sh` *oaam_admin_server*

   The default server name for the OAAM Administration Server is `oaam_admin_server1`.

14.2.1.6 Importing the OAAM Snapshot

A full snapshot of policies, rules, challenge questions, dependent components, and configurations is shipped with Oracle Adaptive Access Manager. This snapshot is required for the minimum configuration of OAAM. Import the snapshot into the system by following the instructions in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.

14.2.1.7 Shutting Down the OAAM Administration Server

Shut down the OAAM Administration Server.

1. Open a command prompt and change to the application server's bin directory:

   For example:

   WAS_HOME`/profiles/`*Custom01*`/bin`

2. Enter the following command:

   `./stopServer.sh` *oaam_admin_server* `-username` *username* `-password` *password*

   The default server name for the OAAM Administration Server is `oaam_admin_server1`.

14.2.1.8 Creating a Datasource

Use the IBM WebSphere Administrative Console to create a JDBC data source with JNDI name `jdbc/OAAM_SERVER_DB_DS`. The data source should be created at Cell level. Ensure that the OAAM managed server is selected in the cell scope. For information on creating a data source on IBM WebSphere, see Section 3.2.5, "Creating a Data Source in an IBM WebSphere Cell."

14.2.1.9 Deploying the Shared Library

Shared libraries are files used by multiple applications. Access Manager uses shared library files in OAAM Basic Integration with Access Manager; therefore, you must create the shared library in the scope of the application/managed server where Access Manager is deployed, and Access Manager must refer to this shared library.

14.2.1.9.1 Creating the Shared Library To make library files available to multiple applications, create a shared library by following these steps:

1. Log in to the IBM WebSphere Administrative Console.
2. In the console navigation tree, expand **Environment** and then select **Shared libraries**.
3. In the Shared Libraries page, select **Show scope selection drop-down list with the all scopes option**, and then, select a shared library scope and click **Apply**. For example, for Access Manager, select the scope of Oracle Access Manager as server scope.
4. Above the table, under Preferences, click the **New** button to create a new shared library in the scope as selected in step 3.
5. In the settings page of the shared library, specify a name for the shared library in the **Name** field. For example `oaam_native_shared_library`.
6. In the **Classpath** text box, specify the absolute paths of all the JAR files present in the following directory:

   ```
   MW_HOME/Oracle_IDM1/oaam/oaam_libs/was_native_jar
   ```

   These are the paths that the product searches for classes and resources of the shared library.

   > **Note:** Each entry should be in a new line, do not use any path separator like ";" or ":".

7. Click **OK** and then **Save**.

14.2.1.9.2 Adding the Shared Library Reference to Application To add the Shared Library reference to the application:

1. Log in to the IBM WebSphere Administrative Console.
2. In the console navigation tree, expand **Applications**, and then **Application Types** and click **WebSphere enterprise applications** to open the list of applications.
3. In the Enterprise Applications page, click the application to which you want to associate the shared library.

4. Under Configurations, click **Shared library references** to access the Shared library references page.
5. In the Shared Library Mapping for Modules section, select the checkbox next to the application you want to associate to the shared library, and then click the **Reference Shared Libraries** button above the table.
6. Select one or more shared libraries that the application will use in the **Available** list and add them to the **Selected** list. Then click **OK**.
7. Save the changes to the configuration.

14.2.1.10 Synchronizing the Node and Restarting the Server

Synchronize the node and restart the server that host the application referencing the shared library.

14.2.1.11 Setting the OAAM Image Directory for Virtual Authentication Devices

For images to be displayed in the virtual authentication devices during the OAAM Registration flow, perform the following steps:

1. Log in to the OAAM Administration Console with the Environment Administrator role.
2. In the Navigation pane, double-click **Properties** under the **Environment** node. The **Properties Search** page is displayed.
3. Enter `bharosa.image.dirlist` in the **Name** field and click **Search**.
4. Click to select the property in the Search Results section.
5. Add `${OAM_ORACLE_HOME}/../oaam/oaam_images` as comma-separated value in the Value column. That is, change the value `${oracle.oaam.home}/oaam_images` to `${oracle.oaam.home}/oaam_images, ${OAM_ORACLE_HOME}/../oaam/oaam_images`.
6. Click **Save**.

   A confirmation dialog is displayed.
7. Click **OK** to dismiss the dialog.

14.2.1.12 Testing the Configuration

To test the configuration:

1. To verify the configuration, remote register two agents, each protecting a resource.
2. Use the Oracle Access Management Console to associate the first resource with `OAAMBasic` for the authentication flow. Associate the second resource with the `LDAPScheme`.

   See Also: "Managing Authentication Schemes" in Oracle Fusion Middleware Administrator's Guide for Oracle Access Management.
3. Access the protected resource configured earlier to verify the configuration.

   You are prompted to enter a user name. Then, on a separate screen you are prompted for the password.

   Once the user name and password are validated you are asked to select and answer three challenge questions. Once completed you are taken to the protected application.

14.2.2 Configuring OAAM Advanced Integration with Access Manager

Integrating Oracle Adaptive Access Manager with Oracle Access Manager provides an enterprise with advanced access security features that greatly improve the level of protection for applications. OAAM Advanced integration with Access Manager can involve scenarios with or without Oracle Identity Manager.

Integration with Oracle Identity Manager provides users with richer password management functionality, including secure "Forgot Password" and "Change Password" flows. For details about integrating with Oracle Identity Manager, see Section 14.3, "Integrating Access Manager, OAAM, and OIM on IBM WebSphere."

If Oracle Identity Manager is not part of your environment, follow the integration procedure described in this section.

14.2.2.1 OAAM Advanced Integration with Access Manager Roadmap

Table 14–2 summarizes the steps to configure OAAM Advanced Integration with Access Manager on IBM WebSphere.

Table 14–2 Integration Flow for Access Manager and Oracle Adaptive Access Manager

No	Task	Information
1	Verify that all required components have been installed and configured prior to integration.	For information, see "OAAM Advanced Integration with Access Manager Prerequisites".
2	Ensure the Oracle Access Management and OAAM Administration Consoles and managed servers are running.	For information, see "Restarting the Servers".
3	Create the OAAM users. Before you can access the OAAM Administration Console, you must create administration users.	For information, see "Creating Users and Groups".
4	Import the OAAM base snapshot. A full snapshot of policies, dependent components and configurations is shipped with Oracle Adaptive Access Manager. For Oracle Adaptive Access Manager to be functional, you must import the base OAAM snapshot into the system.	For information, see "Importing Base Snapshot in OAAM".
5	Validate that Access Manager was set up correctly. You should be able to log in to the Oracle Access Management Console successfully.	For information, see "Validating Initial Configuration of Access Manager".
6	Validate that OAAM was set up correctly.	For information, see "Validating Initial Configuration of Oracle Adaptive Access Manager".
7	Register the WebGate. The WebGate is an out-of-the-box access client. This Web server access client intercepts HTTP requests for Web resources and forwards these to the OAM Server 11*g*.	For information, see "Provisioning WebGate Using the Oracle Access Management Console".
8	Register the OAAM server to act as a trusted partner application to Access Manager. A partner application is any application that delegates the authentication function to Access Manager 11*g*.	For information, see "Setting Up Access Manager for Integration with OAAM and Register OAAM as Thirdparty in Access Manager".
9	Specify the Agent password in multiple places. OAAM needs this agent password in order to use the agent profile for integration.	For information, see "Setting the Agent Password".
10	Verify TAP partner registration using the Oracle Access Management tester.	For information, see "Verifying TAP Partner Registration".

Table 14–2 (Cont.) Integration Flow for Access Manager and Oracle Adaptive Access

No	Task	Information
11	Set up TAP integration properties in OAAM.	For information, see "Setting Up OAAM for TAP Integration".
12	Configure the integration to use OAAM TAPScheme to protect Identity Management product resources in the IAMSuiteAgent application domain.	For information, see "Moving the /oamTAPAuthenticate URL".
13	Update the authentication scheme in the policy-protected resource policy to protect a resource with the OAAM TAPScheme.	For information, see "Updating the Authentication Scheme in the Policy-Protected Resource Policy".
14	Validate the Access Manager and Oracle Adaptive Access Manager integration.	For information, see "Validating the Access Manager and Oracle Adaptive Access Manager Integration".

14.2.2.2 OAAM Advanced Integration with Access Manager Prerequisites

Prior to configuring Access Manager with Oracle Adaptive Access Manager, you must have installed all the required components, including any dependencies, and configured the environment in preparation of the integration tasks. For information, see "Prerequisites for OAAM Advanced Integration with Access Manager" in *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.

You will be installing IBM WebSphere instead of WebLogic. For information on the system requirements and certifications, necessary software media and downloads, and installation instructions to install and configure Oracle Fusion Middleware with IBM WebSphere, see Chapter 2, "Installing and Configuring Oracle Identity and Access Management on IBM WebSphere."

14.2.2.3 Restarting the Servers

1. Start the IBM Deployment Manager.

   To start the deployment manager, navigate to the following directory in the IBM WebSphere home and enter the following command:

   ```
   (UNIX) WAS_HOME/profiles/deployment_mgr_profile_name/bin/startManager.sh
   ```

   For example, on a UNIX operating system:

   ```
   /opt/IBM/WebSphere/AppServer/profiles/Dmgr01/bin/startManager.sh
   ```

   WAS_HOME is the path to the `AppServer` directory where IBM WebSphere is installed.

   If you are integrating Access Manager and OAAM, and OAAM is in a different IBM WebSphere Cell, you must also start the Deployment Manager located in the IBM WebSphere profile where OAAM is installed.

2. Start the managed server hosting the OAM Server.

   ```
   OAM_PROFILE/bin/startServer.sh oam_server1
   ```

   OAM_PROFILE is the IBM WebSphere profile where OAM is installed.

   For example:

   ```
   OAM_PROFILE - $WAS_HOME/profiles/Custom01
   ```

3. Start the managed server hosting OAAM Admin Server.

   To start the OAAM Administration Server, navigate to the following directory in the IBM WebSphere home and enter the following command:

   (UNIX) OAAM_PROFILE/bin/startServer.sh *oaam_admin_server*

   The default server name for the OAAM Administration Server is oaam_admin_server1.

   For example, on a UNIX operating system:

   ```
   /opt/IBM/WebSphere/AppServer/profiles/Custom01/bin/startServer.sh
   oaam_admin_server1
   ```

   OAAM_PROFILE is the IBM WebSphere profile where OAAM Administration Server is installed.

   For example, on the UNIX operating system, OAAM_PROFILE:

   WAS_HOME/profiles/*profile_name*

4. Start the managed server hosting the OAAM runtime server.

   To start the OAAM runtime server, navigate to the following directory in the IBM WebSphere home and enter the following command:

   (UNIX) *OAAM_PROFILE*/bin/startServer.sh *oaam_server_server*

   The default server name for the OAAM runtime server is oaam_server_server1.

   OAAM_Profile is the IBM WebSphere profile where OAAM Server is installed.

   For example, on a UNIX operating system:

   ```
   /opt/IBM/WebSphere/AppServer/profiles/Custom01/bin/startServer.sh
   oaam_server_server1
   ```

   For example, on a UNIX operating system, OAAM_PROFILE:

   WAS_HOME/profiles/*profile_name*

14.2.2.4 Creating Users and Groups

By default there is not a user that has the correct privileges to log in to the OAAM Administration Console. You must create a user that has the correct privileges to log in to the OAAM Administration Console and then grant the necessary groups to the user. For details on creating user and groups in IBM WebSphere, see the IBM WebSphere documentation. Users and groups should be defined in the LDAP directory configured for the IBM WebSphere cell. An example of an OAAM user is oaamadmin. For information on the OAAM roles, see Section 8.1.3, "Creating User with Privileges to Log into the OAAM Administration Console."

14.2.2.5 Importing Base Snapshot in OAAM

A full snapshot of policies, rules, challenge questions, dependent components, and configurations is shipped with Oracle Adaptive Access Manager. This snapshot is required for the minimum configuration of OAAM. Import the snapshot into the system by following the instructions in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*

14.2.2.6 Validating Initial Configuration of Access Manager

Verify that Access Manager is set up correctly by accessing the Welcome to Oracle Access Management page.

1. Log in to the Oracle Access Management Console:

   `http://oam_adminserver_host:oam_adminserver_port/oamconsole`

   You should be redirected to the OAM Server for login.

2. Provide the administrator user name and password.

   If the login is successful, the Welcome to Oracle Access Management page is displayed.

14.2.2.7 Validating Initial Configuration of Oracle Adaptive Access Manager

Verify that Oracle Adaptive Access Manager is set up correctly by accessing the OAAM Server.

1. Log in to the OAAM Server.

   `http://host:port/oaam_server`

2. Provide any user name and click **Continue**.
3. Provide the password as `test` because the Access Manager and Oracle Adaptive Access Manager integration has not yet been performed. You must change the password immediately after the integration.
4. Click the **Enter** button on the virtual authentication device.
5. Click **Continue** to register the new user.
6. Click **Continue** to accept the security device.
7. Choose questions and provide answers to register for Knowledge Based Authentication (KBA).

   A successful login indicates that you have configured the initial configuration correctly.

Note: The test login URL `/oaam_server` is used to verify that the OAAM configuration is working before proceeding with the integration of Access Manager. This URL is not intended for use after the integration of Access Manager and OAAM.

After integration, if the user navigates to the URL and enters his username, he is directed to the page where the password is entered. After submitting the password, the login will fail and the following error will be displayed:

```
Error Sorry, the identification you entered was not recognized.
Please try again
```

14.2.2.8 Provisioning WebGate Using the Oracle Access Management Console

Agents communicate with the OAM Server to check protected resources and configured access policies. Registering an agent sets up the required trust mechanism between the agent and the OAM Server.

Ensure that the following are installed and configured before registering the WebGate:

- IBM HTTP Server and its required plugins

  For information on installing and configuring IBM HTTP (IHS), refer to the IBM HTTP product documentation.

- IHS WebGate

  For information on installing the IHS 11*g* WebGate for Access Manager, see the "Installing and Configuring IHS 11g WebGate for OAM" chapter in *Oracle Fusion Middleware Installing WebGates for Oracle Access Manager*.

14.2.2.8.1 Registering the WebGate as a Partner

For information on registering the IHS 11*g* WebGate, see "Registering the New IHS 11*g* WebGate" in *Oracle Fusion Middleware Installing WebGates for Oracle Access Manager*.

14.2.2.8.2 Copy the Access Management 11*g* SSO Agent Artifacts

Copy the Access Management 11*g* SSO Agent artifacts created after registering the Agent to the IHS WebGate configuration location. For information, see "Copying Generated Files and Artifacts to the HTTP Server WebGate Instance Location" in *Oracle Fusion Middleware Installing WebGates for Oracle Access Manager*.

14.2.2.8.3 Starting the IBM HTTP Server

Start the IHS Server. For information, see "Starting the IHS Server and Accessing the IHS Resource" in *Oracle Fusion Middleware Installing WebGates for Oracle Access Manager*.

Once you start the IHS Web Server, log in to the IHS Web Server. For example:

```
http://machine_name.my.company.com:port
```

WebGate intercepts the request and redirects you to the Oracle Access Management Console. Enter the username and password, and you are redirected to the IBM HTTP Server.

14.2.2.9 Setting Up Access Manager for Integration with OAAM and Register OAAM as Thirdparty in Access Manager

To register the OAAM Server as a trusted partner application to Access Manager, follow the steps in "Registering the OAAM Server as a Partner Application to Access Manager" in the *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.

> **Note:** Ensure that the IBM WebSphere Deployment Manager where Access Manager is installed is running.

1. Set up the environment for WSADMIN.
2. Navigate to the *IAM_ORACLE_HOME*`/common/bin` directory.
3. Execute `wsadmin.sh` to enter the `wsadmin`.

   For example:

   ```
   ./wsadmin.sh -port 8879 -user wasadmin -password some-value -conntype soap
   -lang jython
   ```

   port is SOAP PORT for deployment manager

For information on figuring the SOAP port, see the IBM WebSphere documentation.

4. In another terminal window, create the `keystore` directory by executing the following:

```
mkdir IAM_ORACLE_HOME/TAP/TapKeyStore
```

5. Using the `WSADMIN` shell, run the `Oam.registerThirdPartyTAPPartner` command:

```
Oam.registerThirdPartyTAPPartner(partnerName="partnerName",
keystoreLocation= "path_to_keystore" , password="keystore_password",
tapTokenVersion="v2.0", tapScheme="TAPScheme", tapRedirectUrl="OAAM login URL")
```

The command registers any third party as a Trusted Authentication Protocol (TAP) Partner.

An example is provided below.

```
Oam.registerThirdPartyTAPPartner(partnerName = "OAAMTAPPartner",
keystoreLocation= "IAM_ORACLE_HOME/TAP/TapKeyStore/mykeystore.jks" ,
password="password", tapTokenVersion="v2.0", tapScheme="TAPScheme",
tapRedirectUrl="http://OAAM_ Managed_server_host:14300/
oaam_server/oamLoginPage.jsp")
```

Table 14–3 TAP Partner Setup

Parameter	Details
partnerName	The name of the partner should be unique. It can be any name used for identifying the third party partner. If the partner exists in Access Manager, the configuration will be overwritten.
keystoreLocation	The keystore location is an existing location. If the directory path specified is not present, an error occurs. You must provide the complete path including the keystore file name. In the example shown earlier, the keystore location was `IAM_ORACLE_HOME/TAP/TapKeyStore/mykeystore.jks`. Another example is `keystoreLocation= "/scratch/jsmith/dwps1tap/TapKeyStore/mykeystore.jks"`. When you run the command `registerThirdPartyTAPPartner`, the keystore file is created in that location specified.
password	The keystore password used to encrypt the keystore. The keystore is created by running command "registerThirdPartyTAPPartner" in the location as specified for parameter "keystoreLocation". Make a note of the password as you will need it later.
tapTokenVersion	Version of the Trusted Authentication Protocol. `tapTokenVersion` is always `v2.0` for 11.1.1.5.0 and 11.1.2.0. If using IDContext Claims, it is `v2.1`.
tapScheme	Trusted Authentication Protocol Authentication Scheme (TAPScheme out of the box.) This is the authentication scheme that will be updated. If you want two tap partners with different tapRedirectUrls, create a new authentication scheme using the Oracle Access Management Console and use that scheme here. The authentication scheme will be created automatically while you are running the `registerThirdPartyTAPPartner` command in the instructions above. The name of `TAPScheme` will be passed as parameter to that command. The example command has `tapScheme="TAPScheme"`.

Table 14–3 (Cont.) TAP Partner Setup

Parameter	Details
tapRedirectUrl	Third party access URL. The TAP redirect URL should be accessible. If it is not, registration of the partner fails with the message: `Error! Hyperlink reference not valid.` `tapRedirectUrl` is constructed as follows: `http://`*`oaamserver_host`*`:`*`oaamserver_port`*`/oaam_server/oamLoginPage.jsp` Ensure that the OAAM server is running; otherwise registration will fail. The credential collector page will be served by the OAAM Server. The authentication scheme created by `registerThirdPartyTAPPartner` (TAPScheme) points to the OAAM Server credential collector page as the redirectURL.

14.2.2.10 Setting the Agent Password

You will need to specify the Agent password in multiple places. OAAM needs this agent password in order to use the agent profile for integration.

14.2.2.10.1 Adding a Password to the IAMSuiteAgent Profile in the Oracle Access Management Console When Access Manager is installed, a default agent profile called IAMSuiteAgent is created. This profile is used by OAAM when integrating with Access Manager. When the IAMSuiteAgent profile is first created, it has no password. You must set a password before the profile can be used by OAAM for integration. To do this, proceed as follows:

1. Log in to the Oracle Access Management Console.

   `http://`*`oam_adminserver_host:oam_adminserver_port`*`/oamconsole`

2. Enter the username and password.
3. Select the **System Configuration** tab.
4. Expand **Access Manager Settings**, and then **SSO Agents**.
5. Double-click **OAM Agent**.

   The WebGate page opens in the right hand pane.

6. Click **Search** to list all WebGate agents including **IAMSuiteAgent**.
7. Double-click **IAMSuiteAgent** to edit the properties.
8. Specify the password in the **Access Client Password** field and click **Apply** to save the changes.

   This is a required step.

14.2.2.11 Verifying TAP Partner Registration

To verify the TAP partner registration, follow the instructions below.

14.2.2.11.1 Verifying the Challenge URL To validate the Access Manager configuration, perform the following steps:

1. Log in to the Oracle Access Management Console.

   `http://`*`oam_adminserver_host`*`:`*`oam_adminserver_port`*`/oamconsole`

2. Enter credentials.
3. Click the **Policy Configuration** tab in the left pane of the console.

4. In the left pane, expand the **Authentication Schemes** node.
5. Double-click the **TAPScheme** authentication scheme.
6. Verify that the **Challenge Method** is `DAP` and the **Authentication Module** is `DAP`.
7. Verify that **Challenge URL** shows part of the value of the tapRedirectUrl that had been specified when OAAM was registered with Access Manager as a partner application. For example, if the `tapRedirectUrl` is `http://OAAM_Managed_server_host:14300/oaam_server/oamLoginPage.jsp`, then **Challenge URL** should show `/oaam_server/oamLoginPage.jsp`. The host and port part of the URL is parameterized in Challenge Parameter. In the Challenge Parameters field, you will see both `TAPPartnerId=OAAMPartner` and `SERVER_HOST_ALIAS=HOST_ALIAS_1`.
8. Check the challenge parameters are set correctly.

14.2.2.11.2 Adding the MatchLDAPAttribute Challenge Parameter in the TAPScheme

You must add the `MatchLDAPAttribute` challenge parameter and set it to the `User Name Attribute` as specified in the LDAP Identity Store.

1. Log in to the Oracle Access Management Console.

   `http://oam_adminserver_host:oam_adminserver_port/oamconsole`

2. Enter credentials.
3. Click the **Policy Configuration** tab to the left of the screen.
4. Expand the **Authentication Schemes** node.
5. Double-click **TAPScheme** authentication scheme.
6. To add another parameter to an existing parameter, position your cursor in the **Challenge Parameter** field and press **Enter** using your keyboard.
7. In the new line, add an entry for the challenge parameter.

   For example, `MatchLDAPAttribute=uid`

   `MatchLDAPAttribute` must be set to the `User Name Attribute` as specified in the LDAP Identity Store. For example, `uid`, `mail`, `cn`, and so on.

   Note: The challenge parameter is case-sensitive.

   For information, see "Managing User Identity Stores" in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*.
8. Click **Apply** to submit the change.
9. Dismiss the Confirmation window.

14.2.2.11.3 Validating the IAMSuiteAgent Setup

To validate the IAMSuiteAgent setup, proceed as follows:

1. Launch Oracle Access Management tester.

   IAM_ORACLE_HOME`/../`*jdk_version*`/bin/java -jar` *IAM_ORACLE_HOME*`/oam/server/tester/oamtest.jar`

   The Oracle Access Management Tester Console appears.
2. In the Server Connection section provide server connection details:

a. **IP Address**: Access Manager Managed Server Host

b. **Port**: Oracle Access Management Oracle Access Protocol (OAP) Port

c. **Agent ID**: `IAMSuiteAgent`

d. **Agent Password**: `Password` provided in Section 14.2.2.10, "Setting the Agent Password."

The Server Connection section provides fields for the information required to establish a connection to the OAM Server.

3. Click **Connect**.

   If you can connect to the server, the next section, **Protected Resource URI**, will be enabled.

4. The Protected Resource URI section provides information about a resource whose protected status needs to be validated.

   In this section, provide the protected resource URI as follows:

   a. **Host**: `IAMSuiteAgent`

   b. **Port**: `80`

   c. **Resource**: `/oamTAPAuthenticate`

   Note: You can test any other resource protected using `TAPScheme` other than `oamTAPAuthenticate`.

5. Click **Validate**

   The Validate button is used to submit the Validate Resource server request. If the validation is successful, the next section for **User Identity** will be enabled.

6. In the User Identity section, provide `User Identity` and click **Authenticate**. If the authentication is successful, the setup is successful.

   This section provides information about a user whose credentials need to be authenticated. The Authenticate button is used to submit the Authenticate User server request.

14.2.2.12 Setting Up OAAM for TAP Integration

To set up OAAM for TAP Integration, proceed as follows:

1. Set the environment variable `APP_SERVER_TYPE_CODE` to `2`.

```
export APP_SERVER_TYPE_CODE=2
```

2. Run `setupOAMTapIntegration.sh` to configure Access Manager for TAP Integration as documented in "Setting Up Access Manager TAP Integration Properties in OAAM" in *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.

For information on running CLI scripts, see Section 8.1.4, "Setting Up the CLI Environment for OAAM on IBM WebSphere."

14.2.2.13 Moving the /oamTAPAuthenticate URL

If you are planning to use the IAMSuiteAgent application domain to set as the TAPScheme, you must move the `/oamTAPAuthenticate` URL into a separate authentication policy.

To use TAPscheme for Identity Management product resources in the IAM Suite domain, `Protected HigherLevel Policy`, the following configuration must be performed:

1. Log in to the Oracle Access Management Console.
2. In the navigation tree, double-click **Application Domains**, then click **Search**, and in the search results, click **IAM Suite**.
3. Click the **Authentication Policies** tab.
4. Click **Protected Higher Level Policy**.
5. In the Resources window click **/oamTAPAuthenticate**.
6. Click **Delete**, and then **Apply**.
7. Create a new Authentication Policy in the IAMSuite application domain.
8. For authentication scheme, choose **LDAP Scheme**.
9. In the Resources window, click **Add**.
10. Select the resource **/oamTAPAuthenticate.**
11. Click **Apply**.

For Access Manager to be able to override the resource URL before handing it off to OAAM, you must set up the `TAPOverrideResource` challenge parameter in `TAPScheme`.

1. Log in to the Oracle Access Management Console:

   `http://`*`oam_adminserver_host`*`:`*`oam_adminserver_port`*`/oamconsole`

2. Click the **Policy Configuration** tab to the left of the screen.
3. Expand the **Authentication Schemes** node.
4. Double-click **TAPScheme** authentication scheme.
5. To add another parameter to an existing parameter, position your cursor in the **Challenge Parameter** field and press **Enter** using your keyboard.
6. In the new line, add `TAPOverrideResource=http://IAMSuiteAgent:80/oamTAPAuthenticate` for a challenge parameter of TAPScheme.
7. Click **Apply**.

14.2.2.14 Updating the Authentication Scheme in the Policy-Protected Resource Policy

To protect a resource with the OAAM `TAPScheme`, you must edit the policy-protected resource policy and update the authentication scheme. This section provides general steps to do this.

For detailed instructions, see the *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.

1. Log in to Oracle Access Management Administration Console.

   `http://`*`oam_adminserver_host`*`:`*`oam_adminserver_port`*`/oamconsole`

2. Check for the application domain that was created as part of the 11*g* WebGate registration.
3. Edit the authentication policy-protected resources policy.

4. Update the authentication scheme to `TAPScheme` specified as the `tapScheme` parameter in `registerThirdPartyTAPPartner` command.
5. Click **Apply** to save changes.

14.2.2.15 Validating the Access Manager and Oracle Adaptive Access Manager Integration

Try to access the protected resource. You should be redirected to OAAM for registration and challenge. The OAAM login page is shown instead of the Access Manager login page.

14.3 Integrating Access Manager, OAAM, and OIM on IBM WebSphere

Integration with Oracle Identity Manager provides users with richer password management functionality, including secure "Forgot Password" and "Change Password" flows.

14.3.1 Access Manager, OAAM, and OIM Integration Roadmap

Table 14–4 lists the high-level tasks for integrating Access Manager, Oracle Adaptive Access Manager, and Oracle Identity Manager.

Table 14–4 Integration Flow for Access Manager, OAAM, and OIM

No	Task	Information
1	Verify that all required components have been installed and configured prior to integration.	For information, see *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite* and the IBM WebSphere documentation.
2	Install Access Manager, Oracle Adaptive Access Manager, and IBM WebSphere servers.	For information, see Chapter 2, "Installing and Configuring Oracle Identity and Access Management on IBM WebSphere."
3	Integrate Access Manager and Oracle Identity Manager.	For information, see *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.
4	Enable LDAP synchronization for Oracle Identity Manager. This is required for integration between Access Manager, Oracle Adaptive Access Manager, and Oracle Identity Manager.	For information, see *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.
5	Integrate Access Manager and Oracle Adaptive Access Manager.	For information, see Section 14.2.2, "Configuring OAAM Advanced Integration with Access Manager."
6	Set up the integration between Oracle Adaptive Access Manager and OIM.	For information, see Section 14.3.7, "Integrating Oracle Identity Manager and Oracle Adaptive Access Manager."
7	Migrate OAAM policies to the policy store	For information, see Section 14.3.8, "Migrating OAAM Policies."
8	Enable OAAM to generate HTTP post-based messages to Access Manager	For information, see Section 14.3.9, "Enabling OAAM to Generate HTTP Post-Based Messages to Access Manager."

14.3.2 Access Manager, Oracle Adaptive Access Manager, and OIM Integration Prerequisites

Prior to integrating Access Manager, Oracle Adaptive Access Manager, and Oracle Identity Management, you must have installed all the required components, including any dependencies, and configured the environment in preparation of the integration tasks that follow.

For information on required components that must be installed and configured before the Access Manager, Oracle Adaptive Access Manager, and Oracle Identity Management integration tasks are performed, see *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.

The steps below are based on the assumption that Access Manager and Oracle Identity Manager are integrated using the out-of-the box integration.

14.3.3 Installing Access Manager, Oracle Adaptive Access Manager, and Oracle Identity Manager

For information on installing Access Manager, OAAM, and OIM, see *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.

For this integration, Oracle Identity Manager and OAAM must reside in a single cell.

When you run the `was_config` command to extend an Oracle Adaptive Access Manager cell to include the Oracle Identity Manager template, you may see the following warning:

```
Conflict detected
CFGFWK -42001: The following duplicate elements exists in a configuration,
discarding new elements from a incomming template
```

You can safely ignore this error message.

14.3.4 Integrating Access Manager and Oracle Identity Manager

Integration between Oracle Identity Manager and Access Manager is required for integration between Access Manager, Oracle Adaptive Access Manager, and Oracle Identity Manager. For more information, see *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.

14.3.5 Enabling LDAP Synchronization for Oracle Identity Manager

Enabling LDAP synchronization for Oracle Identity Manager is required for integration between Access Manager, Oracle Adaptive Access Manager, and Oracle Identity Manager. For more information, see *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.

14.3.6 Integrating Access Manager and Oracle Adaptive Access Manager

Note: In the integration of Access Manager, Oracle Identity Management, and Oracle Adaptive Access Manager, the `IdentityManagerAccessGate` profile should already exist since it is configured during the Access Manager and Oracle Identity Management integration. For details, see *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.

Configure the Access Manager and Oracle Adaptive Access Manager integration so that the OAAM server acts as a trusted partner application. For information on integrating Oracle Adaptive Access Manager and Access Manager, refer to "Integrating Access Manager and OAAM on IBM WebSphere"

14.3.7 Integrating Oracle Identity Manager and Oracle Adaptive Access Manager

This section describes how to integrate Oracle Identity Management and Oracle Adaptive Access Manager for the three-way integration of Access Manager, Oracle Identity Management, and Oracle Adaptive Access Manager:

- Adding OAAM Users and Groups from the OIM Console
- Setting Oracle Identity Manager Properties for Oracle Adaptive Access Manager
- Updating OAAM Properties to Enable Integration Between Oracle Identity Manager and OAAM
- Configuring Oracle Identity Manager Credentials in the Credential Store Framework

14.3.7.1 Adding OAAM Users and Groups from the OIM Console

To be able to access the OAAM Admin Console, OAAM user and OAAM roles must be created in the identity store OIM is pointing to. You can add the users and groups from the Oracle Identity Manager System Administrative Console. For information, see "Managing Users" and "Managing Roles" in *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Manager*.

14.3.7.2 Setting Oracle Identity Manager Properties for Oracle Adaptive Access Manager

In Oracle Identity Manager, the `OIM.ChangePasswordURL` and `OIM.ChangePasswordURL` properties must be set to valid OAAM URLs, and `OIM.DisableChallengeQuestions` must be set to `true` for Oracle Adaptive Access Manager to provide the challenge questions functionality instead of Oracle Identity Manager.

To modify Oracle Identity Manager properties, take these steps:

1. Log in to the Oracle Identity Manager System Administrative Console.
2. Click **Configuration** in **System Management** and under **System Management**, click the **System Configuration** link.
3. In the pop-up window, click on **Advanced Search**.
4. Set the following properties and click **Save**.

> **Note:** For the URLs, use the hostnames as they were configured in Access Manager. For example, if a complete hostname (with domain name) was provided during Access Manager configuration, use the complete hostname for the URLs.

Table 14–5 Oracle Identity Manager Redirection

Property	Description and Value
OIM.DisableChallengeQuestions	TRUE

Table 14–5 (Cont.) Oracle Identity Manager Redirection

Property	Description and Value
OIM.ChangePasswordURL	The URL for the change password page in Oracle Adaptive Access Manager is: `http://oaam_server_managed_server_host:oaam_server_managed_server_port/oaam_server/oimChangePassword.jsp` In a high availability (HA) environment, set this property to point to the virtual IP URL for the OAAM server.
OIM.ChallengeQuestionModificationURL	The URL for the challenge questions modification page in Oracle Adaptive Access Manager is: `http://oaam_server_managed_server_host:oaam_server_managed_server_port/oaam_server/oimResetChallengeQuestions.jsp`

5. Restart the Oracle Identity Manager managed server.

14.3.7.3 Updating OAAM Properties to Enable Integration Between Oracle Identity Manager and OAAM

To set OAAM properties for Oracle Identity Manager:

1. Log in to the OAAM Administration Console:

   `http://oaam_managed_server_host:oaam_admin_managed_server_port/oaam_admin`

   You must log in as a user with access to the Properties Editor.

2. In the navigation tree, double-click **Properties** under the Environment node. The Properties search page is displayed.

3. Enter the name of the property you want to set in the **Name** field and click **Search**.

   Note: If the search for a property displays no records, you must create the property. For instructions on creating a property, see "Creating a New Database Type Property" in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.

4. Click to select the property in the Search Results section.

5. In the **Value** field, enter the new value and click **Save**.

   A confirmation dialog is displayed.

6. Click **OK** to dismiss the dialog.

For the following properties, set the values according to your deployment:

Table 14–6 Oracle Identity Manager Integration Properties

Property Name	Property Values
bharosa.uio.default.user.management.provider.classname	`com.bharosa.vcrypt.services.OAAMUserMgmtOIM`

Table 14–6 (Cont.) Oracle Identity Manager Integration Properties

Property Name	Property Values
oaam.oim.auth.login.config	`${oracle.oaam.home}/../designconsole/config/`**`authws.conf`** Note: `authws.conf` is not the out-of-the-box value.
oaam.oim.url	`corbaloc:iiop:`*`host`*`:`*`port`* *port* is the bootstrap port of the OIM server
oaam.oim.xl.homedir	`${oracle.oaam.home}/../designconsole`
bharosa.uio.default.signon.links.enum.selfregistration.url	The URL for Self Registrations is as follows: `http://`*`OIM-Managed-Server-Host`*`:`*`OIM-Managed-Server-Port`*`/identity/faces/register?&backUrl=`*`backURL`*
bharosa.uio.default.signon.links.enum.trackregistration.url	The URL for Track Registrations is as follows: `http://`*`OIM-Managed-Server-Host`*`:`*`OIM-Managed-Server-Port`*`/identity/faces/trackregistration?&backUrl=`*`backURL`*
bharosa.uio.default.signon.links.enum.trackregistration.enabled	true
bharosa.uio.default.signon.links.enum.selfregistration.enabled	true
oaam.oim.csf.credentials.enabled	true This property enables the configuring of credentials in the Credential Store Framework as opposed to maintaining them using the Properties Editor. This step is performed so that credentials can be securely stored in CSF.
oaam.oim.passwordflow.unlockuser	true This property enables automatic unlocking of the user in the Forgot Password flow.
oaam.oim.initial.context.factory	`com.ibm.websphere.naming.WsnInitialContextFactory`

14.3.7.4 Configuring Oracle Identity Manager Credentials in the Credential Store Framework

Oracle Adaptive Access Manager must have the credentials of an OIM Administrator in order to perform various activities. A key for Oracle Identity Manager WebGate credentials is created in MAP `oaam`. So that the OIM credentials can be securely stored in the Credential Store Framework, follow the steps below to add a password credential to the OAAM domain.

1. Log in to the Oracle Fusion Middleware Enterprise Manager Console:

   `http://`*`websphere_host`*`:`*`OracleAdminServer_port`*`/em`

where *port* is OracleAdminServer HTTP port.

You must log in as an IBM WebSphere Administrator. For example, `wasadmin`.

2. Expand the **Websphere Cell** icon in the navigation tree in the left pane.
3. Select your IBM WebSphere cell, right-click, and select the menu option **Security** and then the option **Credentials** in the submenu.
4. Click **oaam** to select the map, then click **Create Key**.
5. In the pop-up dialog, ensure that **Select Map** is **oaam**.
6. Provide the following properties and click **OK**.

Table 14–7 Oracle Identity Manager Credentials

Property	Value
Map Name	`oaam`
Key Name	`oim.credentials`
Key Type	Password
UserName	User name of Oracle Identity Manager Administrator
Password	Password of Oracle Identity Manager Administrator

14.3.8 Migrating OAAM Policies

Follow these steps to migrate the OAAM policies:

1. From the IBM WebSphere Administrative Console, click **System administration** > **Deployment manager** to display the Configuration tab.
2. Expand Java and Process Management, and then click **Process definitions**.
3. Under Additional Properties, click **Java Virtual Machine**.
4. Under Additional Properties, click **Custom Properties**.

   All custom properties are listed on the next page.
5. Click the custom property **ws.ext.dirs** to edit its value. If the custom property is not displayed in the same page, you can use the **Next** button at the bottom of the page to navigate to the property or to search for the property in the list.
6. For the entry named **ws.ext.dirs**, which is the deployment manager's JVM custom property, edit the value in the Value field to append the absolute path to the location where the `oamAuthnProvider.jar` file is stored. Use delimiter as ":" without quotes.

   The value to append is

   ```
   ${oracle.as.jrf_11.1.1.6.0_oracle_common_ORACLE_HOME}/modules/
   oracle.oamprovider_11.1.1/oamAuthnProvider.jar
   ```

   where `${oracle.as.jrf_11.1.1.6.0_oracle_common_ORACLE_HOME}` is the absolute path to the `oracle_common` directory, i.e., Oracle Common home.

   Note: The value of `ws.ext.dirs` contains the path to the Oracle Common home. Use this path value as part of the new value that you will be appending. Refer to the example below.

   The example is as follows:

   If the existing value of `ws.ext.dirs` is:

```
${oracle.as.jrf_11.1.1.6.0_oracle_common_ORACLE_HOME}/modules/
oracle.jrf_11.1.1/jrf-was.jar
```

Then the new value after appending the path to `oamAuthnProvider.jar` is:

```
${oracle.as.jrf_11.1.1.6.0_oracle_common_ORACLE_HOME}/modules/
oracle.jrf_11.1.1/jrf-was.jar:${oracle.as.jrf_11.1.1.6.0_oracle_common_ORACLE_
HOME}/modules/oracle.oamprovider_11.1.1/oamAuthnProvider.jar
```

7. Click **Save** to save the changes.
8. Restart the Deployment Manager.
9. Verify that the policies were migrated successfully by using Oracle Enterprise Manager Fusion Middleware Control.

   For example:

   From Oracle Enterprise Manager Fusion Middleware Control, right-click **Cell_Websphere**. Navigate to **Security** > **Application Policies**. Search with application as `oaam_admin_11.2.0.0`. If policies are migrated, you should see rows with names `OAAM*Group`.

14.3.9 Enabling OAAM to Generate HTTP Post-Based Messages to Access Manager

Access Manager and OAAM communication involves transferring information required to perform authentication, preserving Access Manager context data, and providing the TAP token. In cases where context data is large, such as form data, OAAM can be configured to generate HTTP POST-based responses back to Access Manager to preserve up to 8K of the client application's form data.

To enable OAAM to generate POST-based responses so that Access Manager's context data is preserved, you must set `oaam.uio.oam.dopost` to `true`.

1. Log in to the OAAM Administration Console:

   `http://`*`oaam_managed_server_host`*`:`*`oaam_admin_managed_server_port`*`/oaam_admin`

   You must log in as a user with access to the Properties Editor.
2. In the navigation tree, double-click **Properties** under the Environment node. The Properties search page is displayed.
3. Enter `oaam.uio.oam.dopost` in the **Name** field and click **Search**.

   Note: If the search for a property displays no records, you must create the property. For instructions on creating a property, see "Creating a New Database Type Property" in *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.
4. Click to select the property in the Search Results section.
5. Enter `true` in the **Value** field and click **Save**.

   A confirmation dialog is displayed.
6. Click **OK** to dismiss the dialog.

A

Fusion Middleware Control Page Reference

This appendix describes the features and options available on the Fusion Middleware Control pages that appear when you are managing an IBM WebSphere cell that was configured for Oracle Fusion Middleware.

This appendix contains the following sections:

- Understanding the Information on the IBM WebSphere Cell Home Page
- Understanding the Information on the WebSphere Application Server Home Page
- Understanding the Information on the IBM WebSphere Application Deployment Home Page

A.1 Understanding the Information on the IBM WebSphere Cell Home Page

The Cell home page is divided into the following regions:

- Summary Region of the Cell Home Page
- Deployments Region of the Cell Home Page
- Servers Region of the Cell Home Page
- Clusters Region of the Cell Home Page

Summary Region of the Cell Home Page

The Summary region of the Cell home page provides general information about the cell, as well as a link to the IBM WebSphere Administrative Console, which you can use to manage the cell.

Table A–1 describes the fields available in the General section of the Summary region.

Table A–1 Fields Available in the General Section of the Summary Region

Element	Description
Cell Name	The name given to the cell when the cell was configured with the Oracle Fusion Middleware Configuration Wizard.
Version	The version of IBM WebSphere that was used to configure the Cell. Note that this version number can also identify which set of patches have been applied to the IBM WebSphere installation.

Table A–1 (Cont.) Fields Available in the General Section of the Summary Region

Element	Description
Administrative Console Port	The non-secure port used to access the IBM WebSphere Administrative Console. Specifically, this is the port identified by *WC_Adminhost_port* in the following URL: `http://hostname:WC_Adminhost_port/ibm/console`
Administrative Console Secure Port	The secure port used to access the IBM WebSphere Administrative Console. Specifically, this is the port identified by *WC_Adminhost_secure port* in the following URL: `https://hostname:WC_Adminhost_secure_port/ibm/console`
SOAP Connector Port	The port used for communications with the administrative server via the Simple Object Access Protocol (SOAP).
Bootstrap Port	This is the value of the bootstrap port for the administrative server. This port is required when you are installing the IBM WebSphere Application Client software and when using utilities such as the IBM WebSphere `dumpNameSpace` tool.
Deployment Mode	The deployment mode of the IBM WebSphere software. For example, this field indicates whether this is an IBM WebSphere Application Server - Network Deployment installation or an IBM WebSphere Application Server deployment.

Deployments Region of the Cell Home Page

This region lists the applications that have been deployed to the servers in the cell. Each application deployment is listed, as well as the deployment name, status, and target servers where the deployment is running.

Click the name of an application deployment to display the WebSphere Application Deployment home page, which provides more information about each application deployment.

The chart identifies the percentage of deployments that are currently up and running, as opposed to those that are down or not available.

Internal applications are those that are required by Oracle Fusion Middleware. The internal applications are deployed automatically and are required by the Oracle Fusion Middleware products you installed and configured in the cell.

Servers Region of the Cell Home Page

This region lists the servers in the cell. The chart identifies the percentage of servers that are up and running, as opposed to those that are down or not available.

For each server, the region lists the server name, status, and--if it resides in a cluster--the name of the cluster.

Clusters Region of the Cell Home Page

This region lists the clusters currently configured in the cell. For each cluster, it provides the cluster name, status, and a list of the servers in the cluster.

A.2 Understanding the Information on the WebSphere Application Server Home Page

The WebSphere Application Server home page is divided into the following regions:

Summary Region of the WebSphere Application Server Home Page

The Summary region of the WebSphere Application Server home page provides general information about the server, as well as a link to the IBM WebSphere Administrative Console, which you can use to manage the server.

Table A–2 describes the fields available in the General section of the Summary region.

Table A–2 Fields Available in the General Section of the Summary Region of the Applicatin Server Page

Element	Description
Cell Name	The name given to the cell when the cell was configured with the Oracle Fusion Middleware Configuration Wizard.
Node Name	The name of the node that contains this server.
Version	The version of IBM WebSphere that was used to configure the Cell. Note that this version number can also identify which set of patches have been applied to the IBM WebSphere installation.
WebSphere Home	The full path of the directory where the current IBM WebSphere software was installed and configured.
Host	The fully-qualified name of the host where the server is currently running.

Deployments Region of the WebSphere Application Server Home Page

This region lists the applications that have been deployed to the server. Each application deployment is listed, including the deployment name and status.

Click the name of an application deployment to display the WebSphere Application Deployment home page, which provides more information about each application deployment.

The chart identifies the percentage of deployments that are currently up and running, as opposed to those that are down or not available.

Internal applications are those that are required by Oracle Fusion Middleware. The internal applications are deployed automatically and are required by the Oracle Fusion Middleware products you installed and configured in the cell.

A.3 Understanding the Information on the IBM WebSphere Application Deployment Home Page

The Application Deployment page is divided into the following sections:

Summary Region on the IBM WebSphere Application Deployment Page

The Summary region of the WebSphere Application Deployment home page provides general information about the application, as well as a link to the IBM WebSphere Administrative Console, which you can use to manage the application.

Table A–3 describes the fields available in the General section of the Summary region.

Table A–3 Fields Available in the General Section of the Summary Region of the Application Deployment Page

Element	Description
Application Type	The type of application. For example, this field indicates whether the application was deployed as an enterprise archive (EAR) or other archive type.
Cell Name	The name given to the cell when the cell was configured with the Oracle Fusion Middleware Configuration Wizard.
Node Name	The name of the node that contains the server where the application was deployed.
Deployed On	The name of the server where this instance of the application is deployed.