Oracle® Identity Manager

Installation and Configuration Guide for JBoss Application Server

Release 9.1.0

E10369-04

December 2009

ORACLE®

Oracle Identity Manager Installation and Configuration Guide for JBoss Application Server, Release 9.1.0

E10369-04

Primary Author: Lyju Vadassery

Contributing Author: Debapriya Datta

Contents

4 Installing and Configuring a Database For Oracle Identity Manager

5 Installing Oracle Identity Manager on Windows

6 Installing Oracle Identity Manager on UNIX

7 Postinstallation Configuration for Oracle Identity Manager and JBoss Application Server

Preface

This guide explains the procedure to install Oracle Identity Manager release 9.1.0 on JBoss Application Server.

Audience

This guide is intended for system administrators of Oracle Identity Manager.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Deaf/Hard of Hearing Access to Oracle Support Services

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at http://www.fcc.gov/cgb/consumerfacts/trs.html, and a list of phone numbers is available at http://www.fcc.gov/cgb/dro/trsphonebk.html.

Related Documents

For more information, see the following documents in the Oracle Identity Manager documentation set:

- *Oracle Identity Manager Release Notes*
- *Oracle Identity Manager Installation and Configuration Guide for BEA WebLogic Server*
- *Oracle Identity Manager Installation and Configuration Guide for IBM WebSphere Application Server*
- *Oracle Identity Manager Installation and Configuration Guide for Oracle Application Server*
- *Oracle Identity Manager Best Practices Guide*
- *Oracle Identity Manager Globalization Guide*
- *Oracle Identity Manager Design Console Guide*
- *Oracle Identity Manager Administrative and User Console Guide*
- *Oracle Identity Manager Administrative and User Console Customization Guide*
- *Oracle Identity Manager Tools Reference*
- *Oracle Identity Manager Audit Report Developer's Guide*
- *Oracle Identity Manager Integration Guide for Crystal Reports*
- *Oracle Identity Manager API Usage Guide*
- *Oracle Identity Manager Concepts*
- *Oracle Identity Manager Reference*

Documentation Updates

Oracle is committed to delivering the best and most recent information available. For information about updates to the Oracle Identity Manager documentation set, visit Oracle Technology Network at:

`http://www.oracle.com/technology/documentation`

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), and names of files, directories, attributes, and parameters.

Convention	Meaning
**_HOME*	This convention represents the directory where an application is installed. The directory where you install Oracle Identity Manager server is referred to as *OIM_HOME*. Each Oracle Identity Manager component includes an abbreviation: *OIM_DC_HOME* for the Design Console and *OIM_RM_HOME* for the Remote Manager.

<Entry 1>.<Entry 2>.<Entry 3>	This convention represents nested XML entries that appear in files as follows:
	`<Entry 1>`
	`<Entry 2>`
	`<Entry 3>`

1

Overview of the Installation Procedure

Installing Oracle Identity Manager release 9.1.0 on JBoss Application Server involves the following steps:

1. Preparing for the installation: See Chapter 2, "Planning the Installation".
2. Setting up JBoss Application Server for Oracle Identity Manager: See Chapter 3, "Installing and Configuring JBoss Application Server for Oracle Identity Manager".
3. Setting up a database for Oracle Identity Manager: See Chapter 4, "Installing and Configuring a Database For Oracle Identity Manager".
4. Installing a single Oracle Identity Manager instance: See one of the following chapters based on the operating system:
   - Chapter 5, "Installing Oracle Identity Manager on Windows"
   - Chapter 6, "Installing Oracle Identity Manager on UNIX"
5. Performing basic Oracle Identity Manager and JBoss Application Server configuration tasks related to the installation setup: See Chapter 7, "Postinstallation Configuration for Oracle Identity Manager and JBoss Application Server".
6. Starting Oracle Identity Manager and accessing the Administrative and User Console: See Chapter 8, "Starting and Stopping Oracle Identity Manager".
7. Deploying Oracle Identity Manager in a clustered JBoss Application Server installation: See Chapter 9, "Deploying in a Clustered JBoss Application Server Configuration".
8. Installing, configuring, and starting the Oracle Identity Manager Design Console: See Chapter 10, "Installing and Configuring the Oracle Identity Manager Design Console".
9. Installing, configuring, and starting the Oracle Identity Manager Remote Manager: See Chapter 11, "Installing and Configuring the Oracle Identity Manager Remote Manager".
10. Troubleshooting the Oracle Identity Manager installation: See Chapter 12, "Troubleshooting the Oracle Identity Manager Installation".

2

Planning the Installation

Oracle recommends that you familiarize yourself with the components required for deployment before installing Oracle Identity Manager. Oracle also recommends that you install and use the included Diagnostic Dashboard to ensure that your system is ready for Oracle Identity Manager installation. See "Using the Diagnostic Dashboard" on page 2-4 for details of installing the Diagnostic Dashboard.

A basic Oracle Identity Manager installation consists of the following:

- A database server
- An application server
- An Oracle Identity Manager installation running on the application server
- A Design Console
- An Administrative and User Console running on a Web browser

This chapter contains the following topics:

2.1 Host Requirements for Oracle Identity Manager Components

This section lists the minimum host system requirements for the various components in an Oracle Identity Manager environment.

> **Note:** Check the Oracle Identity Manager Release Notes for the requirements and supported configurations specific to each version of the Oracle Identity Manager product.

You must obtain the enterprise versions of the application server and database software, complete with valid licenses. Oracle Identity Manager does not include this software.

The Oracle Identity Manager installation program might conflict with other installed applications, utilities, or drivers. Try to remove all nonessential software and drivers from the installation computer before loading Oracle Identity Manager. This practice also ensures that the database host can create the database schema.

2.1.1 Oracle Identity Manager Server (Host) Requirements

Table 2–1 lists the minimum host requirements for the Oracle Identity Manager server and the guidelines for a basic installation.

Table 2–1 Oracle Identity Manager Server Requirements

Server Platform	Item
Microsoft Windows and Linux	▪ Processor Type: Intel Xeon or Pentium IV ▪ Processor Speed: 2.4 GHz or higher, 400 MHz FSB or higher ▪ Number of Processors: 1 ▪ Memory: 2 GB for each Oracle Identity Manager instance ▪ Hard Disk Space: 1 GB (initial size)
Solaris	▪ Server: Sun Fire V210 ▪ Number of Processors: 1 ▪ Memory: 2 GB for each Oracle Identity Manager instance ▪ Hard Disk Space: 1 GB (initial size)

2.1.2 Database Server Host Requirements

Table 2–2 provides sample database minimum host requirements for selective supported operating systems and should be considered only as guidelines. See database documentation for the specific database host requirements.

Table 2–2 Sample Database Server Host Requirement

Database Server Platform	Item
Microsoft Windows and Linux	▪ Processor Type: Intel Xeon ▪ Processor Speed: 2.4 GHz or higher, 400 MHz FSB or higher ▪ Number of Processors: 2 ▪ Memory: 4 GB total or 2 GB for each CPU ▪ Hard Disk Space: 40 GB (initial size) for Windows, 20 GB (initial size) for UNIX
Solaris	▪ Server: Sun Fire V250 ▪ Number of Processors: 2 ▪ Memory: 4 GB total or 2 GB for each CPU ▪ Hard Disk Space: 40 GB (initial size) ▪ Number of Hard Disks: 1 Disk

2.1.3 Design Console Host Requirements

Table 2–3 lists the minimum host requirements for the Oracle Identity Manager Design Console.

Table 2–3 Design Console Host Requirements

Design Console Platform	Item
Microsoft Windows	■ Processor Type: Intel Pentium IV ■ Processor Speed: 1.4 GHz or higher ■ Number of Processors: 1 ■ Memory: 512 MB ■ Hard Disk Space: 300 MB

2.1.4 Remote Manager Host Requirements

Table 2–4 lists the minimum host requirements for the Oracle Identity Manager Remote Manager.

Table 2–4 Remote Manager Host Requirements

Remote Manager Platform	Item
Microsoft Windows and Linux	■ Processor Type: Intel Pentium IV ■ Processor Speed: 1.4 GHz or higher ■ Number of Processors: 1 ■ Memory: 512 MB ■ Hard Disk Space: 1 GB
Solaris	■ Server: Sun Fire V100 Server ■ Number of Processors: 1 ■ Memory: 512 MB ■ Hard Disk Space: 10 GB
AIX	■ Processor Type: PowerPC ■ Number of Processors: 1 ■ Memory: 512 MB ■ Hard Disk Space: 10 GB

2.2 Planning for Non-English Oracle Identity Manager Environments

If you are deploying Oracle Identity Manager components in non-English environments, then review the following guidelines and requirements:

- Before installing any of the Oracle Identity Manager components, ensure that the regional and language settings (locale) on the target system meet the following requirements:
  - An appropriate language version of the operating system is installed.
  - Specific language settings are properly configured.
- See *Oracle Identity Manager Globalization Guide* information about configuring localized deployments and to ensure that you meet the character restrictions for various components and attributes.
- For Oracle database globalization support, you must configure the database for Unicode. See "Creating an Oracle Database" on page 4-1 for more information.

2.3 Installation Worksheet

Table 2–5 provides information about the configuration attributes that you must set during Oracle Identity Manager installation. Print this worksheet and use it to take notes during the installation. Enter information specific to your installation in the User Selection column.

Table 2–5 Installation Worksheet

Item	Default	User Selection
The base directory for installing Oracle Identity Manager	Windows: C:\oracle UNIX: opt/oracle	
The name or IP address of the computer where the Oracle Identity Manager database is installed	No default value	
The TCP port number on which the database listens for connections	1433 for Microsoft SQL Server **Note:** Microsoft SQL Server is not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components. 1521 for Oracle	
The name of the database for your installation	No default value	
The name and password of the database account that Oracle Identity Manager uses to access the database	No default value	
The JDK installation directory	Windows: `C:\j2sdk`*`version`* UNIX: `/opt/j2sdk`*`version`*	
The JBoss Application Server installation directory	Windows: `C:\jboss-`*`version`* UNIX: `/opt/jboss-`*`version`*	

2.4 Using the Diagnostic Dashboard

The Diagnostic Dashboard is a Web application that runs on the application server. It checks your pre and postinstallation environments for components required by Oracle Identity Manager. Oracle recommends that you install the Diagnostic Dashboard before installing Oracle Identity Manager.

2.4.1 Installing the Diagnostic Dashboard

The Diagnostic Dashboard files are located in the `DiagnosticDashboard directory` on the Oracle Identity Manager Installer CD media.

You must deploy the Diagnostic Dashboard Web application on your application server. For more information, see *Oracle Identity Manager Administrative and User Console Guide*.

Note: If you install the Diagnostic Dashboard XIMDD application on JBoss Application Server before you install Oracle Identity Manager, you might encounter an exception when you start JBoss Application Server. To avoid this exception, set the `UseJBossWebLoader` property in the `jboss-4.0.3SP1\server\default\deploy\jbossweb-tomcat55.sar\META-INF\jboss-service.xml` file to `true`.

2.4.2 Verifying Your Preinstallation Environment

You can use the Diagnostic Dashboard to verify that the components required to install Oracle Identity Manager are present:

- A supported Java Virtual Machine (JVM)
- A supported database
- Microsoft SQL Server JDBC libraries (only if you use Microsoft SQL Server)

Note: Microsoft SQL Server is not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

See Also: *Oracle Identity Manager Administrative and User Console Guide* for information about the Diagnostic Dashboard

3

Installing and Configuring JBoss Application Server for Oracle Identity Manager

This chapter explains how to set up JBoss Application Server before installing Oracle Identity Manager.

You must perform the following tasks described in this chapter:

- Installing the Java JDK
- Installing JBoss Application Server
- Setting Memory Parameters

Note:

- See Chapter 9, "Deploying in a Clustered JBoss Application Server Configuration" for information about preparing to deploy Oracle Identity Manager in a JBoss Application Server cluster.

- JBoss Application Server clustered environments are not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components

---

3.1 Installing the Java JDK

You must have a certified version of the Java JDK installed to deploy Oracle Identity Manager on JBoss Application Server. See *Oracle Identity Manager Release Notes* to learn the certified version of the Java JDK and then use the following steps to verify that the correct version of the Java JDK has been installed on the system:

1. Open a console window.
2. Enter `java -version`

   The information that appears might look like the following:

```
C:\>java -version
java version "1.4.2_15"
Java(TM) 2 Runtime Environment, Standard Edition (build 1.4.2_15-b02)
Java HotSpot(TM) Client VM (build 1.4.2_15-b02, mixed mode)
```

3.2 Installing JBoss Application Server

Install JBoss Application Server on the computer on which you are going to install Oracle Identity Manager. See your JBoss Application Server documentation for detailed installation procedures.

Note: You can obtain JBoss Application Server from:
http://www.jboss.org

3.2.1 Setting Environment Variables

Perform the following procedures to set your environment variables:

For Microsoft Windows

To set environment variables on Microsoft windows:

1. From the Windows **Start Menu**, select **Settings**, select **Control Panel**, select **System**, select **Advanced**, then select **Environment Variables**.
2. In the **System Variables** list, select **Path**, then click **Edit**.

   In the Variable Value field, add the location of your JDK to the beginning of the existing path. For example, if your existing path is the following:

   ```
   %SystemRoot%\system32;%SystemRoot%;C:\Program Files;
   ```

   Change it to the following:

   ```
   c:\j2sdk1.4.2_15\bin;%SystemRoot%\system32;%SystemRoot%;C:\Program Files
   ```

   Click **OK** to commit your change.
3. In the **System Variables** list, search for JAVA_HOME.

   If JAVA_HOME does not exist, complete Step a. If JAVA_HOME exists, complete Step b.

   a. Click **New**. In the **Variable Name** field, enter `JAVA_HOME`. In the **Variable Value** field, enter the path to the JDK. Click **OK** to commit your entries, then click **OK** twice more to close the Environment Variables and System Properties windows, respectively.

   b. Click **Edit**. Verify that the path to the JDK exists in the **Variable Value** field. If it does not exist, enter the path in the **Variable Value** field. Click **OK** to commit your entry, then click **OK** twice more to close the Environment Variables and System Properties windows, respectively.

      For example: JAVA HOME=C:\j2sdk1.4.2_15

Note: A window might appear displaying a message asking if you want to update the JDK. Close this window without updating the JDK.

For UNIX

To set environment variables on UNIX:

1. Set the JAVA HOME variable, for example:

   ```
   export JAVA_HOME=/opt/j2sdk1.4.2_15
   ```

2. Export the path to the JAVA HOME variable, for example:

```
export PATH=$JAVA_HOME/bin:$PATH
```

3.3 Setting Memory Parameters

After installing JBoss Application Server, you must change the JVM memory settings for production environments or when you are processing large volumes in the nonproduction mode of the Oracle Identity Manager installation. The instructions for setting the memory parameters, which appear in the following sections, depend on whether the application server host is running on Microsoft Windows, or UNIX.

3.3.1 Setting Memory Allocation for Microsoft Windows

To set JBoss Application Server memory on a Microsoft Windows host:

1. Open the *JBOSS_HOME*\bin\run.bat file in a text editor.
2. Locate the line that contains the following:

```
set JAVA_OPTS=%JAVA_OPTS% -Xms128m -Xmx512m
```

3. Change the memory settings to the following values, which are recommended for production deployments:

```
set JAVA_OPTS=%JAVA_OPTS% -Xms1280m -Xmx1280m -XX:PermSize=128m
-XX:MaxPermSize=256m
```

4. Save and close the run.bat file.

3.3.2 Setting Memory Allocation for UNIX

To set the memory allocation for JBoss Application Server on UNIX:

1. Open the *JBOSS_HOME*/bin/run.conf file in a text editor.
2. Locate the following line:

```
JAVA_OPTS="-server -Xms128m -Xmx 128m"
```

3. Change the memory settings to the following values, which are recommended for production deployments:

```
JAVA_OPTS="-server -Xms1280m -Xmx1280m -XX:PermSize=128m -XX:MaxPermSize=256m"
```

4. Save and close the run.conf file.

3.4 Configuring JBoss Application Server

This section describes how to configure JBoss Application Server. It contains these topics:

- Removing JBoss Application Server Components
- Removing Files and Directories

3.4.1 Removing JBoss Application Server Components

For a JBoss Application Server installation, some JBoss components are not required by Oracle Identity Manager. These files vary for standalone and clustered installations.

You can remove the following components:

- Cache invalidation service (keep for a clustered installation)
- J2EE client deployer service
- Integrated HAR deployer and Hibernate session Management services
- JMX Console
- Management console
- Console/e-mail monitor alerts
- UUID key Generation
- JBoss scheduler manager
- Scheduler service
- Test queues and topics
- Mail service
- HTTP Invoker
- CORBA/IIOP
- AOP Application
- Web services support

3.4.2 Removing Files and Directories

The following sections list the files and directories that you can remove after installing Oracle Identity Manager.

3.4.2.1 Non-Clustered Installations

Remove the following files from *JBOSS_HOME*/server/default/deploy/:

- cache-invalidation-service.xml
- client-deployer-service.xml
- monitoring-service.xml
- scheduler-service.xml
- schedule-manager-service.xml
- mail-service.xml
- uuid-key-generator.sar

Remove the following directories from *JBOSS_HOME*/server/default/deploy/:

- jboss-hibernate.deployer

> **Note:** Remove this directory only if it exists.

- jmx-console.war

- management
- http-invoker.sar
- jboss-aop.deployer
- jboss-ws4ee.sar

Remove the following file:

JBOSS_HOME/server/default/deploy/jms/jbossmq-destinations-service.xml.

Open the *JBOSS_HOME*/server/default/conf/jboss-service.xml file and remove the following attribute:

```
<attribute name'"RMI_IIOPService">jboss:service=CorbaORB</attribute>
```

3.4.2.2 Clustered Installations

Remove the following files from *JBOSS_HOME*/server/all/deploy/:

- client-deployer-service.xml
- monitoring-service.xml
- scheduler-service.xml
- schedule-manager-service.xml
- httpha-invoker.sar
- mail-service.xml
- jboss-aop.deployer
- jboss-ws4ee.sar

Remove the following directories from the *JBOSS_HOME*/server/all/deploy/:

- jboss-hibernate.deployer
- jmx-console.war
- management
- uuid-key-generator.sar

Remove the following file:

JBOSS_HOME/server/all/deploy-hasingleton/jms/jbossmq-destinations-service.xml

Open the *JBOSS_HOME*/server/all/conf/jboss-service.xml file and remove the following attribute:

```
<attribute name'"RMI_IIOPService">jboss:service=CorbaORB</attribute>
```

4

Installing and Configuring a Database For Oracle Identity Manager

Oracle Identity Manager requires a database. You must install and configure your database before you begin the Oracle Identity Manager installation. Refer to the topic that applies to your database:

- Using an Oracle Database for Oracle Identity Manager
- Using Oracle RAC Databases for Oracle Identity Manager
- Using a Microsoft SQL Server Database for Oracle Identity Manager

4.1 Using an Oracle Database for Oracle Identity Manager

To use Oracle Database as your database, you must perform the tasks described in the following sections:

- Installing Oracle Database
- Creating an Oracle Database
- Preparing the Oracle Database

4.1.1 Installing Oracle Database

Install Oracle9*i* Database or Oracle Database 10*g* release 2 by referring to the documentation delivered with Oracle Database. See *Oracle Identity Manager Release Notes* for the specific supported versions. Oracle recommends using the Basic installation.

Note: If you choose the Custom installation, you must include the JVM option, which is required for XA transaction support.

4.1.2 Creating an Oracle Database

You must create a new Oracle database instance for Oracle Identity Manager. When creating the database, ensure that you configure the Oracle JVM feature and enable query rewrite.

You can use the Database Configuration Assistant (DBCA) tool to create the database. To configure the Oracle JVM feature, select the Oracle JVM feature on the Standard Database Features page of the DBCA.

To enable the database for query rewrite, set the initialization parameters `QUERY_REWRITE_ENABLED` to `TRUE` and `QUERY_REWRITE_INTEGRITY` to `TRUSTED` in the **All Initialization Parameters** field of the DBCA.

> **Note:** For the Oracle Identity Manager installation, Oracle recommends that you configure a minimum block size of 8K for Oracle Database.

See Oracle Database documentation for detailed instructions on creating a database instance.

4.1.2.1 Configuring the Database for Globalization Support

For globalization support for Oracle Identity Manager, Oracle recommends configuring the database for Unicode. To configure the database for Unicode, perform the following steps:

1. Select **AL32UTF8** in the Character Sets tab of the DBCA. This character set supports the Unicode standard.
2. Set the `NLS_LENGTH_SEMANTICS` initialization parameter to `CHAR` in the **All Initialization Parameters** field of the DBCA.

> **See Also:** *Oracle Identity Manager Globalization Guide* for information about globalization support for Oracle Identity Manager

4.1.3 Preparing the Oracle Database

After you install Oracle Database and created a database instance, you must prepare it for Oracle Identity Manager by completing the following tasks:

- Verify that query rewrite is enabled

> **Note:** Query rewrite is applicable only if you are using Oracle Database Enterprise Edition.

- Enable XA transactions support

> **Note:** Java Virtual Machine (JVM) is required to enable XA transaction support. If you did not install the Oracle JVM component during Oracle Database installation, then you must install it now. See the Oracle Database documentation for specific instructions.

- Create at least one tablespace for storing Oracle Identity Manager data
- Create a database user account for Oracle Identity Manager

You can perform the preceding tasks to prepare your Oracle database for Oracle Identity Manager by running one of the following scripts:

- On UNIX, run the following:

  ```
  prepare_xl_db.sh
  ```

- On Microsoft Windows, run the following:

  ```
  prepare_xl_db.bat
  ```

Both of these scripts ship with the Oracle Identity Manager Installer and are located in the `/installServer/Xellerate/db/oracle/` directory.

You must observe the following prerequisites when using the prepare_xl_db script:

- The script must be run by a user holding DBA privileges (for example, the oracle user on UNIX typically holds these privileges).
- The script must be run on the computer on which the database resides.

To prepare your Oracle database for Oracle Identity Manager, complete the steps associated with the operating system on the computer hosting your Oracle database.

4.1.3.1 Preparing the Database on UNIX

To prepare the database on UNIX:

1. Copy the scripts `prepare_xl_db.sh` and `xell_db_prepare.sql` from the distribution CD to a directory on the computer hosting your database on which you (as the account user performing this task) have write permission.
2. Run the following command to enable permission to run the script:

   ```
   chmod 755 prepare_xl_db.sh
   ```

3. Run the `prepare_xl_db.sh` script by entering the following command:

   ```
   ./prepare_xl_db.sh
   ```

4. Provide information appropriate for your database and host computer when the script prompts you for the following items:
   - The location of your Oracle home (*ORACLE_HOME*)
   - The name of your database (*ORACLE_SID*)
   - The name of the Oracle Identity Manager database user to be created
   - The password for the Oracle Identity Manager database user
   - The name of the tablespace to be created for storing Oracle Identity Manager data
   - The directory in which to store the data file for the Oracle Identity Manager tablespace
   - The name of the data file (you do not need to append the .dbf extension)
   - The name of the temporary tablespace
5. Check the `prepare_xl_db.lst` log file located in the directory from which you ran `prepare_xl_db.sh` to see the execution status and additional information.

Note: If you encounter errors after running the `prepare_xl_db.sh` script, run the following command to ensure that `prepare_xl_db.sh` is executable on UNIX and Linux and then run the `prepare_xl_db.sh` script again.

```
$ dos2unix prepare_xl_db.sh
```

4.1.3.2 Preparing the Database on Microsoft Windows

To prepare the database on Microsoft Windows:

1. Copy the scripts `prepare_xl_db.bat` and `xell_db_prepare.sql` from the distribution CD to a directory on the computer hosting your database on which you (as the account user performing this task) have write permission.
2. Open a command window, navigate to the directory to which you just copied the scripts, then run `prepare_xl_db.bat` with the following arguments:

   ```
   prepare_xl_db.bat ORACLE_SID ORACLE_HOME
   XELL_USER XELL_USER_PWD TABLESPACE_NAME
   DATAFILE_DIRECTORY DATAFILE_NAME
   XELL_USER_TEMP_TABLESPACE SYS_USER_PASSWORD
   ```

   For example, the string you enter on the command line might look like the following:

   ```
   prepare_xl_db.bat XELL C:\oracle\ora92 xladm xladm
   xeltbs C:\oracle\oradata xeltbs_01 TEMP manager
   ```

   Table 4–1 lists the options used in the preceding example of `prepare_xl_db.bat`.

Table 4–1 Options for the prepare_xl_db.bat Script

Argument	Description
`XELL`	Name of the database
`C:\oracle\ora92`	Directory where the Oracle database is installed
`xladm`	Name of the Oracle Identity Manager user to be created
`xladm`	Password for the Oracle Identity Manager user
`xeltbs`	Name of the tablespace to be created
`C:\oracle\oradata`	Directory where the data files will be placed
`xeltbs_01`	Name of the data file (you do not need to include the .dbf extension)
`TEMP`	Name of the temporary tablespace that already exists in your database
`manager`	Password for the SYS user

3. Check the `prepare_xell_db.lst` log file located in the directory from which you ran `prepare_xl_db.bat` to see the execution status and additional information.

4.1.3.3 Evaluating Script Results

If the script returns a message indicating successful execution, you can continue to the next task, which is installing Oracle Identity Manager.

If the script does not succeed, you must manually fix all fatal (nonrecoverable) errors so that the database is prepared successfully.

You can ignore all non-fatal errors. For example, when the script tries to drop a non-existent view, it will return the error "ORA-00942: table or view does not exist".

Ensure to scan all the errors in the log file and ignore or resolve them on an individual basis. Remember that you must successfully prepare the database for Oracle Identity Manager before you can install Oracle Identity Manager.

4.1.4 Removing Oracle Identity Manager Entries from an Oracle Database

To remove Oracle Identity Manager entries from an Oracle database after removing (deinstalling) the Oracle Identity Manager product, drop the database user holding the Oracle Identity Manager schema.

4.2 Using Oracle RAC Databases for Oracle Identity Manager

This section explains how to deploy Oracle Real Application Clusters (Oracle RAC) databases for Oracle Identity Manager and contains the following sections:

- Installing Oracle Identity Manager for Oracle RAC
- Oracle RAC Net Services
- JDBC and Oracle RAC
- Configuring JBoss Application Server for Oracle RAC

4.2.1 Installing Oracle Identity Manager for Oracle RAC

Oracle RAC is a cluster database with a shared cache architecture that provides highly scalable and available database solutions. Oracle RAC consists of multiple database instances on different computers acting in tandem to provide these features.

Note: The Oracle Identity Manager Installer program does not provide support for Oracle RAC. To deploy Oracle Identity Manager for Oracle RAC, you must install Oracle Identity Manager on a single database instance in Oracle RAC and then change the application server settings, specifically the connection pool parameters, to use the Oracle RAC JDBC connection string.

Use the following steps to install Oracle Identity Manager for Oracle RAC:

1. Ensure Oracle RAC is properly set up and configured with the Oracle Identity Manager schema owner.
2. Start the Oracle Identity Manager Installer.
3. On the Database Parameters page of the installer, enter the host name, port number, and database name of a single database instance in Oracle RAC.
4. Complete the Oracle Identity Manager installation by performing the steps in the installer.
5. Configure your application server for Oracle RAC by referring to Configuring JBoss Application Server for Oracle RAC.

4.2.2 Oracle RAC Net Services

The net services name entry for an Oracle RAC database differs from that of a conventional database. The following is an example of the net services name entry for an Oracle RAC database:

```
racdb=
      (DESCRIPTION=
              (LOAD_BALANCE=off)
              (FAILOVER=on)
              (ADDRESS_LIST=
                      (ADDRESS=(protocol=tcp)(host=node1-vip)(port=1521))
```

```
                (ADDRESS=(protocol=tcp)(host=node2-vip)(port=1521)))
(CONNECT_DATA=
        (SERVER=DEDICATED)
        (SERVICE_NAME=racdb)))
```

Table 4–2 lists and describes the parameters in a net services name entry for an Oracle RAC database.

Table 4–2 Parameters for Oracle RAC Database Net Services Name Entries

Parameter	Description
LOAD_BALANCE	Specifies whether client load balancing is enabled (on) or disabled (off). The default setting is on.
FAILOVER	Specifies whether failover is enabled (on) or disabled (off). The default setting is on.
ADDRESS_LIST	Specifies the list of all the nodes in Oracle RAC, including their host names and the ports they listen on.

4.2.3 JDBC and Oracle RAC

JDBC client applications using the Thin driver to connect to an Oracle RAC database must use the Oracle RAC net services name as a part of the JDBC URL. The entire Oracle RAC net services name is concatenated and the entire string is used in the JDBC URL so the client application can connect to Oracle RAC.

The following sample code shows how a JDBC URL is used to connect to an Oracle RAC database:

```
//String url = "jdbc:oracle:thin:@dbhost:1521:dbservice"
String racUrl =
"jdbc:oracle:thin:@(DESCRIPTION=(LOAD_BALANCE=off)(FAILOVER=on)(ADDRESS_LIST=(ADDR
ESS=(protocol=tcp)(host=node1-vip)(port=1521))(ADDRESS=(protocol=tcp)(host=node2-v
ip)(port=1521)))(CONNECT_DATA=(SERVER=DEDICATED)(SERVICE_NAME=racdb)))";

       String strUser = "username";
       String strPW = "password";

       // load Oracle driver
       Class.forName("oracle.jdbc.driver.OracleDriver");

       // create the connection
       con = DriverManager.getConnection(strURL, strUser, strPW);
```

The subsequent sections about configuring application servers for Oracle RAC databases explain how to modify connection pools to use a similar JDBC URL so the application server can communicate with Oracle RAC.

4.2.4 Configuring JBoss Application Server for Oracle RAC

Note: JBoss Application Server clustered environments are not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

This section explains how to configure JBoss Application Server (nonclustered or clustered) for Oracle RAC by ensuring the data sources and connection pools are configured to use the Oracle RAC JDBC connection string.

> **See:** For information about avoiding split branches of a distributed transaction between Oracle Identity Manager and RAC, see *Best Practices for Using XA with RAC* document. You can find this document by searching the Oracle Technology Network at
>
> `http://www.oracle.com/technology/index.html`

Perform the following steps to configure both non-clustered and clustered JBoss Application Server for Oracle RAC:

1. Get the RAC net services name from the tnsnames.ora file.
2. Construct the RAC JDBC URL by referring to JDBC and Oracle RAC.
3. Open the *OIM_HOME*`/xellerate/config/xlconfig.xml` file.
4. Locate the <DirectDB> section and replace the value of the <url>...</url> tag with the RAC JDBC URL. For example, the new tag can be similar to the following:

   ```
   <url>jdbc:oracle:thin:@(DESCRIPTION=(LOAD_BALANCE=off)(FAILOVER=on)(ADDRESS_
   LIST=(ADDRESS=(protocol=tcp)(host=node1-vip)(port=1521))(ADDRESS=(protocol=tcp)
   (host=node2-vip)(port=1521)))(CONNECT_DATA=(SERVER=DEDICATED)(SERVICE_
   NAME=racdb)))</url>
   ```

5. Save and close the *OIM_HOME*`/xellerate/config/xlconfig.xml` file.
6. If you are configuring a nonclustered JBoss Application Server environment, open the *JBOSS_HOME*`/server/default/deploy/xell-ds.xml` file.

   If you are configuring a clustered JBoss Application Server environment, open the *JBOSS_HOME*`/server/all/farm/xell-ds.xml` file for each node in the cluster.

   These files contain entries with the XA and non-XA datasources.
7. Locate the <datasources>.<local-tx-datasource>.<connection-url> entry.
8. Change the value of this entry to the JDBC URL described in step 4.
9. Locate the <datasources>.<xa-datasource>.<xa-datasource-property name="URL"> entry.
10. Change the value of this entry to the JDBC URL described in step 4.
11. Save and close the file.
12. Restart JBoss Application Server. For JBoss clusters, restart all nodes in the cluster.

4.3 Using a Microsoft SQL Server Database for Oracle Identity Manager

> **Note:** Microsoft SQL Server is not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

To use Microsoft SQL Server for your database, you must complete the procedures in the following sections:

- Installing and Configuring Microsoft SQL Server
- Configuring JBoss Application Server for Microsoft SQL Server
- Registering Microsoft SQL Server
- Creating a Microsoft SQL Server Database
- Creating a Microsoft SQL Server Database Account

After you have completed these tasks, you are ready to install the Oracle Identity Manager components.

4.3.1 Installing and Configuring Microsoft SQL Server

To install and configure SQL Server for Oracle Identity Manager:

1. Install Microsoft SQL Server 2000 with Service Pack 3a.

   During installation, choose **mixed authentication mode**, then set the password to **sa**.

   Note: Perform Steps 2 through 4 on the computer hosting the application server.

2. Download the SQL Server 2000 Driver for JDBC Service Pack 3 from `http://www.microsoft.com`.
3. Install SQL Server 2000 Driver for JDBC Service Pack 3.

   Note: Specify a short path for the installation folder, such as C:\JDBCjars, so that you can easily add the path to your CLASSPATH in the next step. If the classpath is more than 256 characters, the installer does not work properly.

4. Locate the JDBC driver files (mssqlserver.jar, msbase.jar, and msutil.jar).

   Add their location to the system CLASSPATH environment variable. If the CLASSPATH environment variable does not exist, you must create it. The string you add should look like the following:

   ```
   C:\jdbc_install_folder\lib\mssqlserver.jar;
   C:\jdbc_install_folder\lib\msbase.jar;
   C:\jdbc_install_folder\lib\msutil.jar;
   ```

   In these sample strings, *jdbc_install_folder* is the location where the SQL Server 2000 Driver for JDBC files is installed.

5. Enable distributed transactions by installing SQL Server JDBC XA procedures.

   Copy the sqljdbc.dll file in the *SQLServer JDBC Driver*\SQLServer JTA\ directory to the following directory:

   ```
   C:\Program Files\Microsoft SQl Server\MSSQL\Binn
   ```

6. Run the script instjdbc.sql.

   Follow the instructions for installing stored procedures for Java Transaction APIs (JTA). These instructions are bundled with the SQL Server 2000 Driver for JDBC (see the jdbcsqlsrv9.html Help file).

7. Ensure that the Distributed Transaction Coordinator (MSDTC) service for your SQL Server is running.

   If necessary, use the SQL Server Service Manager to start it.

   Note: You can set the Distributed Transaction Coordinator to start automatically whenever the operating system is restarted.

4.3.2 Configuring JBoss Application Server for Microsoft SQL Server

After installing JBoss Application Server, set up JBoss to work with SQL Server by copying (not moving) the following JDBC driver files to the lib directory of your default JBoss server:

- mssqlserver.jar
- msbase.jar
- msutil.jar

Copy the files from the SQL Server 2000 Driver for JDBC library directory (the default is C:\Program Files\Microsoft SQL Server 2000 Driver for JDBC\lib) to *JBOSS_HOME*\server\default\lib.

Note:

- For a JBoss Application Server cluster, copy (do not move) the files from the SQL Server 2000 Driver for JDBC library directory to *JBOSS_HOME*\server\all\lib.

- JBoss Application Server clustered environments are not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

4.3.3 Registering Microsoft SQL Server

To register Microsoft SQL Server:

1. Start the SQL Server Enterprise Manager application.

   From the Windows **Start Menu**, select **Programs**, select **Microsoft SQL Server**, then select **Enterprise Manager**.

2. In the left pane of the SQL Server Enterprise Manager application window, select **Console Root**, then select **Microsoft SQL Servers**.

3. Right-click **SQL Server Group** and select **New SQL Server Registration**.

4. In the Register SQL Server Wizard dialog, click **Next**.

5. On the Select a SQL Server page, perform one of the three following sub-steps:

   - Select your server from the list in the right pane, click **Add**, then click **Next**.
   - Select **LOCAL**, then click **Add**, then click **Next**.
   - Enter the host name of your server in the text entry box, click Add, then click Next.

6. On the Select an Authentication Mode page, select **The SQL Server login information that was assigned to me by the administrator [SQL Server Authentication]**, then click **Next**.
7. On the Register Connection Option page, select **Login automatically using my SQL server account information**, then complete the following sub-steps:
   a. In the **Login name** field, enter the account name used to connect to your SQL server. Typically, this is **sa**.
   b. In the **Password** field, enter the password associated with the account name you specified, then click **Next**.
8. On the Select SQL Server Group page, select **Add the SQL Server(s) to an existing SQL Server Group**, select a group from the **Group name** list, then click **Next**.
9. On the Completing the Register SQL Server Wizard page, click **Finish**, then click **Done**.

4.3.4 Creating a Microsoft SQL Server Database

The following procedure describes how to create a new database for Oracle Identity Manager.

> **Note:** In the following procedure uses the name XELL for the database. You are not required to use XELL as the name for the database. This document refers to the name of the database as XELL throughout.

To create a SQL Server database, complete these steps:

1. Start the Microsoft SQL Server Enterprise Manager application. To do so, from the Windows **Start Menu**, select **Programs**, select **Microsoft SQL Server**, then select **Enterprise Manager**.
2. In the left pane of the SQL Server Enterprise Manager application window, select **Console Root**, select **Microsoft SQL Servers**, select the server group to which your server belongs, then double-click the icon representing your server.
3. Right-click **Databases**, then select **New Database**.
4. In the Database Properties dialog, select the **General** tab, then enter `XELL` in the **Name** field.
5. Select the **Data Files** tab. Then, for the **Initial Size** and **Filegroup** columns in the Database files matrix, enter the information from the corresponding columns in Table 4–3.

Table 4–3 Database Files

File Name	Initial Size in Megabytes (MB)	Filegroup Name	Content
XELL_PRIMARY	100	PRIMARY	System objects required for SQL Server operation
XELL_DATA	500	XELL_DATA	Physical data and primary keys
XELL_INDEX	300	XELL_INDEX	Indexes

Table 4–3 (Cont.) Database Files

File Name	Initial Size in Megabytes (MB)	Filegroup Name	Content
XELL_TEXT	500	XELL_TEXT	Large text fields
XELL_UPA	1000	XELL_UPA	Keys for the User Profile Audit component

Note: Table 4–3 lists initial sizes for a production environment. For non-production installations, you can use the default initial sizes provided for the filegroups.

To ensure successful installation of Oracle Identity Manager, filegroup names must be entered exactly as they appear in Table 4–3. You can vary the File Name and Location strings to match the database name and the location of your Microsoft SQL Server installation.

a. Select **Automatically Grow File**.

b. Select **By Percent**, then enter `10` in the associated field.

c. Select **Unrestricted file growth**.

Note: The PRIMARY filegroup contains the system objects required for SQL Server to operate. The XELL_DATA filegroup stores the physical data and primary keys, XELL_INDEX filegroup stores indexes, XELL_TEXT stores large text fields, and XELL_UPA stores the physical data and primary keys of the User Profile Audit component.

6. Select the **Transaction Log** tab, then change the initial size to 500 MB. Leave all the other options on the tab at their default values.

Note: For non-production installations, you can use the default initial size for the log file.

7. Click **OK** to start creating the database.

4.3.5 Creating a Microsoft SQL Server Database Account

The following procedure describes how to create a database account for Oracle Identity Manager and assign appropriate permissions to that account.

Note: The following procedure assumes the account name xladm. If you want an account name other than xladm, then specify that login instead of xladm throughout the following procedure and also when installing Oracle Identity Manager.

To create a Microsoft SQL Server database account and permissions:

1. Start the Microsoft SQL Server Enterprise Manager application.

From the Windows **Start Menu**, select **Programs**, select **Microsoft SQL Server**, then select **Enterprise Manager**.

2. In the left pane of the SQL Server Enterprise Manager application window, select **Console Root**, select **Microsoft SQL Servers**, select the server group to which your server belongs, then double-click the icon representing your server.
3. Select **Security**, right-click **Logins**, then select **New Login**.
4. In the SQL Server Login Properties dialog, select the **General** tab.

   In the **Name** field, enter xladm (or a different account name that you prefer).
5. Select **SQL Server Authentication**. In the **Password** field, enter the password associated with the account name that you specified in Step 4.
6. In the **Database** box within the **Defaults** section, select **XELL** from the list.

   Leave the **Language** box set to <default>.
7. Select the **Database Access** tab. In the upper panel, select the check box associated with **XELL**.
8. In the lower panel, select the check boxes associated with the following:
   - public
   - db_owner
   - db_accessadmin
   - db_securityadmin
   - db_ddladmin
   - db_datareader
   - db_datawriter
9. Click **OK** to commit your changes.

   When prompted, confirm the password and click **OK**.
10. To check your database settings, right-click the icon representing your server, then select **Properties** from the shortcut menu.
11. On the SQL Server Properties page, select the **Security** tab, then verify that Authentication is set to **SQL Server and Windows**.
12. Click the **General** tab, then verify that the check boxes associated with **Autostart SQL Server** and **Autostart MSDTC** are selected.

    If **Autostart SQL Server Agent** is selected, do not change the existing setting, because that setting might be required by other applications. Click **OK** to close the **SQL Server Properties** page.

4.3.6 Removing Oracle Identity Manager Entries from a Microsoft SQL Server Database

To remove Oracle Identity Manager entries from a SQL Server database after removing (deinstalling) the Oracle Identity Manager product, perform the following steps:

1. Delete the Oracle Identity Manager database.
2. Delete the Oracle Identity Manager login.

5

Installing Oracle Identity Manager on Windows

This chapter explains how to install Oracle Identity Manager on Microsoft Windows in a nonclustered installation.

> **See Also:** Chapter 9, "Deploying in a Clustered JBoss Application Server Configuration" for information about deploying Oracle Identity Manager in a clustered installation

You must install Oracle Identity Manager on systems running the application server. Oracle Identity Manager components such as the Remote Manager and Design Console can be installed on separate systems. Each component has its own installer.

This chapter contains the following topics:

- Installing the Database Schema
- Installing Documentation
- Installing Oracle Identity Manager on Microsoft Windows
- Removing Oracle Identity Manager

> **Caution:** Do *not* use a remote client tool, such as Symantec pcAnywhere, to install Oracle Identity Manager products.

5.1 Installing the Database Schema

As part of the installation, the Oracle Identity Manager Installer loads a schema into your database. You only install the database schema once. It is installed the first time you run the Oracle Identity Manager Installer. Each subsequent time you run the installer to deploy other Oracle Identity Manager components you enter information about the database connection to configure the component for the same schema. If required, contact your database administrator (DBA).

> **Note:** During the schema installation, a log file is created in the *OIM_HOME*/logs directory.

5.2 Installing Documentation

The Oracle Identity Manager documentation is installed automatically in the *OIM_HOME* directory. No special input is required. A full documentation set is installed with each Oracle Identity Manager component.

5.3 Installing Oracle Identity Manager on Microsoft Windows

This section describes how to install Oracle Identity Manager on a computer running Microsoft Windows.

Caution: Do not install Oracle Identity Manager on top of an existing Oracle Identity Manager installation. For each new installation, use a different home directory. If you want to reuse the name of an existing Oracle Identity Manager home directory, then back up your original Oracle Identity Manager home by renaming that directory.

Remember at all times that all Oracle Identity Manager components must be installed in different home directories. For example, you cannot install the Remote Manager in the same directory as Oracle Identity Manager.

To install Oracle Identity Manager on a Windows host:

Note: Microsoft SQL Server is not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

---

1. If you are using Microsoft SQL Server as your database, before installing Oracle Identity Manager be sure to copy the following three files located in `C:\Program Files\Microsoft SQL Server 2000 Driver for JDBC\lib\` to the *`JBOSS_HOME`*`\server\default\lib\` directory and add the driver location to the system CLASSPATH environment variable:
   - mssqlserver.jar
   - msbase.jar
   - msutil.jar
2. Insert the Oracle Identity Manager Installation CD into your CD-ROM drive.
3. Using Windows Explorer, access the installServer directory on the installation CD and double-click the setup_server.exe file.
4. Select a language on the Installer page and click **OK**. The Welcome page is displayed.
5. Click **Next** on the Welcome page. The Admin User Information page is displayed.
6. Enter the password that you want to use as the Oracle Identity Manager administrator, confirm the password by entering it again, and then click **Next**. The OIM Application Options page is displayed.
7. Select one of the following applications to install, and then click **Next**:
   - Oracle Identity Manager
   - Oracle Identity Manager with Audit and Compliance Module

See Also: *Oracle Identity Manager Audit Report Developer's Guide* for information about the Audit and Compliance Module

The Target directory page is displayed.

8. Complete one of the following:
   - Install Oracle Identity Manager into the default directory, which is C:\oracle\, click **Next**.
   - Install Oracle Identity Manager into another directory, enter the path in the **Directory** field, then click **Next**.

     or

     Click **Browse**, navigate to the desired location, then click **Next**.

     Note: If the directory path does not exist, then the Base Directory settings field is displayed. Click **OK**. This directory is automatically created. If you do not have write permission to create the default directory for Oracle Identity Manager, then a message is displayed informing you that the installer could not create the directory. Click **OK** to close the message, and then contact your system administrator to obtain the appropriate permissions.

   The Database Server Selection page is displayed.

9. Specify either **Oracle** or **SQL Server** as the type of database that you are using with Oracle Identity Manager and click **Next**. The Database Information page is displayed.

10. Enter all database connectivity information required to install the database schema.

    You install this schema just once, as part of your initial Oracle Identity Manager installation. Thereafter, you configure all the other Oracle Identity Manager components to point to this common schema.

    Note: To install against an existing database, verify that the version of Oracle Identity Manager you are installing is certified with your existing database version. See *Oracle Identity Manager Release Notes* for information about the certified configurations.

    When Oracle Identity Manager is installed against an existing database, a warning message is displayed indicating the database schema already exists and instructing you to copy the .xldatabasekey file from the existing Oracle Identity Manager installation to the new *OIM_HOME*\xellerate\config\ directory after you complete the installation process.

    You should create the \config directory in the new *OIM_HOME*\xellerate\ path if it does not already exist.

    Enter the following database information:

    - In the **Host** field, enter the host name or the IP address of the computer on which the database resides.

- In the **PORT** field, enter the port number on which the database listens for connections. The default port is 1521 for Oracle Database and 1433 for Microsoft SQL Server.
- In the **Database SID** field, enter the name of the database instance.
- In the **User Name** field, enter the user name of the database account that you created for Oracle Identity Manager.
- In the **Password** field, enter the Oracle Identity Manager database user password.
- Click **Next** to commit these settings.

Note: When you set the preceding items, see the configuration settings specified in "Using an Oracle Database for Oracle Identity Manager" on page 4-1 or "Using a Microsoft SQL Server Database for Oracle Identity Manager" on page 4-7 to verify your settings.

The installer checks for database connectivity and if a database schema exists. If the check passes, the installer proceeds to the next step in the process. If the check fails, an error message is displayed.

- Select the appropriate database options:
  - If a database exists, and the connectivity is good, proceed to Step 11.
  - If no connectivity is detected, you are prompted to enter new information or to fix the connection. Click **Next** after entering new information or fixing the connection.

  The Authentication Information page is displayed.

11. Select either the **Oracle Identity Manager Default Authentication** or **SSO** (Single Sign-On) **Authentication** option. If you select Single Sign-On authentication, you must provide the header variable used in the Single Sign-On system in the **Enter the header value for SSO Authentication** field. Click **Next**.

    The Application Server page is displayed.

12. Select **JBoss Application Server**, then click **Next**. The Cluster Information page is displayed.

13. Specify the server configuration (clustered or nonclustered) by using the following:

    - For a nonclustered installation, select **No** and click **Next**.
    - If you are deploying in a clustered installation, select **Yes**, enter the unique partition name, and see Chapter 9, "Deploying in a Clustered JBoss Application Server Configuration" on page 5-1 for more information.

    Note: JBoss Application Server clustered environments are not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in*Oracle Identity Manager Release Notes* for information about certified components.

    The Application Server Information page is displayed.

14. Enter the information about your application server and Java installation as follows:

    - Enter the path to your application server installation directory.

      Alternatively, click **Browse** and navigate to your application server installation directory.

    - Enter the path to the JDK directory.

      Alternatively, click **Browse** and navigate to the JDK directory.

    Then, click **Next**. The Application Server Configuration Backup page is displayed.

15. Back up your application server and click **Next** to start server installation.

16. If the installer detects an existing database, you can choose to use that database.

    Select **Yes**, then click **Next**. If the existing database is not encrypted, you are prompted to encrypt it. Select **Yes**, then click **Next**.

17. The Summary page is displayed. Click **Install** to install the Oracle Identity Manager application.

18. After Oracle Identity Manager is installed, a message is displayed listing the location of the installer log file and the next steps you should perform.

    Click **OK** and complete the post installation steps listed in the message.

19. The Completed page is displayed.

    Click **Finish** to exit the installer.

After installing Oracle Identity Manager, follow the instructions in Chapter 7, "Postinstallation Configuration for Oracle Identity Manager and JBoss Application Server".

5.4 Removing Oracle Identity Manager

To remove an Oracle Identity Manager installation:

1. Stop Oracle Identity Manager if it is running, and stop all Oracle Identity Manager processes.
2. Delete the *OIM_HOME* directory in which you installed Oracle Identity Manager.

6

Installing Oracle Identity Manager on UNIX

This chapter explains how to install Oracle Identity Manager on UNIX in a nonclustered installation.

See Also:

- *Oracle Identity Manager Release Notes* for information about supported UNIX platforms
- Chapter 9, "Deploying in a Clustered JBoss Application Server Configuration" for information about deploying Oracle Identity Manager in a clustered installation

You must install Oracle Identity Manager on systems running JBoss Application Server. Oracle Identity Manager components such as the Remote Manager can be installed on separate systems. Each component has its own installer.

This chapter contains the following topics:

6.1 Installation Prerequisites and Notes

The following is a list of prerequisites and notes for installing Oracle Identity Manager on UNIX:

- The Oracle Identity Manager Installer program requires at least 200 MB of free space in the home directory while installing Oracle Identity Manager. Check the /etc/passwd file to determine the home directory. Note that you cannot work around this requirement by changing the value of the $HOME variable.
- There must be at least 200 MB of free space in the /var/tmp directory.
- Set the `JAVA_HOME` variable before installing Oracle Identity Manager by using the following steps:

  1. Set the JAVA HOME variable, for example:

     ```
     export JAVA_HOME=/opt/j2sdk1.4.2_15
     ```

  2. Export the path to the JAVA HOME variable, for example:

```
export PATH=JAVA_HOME/bin:$PATH
```

See *Oracle Identity Manager Release Notes* for information about the certified versions of Java JDK.

- If you are using Microsoft SQL Server as your database, before installing Oracle Identity Manager ensure that the following three files are in the *JBOSS_HOME*`/server/default/lib/` directory and add the driver location to the system CLASSPATH environment variable:

  Note: Microsoft SQL Server is not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

  - mssqlserver.jar
  - msbase.jar
  - msutil.jar
- The default logging package included by the base RedHat Linux installation causes installation problems and exceptions for Oracle Identity Manager. Before installing Oracle Identity Manager on RedHat Linux, delete the commons-logging-1.0.2 library from the base operating system installation. The commons-logging-1.0.2 library is typically installed with any standard RedHat installation. In addition, ensure that you delete the symbolic links in the `/usr/share/java/` directory. Deleting these symbolic links will force Oracle Identity Manager to use its own internal logger JAR files during installation.
- Do not install Oracle Identity Manager on top of an existing Oracle Identity Manager installation. Use a different Oracle Identity Manager home directory. If you want to reuse the same directory name for the Oracle Identity Manager home directory, then back up your previous Oracle Identity Manager home by renaming the original directory.

  In addition, all Oracle Identity Manager components must be installed in different home directories. For example, you cannot install the Remote Manager in the same directory in which Oracle Identity Manager is installed.

6.2 Installing the Database Schema

As part of the installation, the Oracle Identity Manager Installer loads a schema into the database. You only install the database schema once. It is installed the first time you run the Oracle Identity Manager Installer. Each subsequent time you run the installer to deploy other Oracle Identity Manager components, you enter information about the database connection to configure the component for the same schema. If required, contact your database administrator (DBA).

Note: During the schema installation, a corresponding log file is created under the *OIM_HOME*`/logs/` directory named `dbInstall.log`.

6.3 Installing Documentation

The Oracle Identity Manager documentation is installed automatically in the *OIM_HOME* directory. No special input is required. A full documentation set is installed with each Oracle Identity Manager component.

6.4 Installing Oracle Identity Manager on UNIX

Oracle Identity Manager for UNIX is installed through a console mode installer, which supports the following two input methods:

- Choose from among a list of options

  Each option is numbered and accompanied by brackets ([ ]). To select an option, enter its number. Once selected, the associated brackets display an X ([X]).

- Enter information at a prompt

  Type the information at the prompt and press **Enter**. Default values are enclosed in brackets after a prompt; to accept a default value, press **Enter**.

The installer contains logical sections or panels. You can perform the following actions in the panels:

- When you have selected an item from a list of options, enter zero (**0**) to indicate that the desired item has been selected.
- To move to the next installation panel, enter **1**.
- To go back to the previous panel, enter **2**.
- To cancel the installation, enter **3**.
- To redisplay the current panel, enter **5**.

To install Oracle Identity Manager on UNIX:

1. Insert the Oracle Identity Manager Installation CD into your CD-ROM drive.
2. From the console, change directory (cd) to the installServer directory on the installation CD and run the install_server.sh file by using the following command:

   ```
   sh install_server.sh
   ```

   The installer starts in console mode.

   Note: If you are not installing Oracle Identity Manager from distributed media (CD), you must set the execute bit of all shell scripts in the installServer directory. To set the execute bit for all shell scripts recursively, cd to the installServer directory and run the following command:

   ```
   find . -name "*.sh" -exec chmod u+x {} \;
   ```

3. Choose a language by entering a number from the list of languages.

   Enter **0** to apply the language selection. The Welcome Message panel is displayed.

4. Enter **1** on the Welcome Message panel to display the next panel.

   The Admin User Information panel is displayed.

5. Enter a password you want to use for the Oracle Identity Manager Administrator, confirm the password by entering it again, and then enter **1** to move to the next panel.

   The OIM Application Options panel is displayed.

6. Enter **1** on the OIM Application Options panel to display the next panel.

   The Select the Oracle Identity Manager application to install panel is displayed.

7. Select the application to install:

   - Enter **1** for Oracle Identity Manager.
   - Enter **2** for the Oracle Identity Manager with Audit and Compliance Module.

   Enter **0** when you are finished to the next panel. The Target directory panel is displayed.

8. On the Target directory panel, complete one of the sub-steps that follow:

   - Enter the path to the directory in which you want to install Oracle Identity Manager. For example, enter `/opt/oracle/`.
   - Enter **1** to move to the next panel.

   If the directory does not exist, you are asked to create it. Enter **y** to create the directory.

   Note: For some non-English installations, irrespective of the prompt, only `y` works.

   The Database Server Selection panel is displayed.

   Note: To install against an existing database, verify that the version of Oracle Identity Manager you are installing is certified with your existing database version. See *Oracle Identity Manager Release Notes* to confirm the certified configurations.

   When Oracle Identity Manager is installed against an existing database, a warning message is displayed indicating that the database schema already exists and instructing you to copy the .xldatabasekey file from the existing Oracle Identity Manager installation to the new *`OIM_HOME`*`/xellerate/config` directory after you complete the installation process.

   Create the new *`OIM_HOME`*`/xellerate/config` directory if it does not already exist.

   ---

9. On the Database Server Selection panel, specify the type of database that you are using:

   - Enter **1** for Oracle Database.
   - Enter **2** for Microsoft SQL Server.

   Note: Microsoft SQL Server is not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

- Enter **0** when you are finished.
- Enter **1** to move to the next panel.

10. Enter the database information:
    - Enter the database host name or IP address.
    - Enter the port number, or accept the default.
    - Enter the SID for the database name.
    - Enter the database user name for the account that Oracle Identity Manager uses to connect to the database.
    - Enter the password for the database account that Oracle Identity Manager uses to connect to the database.
    - Enter **1** to move to the next panel.

    The Authentication Information panel is displayed.

11. Select the authentication mode for the Oracle Identity Manager Web application.
    - Enter **1** for Oracle Identity Manager Default Authentication.
    - Enter **2** for SSO Authentication.
    - Enter **0** when you are finished.

    If you select SSO authentication, then you must provide the header variable used in the Single Sign-On system when prompted.

    Enter **1** to move to the next panel.

    The Application Server Selection panel is displayed.

12. Specify your application server type.
    - Enter **4** for JBoss Application Server.
    - Enter **0** when you are finished.
    - Enter **1** to move to the next panel.

    The Cluster Information panel is displayed.

13. Provide the following information regarding deploying in a cluster:
    - Enter **1** for Yes (clustered) and enter the unique partition name at the prompt.
    - Enter **2** for No (non-clustered).
    - Enter **0** when you are finished.
    - Enter **1** to move to the next section.

    The Application Server Information panel is displayed.

Important:

- If you are deploying in a clustered installation, select **Yes** and see Chapter 9, "Deploying in a Clustered JBoss Application Server Configuration" on page 9-1 for more information.

- JBoss Application Server clustered environments are not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

---

14. In the Application Server Information panel:
    - Provide the location where the application server is installed
    - Provide the location where the JDK is installed
    - Enter **1** to move to the next section.
15. When you receive a message about backing up the application server installation, enter **1** to move to the next section. The Summary panel is displayed.
16. On the Summary panel, enter **1** to begin installation.
17. After the installation is finished, the Completed panel is displayed. Enter **3** to finish and exit.

After installing Oracle Identity Manager, follow the instructions in Chapter 7, "Postinstallation Configuration for Oracle Identity Manager and JBoss Application Server".

6.5 Removing Oracle Identity Manager

To remove an Oracle Identity Manager installation:

1. Stop Oracle Identity Manager if it is running and stop all Oracle Identity Manager processes.
2. Delete the *OIM_HOME* directory in which you installed Oracle Identity Manager.

7

Postinstallation Configuration for Oracle Identity Manager and JBoss Application Server

After installing Oracle Identity Manager, you must complete some postinstallation tasks before using the application. Depending on the deployment, you might choose not to perform some of these tasks. The following is a list of the postinstallation tasks documented in this chapter:

- Default JMS Queue Configuration
- Reserving JBoss Application Server Ports on Microsoft Windows Installation
- Changing Keystore Passwords
- Setting Log Levels
- Enabling Single Sign-On (SSO) for Oracle Identity Manager
- Configuring Multiple JBoss Application Server Installations to Use a Single Database
- Configuring Custom Authentication
- Setting the Compiler Path for Adapter Compilation
- Encrypting Oracle Identity Manager Database Password in the xell-ds.xml File for JBoss Application Server
- Deploying the SPML Web Service
- Tuning JDBC Connection Pools

Note: The examples in this chapter are Microsoft Windows-based, however the postinstallation tasks apply to UNIX as well.

7.1 Default JMS Queue Configuration

In releases earlier than 9.1.0, Oracle Identity Manager uses a single JMS queue (named xlQueue) for all asynchronous operations including requests, reconciliation, attestation, and offline tasks. Release 9.1.0 onward, by default, Oracle Identity Manager uses separate JMS queues for specific operations to optimize JMS queue processing. The following list shows the JMS queues in the default configuration and indicates the operation related to each queue:

- `xlQueue` (for request operations)

- `xlReconQueue` (for reconciliation operations)
- `xlAuditQueue` (for auditing operations)
- `xlAttestationQueue` (for attestation operations)
- `xlProcessQueue` (for use in a future release)

7.2 Reserving JBoss Application Server Ports on Microsoft Windows Installation

Perform the following steps to reserve the necessary ports for JBoss Application Server on Microsoft Windows installation:

1. Select **Run** from the Windows **Start** menu. The Run dialog box is displayed.
2. Enter `regedt32` in the Run dialog box and click **OK**. The Registry Editor window is displayed.
3. Navigate to the following registry key:

   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters

4. If it does not already exist, create a ReservedPorts value, as follows:
   a. Point to **New** on the **Edit** menu and click **Multi-String Value**.
   b. Enter `ReservedPorts` as the value name and press **Enter**.
5. Double-click the **ReservedPorts** value. The Edit Multi-String dialog box is displayed.
6. In the Edit Multi-String dialog box, enter `1098-1434` in the **Value data** box.
7. Click **OK** to close the Edit Multi-String dialog box, and then close the Registry Editor window.

7.3 Changing Keystore Passwords

During installation, the passwords for the Oracle Identity Manager keystores are set to `xellerate`. The Installer scripts and installation log contain this default password. It is strongly recommended that you change the keystore passwords for all production installations.

To change the keystore passwords, you must change the storepass of .xlkeystore and the keypass of the xell entry in .xlkeystore—and these two values must be identical. Use the keytool and the following steps to change the keystore passwords:

1. Open a command prompt on the Oracle Identity Manager host computer.
2. Navigate to the *OIM_HOME*\xellerate\config directory.
3. Run the keytool with the following options to change the storepass:

   JAVA_HOME\jre\bin\keytool -storepasswd -new *new_password* -storepass xellerate -keystore .xlkeystore -storetype JKS

4. Run the keytool with the following options to change the keypass of the xell entry in .xlkeystore:

   JAVA_HOME\jre\bin\keytool -keypasswd -alias xell -keypass xellerate -new *new_password* -keystore .xlkeystore -storepass *new_password*

Note: Replace *new_password* with the same password entered in step 3.

Table 7–1 lists the options used in the preceding example of keytool usage.

Table 7–1 Command Options for the keytool Utility

Option	Description
JAVA_HOME	Location of the Java directory associated with the application server
new_password	New password for the keystore
`-keystore` *option*	Keystore whose password you are changing (.xlkeystore for Oracle Identity Manager or .xldatabasekey for the database)
`-storetype` *option*	JKS for .xlkeystore and JCEKS for .xldatabasekey

5. Open *OIM_HOME*`\xellerate\config\xlconfig.xml` in a text editor.
6. Edit the <xl-configuration>.<Security>.<XLPKIProvider>.<KeyStore> section, <xl-configuration>.<Security>.<XLPKIProvider>.<Keys> section, and <RMSecurity>.<KeyStore> section to specify the keystore password as follows:

   Note: Change the <XLSymmetricProvider>.<KeyStore> section of the configuration file to update the password for the database keystore (.xldatabasekey).

   - Change the password tag to `encrypted="false"`.
   - Enter the password (in the clear), for example:

```
<Security>
<XLPKIProvider>
<KeyStore>
      <Location>.xlkeystore</Location>
      <Password encrypted="false">new_password</Password>
      <Type>JKS</Type>
      <Provider>sun.security.provider.Sun</Provider>
</KeyStore>
<Keys>
<PrivateKey>
<Alias>xell</Alias>
<Password encrypted="false">new_password</Password>
</PrivateKey>
</Keys>
<RMSecurity>
<KeyStore>
<Location>.xlkeystore</Location>
<Password encrypted="false">new_password</Password>
<Type>JKS</Type>
<Provider>sun.security.provider.Sun</Provider>
</KeyStore>
```

7. Save and close the xlconfig.xml file.
8. Restart the application server.

When you stop and start the application server, a backup of the configuration file is created. The configuration file (with the new password) is read in, and the password is encrypted in the file.

9. If all of the preceding steps have succeeded, you can delete the backup file.

Note: On UNIX, you might also want to clear the shell's command history by using the following command:

```
history -c
```

7.4 Setting Log Levels

Oracle Identity Manager uses log4j for logging. For JBoss-based installations, logging is configured in the log4j.xml file.

By default, the log level is set to Warning, except for DDM, for which the log level is set to Debug by default. You can change the log level universally for all components or for an individual component. For normal operation of Oracle Identity Manager, this postinstallation configuration step is not required.

7.4.1 Oracle Identity Manager Component Logging

The components are listed in the *OIM_HOME*\xellerate\config\log.properties file in the XELLERATE section. They are:

```
log4j.logger.XELLERATE=WARN
log4j.logger.XELLERATE.DDM=DEBUG
log4j.logger.XELLERATE.ACCOUNTMANAGEMENT=DEBUG
log4j.logger.XELLERATE.SERVER=DEBUG
log4j.logger.XELLERATE.RESOURCEMANAGEMENT=DEBUG
log4j.logger.XELLERATE.REQUESTS=DEBUG
log4j.logger.XELLERATE.WORKFLOW=DEBUG
log4j.logger.XELLERATE.WEBAPP=DEBUG
log4j.logger.XELLERATE.SCHEDULER=DEBUG
log4j.logger.XELLERATE.SCHEDULER.Task=DEBUG
log4j.logger.XELLERATE.ADAPTERS=DEBUG
log4j.logger.XELLERATE.JAVACLIENT=DEBUG
log4j.logger.XELLERATE.POLICIES=DEBUG
log4j.logger.XELLERATE.RULES=DEBUG
log4j.logger.XELLERATE.DATABASE=DEBUG
log4j.logger.XELLERATE.APIS=DEBUG
log4j.logger.XELLERATE.OBJECTMANAGEMENT=DEBUG
log4j.logger.XELLERATE.JMS=DEBUG
log4j.logger.XELLERATE.REMOTEMANAGER=DEBUG
log4j.logger.XELLERATE.CACHEMANAGEMENT=DEBUG
log4j.logger.XELLERATE.ATTESTATION=DEBUG
log4j.logger.XELLERATE.AUDITOR=DEBUG
```

7.4.2 Setting Log Levels for JBoss Application Server

The log4j.xml file is used for all logging with JBoss Application Server; therefore, Oracle Identity Manager components use an Xellerate tag. The log4j.xml file contains a general setting for Xellerate:

```
<category name="XELLERATE">
  <priority value="WARN" />
```

```
</category>
```

You can change the log level for all components by editing the `priority value` of the general setting, or for a specific component by adding a new logging category element.

The available categories are listed in the log.properties file in the XELLERATE section. See Oracle Identity Manager Component Logging on page 7-4 for more information.

For example, to change the level for Oracle Identity Manager, add the following element to the log4j.xml file:

```
<category name="XELLERATE.SERVER">
  <priority value="WARN" />
  <appender-ref ref="FILE"/>
</category>
```

To set Oracle Identity Manager log levels in JBoss Application Server:

1. Open the *JBOSS_HOME*`\server\default\conf\log4j.xml` file in a text editor.
2. Insert an element for the desired component.
3. Set the `priority value` to the appropriate level for the desired components.

   The following is a list of the supported log levels, appearing in descending order of information logged (DEBUG logs the most information and FATAL logs the least information):

   - DEBUG
   - INFO
   - WARN
   - ERROR
   - FATAL
4. Save your changes.

7.5 Enabling Single Sign-On (SSO) for Oracle Identity Manager

The following procedure describes how to enable Single Sign-On for Oracle Identity Manager with ASCII character logins. To enable Single Sign-On with non-ASCII character logins, use the following procedure—but include the additional configuration setting described in step 4.

> **See Also:** *Oracle Identity Manager Best Practices Guide* for additional information about configuring Single Sign-On for Oracle Identity Manager with Oracle Access Manager.

> **Note:** Header names comprised only of alphabetic characters are certified. Oracle recommends that you do not use special characters or numeric characters in header names.

To enable Single Sign-On for Oracle Identity Manager:

1. Stop the application server gracefully.

2. Open *OIM_HOME*/xellerate/config/xlconfig.xmlin a text editor.
3. Locate the following Single Sign-On configuration (the following are the default settings without Single Sign-On):

```
<web-client>
<Authentication>Default</Authentication>
<AuthHeader>REMOTE_USER</AuthHeader>
</web-client>
```

4. Edit the Single Sign-On configuration to the following and replace *SSO_HEADER_NAME* with the appropriate header configured in your Single Sign-On system:

```
<web-client>
<Authentication>SSO</Authentication>
<AuthHeader><SSO_HEADER_NAME></AuthHeader>
</web-client>
```

   To enable Single Sign-On with non-ASCII character logins, you must include a decoding class name to decode the non-ASCII header value. Add the decoding class name and edit the Single Sign-On configuration as follows:

```
<web-client>
<Authentication>SSO</Authentication>
<AuthHeader>SSO_HEADER_NAME</AuthHeader>
<AuthHeaderDecoder>com.thortech.xl.security.auth.CoreIDSSOAuthHeaderDecoder</Au
thHeaderDecoder>
</web-client>
```

   Replace *SSO_HEADER_NAME* with the appropriate header configured in your Single Sign-On system.

5. Change your application server and Web server configuration to enable Single Sign-On by referring to your application and Web server vendor documentation.
6. Restart the application server.

7.6 Configuring Multiple JBoss Application Server Installations to Use a Single Database

When two or more non-clustered JBoss Application Server installations connected to a load balancer point to a single database, you must configure the individual JBoss Application Server instances to use different JMS tables.

Complete the following procedure on the second and all other JBoss Application Server instances by using the same Oracle Identity Manager database so they use different JMS tables:

1. Open the *JBOSS_HOME*\server\default\deploy\jms*database_name*-jdbc2-service.xml in a text editor.

   Note that *database_name* refers to the common database used by multiple JBoss Application Server instances.

2. In all the queries and statements in the sqlProperties section of the *database_name*-jdbc2-service.xml, change the names of the tables represented by JMS_MESSAGES and JMS_TRANSACTIONS to new, unique, and valid values.
3. Add the following statements to the end of the file:

```
DELETE_TEMPORARY_MESSAGES = DELETE FROM NEW_JMS_MESSAGES_NAME
    WHERE TXOP='T'
CREATE_IDX_MESSAGE_TXOP_TXID = CREATE INDEX
    NEW_JMS_MESSAGES_NAME_TXOP_TXID ON NEW_JMS_MESSAGES_NAME (TXOP, TXID)
CREATE_IDX_MESSAGE_DESTINATION = CREATE INDEX
    NEW_JMS_MESSAGES_NAME_DESTINATION ON NEW_JMS_MESSAGES_NAME(DESTINATION)
```

Note: *NEW_JMS_MESSAGES_NAME* represents the new name of the JMS_MESSAGES tables you changed in step 2.

4. Save and close the file.

7.7 Configuring Custom Authentication

This section describes how to use custom authentication solutions with Oracle Identity Manager.

Oracle Identity Manager deploys a Java Authentication and Authorization Service (JAAS) module to authenticate users. For unattended logins, which require offline message processing and scheduled task execution, Oracle Identity Manager uses signature-based authentication. Although you should use JAAS to handle signature-based authentication, you can create a custom authentication solution to handle standard authentication requests.

Note: The Oracle Identity Manager JAAS module must be deployed on your application server and should be the first invoked authenticator.

To enable custom authentication on JBoss Application Server, you implement a custom authentication module class. Oracle Identity Manager will then delegate standard authentication requests to the custom authentication module.

To implement custom authentication for JBoss Application Server:

1. Open the *OIM_HOME*\config\xlconfig.xml file in a text editor and add the following element, replacing *CustomLoginModule* with the name of your custom login module class.

```
<login-module>
     <thirdPartyLoginModule>CustomLoginModule</thirdPartyLoginModule>
</login-module>
```

2. Optional. Specify configuration properties for the custom authentication module by opening the *JBOSS_HOME*\config\login-config.xml file in a text editor and adding a module option within the <login-module> element for the com.thortech.xl.security.jboss.UsernamePasswordLoginModule package. Oracle Identity Manager will delegate the specified properties to the custom authentication module class. For example, the following rolesProperties module option will be delegated to the custom authentication module class.

```
<application-policy name = "xellerate">
  <authentication>
    <login-module
      code="com.thortech.xl.security.jboss.XLClientLoginModule"
```

```
        flag="required">
      </login-module>
      <login-module code=
          "com.thortech.xl.security.jboss.UsernamePasswordLoginModule"
           flag = "required" >
        <module-option name =
          "unauthenticatedIdentity">Unknown</module-option>
        <module-option name =
          "data-source">java:/jdbc/xlDS</module-option>
        <module-option name =
          "rolesProperties">customRoleProperties</module-option>
      </login-module>
    </authentication>
  </application-policy>
```

3. Copy your custom authentication JAR file to the *JBOSS_HOME*\server\default\lib directory.

7.7.1 Protecting the JNDI Namespace

When you specify a custom authentication solution, you should also protect the Java Naming and Directory Interface (JNDI) namespace to ensure that only designated users have permission to view resources. The primary purpose of protecting the JNDI namespace is to protect Oracle Identity Manager from any malicious applications that might be installed in the same application server instance. Even if no other applications, malicious or otherwise, are installed in the same application server instance as Oracle Identity Manager, you should protect your JNDI namespace as a routine security measure.

To protect your JNDI namespace and configure Oracle Identity Manager to access it:

1. Open the *OIM_HOME*\config\xlconfig.xml file in a text editor and add the following elements to the <Discovery> element:

   ```
   <java.naming.security.principal>
   <java.naming.security.credentials>
   ```

2. To optionally encrypt the JNDI password, add an encrypted attribute that is assigned a value of true to the <java.naming.security.credentials> element, and assign the password as the element's value, as follows:

   ```
   <java.naming.security.credentials
     encrypted="true">password</java.naming.security.credentials>
   ```

3. Add the following elements to the <Scheduler> element:

   ```
   <CustomProperties>
     <org.quartz.dataSource.OracleDS.java.naming.security.principal>user
     </org.quartz.dataSource.OracleDS.java.naming.security.principal>
     <org.quartz.dataSource.OracleDS.java.naming.security.credentials>pwd
     </org.quartz.dataSource.OracleDS.java.naming.security.credentials>
   </CustomProperties>
   ```

7.7.2 Increasing the Transaction Timeout

You might have to increase the transaction timeout values for JBoss Application Servers, as the default values can be low for certain transactions. Oracle recommends the values specified in the following step:

Open the *JBOSS_HOME*\server\default\conf\jboss-service.xml file and ensure that transaction timeout attribute value is set to 1200 as follows:

```
<attribute name="TransactionTimeout">1200</attribute>
```

Note:

- For clustered JBoss Application Server, the corresponding file is located at *JBOSS_HOME*\server\all\conf\jboss-service.xml.

- JBoss Application Server clustered environments are not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

7.8 Setting the Compiler Path for Adapter Compilation

To compile adapters or import Deployment Manager XML files that have adapters, you must set the compiler path. To set the compiler path for adapter compilation, you must first install the Design Console. See Chapter 10, "Installing and Configuring the Oracle Identity Manager Design Console" for instructions on installing the Design Console and then setting the compiler path for adapter compilation.

7.9 Encrypting Oracle Identity Manager Database Password in the xell-ds.xml File for JBoss Application Server

By default, JBoss Application Server does not encrypt data source passwords, as described in the JBoss document at

http://wiki.jboss.org/wiki/Wiki.jsp?page=EncryptingDataSourcePasswords

This section describes how to encrypt the Oracle Identity Manager database password in JBoss Application Server deployments. Specifically, you must perform the following steps to manually encrypt a password, and then modify the `xell-ds.xml` and `login-config.xml` files so that they can access the encrypted form of the password instead of the clear text version:

Note: For a clustered installation, repeat this procedure on all the nodes of the cluster.

1. Open a console window and navigate to the *JBOSS_HOME* directory.
2. Run one of the following commands to encrypt the Oracle Identity Manager database password. In this command, replace *password* with the actual password that you want to encrypt.

   UNIX/Linux

   ```
   java -cp "JBOSS_HOME/lib/jboss-jmx.jar:lib/jboss-common.jar:server/
   default/lib/jboss-jca.jar:server/default/lib/jbosssx.jar"
   org.jboss.resource.security.SecureIdentityLoginModule password
   ```

   Microsoft Windows

   ```
   java -cp "JBOSS_HOME/lib/jboss-jmx.jar;lib/jboss-common.jar;server/
   default/lib/jboss-jca.jar;server/default/lib/jbosssx.jar"
   org.jboss.resource.security.SecureIdentityLoginModule password
   ```

3. The command you run in the previous step returns an encoded form of the password you specify. For example, the password `Welcome1` is encoded as `3146f9cc50afd6a6df8592078de921bc`. Highlight and copy the encoded password.
4. Open the *JBOSS_HOME*`/server/default/deploy/xell-ds.xml` file in a text editor.
5. Delete the `<user-name>` and `<password>` elements from the `<local-tx-datasource>` element.
6. Add the following `<security-domain>` element to the end of the `<local-tx-datasource>` element:

```
<security-domain>EncryptDBPassword</security-domain>
```

7. Delete the `<xa-datasource-property name="User">` and `<xa-datasource-property name="Password">` elements from the `<xa-datasource>` element.
8. Add the following `<security-domain>` element to the end of the `<xa-datasource>` element:

```
<security-domain>EncryptXADBPassword</security-domain>
```

9. Save and close the *JBOSS_HOME*`/server/default/deploy/xell-ds.xml` file.
10. Open the *JBOSS_HOME*`/server/default/conf/login-config.xml` file in a text editor.
11. Add the following elements to the `<application-policy>` element:

Note: Replace *datasource_username* with the datasource user name and *encoded_password* with the encoded password you copy in Step 3.

```
<application-policy name = "EncryptDBPassword">
 <authentication>
  <login-module code = "org.jboss.resource.security.SecureIdentityLoginModule"
flag = "required">
   <module-option name = "username">datasource_username</module-option>
   <module-option name = "password">encoded_password</module-option>
   <module-option name =
"managedConnectionFactoryName">jboss.jca:service=LocalTxCM,name=jdbc/xlDS</modu
le-option>
  </login-module>
 </authentication>
</application-policy>

<application-policy name = "EncryptXADBPassword">
 <authentication>
  <login-module code = "org.jboss.resource.security.SecureIdentityLoginModule"
flag = "required">
   <module-option name = "username">datasource_username</module-option>
   <module-option name = "password">encoded_password</module-option>
   <module-option name =
"managedConnectionFactoryName">jboss.jca:service=XATxCM,name=jdbc/xlXADS</modul
e-option>
  </login-module>
```

```
 </authentication>
</application-policy>
```

12. Save and close the *JBOSS_HOME*/server/default/deploy/login-config.xml file.

7.10 Deploying the SPML Web Service

Organizations can have multiple provisioning systems that exchange information about the modification of user records. In addition, there can be applications that must interact with multiple provisioning systems. The SPML Web Service provides a layer over Oracle Identity Manager to interpret SPML requests and convert them to Oracle Identity Manager calls.

The SPML Web Service is packaged in a deployable Enterprise Archive (EAR) file. This file is generated when you install Oracle Identity Manager.

Because the EAR file is generated while you install Oracle Identity Manager, a separate batch file in the Oracle Identity Manager home directory runs the scripts that deploy the SPML Web Service on the application server on which Oracle Identity Manager is running. You must run the batch file to deploy the SPML Web Service.

For details about the SPML Web Service, see Chapter 12, "The SPML Web Service" in *Oracle Identity Manager Tools Reference*.

7.11 Tuning JDBC Connection Pools

To implement tuning for the JDBC connection pools used by Oracle Identity Manager, open *JBOSS_HOME*/server/default/deploy/xell-ds.xml file and implement the following changes:

Note:

- For a clustered installation of Oracle Identity Manager on JBoss Application Server, the xell-ds.xml file can be located at *JBOSS_HOME*/server/all/farm.

- JBoss Application Server clustered environments are not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

- It is strongly recommended that you implement the suggested tuning for the JDBC connection pools used by Oracle Identity Manager. This can be further tuned based on the application usage.

---

1. For jdbc/xlDS pool, insert the following before the line </local-tx-datasource>:

```
<min-pool-size>30</min-pool-size>
<max-pool-size>50</max-pool-size>
<blocking-timeout-millis>15000</blocking-timeout-millis>
<idle-timeout-minutes>15</idle-timeout-minutes>
```

2. For jdbc/xlXADS pool, insert the code mentioned in step 1 before the line </xa-datasource>.

3. Restart JBoss Application Server.

8

Starting and Stopping Oracle Identity Manager

This chapter describes how to start and stop Oracle Identity Manager, and how to access the Administrative and User Console. This chapter contains the following topics:

> **Important:** You must complete all post-installation steps in Chapter 7, "Postinstallation Configuration for Oracle Identity Manager and JBoss Application Server" on page 7-1 before starting Oracle Identity Manager.

8.1 Removing Backup xlconfig.xml Files After Starting or Restarting

After you start any Oracle Identity Manager component for the first time, or after you change any passwords in the xlconfig.xml file, Oracle Identity Manager encrypts and saves the passwords. Oracle Identity Manager also creates a backup copy of the xlconfig.xml file before saving changes to the file. These backup files contain old passwords in plaintext. The backup file are named xlconfig.xml.*x*, where *x* is the latest available number, for example xlconfig.xml.0, xlconfig.xml.1, and so on.

> **Note:** You must remove these backup files after starting any Oracle Identity Manager component for the first time, or on restarting after changing any passwords in xlconfig.xml once you have established that the new password is working properly.

8.2 Starting Oracle Identity Manager

This section describes how to start Oracle Identity Manager on Microsoft Windows, UNIX.

To start Oracle Identity Manager:

1. Verify that your database is up and running.

2. Start Oracle Identity Manager by running one of the following scripts appropriate for your operating system. Running the Oracle Identity Manager start script also starts JBoss Application Server.

   On Microsoft Windows

   OIM_HOME\xellerate\bin\xlStartServer.bat

   On UNIX

   OIM_HOME/xellerate/bin/xlStartServer.sh

8.3 Stopping Oracle Identity Manager

To stop Oracle Identity Manager gracefully, you stop the JBoss Application Server by running one of the following scripts appropriate for your operating system.

On Microsoft Windows

JBOSS_HOME\bin\shutdown.bat -S

On UNIX

JBOSS_HOME/bin/shutdown.sh -S

8.4 Accessing the Administrative and User Console

After starting the JBoss Application Server and Oracle Identity Manager, you can access the Administrative and User Console.

To access the Administrative and User Console:

1. Browse to the following URL by using a Web browser:

   http://*hostname*:*port*/xlWebApp

   In this URL, *hostname* represents the name of the computer hosting the application server and *port* refers to the port on which the server is listening. The default port number for JBoss Application Server is 8080.

   Note: The application name, xlWebApp, is case-sensitive.

   For example:

   http://localhost:8080/xlWebApp

2. After the Oracle Identity Manager login page is displayed, log in with your user name and password.

8.5 Using the Diagnostic Dashboard to Verify Installation

The Diagnostic Dashboard verifies each component in your postinstallation environment by testing for:

- A trusted store
- Single Sign-On Configuration
- Messaging capability
- A task scheduler

- A Remote Manager

The Diagnostic Dashboard also checks for all supported versions of components along with their packaging.

> **Note:** See "Using the Diagnostic Dashboard" on page 2-4 for more information.

9

Deploying in a Clustered JBoss Application Server Configuration

Note: JBoss Application Server clustered environments are not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

This chapter describes how to deploy Oracle Identity Manager in a clustered JBoss Application Server environment.

This chapter discusses the following topics:

- Overview of Installation in a Clustered Installation
- Installing Oracle Identity Manager on the First Node
- Copying Oracle Identity Manager to Additional JBoss Application Server Nodes
- Setting up the Load Balancer for JBoss Application Server
- Installing and Configuring a Database for Oracle Identity Manager
- Configuring Oracle Identity Manager on the JBoss Application Server Cluster
- Configuring the JBoss Application Server Cluster to Use a Common Database
- Starting the JBoss Application Server Cluster

Caution: Deploying an application in a clustered installation is a complex procedure. This document assumes that you have expertise in installing and using applications in a JBoss Application Server cluster. These instructions provide the Oracle Identity Manager-specific details only. They are not complete instructions for setting up a JBoss Application Server cluster. For more information about clustering, see JBoss Application Server documentation.

9.1 Overview of Installation in a Clustered Installation

To install Oracle Identity Manager on a JBoss Application Server cluster, you must complete the following general tasks:

1. Install Oracle Identity Manager on the first node in your JBoss Application Server cluster.

See "Installing Oracle Identity Manager on the First Node" on page 9-2 for more information.

2. Copy the JBoss Application Server and Oracle Identity Manager installation directories from the first node in your JBoss Application Server cluster to all other nodes. Ensure to maintain the original directory structure throughout this process.

   See "Copying Oracle Identity Manager to Additional JBoss Application Server Nodes" on page 9-2 for more information.

3. If you are using Microsoft SQL Server as your database, locate the JDBC driver files (mssqlserver.jar, msbase.jar, and msutil.jar) and copy them to the *JBOSS_HOME*`/server/all/lib` directory.

   Note: Microsoft SQL Server is not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

   See "Configuring JBoss Application Server for Microsoft SQL Server" on page 4-9 for more information.

4. Set up the load balancer for your JBoss Application Server cluster.

   See "Setting up the Load Balancer for JBoss Application Server" on page 9-3 for more information.

5. Perform postinstallation configuration of Oracle Identity Manager on the JBoss Application Server cluster.

   See "Configuring Oracle Identity Manager on the JBoss Application Server Cluster" on page 9-6 for more information.

6. Start the cluster.

   See "Starting the JBoss Application Server Cluster" on page 9-8 for more information.

9.2 Installing Oracle Identity Manager on the First Node

Follow the installation steps for Oracle Identity Manager in "Installing Oracle Identity Manager on Microsoft Windows" on page 5-2 or "Installing Oracle Identity Manager on UNIX" on page 6-3 to install Oracle Identity Manager on the initial node in your JBoss Application Server cluster.

9.3 Copying Oracle Identity Manager to Additional JBoss Application Server Nodes

Ensure that the name and path of the *JAVA_HOME* directory used by Oracle Identity Manager is the same across all the nodes of the cluster.

Then, for each additional node in your JBoss Application Server cluster, copy the JBoss and Oracle Identity Manager installation directories from the first node to all other nodes, making sure to maintain the original directory structure and hierarchy throughout this process.

9.4 Setting up the Load Balancer for JBoss Application Server

The procedure for installing a load balancer for your JBoss Application Server cluster varies according to the operating system running on the host computers on which the JBoss Application Server nodes are installed.

9.4.1 Setting Up a Load Balancer for JBoss Application Server on Microsoft Windows

To set up a load balancer on Microsoft Windows:

1. Download the latest distribution package for the Apache2 Web server from Apache.org, then install the Apache server in a directory that this document henceforth refers to as *APACHE_HOME*.
2. Download the latest distribution package mod_jk 1.2.x from the Tomcat connector section page at the following URL: http://tomcat.apache.org/download-connectors.cgi.
3. Copy the library named mod_jk.so to the *APACHE_HOME*\modules directory.
4. Set up Apache to use modjk by adding the following line (and the accompanying comment line) as the last line of the *APACHE_HOME*\conf\httpd.conf file:

   ```
   # Include mod_jk configuration file

   Include conf/mod_jk.conf
   ```

5. In the directory *APACHE_HOME*\conf, create a configuration file to forward requests to JBoss Application Server instances.

   Name this file as mod_jk.conf and populate with the following lines:

   ```
   # Load mod_jk module
        # Specify the filename of the mod_jk lib
        LoadModule jk_module modules/mod_jk.so
        # Where to find workers.properties
        JkWorkersFile conf/workers.properties
        # Where to put jk logs
        JkLogFile logs/mod_jk.log

        # Set the jk log level [debug/error/info]
        JkLogLevel info
        # Select the log format
        JkLogStampFormat  "[%a %b %d %H:%M:%S %Y]"
        # JkOptions indicates to send SSK KEY SIZE
        JkOptions +ForwardKeySize +ForwardURICompat -ForwardDirectories
        # JkRequestLogFormat
        JkRequestLogFormat "%w %V %T"
        # Mount your applications
        JkMount /application/* loadbalancer
        # You can use external file for mount points.
        # It will be checked for updates each 60 seconds.
        # The format of the file is: /url=worker
        # /examples/*=loadbalancer
        JkMountFile conf/uriworkermap.properties
        # Add shared memory.
        # This directive is present with 1.2.10 and
        # later versions of mod_jk, and is needed for
        # for load balancing to work properly
        JkShmFile logs/jk.shm
        # Add jkstatus for managing runtime data
        <Location /jkstatus/>
        JkMount status
   ```

```
Order deny,allow
Deny from all
Allow from all
</Location>
```

6. Review the directive descriptions located at the Apache Tomcat Connector Documentation Index Web site at the following URL:

   http://tomcat.apache.org/connectors-doc/

   Ensure to follow the guidelines concerning Apache cache size.

   In the *APACHE_HOME*\conf directory, create a file named workers.properties and populate it with the following lines:

```
# Define list of workers that will be used
     # for mapping requests
       worker.list=loadbalancer,status
     # Define Node1
     # modify the host as your host IP or DNS name.
       worker.node1.port=8009
       worker.node1.host=IP of node1
       worker.node1.type=ajp13
       worker.node1.lbfactor=1
     # worker.node1.local_worker=1 (1)
       worker.node1.cachesize=10
     # Define Node2
     # modify the host as your host IP or DNS name.
       worker.node2.port=8009
       worker.node2.host= IP of node2
       worker.node2.type=ajp13
       worker.node2.lbfactor=1
     # worker.node2.local_worker=1 (1)
       worker.node2.cachesize=10
     # Load-balancing behavior
       worker.loadbalancer.type=lb
       worker.loadbalancer.balance_workers=node1,node2
       worker.loadbalancer.sticky_session=1
     # worker.loadbalancer.local_worker_only=1
     # worker.list=loadbalancer
```

7. If your JBoss Application Server cluster contains more than two nodes, then you must add extra lines to the workers.properties file in the *APACHE_HOME*\conf directory.

   For example, if you have three nodes, you must add the following lines.

```
  # modify the host as your host IP or DNS name.
       worker.node3.port=8009
       worker.node3.host= IP of node3
       worker.node3.type=ajp13
       worker.node3.lbfactor=1
     # worker.node3.local_worker=1 (1)
   worker.node3.cachesize=10
```

   For each subsequent node, you must add the preceding group of lines again, except that you must change all references to node3, node4, node5, or node 6, and so on as appropriate.

8. In the *APACHE_HOME*\conf directory, create the uriworkermap.properties file, which will hold the URL mappings Apache forwards to Tomcat.

This file enables mod_jk to forward to Tomcat requests from /mx-console, /web-console, /xlWebApp, /xlScheduler as well as /Nexaweb. The syntax for each line is `/url=worker_name`. Paste this example into the file you created:

```
# Simple worker configuration file
     # Mount the Servlet context to the ajp13 worker
     /jmx-console=loadbalancer
     /jmx-console/*=loadbalancer
     /web-console=loadbalancer
     /web-console/*=loadbalancer
     /xlWebApp=loadbalancer
     /xlWebApp/*=loadbalancer
     /xlScheduler=loadbalancer
     /xlScheduler/*=loadbalancer
     /Nexaweb=loadbalancer
     /Nexaweb/*=loadbalancer
```

9. Start Apache by starting Microsoft Windows Explorer, navigating to the directory *APACHE_HOME*\bin, then double clicking Apache.exe.

9.4.2 Setting Up a Load Balancer for JBoss Application Server on UNIX

To set up the load balancer on UNIX:

1. Download the binary file for Apache 2.0 for UNIX from the following URL:

   http://httpd.apache.org/download.cgi

2. Run the following commands to install Apache:

   a. `tar xvfz httpd-2.0.54.tar.gz`

   b. `cd httpd-2.0.54`

   c. `./configure --prefix=/opt/apache2 --enable-module=so`

   d. `make`

   e. `make install`

3. Download the `jakarta-tomcat-connectors-1.2.14-src.tar.gz` file from the Apache Software Foundation Web site by searching at the following URL: http://www.apache.org/

4. Run the following commands to install the connector:

   a. `tar xzvf jakarta-tomcat-connectors-1.2.14-src.tar.gz`

   b. `cd jakarta-tomcat-connectors-1.2.14-src/jk/native`

   c. `chmod 755 buildconf.sh`

   d. `./buildconf.sh`

   e. `./configure --with-apxs=/opt/apache2/bin/apxs`

   f. `make`

   g. `make install`

   h. `cd / jakarta-tomcat-connectors-jk1.2.14-src/jk/native/apache-2.0/`

   i. `cp mod_jk.so /opt/apache2/modules/`

5. Complete steps 4 to 8 in the procedure "Setting Up a Load Balancer for JBoss Application Server on Microsoft Windows" on page 9-3 as they are the same steps for Microsoft Windows and UNIX.
6. Navigate to the *APACHE_HOME*/bin/ directory, then run the following command:

   ```
   ./apachectl start
   ```

9.5 Installing and Configuring a Database for Oracle Identity Manager

Refer to Chapter 4, "Installing and Configuring a Database For Oracle Identity Manager" for information.

9.6 Configuring Oracle Identity Manager on the JBoss Application Server Cluster

After you install Oracle Identity Manager on your JBoss Application Server cluster, you must perform certain configuration steps on each node in the cluster.

To configure Oracle Identity Manager on your JBoss Application Server cluster:

1. For each successive node in the cluster, navigate to the directory *JBOSS_HOME*/server/all/deploy/jbossweb-tomcat55.sar/, open server.xml in a text editor, and perform the following steps:
   a. Locate the following string:

      ```
      <Engine name="jboss.web" defaultHost="localhost" jvmRoute
      ```

   b. Change the value of jvmRoute to the name of the node associated with the computer on which you are currently working. (The name of the node should be node1, node2, or node3, and so on as listed in the workers.properties file associated with the computer on which you are currently working).

      For Example:

      ```
      <Engine name="jboss.web" defaultHost="localhost"
      jvmRoute="node1">
      ```

2. For each successive node in your cluster, navigate to the directory *JBOSS_HOME*/server/all/deploy, and open the following files:

   cluster-service.xml

   tc5-cluster-service.xml

   a. Comment out the following block in both of the preceding files:

      ```
      <!--
      <Config>
                  <UDP mcast_addr="228.1.2.3" mcast_port="45566" ip_ttl="8"
      ip_mcast="true" mcast_send_buf_size="800000" mcast_recv_buf_size="150000"
      ucast_send_buf_size="800000" ucast_recv_buf_size="150000"
      loopback="false"/>
                  <PING timeout="2000" num_initial_members="3" up_thread="true"
      down_thread="true"/>
                  <MERGE2 min_interval="10000" max_interval="20000"/>
                  <FD shun="true" up_thread="true" down_thread="true"
      timeout="2500" max_tries="5"/>
                  <VERIFY_SUSPECT timeout="3000" num_msgs="3" up_thread="true"
      down_thread="true"/>
                  <pbcast.NAKACK gc_lag="50"
      ```

```
retransmit_timeout="300,600,1200,2400,4800" max_xmit_size="8192"
up_thread="true" down_thread="true"/>
            <UNICAST timeout="300,600,1200,2400,4800"
window_size="100" min_threshold="10" down_thread="true"/>
            <pbcast.STABLE desired_avg_gossip="20000"
               up_thread="true" down_thread="true"/>
            <FRAG frag_size="8192" down_thread="true" up_thread="true"/>
            <pbcast.GMS join_timeout="5000" join_retry_timeout="2000"
               shun="true" print_local_addr="true"/>
            <pbcast.STATE_TRANSFER up_thread="true" down_thread="true"/>
         </Config>
 -->
```

b. Uncomment the following block in both files:

```
 <Config>
            <TCP bind_addr="thishost" start_port="7800" loopback="true"/>
            <TCPPING initial_hosts="thishost[7800],otherhost[7800]"
port_range="3" timeout="3500" num_initial_members="3" up_thread="true"
down_thread="true"/>
            <MERGE2 min_interval="5000" max_interval="10000"/>
            <FD shun="true" timeout="2500" max_tries="5" up_thread="true"
              down_thread="true" />
            <VERIFY_SUSPECT timeout="1500" down_thread="false"
up_thread="false" />
           <pbcast.NAKACK down_thread="true" up_thread="true"
gc_lag="100" retransmit_timeout="3000"/>
            <pbcast.STABLE desired_avg_gossip="20000" down_thread="false"
              up_thread="false" />
            <pbcast.GMS join_timeout="5000" join_retry_timeout="2000"
              shun="false"  print_local_addr="true" down_thread="true"
              up_thread="true"/>
            <pbcast.STATE_TRANSFER up_thread="true" down_thread="true"/>
         </Config>
```

c. Within the block listed in Step b, replace `thishost` with the IP of the computer on which you are currently working.

The entire IP list must be surrounded by double quotes. For example: `TCPbind_addr="192.168.161.20"`.

d. Within the block listed in Step b, replace `otherhost` with the IP of the other computer in the cluster, or, if the cluster contains more than two nodes, replace `otherhost` with a comma-delimited list of all the IPs.

The IP must be surrounded by double quotes.

3. For each successive node in the cluster, modify the *OIM_HOME*`/xellerate/config/xlconfig.xml` file.

Locate the setting for the `java.naming.provider.url` in the `<Discovery>` section and insert a comma-delimited list of URLs corresponding to all the nodes in cluster.

For example, you might change a string as shown in the following sample:

```
<java.naming.provider.url>
    jnp://localhost:1100
  </java.naming.provider.url>
to the following string:
 <java.naming.provider.url>
    jnp://<IP of node1>:1100,<IP of node 2>:1100
  </java.naming.provider.url>
```

9.7 Configuring the JBoss Application Server Cluster to Use a Common Database

Perform the following steps on the second and remaining cluster members to configure the cluster to use a common database:

1. Open the *JBOSS_HOME*/server/all/deploy-hasingleton/jms/*database_name*-jdbc2-service.xml in a text editor.

   Note that *database_name* refers to the common database used by the cluster.

2. In all the queries and statements in the sqlProperties section of the *database_name*-jdbc2-service.xml, change the names of the tables represented by JMS_MESSAGES and JMS_TRANSACTIONS to new, unique, and valid values.

3. Add the following statements to the end of the file:

```
DELETE_TEMPORARY_MESSAGES = DELETE FROM NEW_JMS_MESSAGES_NAME
    WHERE TXOP='T'
CREATE_IDX_MESSAGE_TXOP_TXID = CREATE INDEX
    NEW_JMS_MESSAGES_NAME_TXOP_TXID ON NEW_JMS_MESSAGES_NAME (TXOP, TXID)
CREATE_IDX_MESSAGE_DESTINATION = CREATE INDEX
    NEW_JMS_MESSAGES_NAME_DESTINATION ON NEW_JMS_MESSAGES_NAME(DESTINATION)
```

   Note: *NEW_JMS_MESSAGES_NAME* represents the new name of the JMS_MESSAGES tables you changed in step 2.

4. Save and close the file.

9.8 Starting the JBoss Application Server Cluster

To start the JBoss Application Server cluster on which you have installed and configured Oracle Identity Manager:

1. Initially, start only one node in the cluster (commonly referred to as the master node).

   Navigate to the directory *OIM_HOME*/xellerate/bin, then run one of the following commands, as appropriate for the operating system on the computer hosting JBoss Application Server and Oracle Identity Manager:

   On Microsoft Windows:

   ```
   xlStartServer.bat
   ```

   On UNIX:

   ```
   xlStartServer.sh
   ```

2. On each remaining computer in the cluster, navigate to the directory *OIM_HOME*/xellerate/bin. Then run one of the following commands, as appropriate for the operating system on the computer hosting JBoss Application Server and Oracle Identity Manager:

   On Microsoft Windows:

   ```
   xlStartServer.bat
   ```

On UNIX:

```
xlStartServer.sh
```

3. Access the Administration console by opening a browser and pointing it to the following URL

   http://*IP_of_computer_in_which_apache_server_is_running*/xlWebApp

10

Installing and Configuring the Oracle Identity Manager Design Console

This chapter explains how to install the Oracle Identity Manager Design Console, which is a Java client. You have the option to install the Design Console on the same computer as your Oracle Identity Manager installation or on a separate computer.

This chapter discusses the following topics:

10.1 Requirements for Installing the Design Console

Verify that your environment meets the following requirements for Design Console installation:

- You must have a running installation of Oracle Identity Manager.
- If you are installing the Design Console on a computer other than the host for the application server, you must know the host name and port number of the computer hosting that application server.
- The Design Console host must be able to ping the application server host by using both the IP address and the host name.
- For clustered Oracle Identity Manager installations, you must know the host name and port number of the Web server.

Note: If you cannot resolve the host name of the application server, then try adding the host name and IP address in the hosts file in the C:\winnt\system32\drivers\etc\ directory.

10.2 Installing the Design Console

> **Note:** All Oracle Identity Manager components must be installed in different home directories. If you are installing the Design Console on a computer that is hosting another Oracle Identity Manager component, such as Oracle Identity Manager or the Remote Manager, then you must specify a different installation directory for the Design Console.

To install the Design Console on a Microsoft Windows host:

1. Insert the Oracle Identity Manager Installation CD into your CD-ROM drive.
2. Using Microsoft Windows Explorer, navigate to the installServer directory on the installation CD.
3. Double-click the setup_client.exe file.
4. Choose a language from the list on the Installer page. The Welcome page is displayed.
5. On the Welcome page, click **Next**.
6. On the Target directory page, complete one of the following sub-steps:
   a. The default directory for the Design Console is C:\oracle. To install the Design Console in this directory, click **Next**.
   b. To install the Design Console in another directory, specify the path of the directory in the **Directory** field, and then click **Next**.

   > **Note:** If the directory path that you select does not exist, then the Base Directory settings field is displayed. Click **OK**. This directory is automatically created. If you do not have write permission to create the default directory, then a message is displayed informing you that the installer could not create the directory. Click **OK** to close the message and then contact your system administrator to obtain the appropriate permissions.

7. On the Application Server page, select JBoss, then click **Next**. The next page prompts you to specify the JRE to use with Design Console.
8. Select the JRE that is installed with Oracle Identity Manager or specify an existing JRE. Then, click **Next**. The Application Server configuration page is displayed.
9. On the Application Server Host Information page, enter the information appropriate for the application server hosting your Oracle Identity Manager installation:
   a. In the first field, enter the host name or IP address.

   > **Note:** The host name is case-sensitive.

   b. In the second field, enter the naming port for the application server on which Oracle Identity Manager is deployed.
   c. Click **Next**.

10. On the Graphical Workflow Rendering Information page, enter the application server configuration information:
    a. Enter the Oracle Identity Manager server IP address.
    b. Enter the port number.
    c. Select **Yes** or **No** to specify whether or not the Design Console must use Secure Sockets Layer (SSL).
    d. Click **Next**.
11. On the Shortcut page, select (or deselect) the check boxes for the shortcut options according to your preferences:
    a. Choose to create a shortcut to the Design Console on the Start Menu.
    b. Choose to create a shortcut to the Design Console on the desktop.
    c. Click **Next** when you are satisfied with the check box settings.
12. On the Summary page, click **Install** to begin the Design Console installation.
13. The final installation page displays a reminder to copy certain application server-specific files to your Oracle Identity Manager installation. Follow these instructions and then click **OK**.
14. Click **Finish** to complete the installation process.

10.3 Postinstallation Requirements for the Design Console

For both clustered and non-clustered installations, copy the *JBOSS_HOME*\client\jbossall-client.jar file from the computer hosting Oracle Identity Manager to the *OIM_DC_HOME*\xlclient\ext directory on the computer on which you are installing the Design Console instance.

To complete installation for clustered installations:

1. Change the <Discovery> settings in the *OIM_DC_HOME*\xlclient\Config\xlconfig.xml file for all Design Console installations.

   For example, you would change a string like the following:

   ```
   <java.naming.provider.url>
     jnp://localhost:1100
   </java.naming.provider.url>
   ```

   to the following string:

   ```
   <java.naming.provider.url>
     jnp://IP_of_node1:1100,IP_of_node2:1100
   </java.naming.provider.url>
   ```

2. Add the following tag to Discovery.CoreServer section of the *OIM_DC_HOME*\xlclient\Config\xlconfig.xml file:

   ```
   <jnp.partitionName>MyPartition</jnp.partitionName>
   ```

   MyPartition represents the partition name you specified during Oracle Identity Manager on JBoss Application Server clusters.

Note: JBoss Application Server clustered environments are not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

3. To configure Workflow Visualization to access all available nodes in the cluster:
   a. Open the *OIM_DC_HOME*\xlclient\Config\xlconfig.xml and locate the following statement:

      ```
      <ApplicationURL>...</ApplicationURL>
      ```

   b. Replace the application server URL with the IP address and port of the Web server, as follows:

      ```
      <ApplicationURL>http://webserverIP/xlWebApp/LoginWorkflowRenderer.do
      </ApplicationURL>
      ```

4. In the configuration XML file, change the multicast address to match that of Oracle Identity Manager:
   a. Open the following file:

      OIM_HOME\xellerate\config\xlconfig.xml

   b. Search for the <MultiCastAddress> element, and copy the value assigned to this element.
   c. Open the following file:

      OIM_DC_HOME\xlclient\Config\xlconfig.xml

   d. Search for the <Cache> element, and replace the value of the <MultiCastAddress> element inside this element with the value that you copy in Step b.

10.4 Starting the Design Console

To start the Design Console, double-click *OIM_DC_HOME*\xlclient\xlclient.cmd or select Design Console from the Microsoft Windows Start menu or desktop.

10.5 Setting the Compiler Path for Adapter Compilation

In the System Configuration form of the Design Console, you must set the XL.CompilerPath system property to include the path of the bin directory inside the JDK directory (*JDK_HOME*\bin) that is used by the application server on which Oracle Identity Manager is deployed.

Then, restart Oracle Identity Manager.

See Also: The "Rule Elements, Variables, Data Types, and System Properties" section in *Oracle Identity Manager Reference*

10.6 Configuring SSL Communication With the Design Console (Optional)

After installing the Oracle Identity Manager Design Console, you might want to configure it to communicate to your Oracle Identity Manager over SSL. Use the

following procedure to configure communication from your Design Console to Oracle Identity Manager over SSL.

1. Stop Oracle Identity Manager.
2. Perform the following backup tasks:
   - Create a backup of the *OIM_HOME* directory in which you installed Oracle Identity Manager.
   - Create a backup of the *OIM_DC_HOME* directory in which you installed the Oracle Identity Manager Design Console.
   - Create a backup of the *JBOSS_HOME* directory in which you installed JBoss Application Server.
3. Export the Oracle Identity Manager certificate by using the following commands:
   a. `cd OIM_HOME\config`
   b. `%JAVA_HOME%\bin\keytool -export -file xlserver.cer -keystore .xlkeystore -storepass xellerate -alias xell`

      The xlserver.cer file is created in the config folder.
4. Open the *OIM_HOME*`\config\xljbossssl-service.xml` file:
   a. Find the following line:

      ```
      <attribute name="KeyStorePass"><XDtConfig:configParameter
      ValueparamName="KeyStorePass"/></attribute>
      ```

   b. Change the line to the following:

      ```
      <attribute name="KeyStorePass">xellerate</attribute>
      ```

5. Change the installation profile by using the following commands:
   a. `cd OIM_HOME\profiles`
   b. Open the jboss.profile file and set the following properties:
      - `configure.ssl.invoker=true`
      - `jboss.ssl.invocation=true`
      - `jboss.ssl.port=10443`
      - `jboss.ssl.clustered.port=10444`
      - `jboss.stateful.invoker=xl-stateful-rmi-invoker`
      - `jboss.stateless.invoker=xl-stateless-rmi-invoker`
6. Run the setup command by using the following commands:
   a. `cd OIM_HOME\setup`
   b. `setup_jboss.cmd database_password`

Note:

- For non-clustered installation, *JBOSS_DIR* refers to *JBOSS_HOME*\server\default and for clustered installation it refers to *JBOSS_HOME*\server\all.

- JBoss Application Server clustered environments are not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

7. Edit the login-config.xml file by using the following commands:

   a. cd *JBOSS_DIR*\conf

   b. Open the login-config.xml file and find the XML tags toward the end in the file that look like the following:

```
<policy>
...
...
...
       <application-policy name= "xellerate">
              <authentication>
              ....
              ....
             </authentication>
       </application-policy>
</policy>
```

   c. You will see two application-policy entries. Remove the last entry.

Note: Ensure that you remove the lines starting with <application-policy name="xellerate"> and ending through </application-policy>. Do not remove the last line ending with </policy>.

8. Copy the *OIM_HOME*\config\xlserver.cer file to *OIM_DC_HOME*\java\lib\security on all Design Console systems that will communicate with Oracle Identity Manager.

   Use the following command to copy the xlserver.cer file:

```
..\..\bin\keytool -import -file xlserver.cer -keystore
cacerts -storepass changeit -trustcacerts -alias xell
```

   When prompted, enter yes to trust the certificate.

9. Copy the *OIM_HOME*\config\.xlkeystore file to the *JBOSS_DIR*\conf\ directory.

10. Copy the cacerts file from the *OIM_DC_HOME*\java\lib\security directory to the *JBOSS_DIR*\conf\ directory.

11. Open the *JBOSS_HOME*\deploy\jbossweb-tomcat55.sar\server.xml file:

    a. Find the line that starts with:

```
<!-- SSL/TLS Connector configuration using the admin devl
guide keystore -->
```

b. Edit the lines in this entry so that it is displayed as follows:

```
<!-- SSL/TLS Connector configuration using the admin devl guide keystore
-->
        <Connector port="8443" address="${jboss.bind.address}"
              maxThreads="100" minSpareThreads="5" maxSpareThreads="15"
              scheme="https" secure="true" clientAuth="false"
              keystoreFile="${jboss.server.home.dir}/conf/.xlkeystore"
              keystorePass="xellerate"
              truststoreFile="${jboss.server.home.dir}/conf/cacerts"
              truststorePass="changeit"
             sslProtocol = "TLS" />
```

c. Uncomment the entry.

d. Save and close the updated `server.xml` file.

12. Open the *OIM_DC_HOME*`\config\xlconfig.xml` in a text editor.

Change

```
<ApplicationURL>http://HOSTNAME:8080/xlWebApp/loginWorkflowRenderer.do
    </ ApplicationURL>
```

To:

```
<ApplicationURL>https://HOSTNAME:8443/xlWebApp/loginWorkflowRenderer.do
    </ ApplicationURL>
```

Note:

- It is assumed that the JBOSS application server uses `8080` as the HTTP port and `8443` as the HTTPS port.
- For clustered JBOSS installations, the value for `<ApplicationURL>` in *OIM_DC_HOME*`\config\xlconfig.xml` can point to one application server URL or it can point to the Web server URL. In the second case, you must trust the Web server certificate from the Web server as described in step 7 of this procedure.
- JBoss Application Server clustered environments are not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

---

13. Restart Oracle Identity Manager for the changes to take effect.

10.7 Removing the Design Console Installation

To remove the Design Console installation:

1. Stop Oracle Identity Manager and the Design Console if they are running.

2. Stop all Oracle Identity Manager processes.

3. Delete the *OIM_DC_HOME* directory in which you installed the Design Console.

11

Installing and Configuring the Oracle Identity Manager Remote Manager

This chapter explains how to install Oracle Identity Manager Remote Manager. It contains the following sections:

11.1 Installing the Remote Manager on Microsoft Windows

Complete the following steps to install the Remote Manager on a Microsoft Windows host.

> **Important:** All Oracle Identity Manager components must be installed in different home directories. If you are installing the Remote Manager on a computer that is hosting another Oracle Identity Manager component (the server or the Design Console), specify an installation directory that has not been used.

1. Insert the Oracle Identity Manager Installation CD into your CD-ROM drive.
2. Using Microsoft Windows Explorer, navigate to the installServer directory in the installation CD.
3. Double-click the setup_rm.exe file.
4. Choose a language from the list on the Installer page. The Welcome page is displayed.
5. On the Welcome page, click **Next**.
6. On the Target directory page, complete one of the following sub-steps:
   a. The default directory for Oracle Identity Manager products is C:\oracle. To install the Remote Manager in this directory, click **Next**.
   b. To install Remote Manager in a different directory, specify the path of the directory in the **Directory name** field, and then click **Next**.

> **Note:** If the directory path that you specified does not exist, then the Base Directory settings field is displayed. Click **OK**. The directory is automatically created. If you do not have write permission to create the default directory for Oracle Identity Manager, then a message is displayed informing you that the installer could not create the directory. Click **OK** to close the message, and then contact your system administrator to obtain the appropriate permissions.

7. Select either the JRE that is installed with Oracle Identity Manager or specify an existing JRE. Click **Next**. The Remote Manager Configuration page is displayed.
8. On the Remote Manager Configuration page, enter the appropriate information for the Remote Manager:
   a. Enter the service name. The default value is RManager.
   b. Enter the Remote Manager binding port. The default value is 12346.
   c. Enter the Remote Manager Secure Sockets Layer (SSL) port. The default value is 12345.
   d. Click **Next**.
9. On the Shortcut page, select (or deselect) the check boxes for the following shortcut options according to your preferences:
   a. Choose to create a shortcut for the Remote Manager on the desktop.
   b. Choose to create a shortcut for the Remote Manager on the Start Menu.
10. Click **Next** when you are satisfied with the check box settings.
11. On the Summary page, review the configuration details, and then click **Install** to begin the installation.
12. After the installation has completed, click **Finish** on the Completed page to exit.

11.2 Installing the Remote Manager on UNIX

To install the Remote Manager on UNIX:

> **Note:** Before installing the Remote Manager, you must set the JAVA_HOME variable to the JRE included with the Remote Manager installer.

1. Insert the Oracle Identity Manager Installation CD into your CD-ROM drive.
2. From the File Manager, access the installServer directory in the installation CD.
3. Run the install_rm.sh file. The command-line installer starts.
4. Choose a language from the list by entering a number and then entering **0** to apply the language. The Welcome panel is displayed.
5. On the Welcome panel, enter **1** to move to the next panel. The Target directory panel is displayed.
6. On the Target directory panel, enter the path to the directory in which you want to install the Oracle Identity Manager Remote Manager. The default directory is /opt/oracle.

- Enter **1** to move to the next panel.
- If the directory does not exist, you are asked to create it. Enter **y** to create the directory.

> **Note:** All Oracle Identity Manager components must be installed in different home directories. If you are installing the Remote Manager on a computer that is hosting Oracle Identity Manager, then you must specify a unique installation directory.

7. Specify the JRE to use with Remote Manager:
   - Enter **1** to install the JRE included with Oracle Identity Manager.
   - Enter **2** to use an existing JRE at a specified location.

   After specifying the JRE, enter **0** to accept your selection and then enter **1** to move to the next panel.
8. On the Remote Manager Configuration panel, enter the Remote Manager configuration information:
   a. Enter the Service Name, or press **Enter** to accept the default value.
   b. Enter the Remote Manager binding port, or press **Enter** to accept the default value.
   c. Enter the Remote Manager SSL port, or press **Enter** to accept the default value.
   d. Enter **1** to move to the next panel.

      The Remote Manager installation summary panel is displayed.
9. Check the information.
   - Enter **2** to go back and make changes.
   - Enter **1** to start the installation.

     Oracle Remote Manager installs and the Post Install Summary panel is displayed.
10. Enter **3** to finish the installation.

11.3 Configuring the Remote Manager

The Remote Manager and Oracle Identity Manager communicate using SSL. If you are using Remote Manager, you must enable a trust relationship between Oracle Identity Manager and the Remote Manager. (The server must trust the Remote Manager certificate).

You also have the option to enable client-side authentication (where the Remote Manager checks the server's certificate). Import the Remote Manager's certificate into the keystore of Oracle Identity Manager and make it trusted. For client-side authentication, import the certificate for Oracle Identity Manager into the keystore for your Remote Manager, then make that certificate trusted. You must also manually edit the configuration file associated with the server, and depending on the options you selected during Remote Manager installation, the Remote Manager configuration file as well.

This section discusses the following topics:

- Trusting the Remote Manager Certificate
- Enabling Client-side Authentication for Remote Manager

11.3.1 Changing the Remote Manager Keystore Passwords

During installation, the password for the Remote Manager keystore is set to `xellerate`. Oracle recommends changing the keystore passwords for all production installations.

To change the keystore passwords, you must change the storepass of .xlkeystore and the keypass of the xell entry in .xlkeystore—and these two values must be identical. Use the keytool utility and the following steps to change the keystore passwords:

1. Open a command prompt on the Oracle Identity Manager host computer.
2. Navigate to the *OIM_RM_HOME*`\xellerate\config` directory.
3. Run the keytool utility with the following options to change the storepass:

   ```
   JAVA_HOME\jre\bin\keytool -storepasswd -new new_password -storepass xellerate
   -keystore .xlkeystore -storetype JKS
   ```

4. Run the keytool utility with the following options to change the keypass of the xell entry in .xlkeystore:

   ```
   JAVA_HOME\jre\bin\keytool -keypasswd -alias xell -keypass xellerate
   -new new_password -keystore .xlkeystore -storepass xellerate
   ```

   JAVA_HOME represents the location of the Java installation associated with the Remote Manager installation.
5. Open *OIM_RM_HOME*`\xlremote\config\xlconfig.xml` in a text editor.
6. Edit the `<RMSecurity>.<KeyStore>` section to specify the keystore password as follows:
   - Change the password tag to `encrypted="false"`.
   - Enter the password, for example:

     ```
     <RMSecurity>
     <KeyStore>
     <Location>.xlkeystore</Location>
     <Password encrypted="false">new_password</Password>
     <Type>JKS</Type>
     <Provider>sun.security.provider.Sun</Provider>
     </KeyStore>
     ```

Note: If you are using client-side authentication for the Remote Manager, enter the Oracle Identity Manager's keystore password in the <RMSecurity>.<TrustStore> section of *OIM_RM_HOME*`/xlremote\config\xlconfig.xml` as follows:

```
<TrustStore>
<Location>.xlkeystore</Location>
<Password encrypted="false">OIM_Server_keystore_password</Password>
<Type>JKS</Type>
<Provider>sun.security.provider.Sun</Provider>
</TrustStore>
```

---

7. Save and close the xlconfig.xml file.
8. Restart the Remote Manager.
9. Open *OIM_HOME*\xellerate\config\xlconfig.xml in a text editor.
10. Edit the <RMSecurity>.<TrustStore> section to specify the new Remote Manager keystore password as follows:
    - Change the password tag to encrypted="false".
    - Enter the password (in the clear), for example:

```
<TrustStore>
<Location>.xlkeystore</Location>
<Password encrypted="false">new_password</Password>
<Type>JKS</Type>
<Provider>sun.security.provider.Sun</Provider>
</TrustStore>
```

11. Save and close the xlconfig.xml file, then restart Oracle Identity Manager.

11.3.2 Trusting the Remote Manager Certificate

To establish a trust relationship between Oracle Identity Manager and the Remote Manager:

1. Copy the Remote Manager certificate to the server computer. On the Remote Manager computer, locate the *OIM_RM_HOME*\xlremote\config\xlserver.cert file and copy it to the server computer.

> **Note:** The server certificate in the *OIM_HOME* directory is also named xlserver.cert. Ensure that you do not overwrite that certificate.

2. Open a command prompt on the server computer.
3. To import the certificate by using the keytool utility, use the following command:

```
JAVA_HOME\jre\bin\keytool -import -alias rm_trusted_cert -file
RM_cert_location\xlserver.cert -trustcacerts -keystore
OIM_HOME\xellerate\config\.xlkeystore -storepass xellerate
```

   JAVA_HOME is the location of the Java directory for your application server, the value of alias is an arbitrary name for the certificate in the store, and *RM_cert_location* is the location where you copied the certificate.

> **Note:** If you changed the keystore password, substitute that for xellerate for the value of the storepass variable.

4. Enter **Y** at the prompt to trust the certificate.
5. Open the *OIM_HOME*\xellerate\config\xlconfig.xml file in a text editor.
6. Locate the <RMIOverSSL> property and set it to true, for example:

```
<RMIOverSSL>true</RMIOverSSL>
```

7. Locate the <KeyManagerFactory> property. If you are using the IBM JRE, set the value to IBMX509. For all other JREs, set the value to SUNX509. For example:

```
<KeyManagerFactory>IBMX509</KeyManagerFactory>
```

or

```
<KeyManagerFactory>SUNX509</KeyManagerFactory>
```

8. Save the file.
9. Restart Oracle Identity Manager.

11.3.2.1 Using Your Own Certificate

To configure the Remote Manager by using your own certificate on the Remote Manager system:

1. Import your custom key in a new keystore (new_keystore_name) other than.xlkeystore. Remember the password (new_keystore_pwd) that you use for the new keystore.
2. Copy this new keystore to the *OIM_RM_HOME*\xlremote\config\ directory.
3. Open *OIM_RM_HOME*\xlremote\config\xlconfig.xml in a text editor.
4. Locate the <RMSecurity> tag and change the value in the <Location> and <Password> tags as follows:
   - If you are using the IBM JRE, change the values to:

     ```
     <KeyStore>
          <Location>new_keystore_name</Location>
          <Password encrypted="false">new_keystore_pwd</Password>
          <Type>JKS</Type>
          <Provider>com.ibm.crypto.provider.IBMJCE</Provider>
     </KeyStore>
     ```

   - For all other JREs, change the values to:

     ```
     <KeyStore>
          <Location>new_keystore_name</Location>
          <Password encrypted="false">new_keystore_pwd</Password>
          <Type>JKS</Type>
          <Provider>sun.security.provider.Sun</Provider>
     </KeyStore>
     ```

5. Restart the Remote Manager server and open the xlconfig.xml file to ensure that the password for the new keystore was encrypted.

To configure the Remote Manager by using your own certificate on the Oracle Identity Manager server:

1. Import the same certificate key used in the Remote Manager system to a new keystore (new_svrkeystore_name) other than .xlkeystore. Remember the password (new_svrkeystor_pwd) that you use for the new keystore.
2. Copy this new keystore to the *OIM_HOME*\xellerate\config directory.
3. Open *OIM_HOME*\xellerate\config\xlconfig.xml in a text editor.
4. Locate the <RMSecurity> tag and change the value in the <Location> and <Password> tags as follows:

   ```
   <TrustStore>
      <Location>new_svrkeystore_name</Location>
      <Password encrypted="false">new_svrkeystor_pwd</Password>
      <Type>JKS</Type>
   ```

```
   <Provider>sun.security.provider.Sun</Provider>
</TrustStore>
```

5. Restart Oracle Identity Manager and open the `xlconfig.xml` file to ensure that the password for the new keystore is encrypted.

11.3.3 Enabling Client-side Authentication for Remote Manager

To enable client-side authentication:

1. On the computer hosting the Remote Manager, open *OIM_RM_HOME*\xlremote\config\xlconfig.xml in a text editor.
2. Set the <ClientAuth> property to `true`, for example:

   ```
   <ClientAuth>true</ClientAuth>
   ```

3. Ensure the <RMIOverSSL> property is set to `true`, for example:

   ```
   <RMIOverSSL>true</RMIOverSSL>
   ```

4. Locate the <KeyManagerFactory> property.

   If you are using the IBM JRE, set the value to `IBMX509`. For example:

   ```
   <KeyManagerFactory>IBMX509</KeyManagerFactory>
   ```

   For all other JREs, set the value to `SUNX509`.

   ```
   <KeyManagerFactory>SUNX509</KeyManagerFactory>
   ```

5. Save the file.
6. Copy the server certificate to the Remote Manager computer. On the server computer, locate the *OIM_HOME*\xellerate\config\xlserver.cert file and copy it to the Remote Manager computer.

   Note: The Remote Manager certificate is also named xlserver.cert. Ensure that you do not overwrite that certificate.

7. Open a command prompt on the Remote Manager computer.
8. Import the certificate by using the following keytool command:

   ```
   JAVA_HOME\jre\bin\keytool -import -alias trusted_server_cert -file
   server_cert_location\xlserver.cert -trustcacerts -keystore
   XL_RM_HOME\xlremote\config\.xlkeystore -storepass xellerate
   ```

   JAVA_HOME is the location of the Java directory for your Remote Manager, the value of `alias` is an arbitrary name for the certificate in the store, *OIM_RM_HOME* is the home directory for the Remote Manager, and *server_cert_location* is the location to which you copied the server certificate.

   Note: If you changed the keystore password, substitute that value for xellerate, which is the default value of the storepass variable.

9. Enter **Y** at the prompt to trust the certificate.
10. Restart the Remote Manager.

11.4 Starting the Remote Manager

Use the following script to start the Remote Manager:

- On Microsoft Windows:

  OIM_RM_HOME\xlremote\remotemanager.bat

- On UNIX:

  OIM_RM_HOME/xlremote/remotemanager.sh

11.5 Removing the Remote Manager Installation

To remove the Remote Manager installation:

1. Stop Oracle Identity Manager and the Remote Manager if they are running.
2. Stop all Oracle Identity Manager processes.
3. Delete the *OIM_RM_HOME* directory in which you installed the Remote Manager.

12

Troubleshooting the Oracle Identity Manager Installation

This chapter describes problems that can occur during the Oracle Identity Manager Installation and contains the following topics:

- Task Scheduler fails in a Clustered Installation
- Default Login Does Not Work

> **Note:** You can use the Diagnostic Dashboard tool for assistance when you troubleshoot the Oracle Identity Manager Installation. See *Oracle Identity Manager Administrative and User Console Guide* for detailed information.

12.1 Task Scheduler fails in a Clustered Installation

The Task Scheduler fails to work properly when the cluster members (computers that are part of the cluster) have different settings on their system clocks. Oracle highly recommends that the system clocks for all cluster members be synchronized within a second of each other.

12.2 Default Login Does Not Work

If the default login is not working for the Design Console or Administrative and User Console:

- Ensure that you have copied the `jbossall-client.jar` file to the Design Console computer.
- Microsoft SQL Server only:

> **Note:** Microsoft SQL Server is not supported in Oracle Identity Manager release 9.1.0. See "Certified Components" in *Oracle Identity Manager Release Notes* for information about certified components.

  Ensure that the Distributed Transaction Coordinator is running.

A

Java 2 Security for JBoss Application Server

Note: The application might fail to start because of syntax errors in the policy files.

Be careful when you edit the policy files. Oracle recommends that you use the policy tool provided by the JDK for editing the policy files. The tool is available in the following directory:

JAVA_HOME/jre/bin/policytool

To enable Java 2 Security for Oracle Identity Manager:

1. Go to the *$JBOSS_HOME*/bin/ directory and open the run script (`run.bat` for Windows and `run.sh` for UNIX) as follows:

   a. Search for `JAVA_OPTS` and add the following JVM option after `-Dprogram.name=%PROGNAME%`:

```
-Djava.security.manager
-Djava.security.policy= $JBOSS_HOME/server/default/conf/server.policy
-Djboss.home.dir=$JBOSS_HOME
-Djboss.server.home.dir=$JBOSS_HOME/server/default
```

Note: Change *$JBOSS_HOME* to the actual JBoss Application Server directory location.

The following table explains the options.

Option	Description
`-Djava.security.manager`	Enables the Java 2 Security manager.
`-Djava.security.policy`	Specifies the policy file that is to be used for Java 2 Security.
`-Djboss.home.dir`	Specifies the value of the JBoss Application Server installation.
`-Djboss.server.home.dir`	Specifies the location of the JBoss Application Server configuration where Oracle Identity Manager is installed.

2. Go to the *JBOSS_HOME*/server/default/conf directory and modify the server.policy file by copying the Java 2 Security permissions from the Policy File.

Note: If the server.policy file does not exist, you have to create it.

Policy File

The server.policy file consists of the following code:

Note: The instructions to change the code in the policy file are given in comments, which are in bold font.

This server.policy example is for Windows installation, for UNIX ensure to change \\ between the directories name to / in every permission java.io.FilePermission property.

Ensure that you change the multicast IP 231.165.168.131 in this example to reflect the multicast IP address of the Oracle Identity Manager installation. You can find the Oracle Identity Manager multicast IP address in xlconfig.xml.

```
// Oracle Identity Manager Java2 security policy file
// Use -Djava.security.policy=server.policy
// and -Djboss.home.dir=c:/jboss
// and -Djboss.server.home.dir=c:/jboss/server/default

// ********************************************
// Java code and extensions
// ********************************************
// Trust java extensions
grant codeBase "file:${java.home}/lib/ext/-" {
   permission java.security.AllPermission;
};

// Trust core java code
grant codeBase "file:${java.home}/lib/*" {
   permission java.security.AllPermission;
};

// For java.home pointing to the JDK jre directory
grant codeBase "file:${java.home}/jre/lib/-" {
   permission java.security.AllPermission;
};

// ********************************************
// Java code and extensions ends
// ********************************************

// ********************************************
// JBoss Application Server code
// ********************************************

// Trust core JBoss Application Server code
grant codeBase "file:${jboss.home.dir}/bin/-" {
   permission java.security.AllPermission;
```

```
};

grant codeBase "file:${jboss.home.dir}/lib/-" {
   permission java.security.AllPermission;
};

grant codeBase "file:${jboss.server.home.dir}/lib/-" {
   permission java.security.AllPermission;
};

// ****************************************
// JBoss Application Server code ends
// ****************************************

// ****************************************
// JBoss Application Server deployed applications
// ****************************************

// Grant all permissions to the default applications deployed on
// JBoss Application Server. Please change the list depending on whether
// you are deploying on a single or clustered JBoss Application Server
//install.
// ---------------------------------------------
grant codeBase
    "file:${jboss.server.home.dir}/deploy/jboss-aop.deployer/-" {
    permission java.security.AllPermission;
};

grant codeBase
    "file:${jboss.server.home.dir}/deploy/jboss-bean.deployer/-" {
    permission java.security.AllPermission;
};

grant codeBase "file:${jboss.server.home.dir}/deploy/jms/-" {
    permission java.security.AllPermission;
};

grant codeBase
    "file:${jboss.server.home.dir}/deploy/http-invoker.sar/-" {
    permission java.security.AllPermission;
};

grant codeBase
    "file:${jboss.server.home.dir}/deploy/jbossweb-tomcat55.sar/-" {
    permission java.security.AllPermission;
};

grant codeBase
    "file:${jboss.server.home.dir}/deploy/jboss-ws4ee.sar/-" {
    permission java.security.AllPermission;
};

grant codeBase
    "file:${jboss.server.home.dir}/deploy/jmx-console.war/-" {
    permission java.security.AllPermission;
};

grant codeBase "file:${jboss.server.home.dir}/deploy/management/-" {
    permission java.security.AllPermission;
};
```

```
grant codeBase
    "file:${jboss.server.home.dir}/deploy/uuid-key-generator.sar" {
    permission java.security.AllPermission;
};

grant codeBase
    "file:${jboss.server.home.dir}/deploy/jboss-ha-local-jdbc.rar" {
    permission java.security.AllPermission;
};

grant codeBase
    "file:${jboss.server.home.dir}/deploy/jboss-ha-xa-jdbc.rar" {
    permission java.security.AllPermission;
};

grant codeBase
    "file:${jboss.server.home.dir}/deploy/jboss-local-jdbc.rar" {
    permission java.security.AllPermission;
};

grant codeBase
    "file:${jboss.server.home.dir}/deploy/jboss-xa-jdbc.rar" {
    permission java.security.AllPermission;
};

grant codeBase "file:${jboss.server.home.dir}/deploy/mail-ra.rar" {
    permission java.security.AllPermission;
};

// ****************************************
// JBoss Application Server deployed applications ends
// ****************************************

// ***************************************************************
// From here, Oracle Identity Manager application permissions start
// ***************************************************************

// Grant All permissions to nexaweb commons jar file to be loaded from
// $JBOSS_HOME/default/lib/
grant codeBase "file:${jboss.server.home.dir}/lib/nexaweb-common.jar" {
    permission java.security.AllPermission;
};


// OIM codebase permissions
grant codeBase "file:${jboss.server.home.dir}/deploy/XellerateFull.ear" {
      // File permissions

      // Need read,write,delete permissions on $OIM_HOME/config folder
      // to read various config files, write the
      // xlconfig.xml.{0,1,2..} files upon re-encryption and delete
      // the last xlconfig.xml if the numbers go above 9.
      permission java.io.FilePermission "${XL.HomeDir}\\config\\-",
                 "read, write, delete";
      permission java.io.FilePermission "${XL.HomeDir}\\-", "read";

      // Need read,write,delete permissions to generate adapter Java
      // code, delete the .class file when the adapter is loaded into
      // the database
```

```
permission java.io.FilePermission "${XL.HomeDir}\\adapters\\-",
"read,write,delete";

// This is required by the connectors and connector installer
permission java.io.FilePermission
      "${XL.HomeDir}\\ConnectorDefaultDirectory\\-",
      "read,write,delete";
permission java.io.FilePermission
      "${XL.HomeDir}\\connectorResources\\-",
      "read,write,delete";

// Need to read Globalization resource bundle files for various
// locales
permission java.io.FilePermission
      "${XL.HomeDir}\\customResources\\-", "read";

// Need to read code from "JavaTasks", "ScheduleTask",
// "ThirdParty", "EventHandlers" folder
permission java.io.FilePermission
      "${XL.HomeDir}\\EventHandlers\\-", "read";
permission java.io.FilePermission
      "${XL.HomeDir}\\JavaTasks\\-", "read";
permission java.io.FilePermission
      "${XL.HomeDir}\\ScheduleTask\\-", "read";
permission java.io.FilePermission
      "${XL.HomeDir}\\ThirdParty\\-", "read";

// Required by the Generic Technology connector
permission java.io.FilePermission "${XL.HomeDir}\\GTC\\-", "read";

// Server needs read permissions on Nexaweb home directory
//permission java.io.FilePermission "${nexaweb.home}\\-", "read";

// Read permissions on the jboss "tmp" folder, the OIM deploy
// directory and the jboss server "lib" folder.
permission java.io.FilePermission
      "${jboss.server.home.dir}\\tmp\\-", "read";
permission java.io.FilePermission
      "${jboss.server.home.dir}\\deploy\\XellerateFull.ear\\-",
      "read,write";
permission java.io.FilePermission
      "${jboss.server.home.dir}\\lib\\-", "read";

// OIM server invokes the Java compiler. You need "execute"
// permissions on all files.
permission java.io.FilePermission "<<ALL FILES>>", "execute";

// Socket permissions
// Basically we allow all permissions on non-privileged sockets
// The multicast address should be the same as the one in
// xlconfig.xml for Javagroups communication
permission java.net.SocketPermission "*:1024-",
      "connect,listen,resolve,accept";
permission java.net.SocketPermission "231.165.168.131",
      "connect,accept";

// Property permissions
// Read and write OIM properties
// Read XL.*, java.* and log4j.* properties
permission java.util.PropertyPermission "XL.*", "read,write";
```

```
        permission java.util.PropertyPermission "*", "read, write";
        permission java.util.PropertyPermission "java.*", "read";
        permission java.util.PropertyPermission "log4j.", "read";
        permission java.util.PropertyPermission "user.dir", "read";

        // Run-time permissions
        // OIM server needs permissions to create its own class loader,
        // get the class loader, modify threads and register shutdown
        // hooks
        permission java.lang.RuntimePermission "createClassLoader";
        permission java.lang.RuntimePermission "getClassLoader";
        permission java.lang.RuntimePermission "modifyThread";
        permission java.lang.RuntimePermission "modifyThreadGroup";
        permission java.lang.RuntimePermission "shutdownHooks";

        // OIM server needs run-time permissions to generate and load
        // classes in the packages specified below. Also access the
        // declared members of a class.
        permission java.lang.RuntimePermission

"defineClassInPackage.com.thortech.xl.adapterGlue.ScheduleItemEvents";
        permission java.lang.RuntimePermission
              "defineClassInPackage.com.thortech.xl.dataobj.rulegenerators";
        permission java.lang.RuntimePermission
              "defineClassInPackage.com.thortech.xl.adapterGlue";
        permission java.lang.RuntimePermission "accessDeclaredMembers";

        // The following run-time permissions are JBoss specific and will
        // differ between appservers. OIM server needs ability to see
        // current thread caller and credentials, and set the 'Run As'
        // role.
        permission java.lang.RuntimePermission
              "org.jboss.security.SecurityAssociation.getPrincipalInfo";
        permission java.lang.RuntimePermission
              "org.jboss.security.SecurityAssociation.setPrincipalInfo";
        permission java.lang.RuntimePermission
              "org.jboss.security.SecurityAssociation.setRunAsRole";

        // Reflection permissions
        // Give permissions to access and invoke fields/methods from
        // reflected classes.
        permission java.lang.reflect.ReflectPermission
              "suppressAccessChecks";

        // Security permissions for OIM server
        permission java.security.SecurityPermission "*";
        permission javax.security.auth.AuthPermission "doAs";
        permission javax.security.auth.AuthPermission "doPrivileged";
        permission javax.security.auth.AuthPermission "getSubject";
        permission javax.security.auth.AuthPermission "modifyPrincipals";
        permission javax.security.auth.AuthPermission
              "createLoginContext";
        permission javax.security.auth.AuthPermission
              "getLoginConfiguration";
        permission javax.security.auth.AuthPermission
              "setLoginConfiguration";

        // Secure Sockets Layer (SSL) permission (for remote manager)
        permission javax.net.ssl.SSLPermission  "getSSLSessionContext";
    };
```

```
// Nexaweb server codebase permissions
grant codeBase "file:${jboss.server.home.dir}/deploy/Nexaweb.ear" {
      // File permissions
      permission java.io.FilePermission "${user.home}", "read, write";
      permission java.io.FilePermission
            "${jboss.server.home.dir}\\tmp\\-", "read";
      //permission java.io.FilePermission "${nexaweb.home}\\-", "read";

      // Property permissions
      permission java.util.PropertyPermission "*", "read,write";

      // Run-time permissions
      // Nexaweb server needs permissions to create its own class loader,
      // get the class loader, and so on
      permission java.lang.RuntimePermission "createClassLoader";
      permission java.lang.RuntimePermission "getClassLoader";
      permission java.lang.RuntimePermission "setContextClassLoader";
      permission java.lang.RuntimePermission  "setFactory";

      // Nexaweb server security permissions to load the Cryptix
      // extension
      permission java.security.SecurityPermission
            "insertProvider.Cryptix";

      // Socket permissions
      // Permissions on all non-privileged ports.
      permission java.net.SocketPermission "*:1024-",
            "listen, connect, resolve";

      // Security permissions
      permission javax.security.auth.AuthPermission "doAs";
      permission javax.security.auth.AuthPermission "modifyPrincipals";
      permission javax.security.auth.AuthPermission
            "createLoginContext";
};

// The following are permissions given to codebase in the OIM server
// directory
grant codeBase "file:${XL.HomeDir}/-" {
      // File permissions
      permission java.io.FilePermission "${XL.HomeDir}\\config\\-",
            "read";
      permission java.io.FilePermission "${XL.HomeDir}\\JavaTasks\\-",
            "read";
      permission java.io.FilePermission
            "${XL.HomeDir}\\ScheduleTasks\\-", "read";
      permission java.io.FilePermission
            "${XL.HomeDir}\\ThirdParty\\-", "read";
      permission java.io.FilePermission
            "${XL.HomeDir}\\adapters\\-", "read,write,delete";
      permission java.io.FilePermission
            "${jboss.server.home.dir}\\tmp\\-", "read";
      //permission java.io.FilePermission "${nexaweb.home}\\-", "read";

      // Socket permissions
      permission java.net.SocketPermission "*:1024-", "listen";

      // Property permissions
```

```
        // Read XL.* and log4j.* properties
        permission java.util.PropertyPermission "XL.*", "read";
        permission java.util.PropertyPermission "log*", "read";

        // Security permissions
        permission javax.security.auth.AuthPermission "doAs";
        permission javax.security.auth.AuthPermission "modifyPrincipals";
        permission javax.security.auth.AuthPermission "createLoginContext";
    };

    // Minimal permissions are allowed to everyone else
    grant {
       permission java.util.PropertyPermission "*", "read";
       permission java.lang.RuntimePermission "queuePrintJob";
       permission java.net.SocketPermission "*", "connect";
       permission java.lang.RuntimePermission "accessClassInPackage.*";
       permission java.lang.RuntimePermission
            "org.jboss.security.SecurityAssociation.getSubject";
       permission javax.management.MBeanServerPermission "findMBeanServer";
       permission javax.management.MBeanPermission

"org.jboss.mx.modelmbean.XMBean#*[JMImplementation:type=MBeanRegistry]", "*";
       permission javax.security.auth.AuthPermission "createLoginContext.*";

       permission java.io.FilePermission
            "${jboss.server.home.dir}\\tmp\\-", "read,write";

       // For Nexaweb
       permission java.lang.RuntimePermission "getClassLoader";
       permission java.lang.RuntimePermission "setContextClassLoader";
       permission java.util.PropertyPermission "nexaweb.logs", "read,write";
       permission java.util.PropertyPermission
            "sun.net.client.defaultConnectTimeout", "read,write";
       permission java.util.PropertyPermission
            "sun.net.client.defaultReadTimeout", "read,write";

       permission java.lang.RuntimePermission "loadLibrary.*";
       permission java.lang.RuntimePermission "queuePrintJob";
       permission java.net.SocketPermission    "*", "connect";
       permission java.io.FilePermission       "<<ALL FILES>>", "read,write";
       permission java.lang.RuntimePermission   "modifyThreadGroup";

    };
```

Note: To reflect the changes in the code and apply Java 2 Security, you must restart the server.

Index

K

L

N

O

P

R

S

T

X