Oracle® Retail Store Inventory Management

Installation Guide
Release 16.0.3
F28523-02

October 2020

ORACLE

Oracle® Retail Store Inventory Management Installation Guide, Release 16.0.3

Contributors: Nathan Young, Shreyas S Manipura

Value-Added Reseller (VAR) Language

Oracle Retail VAR Applications

The following restrictions and provisions only apply to the programs referred to in this section and licensed to you. You acknowledge that the programs may contain third party software (VAR applications) licensed to Oracle. Depending upon your product and its version number, the VAR applications may include:

(i) the **MicroStrategy** Components developed and licensed by MicroStrategy Services Corporation (MicroStrategy) of McLean, Virginia to Oracle and imbedded in the MicroStrategy for Oracle Retail Data Warehouse and MicroStrategy for Oracle Retail Planning & Optimization applications.

(ii) the **Wavelink** component developed and licensed by Wavelink Corporation (Wavelink) of Kirkland, Washington, to Oracle and imbedded in Oracle Retail Mobile Store Inventory Management.

(iii) the software component known as **Access Via™** licensed by Access Via of Seattle, Washington, and imbedded in Oracle Retail Signs and Oracle Retail Labels and Tags.

(iv) the software component known as **Adobe Flex™** licensed by Adobe Systems Incorporated of San Jose, California, and imbedded in Oracle Retail Promotion Planning & Optimization application.

You acknowledge and confirm that Oracle grants you use of only the object code of the VAR Applications. Oracle will not deliver source code to the VAR Applications to you. Notwithstanding any other term or condition of the agreement and this ordering document, you shall not cause or permit alteration of any VAR Applications. For purposes of this section, "alteration" refers to all alterations, translations, upgrades, enhancements, customizations or modifications of all or any portion of the VAR Applications including all reconfigurations, reassembly or reverse assembly, re-engineering or reverse engineering and recompilations or reverse compilations of the VAR Applications or any derivatives of the VAR Applications. You acknowledge that it shall be a breach of the agreement to utilize the relationship, and/or confidential information of the VAR Applications for purposes of competitive discovery.

The VAR Applications contain trade secrets of Oracle and Oracle's licensors and Customer shall not attempt, cause, or permit the alteration, decompilation, reverse engineering, disassembly or other reduction of the VAR Applications to a human perceivable form. Oracle reserves the right to replace, with functional equivalent software, any of the VAR Applications in future releases of the applicable program.

Contents

Send Us Your Comments ..... **ix**

Preface ..... **xi**

Audience xi

Customer Support xi

Review Patch Documentation xi

Improved Process for Oracle Retail Documentation Corrections xii

Oracle Retail Documentation on the Oracle Help Center docs.oracle.com xii

Conventions xii

1 Preinstallation Tasks **1**

Implementation Capacity Planning 1

Check Supported Database Server Requirements 2

Check Supported Application Server Requirements 3

Check Single Sign-On Requirements 4

Check Directory Server Requirements 4

Check Third-Party Software Dependencies 4

Check Client PC and Web Browser Requirements 4

Supported Oracle Retail Products 5

UNIX User Account Privileges to Install the Software 5

SIM Installation Overview 5

2 RAC and Clustering **7**

3 Database Installation Tasks **9**

Expand the SIM Database Schema Installer Distribution 9

Required Database Character Set 9

Required Tablespaces 9

Create the SIM Database User 10

Run the SIM Database Schema Installer 12

Grant Privileges for the Roles 12

Create Synonyms for Additional users 13

Resolving Errors Encountered During Database Schema Installation 13

Data Seeding 14

Prerequisite 14

Data Seeding Setup 14

Export Seed Data 15

Ftp Export Data Files 15

Import Seed Data 15

4 Database Installation Tasks – Upgrade **17**

Expand the SIM Database Schema Installer Distribution 17

Required Database Character Set 17

Run the SIM Database Schema Upgrade 18

Set Environment Variables 18
Run the Upgrade Scripts 18
Compile Invalid Objects 18
Check Logs 18
Grant Privileges for the Roles 18
Create Synonyms for Additional Users 18
5 Application Installation Tasks 20
Middleware Infrastructure and Weblogic Server12c (12.2.1.4.0) Installation 20
Install RCU Database Schemas 25
Create a New ADF Domain (with managed server and EM) 33
Start the Node Manager 46
Start the AdminServer (admin console) 46
Start the Managed Server 47
Configuration of OID LDAP Provider in Weblogic Domain: 47
Verify OID Authenticator 53
Clustered Installations – Pre-Installation Steps 53
Expand the SIM Application Distribution 54
Loading SIM LDIFs into the OID 55
Set the Environment Variables for the SIM Installer 55
Clustered Installations – Post-Installation Steps 56
SIM Database Authentication Provider set up (to be done after the application deploy) 57
Review and/or Configure Oracle Single Sign-On 60
Create the SIM SSO provider in the SIMDomain 60
SIM Batch Scripts 61
Resolving Errors Encountered During Application Installation 62
Web Help Files 62
Starting and Stopping the Wavelink Server 62
6 Test the SIM Application 65
A Appendix: SIM Database Schema Installer Screens 67
B Appendix: SIM Application WebLogic Server Installer Screens 73
C Appendix: Common Installation Errors 139
EJB Deployment Errors during Installation to WebLogic 139
Output Freezes during Text Mode Installation to WebLogic 139
Database Installer Hangs on Startup 140
Warning: Could not create system preferences directory 140
Warning: Couldn't find X Input Context 140
ConcurrentModificationException in Installer GUI 141
A Second Login Screen Appears After Single Sign-On Login 141
Error Connecting to Database URL 142
GUI screens fail to open when running Installer 142

Log in fails with invalid username/password or user unauthorized errors.....142
D Appendix: Setting up SIM Reports/Tickets in BI Publisher..... 143
BiPublisher 12c – BI Server Component Installation Tasks.....143
BiPublisher 12c only - Installation Process Overview 143
Post install steps for BiPublisher 12C.....143
Installing the SIM BI Publisher Templates 149
Configuring the SIM JDBC connection 149
Verify Oracle BI Publisher Set Up for SIM Reports 151
Configuring SIM for CUPS printers using BIPublisher 12c 152
E Appendix: Single Sign-On for WebLogic 155
What Do I Need for Single Sign-On? 155
Can Oracle Access Manager Work with Other SSO Implementations? 155
Oracle Single Sign-on Terms and Definitions 156
What Single Sign-On is not.....157
How Oracle Single Sign-On Works 157
Installation Overview 158
User Management.....158
F Appendix: Setting Up Password Stores with wallets/credential stores..... 161
About Database Password Stores and Oracle Wallet 161
Setting Up Password Stores for Database User Accounts.....162
Setting up Wallets for Database User Accounts 163
For RMS, RWMS, RPM Batch using sqlplus or sqlldr, RETL, RMS, RWMS, and ARI 163
Setting up RETL Wallets 165
For Java Applications (SIM, ReIM, RPM, RIB, AIP, Alloc, ReSA, RETL).....166
How does the Wallet Relate to the Application? 169
How does the Wallet Relate to Java Batch Program use?.....169
Database Credential Store Administration.....169
Managing Credentials with WSLT/OPSS Scripts 171
listCred 172
updateCred 173
createCred 174
deleteCred 174
modifyBootStrapCredential 174
addBootStrapCredential 176
Quick Guide for Retail Password Stores (db wallet, java wallet, DB credential stores) 177
G Appendix: Tablespace Creation 187
Non-Encrypted Tablespace Creation 187
Encrypted Tablespace Creation 187
Configure a Wallet.....187
Encryption at Tablespace Level 188

H Appendix: Database Parameter File 191
I Appendix: Installation Order 193
Enterprise Installation Order.....193

Send Us Your Comments

Oracle Retail Store Inventory Management, Installation Guide, Release 16.0.3

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document.

Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

Note: Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the Online Documentation available on the Oracle Help Center Web site. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: retail-doc_us@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at www.oracle.com.

Preface

Oracle Retail Installation Guides contain the requirements and procedures that are necessary for the retailer to install Oracle Retail products.

Audience

This Installation Guide is written for the following audiences:

- Database administrators (DBA)
- System analysts and designers
- Integrators and implementation staff

Customer Support

- To contact Oracle Customer Support, access My Oracle Support at the following URL:
- https://support.oracle.com
- When contacting Customer Support, please provide the following:
- Product version and program/module name
- Functional and technical description of the problem (include business impact)
- Detailed step-by-step instructions to re-create
- Exact error message received
- Screen shots of each step you take

Review Patch Documentation

When you install the application for the first time, you install either a base release (for example, 16.0) or a later patch release (for example, 16.0.3). If you are installing the base release or additional patch releases, read the documentation for all releases that have occurred since the base release before you begin installation. Documentation for patch releases can contain critical information related to the base release, as well as information about code changes since the base release.

Improved Process for Oracle Retail Documentation Corrections

To more quickly address critical corrections to Oracle Retail documentation content, Oracle Retail documentation may be republished whenever a critical correction is needed. For critical corrections, the republication of an Oracle Retail document may at times **not** be attached to a numbered software release; instead, the Oracle Retail document will simply be replaced on the Oracle Help Center Web site, or, in the case of Data Models, to the applicable My Oracle Support Documentation container where they reside.

This process will prevent delays in making critical corrections available to customers. For the customer, it means that before you begin installation, you must verify that you have the most recent version of the Oracle Retail documentation set. Oracle Retail documentation is available on the Oracle Help Center at the following URL:

`https://docs.oracle.com/en/industries/retail/index.html`

An updated version of the applicable Oracle Retail document is indicated by Oracle part number, as well as print date (month and year). An updated version uses the same part number, with a higher-numbered suffix. For example, part number E123456-**02** is an updated version of a document with part number E123456-**01**.

If a more recent version of a document is available, that version supersedes all previous versions.

Oracle Retail Documentation on the Oracle Help Center docs.oracle.com

Oracle Retail product documentation is available on the following web site:

`https://docs.oracle.com/en/industries/retail/index.html`

(Data Model documents are not available through docs.oracle.com. You can obtain them through My Oracle Support.)

Conventions

Navigate: This is a navigate statement. It tells you how to get to the start of the procedure and ends with a screen shot of the starting point and the statement "the Window Name window opens."

```
This is a code sample
    It is used to display examples of code
```

Note: In the images or examples below, user details / company name / address / email / telephone number represent a fictitious sample. Any similarity to actual persons, living or dead is purely coincidental and not intended in any manner.

1

Preinstallation Tasks

This chapter discusses the tasks to complete before installation.

> **Note:** Oracle Retail assumes that the retailer has applied all required fixes for supported compatible technologies.

Implementation Capacity Planning

There is significant complexity involved in the deployment of Oracle Retail applications, and capacity planning is site specific. Oracle Retail strongly suggests that before installation or implementation you engage your integrator (such as the Oracle Retail Consulting team) and hardware vendor to request a disk sizing and capacity planning effort.

Sizing estimates are based on a number of factors, including the following:

- Workload and peak concurrent users and batch transactions
- Hardware configuration and parameters
- Data scarcity
- Application features utilized
- Length of time history is retained

Additional considerations during this process include your high availability needs as well as your backup and recovery methods.

Check Supported Database Server Requirements

General Requirements for a database server running SIM include:

Supported on:	Versions Supported:
Database Server OS	OS certified with Oracle Database 12cR1 and 19c Enterprise Edition. Options are: ▪ Oracle Linux 6 and 7 for x86-64 (Actual hardware or Oracle virtual machine). ▪ Red Hat Enterprise Linux 6 and 7 for x86-64 (Actual hardware or Oracle virtual machine). ▪ AIX 7.1 (Actual hardware or LPARs) ▪ Solaris 11.x SPARC (Actual hardware or logical domains)
Database Server 12cR1	Oracle Database Enterprise Edition 12cR1 (12.1.0.2) with the following specifications: **Components:** ▪ Oracle Partitioning ▪ Examples CD **One-offs:** ▪ 20846438: ORA-600 [KKPAPXFORMFKK2KEY_1] WITH LIST PARTITION ▪ 19623450: MISSING JAVA CLASSES AFTER UPGRADE TO JDK 7 ▪ 20406840: PROC 12.1.0.2 THROWS ORA-600 [17998] WHEN PRECOMPILING BY 'OTHER' USER ▪ 20925154: ORA-39126: WORKER UNEXPECTED FATAL ERROR IN KUPW$WORKER GATHER_PARSE_ITEMS JAVA ▪ 19672263: Patch 19672263: GTT SESSION LEVEL STATISTICS RETURNS ORA-20006 **RAC only:** ▪ 21260431: APPSST 12C : GETTING ORA-4031 AFTER 12C UPGRADE ▪ 21373473: INSTANCE TERMINATED AS LMD0 AND LMD2 HUNG FOR MORE THAN 70 SECS **Other components:** ▪ Perl interpreter 5.0 or later ▪ X-Windows interface ▪ JDK 1.7
Database Server 19c	Oracle Database Enterprise Edition 19c (19.3.0.0) with the following components: **Components:** ▪ DB HOME ▪ Examples CD **Other components:** ▪ Perl interpreter 5.0 or later ▪ X-Windows interface ▪ JDK 1.8

Note on 12C JDK: By default, JDK is at 1.6. After installing the 12.1.0.2 binary, apply patch 19623450. Follow the instructions on Oracle Database Java Developer's Guide 12c Release 1 to upgrade JDK to 1.7. The Guide is available at: http://docs.oracle.com/database/121/JJDEV/chone.htm#JJDEV01000.

Check Supported Application Server Requirements

The SIM application must be deployed on Oracle WebLogic 12c (12.2.1.4) with ADF.

Note: SIM is certified to work with only Oracle Internet Directory LDAP server (OID), as specified in the Application Server Requirements section of the SIM Installation Guide. The sample, unsupported .ldif files that SIM includes are provided only as reference.

General requirements for an Oracle WebLogic Server capable of running the SIM application include the following.

Supported on:	Versions Supported:
Application Server OS	OS certified with Oracle Fusion Middleware 12.2.1.4 Options are: ▪ Oracle Linux 6 and 7 for x86-64 (Actual hardware or Oracle virtual machine). ▪ Red Hat Enterprise Linux 6 and 7 for x86-64 (Actual hardware or Oracle virtual machine). ▪ AIX 7.2 (Actual hardware or LPARs) ▪ Solaris 11.x SPARC (Actual hardware or logical domains)
Application Server	Oracle Fusion Middleware 12.2.1.4 **Components:** ▪ FMW 12.2.1.4 Infrastructure (WLS and ADF included) ▪ Oracle Identity Management 12.2.1.4 ▪ Oracle Enterprise Manager 12.2.1.4 **Note:** Oracle Internet Directory (OID) is the supported LDAP directory for Oracle Retail products. For alternate LDAP directories, refer to Oracle WebLogic documentation set. **Java:** ▪ JDK 1.8+ 64 bit **Optional (required for running reports)** ▪ BI Publisher 12.2.1.4 for legacy reports **Note:** OBIEE 12.2.1.4 release does not support standalone BI Publisher installation with new installation (Doc ID 2472158.1),so install OBIEE 12.2.1.3 and upgrade it to 12.2.1.4. Please refer OBIEE 12.2.1.4 Installation guide for compatible FMW infrastructure version - Oracle Business Intelligence Publisher 12.2.1.4.0 **Optional (required for SSO)** ▪ Oracle WebTier 12c (12.2.1.4) Oracle Access Manager (12.1.2.4)

Check Single Sign-On Requirements

If SIM is not being deployed in a Single Sign-On environment, skip this section.

If Single Sign-On is to be used, verify the Oracle Identity Management has been installed along with the components listed in the above Application Server requirements section. Verify the Oracle WebTier Server is registered with the Oracle Access Manager as a partner application.

Check Directory Server Requirements

SIM uses directory server based user authentication and searching. For LDAP, SIM is supported with the following directory servers:

- Oracle Identity Management 12.2.1.4

Check Third-Party Software Dependencies

- Oracle Retail Wireless Foundation Server, provided by Wavelink 5.x.

Check Client PC and Web Browser Requirements

Requirement	Versions
Operating system	Windows 10 **Note**: Oracle Retail assumes that the retailer has ensured its Operating System has been patched with all applicable Windows updates.
Oracle (Sun) Java Runtime Environment (JRE)	Java 8+
Browser	Microsoft Edge 44+ Microsoft Internet Explorer 11 Mozilla Firefox ESR 60+ Chrome 73+ The browser is used to launch the Java WebStart client.

Note: Oracle Retail does not recommend or support installations with less than 128 kb bandwidth available between the PC client and the data center. Limiting the client to less than 128 kb total available bandwidth causes unpredictable network utilization spikes, and performance of the client degrades below requirements established for the product. The 128 kb requirement provides reasonable, predictable performance and network utilization.

Supported Oracle Retail Products

The following Oracle Retail products can be integrated with SIM. Next to each product is an indication of whether it is required or optional for SIM to function properly:

- Retail Integration Bus (RIB) 16.0.3 and all subsequent patches and hot fixes – Required

  Although typically used to integrate SIM with RMS, RIB can also be used to integrate SIM with other merchandising systems.

  > **Note:** RIB requires custom modifications to use a merchandising system other than RMS.

- Retail Merchandising System (RMS) 16.0.3 – Optional
- Oracle Retail Price Management 16.0.3 – Optional
- Oracle Retail Xstore Suite 17.0.2 – Optional

  > **Note:** If integrating with Xstore Point of Service, SSL must be enabled for the SIM Webservices.

The above products can be installed before or after SIM. However, it is helpful to know the connection details for the other products ahead of time so that you can provide them to the SIM application installer, which will configure the connection points for you.

UNIX User Account Privileges to Install the Software

A UNIX user account is needed to install the software. The UNIX user that is used to install the software should have write access to the WebLogic server installation files.

For example, "oretail"

> **Note:** Installation steps will fail when trying to modify files under the WebLogic installation unless the user has write access.

SIM Installation Overview

The following basic steps are required to install and set up SIM for the first time.

1. Install the database (with or without RAC).
2. Install application server (WebLogic) if it has not been installed
3. Install the SIM database schema
4. Set role-based access control. See Chapter 3 of the *Oracle Retail Store Inventory Management Implementation Guide, Volume 1* for instructions.
5. Install the SIM application.
6. Run data-seeding from RMS (Applicable only if SIM integrate with RMS)

2

RAC and Clustering

The Oracle Retail Store inventory Management System has been validated to run in two configurations on Linux:

- Standalone WebLogic and Database installations
- Real Application Cluster Database and WebLogic Clustering

The Oracle Retail products have been validated against a 12.1.0.2 and/or a 19.3.0.0 RAC database. When using a RAC database, all JDBC connections should be configured to use THIN connections rather than OCI connections. Clustering for WebLogic Server 12.2.1.4.0 is managed as an Active-Active cluster accessed through a Load Balancer. Validation has been completed utilizing a RAC 12.1.0.2 and/or a 19.3.0.0 Oracle Internet Directory database with the WebLogic 12.2.1.4.0 cluster. It is suggested that a Web Tier 11.1.1.9 installation be configured to reflect all application server installations if SSO will be utilized.

References for Configuration:

- Oracle Fusion Middleware High Availability Guide, 12c Part Number E95492-01
- Oracle Real Application Clusters Administration and Deployment Guide 19c (19.3) E95728-06

3

Database Installation Tasks

This chapter describes the tasks required for a full database installation.

> **Note:** If the SIM 16.0.x software is already installed, please see **"Database Installation Tasks – Upgrade"** for information on Upgrading to SIM 16.0.3.

Expand the SIM Database Schema Installer Distribution

1. Log in to the UNIX server as a user which has sufficient access to run sqlplus from the Oracle Database installation.
2. Create a new staging directory for the SIM database schema installer distribution (sim16-db.zip). There should be a minimum of 50 MB disk space available for the database schema installation files. This location is referred to as INSTALL_DIR for the remainder of this chapter.
3. Copy sim16-db.zip to <INSTALL_DIR> and extract its contents. This creates a /sim/dbschema subdirectory under INSTALL_DIR.

Required Database Character Set

SIM 16.0.3 databases should be created with the AL32UTF8 database character set. This will ensure support for characters of all languages supported by SIM and ensure proper integration with other Oracle Retail applications.

Required Tablespaces

Before you run the SIM database schema installer, make sure that the required tablespaces have been created in the database As of Release 15, SIM has its own dedicated tablespaces. They are: SIM_DATA, SIM_INDEX, SIM_LOB_DATA, SIM_LOB_INDEX, SIM_ENCRYPTED_DATA, and SIM_ENCRYPTED_INDEX. The SIM_ENCRYPTED_DATA and SIM_ENCRYPTED_INDEX tablespaces hold data which may include Personally Identifiable Information data (PII Data). If you hold the Advanced Security Option license, you can choose to create these two tablespaces with TDE tablespace encryption to protect the PII data. If you do not hold an Advanced Security Option license, you can create the tablespaces as normal tablespaces, but with no encryption. The tablespace names must always be SIM_ENCRYPTED_DATA and SIM_ENCRYPTED_INDEX regardless of whether TDE encryption is used, because the table and index creation scripts look for these specific names.

1. Modify the paths of the script <INSTALL_DIR>/sim/dbschema/dbutils /create_tablespaces.sql. The table below shows the default initial sizes:

TABLESPACE_NAME	Size
SIM_ENCRYPTED_INDEX	12G
SIM_ENCRYPTED_DATA	10G
SIM_INDEX	10G

TABLESPACE_NAME	Size
SIM_DATA	8G
SIM_LOB_DATA	2G
SIM_LOB_INDEX	2G
USERS	2G

2. Once the script has been modified, execute it in SQL*Plus as sys.
   - For Example: SQL> @ create_tablespaces.sql
3. Review create_tablespaces.log for errors and correct as needed.
4. If you do not wish to use TDE tablespace encryption, follow below steps; or for TDE encryption skip to step 5.
   a. Modify the paths of the script <INSTALL_DIR>/sim/dbschema/dbutils /create_encrypted_tablespaces_no_TDE.sql as per your environment.
   b. Run the script using SQL*Plus as sys.
   c. Review Create_encrypted_tablespaces_no_TDE.log for errors and correct as needed.
5. If you hold an Advanced Security Option license and wish to use TDE tablespace encryption
   a. Modify the paths of the script <INSTALL_DIR>/sim/dbschema/dbutils /create_encrypted_tablespaces_TDE.sql as per your environment.
   b. Run the script using SQL*Plus as sys.
   c. Review Create_encrypted_tablespaces_TDE.log for errors and correct as needed.
   d. Refer to Appendix: Tablespace Creation for details about how to create tablespaces in an encrypted format.

Create the SIM Database User

The user in the database which will own the SIM tables (master schema) must be created prior to running the SIM database schema installer.

In addition to SIM database Schema owner, 6 required database users are also need to be created as application data source users, the data source user is granted to appropriate SIM database roles based on user responsibilities.

SIM database master schema and Data source users

Schema	Database Role granted
master schema , <master_user> Example: sim01	SIM database schema owner
Admin user, < master_schema >_ADM Example: sim01_adm	Roles granted: SIM_ADMIN, SIM_BUSINESS_VIEWER.
Security user, <master_schema>_SEC Example: sim01_sec	Roles granted: SIM_SECURITY, SIM_ADMIN_VIEWER
Business viewer, <master_schema>_BSV Example: sim01_bsv	Roles granted: SIM_BUSINESS_VIEWER, SIM_ADMIN_VIEWER

Schema	Database Role granted
Business user, <schema_owner>_BSI	Roles granted: SIM_BUSINESS
MPS user, <schema_owner>_MPS Example: sim01_mps	Roles granted: SIM_MPS (user granted to Message Processing Staging Role)
RIB user, <schema_owner>_RIB Example: sim01_rib	Roles granted: SIM_RIB (user granted to Retail Integration Bus Role)

1. Change the directory to <INSTALL_DIR>/sim/dbschema/dbutils/
2. Create a directory "log" for user creation spool files
   ```
   mkdir log
   ```
3. Create SIM database roles:
   ```
   SQL> @create_roles.sql
   ```
4. Create SIM database master schema owner:

   Note: The below user creation scripts take three arguments on the command line in sqlplus: username, password, and temporary tablespace.

   ```
   SQL> @create_user_sim_owner.sql
   ```
   Example username: sim01
5. Create additional db users as application data source users:

   In addition to SIM database Schema owner, SIM application uses 6 data source users to access to the appropriate database objects based on user responsibilities.

   Run following create data source user scripts to create required data source users, the script also grants the user to appropriate SIM database roles based on user responsibilities.

   ```
   SQL> @create_user_sim_admin.sql
   ```
   Example username: sim01_adm

   ```
   SQL> @create_user_sim_rib.sql
   ```
   Example username: sim01_rib

   ```
   SQL>@create_user_sim_business.sql
   ```
   Example username: sim01_bsi

   ```
   SQL> @create_user_sim_business_viewer.sql
   ```
   Example username: sim01_bsv

   ```
   SQL> @create_user_sim_mps.sql
   ```
   Example username: sim01_mps

```
SQL> @create_user_sim_security.sql
```

Example username: sim01_sec

> **Note:** The grant_privs_<role_name>.sql scripts and create_synonym.sql scripts are provided to grant database users to the proper privileges.

Run the SIM Database Schema Installer

This installer installs the SIM database schema, compile SIM objects, inserts SIM control data.

1. Set the following environment variables:
   - Set the ORACLE_HOME to point to an installation that contains sqlplus. It is recommended that this be the ORACLE_HOME of the SIM database.
   - Set the PATH to: $ORACLE_HOME/bin:$PATH
   - Set the ORACLE_SID to the name of your database
   - Set the NLS_LANG for proper locale and character encoding

     > **Example:** NLS_LANG=AMERICAN_AMERICA.AL32UTF8
2. If you are using an X server such as Exceed, set the DISPLAY environment variable so that you can run the installer in GUI mode (recommended). If you are not using an X server, or the GUI is too slow over your network, unset DISPLAY for text mode.
3. Run the install.sh script in <INSTALL_DIR>/ sim/dbschema path. This launches the installer. After installation is completed, a detailed installation log file is created: <INSTALL_DIR>/ sim/dbschema/logs/sim-install-db.<timestamp>.log.

   > **Note:** Appendix A contains details on every screen and field in the database schema installer.
4. When the installer finishes it prints the values of the database SID and database schema user. Note these values as they are needed later when you run the SIM application installer.

Grant Privileges for the Roles

SIM database objects are owned by SIM database master schema owner, and database object privileges are granted to database users via database roles.

SIM database installer grants SIM object privileges to SIM pre-defined database roles.

> **Note:** The grant_sim_role_privs.sql can be used to grant SIM owned object privileges to SIM database roles as follow:
>
> Change the directory to <INSTALL_DIR>/sim/dbschema/dbscripts/util
>
> SQL> *@grant_sim_role_privs.sql <master schema owner>*

Create Synonyms for Additional users

In addition to SIM database schema owner, six required data source users are also created as part of SIM database creation, the data source users are granted to the proper database roles based on the data source user responsibilities.

Change the directory to <INSTALL_DIR>/sim/dbschema/dbscripts. Run Create synonyms scripts as SIM database owner:

```
SQL>@util/create_synonym_list.sql  <schema_owner>
<list_of_synonym_schema>

For example:

SQL> @util/create_synonym_list.sql sim01
sim01_bsi,sim01_bsv,sim01_adm,sim01_sec,sim01_mps
```

Note: Alternatively, the synonym can be created individually by executing the create_synonym.sql for each user, the script will prompt enter the schema owner and synonym user:

For example:

To create synonym for user "sim01app"

@create_synonym.sql;

Enter schema owner: sim01

Enter synonym schema: sim01app

Resolving Errors Encountered During Database Schema Installation

If the database schema installer encounters any errors, it halts execution immediately and prints to the screen which SQL script it was running when the error occurred. It also writes the path to this script to the .dberrors file. When this happens, you must run that particular script using sqlplus. After you are able to complete execution of the script, delete the .dberrors file and run the installer again. You can run the installer in silent mode so that you do not have to retype the settings for your environment. See Appendix D of this document for instructions on silent mode.

See Appendix F of this document for a list of common installation errors.

Subsequent executions of the installer will skip the SQL scripts which have already been executed in previous installer runs. This is possible because the installer maintains a **.dbhistory** file with a listing of the SQL scripts that have been run. If you have dropped the SIM schema and want to start with a clean install, you can delete the .dbhistory file so that the installer runs through all of the scripts again. It is recommended that you allow the installer to skip the files that it has already run.

Data Seeding

After full fresh install SIM database schema and SIM application installation tasks completed.

> **Note:** Data seeding is only applicable for full SIM install.

The data seeding process seeds store foundation data from RMS into SIM, there are three steps:

1. Export seed data from RMS
2. Move seeded data files via SFTP (if needed) to location that SIM needs to run from
3. Import seed data into SIM

Prerequisite

A full SIM database install has been completed.

Data Seeding Setup

The SIM database installer extracts the data seeding scripts from the sim-database-data-seeding.zip to the following location:

STAGING_DIR/sim/dbschema/data_seeding

This folder is referred to as DATA_SEEDING_DIR for the remainder of this chapter.

Verify the directory and the file permissions:

The recommended permissions for data seeding directories are 775 (rwxrwxr-x).

Set the following environment variables:

- Set ORACLE_SID to the name of SIM database.

  Example:

  ```
  export ORACLE_SID=<SIM_DB_NAME>
  ```

- Set the ORACLE_HOME. It is recommended that this be the ORACLE_HOME of the SIM database.

  Example:

  ```
  export  ORACLE_HOME=/u00/oracle/product/19.3.0.0
  ```

- Set JAVA_HOME

  Example:

  ```
  export JAVA_HOME=  /path/java1.7+_64bit
  ```

Set NLS_LANG

  Example:

  ```
  export NLS_LANG=AMERICAN_AMERICA.AL32UTF8
  ```

- Set the PATH to: $ORACLE_HOME/bin:

  Example:

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

Export Seed Data

Export Foundation Data

```
startDataSeedCli.sh -a 2 -s <rmsDBServer> -p <port> -d <rmsDB>
```

Export Store Data

```
startDataSeedCli.sh -a 3 -s <rmsDBServer > -p <port> -d <rmsDB>
```

Check Export logs

The data seeding process writes master log files into <DATA_SEEDING_DIR>/log directory.

Please check following the master log files:

- export_foundation.log
- export_store.log
- data_seed_common.log

The master log files may have references to sub-process log files:

- <DATA_SEEDING_DIR>/export/foundation/log
- <DATA_SEEDING_DIR>/export/store/log

Zip Export Data Files

Zip the data directories to be uploaded to SIM data seeding import location.

Ftp Export Data Files

Move foundation & store data via SFTP (if needed) to location that SIM needs to run from.

This is implementation dependent step.

Import Seed Data

Login as SIM db schema owner (or SIM DBA), and perform following steps:

1. Put unzipped the data files under DATA_SEEDING_DIR.
2. Run Set Up.
   ```
   startDataSeedCli.sh -a 1 –s <simDBServer> -p <port> -d <simDB>
   ```
3. Run Import Foundation Data.
   ```
   startDataSeedCli.sh -a 4 –s <simDBServer> -p <port> -d <simDB>
   ```
4. Run Import Store Data.
   ```
   startDataSeedCli.sh -a 5 –s <simDBServer> -p <port> -d <simDB>
   ```
5. Run Cleanup.
   ```
   startDataSeedCli.sh -a 6 –s <simDBServer> -p <port> -d <simDB>
   ```
6. Check data seeding logs:

   The data seeding process writes master log files into <DATA_SEEDING_DIR>/log directory.

   Please check following the master log files:

   - import_foundtion.log
   - import_store.log
   - data_seed_common.log

The master log files may have references to sub-process log files:

- <DATA_SEEDING_DIR>/import/foundation/log
- <DATA_SEEDING_DIR>/import/store/log

7. Verify the seeding results files.

   The verification files are located at directory <DATA_SEEDING_DIR>/verify/out:

   - verify_foundation_data.out
   - verify_store_data.out
   - disabled_constraints.out

8. Check disabled database constraints:

   After inspecting the result files, resolve the problematic data. A database administrator will need to manually enable the disabled constraints which are reported.

9. Remove data files:

   After data seeding is finished and you are convinced that your data was correctly seeded, you can remove all data seeding files from <DATA_SEEDING_DIR>

4

Database Installation Tasks – Upgrade

If SIM 16.0.2 is already installed, it is possible to do a patch install from 16.0.2 to 16.0.3.

The upgrade scripts are developed based on the following assumptions:

- These scripts are provided as helper or guidance scripts.
- Industry-standard best practices are followed for schemas, data, upgrades, and migrations.
- Client customization to the base schema is not addressed by these scripts.
- Invalid data might cause referential constraints in invalid state. The bad data must to be handled manually and any invalid constraints must be resolved manually.
- The upgrade drops the database objects such as tables, views, constraints, packages and sequences that do not exist in the new schema. If these objects must be retained, the scripts must be modified accordingly.
- Due to the schema changes (for example, data types in columns), scripts perform validation. If the data type conversion causes errors, the data type change is not applied and is only logged. Manual intervention is required.
- System data is inserted as part of the upgrade scripts. There will be some system data that already exists and overlaps with the new system data. The overlap might cause errors due to unique key constraint.

Expand the SIM Database Schema Installer Distribution

1. Log in to the UNIX server as a user which has sufficient access to run sqlplus from the Oracle Database installation.
2. Create a new staging directory for the SIM database schema distribution (sim16-db.zip).There should be a minimum of 50 MB disk space available for the database schema installation files. This location is referred to as INSTALL_DIR for the remainder of this chapter.
3. Copy sim16-db.zip to <INSTALL_DIR> and extract its contents.
4. Change the directory to <INSTALL_DIR>sim/dbschema
5. Create a directory sim-database-delta, and copy sim-database-delta.zip to sim-database-delta directory
6. Change the directory to <INSTALL_DIR>/sim/dbschema/sim-database-delta, unzip sim-database-delta.zip

Required Database Character Set

SIM 16.0.3 databases should be created with the AL32UTF8 database character set. This will ensure support for characters of all languages supported by SIM and ensure proper integration with other Oracle Retail applications.

Run the SIM Database Schema Upgrade

Set Environment Variables

- Set the ORACLE_HOME to point to SQLPLUS executable
- Set the PATH to: $ORACLE_HOME/bin:$PATH
- Set the ORACLE_SID to the name of your database
- Set the NLS_LANG for proper locale and character encoding
  Example:
  NLS_LANG=AMERICAN_AMERICA.AL32UTF8

Run the Upgrade Scripts

Login database as SIM schema owner, run following script:

@run_all.sql;

Compile Invalid Objects

Login database as SIM schema owner, run following script:

@util/inv_obj_comp.sql;

Check Logs

Check upgrade.log and invalid_object.log files after completion; verify if there is any disabled constraint prior to proceed to next step.

Grant Privileges for the Roles

SIM database objects are owned by SIM database schema owner, and SIM database object privileges are granted to database users via database roles.

Login SQL*Plus as Schema owner,

SQL> @util/grant_sim_role_privs.sql <SIM db schema owner>

For example:

```
SQL> @util/grant_sim_role_privs.sql sim01
```

Note: After running the grant role privileges, remove the generated scripts from the generated_script directory.

Create Synonyms for Additional Users

In addition to SIM database schema owner, six required data source users are also created as part of SIM database creation, the data source users are granted to the proper database roles based on the data source user responsibilities.

Run Create synonyms scripts as SIM database owner:

```
SQL>@util/create_synonym_list.sql  <schema_owner>
<list_of_synonym_schema>
For example:
SQL> @create_synonym_list.sql sim01
sim01_bsi,sim01_bsv,sim01_adm,sim01_sec,sim01_mps
```

Note: Alternatively, the synonym can be created individually by executing the create_synonym.sql for each user, the script will prompt enter the schema owner and synonym user:

For example:

To create synonym for user “sim01app”

@create_synonym.sql;

Enter schema owner: sim01

Enter synonym schema: sim01app

5

Application Installation Tasks

Before proceeding, you must install Oracle WebLogic Server 12c with ADF and any patches listed in the Chapter 1 of this document. The Oracle Retail Store Inventory Management application is deployed to a WebLogic Managed server within the WebLogic installation. It is assumed Oracle Database has already been configured and loaded with the appropriate Store Inventory Management schemas for your installation.

Installing a separate domain is mandated. It can be called "SIMDomain" (or something similar) and will be used to install the managed servers. The ADF libraries should be extended to this domain and the Enterprise Manager application should be deployed.

> **Note:** If this domain is to be setup in a secure mode. Please set up WebLogic as SSL and refer to the SIM Security Guide for details on all items to change to be in secure mode. This would best be done before domain and application install. The domain example below is for unsecured setup.

Middleware Infrastructure and Weblogic Server12c (12.2.1.4.0) Installation

Create a directory to install the WebLogic (this will be the ORACLE_HOME):

Example: mkdir -p /u00/webadmin/products/wls_retail

1. Set the ORACLE_HOME, JAVA_HOME and DOMAIN_HOME environment variables:
   - ORACLE_HOME should point to your WebLogic installation.
   - JAVA_HOME should point to the Java JDK 1.8+. This is typically the same JDK which is being used by the WebLogic domain where application is getting installed.

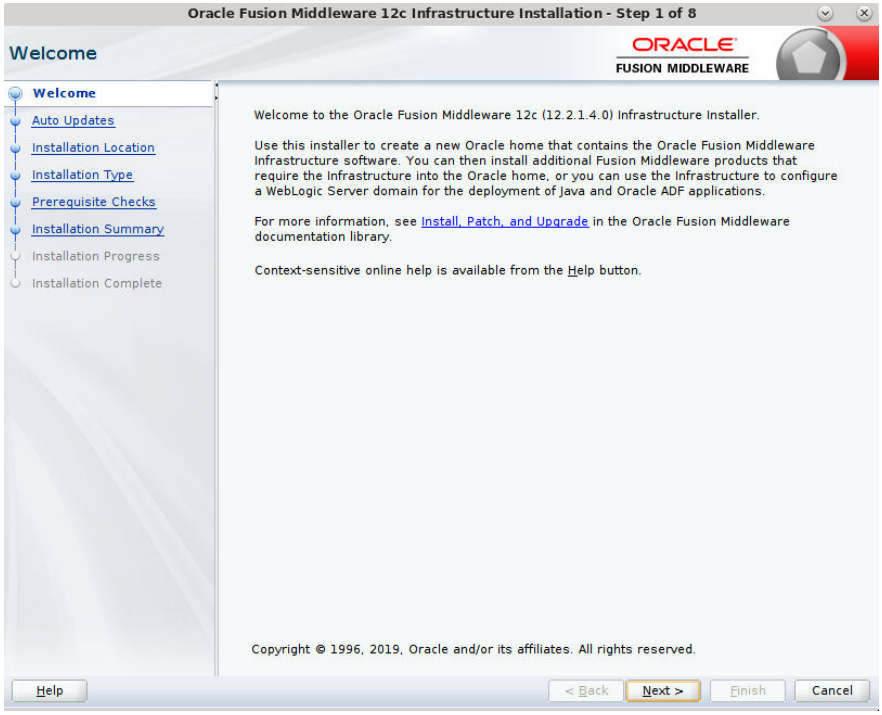
   Example:

   ```
   $export ORACLE_HOME=/u00/webadmin/products/wls_retail
   $export JAVA_HOME=/u00/webadmin/products/jdk_java
   (This should point to the Java which is installed on your server)
   $export PATH=$JAVA_HOME/bin:$PATH
   ```

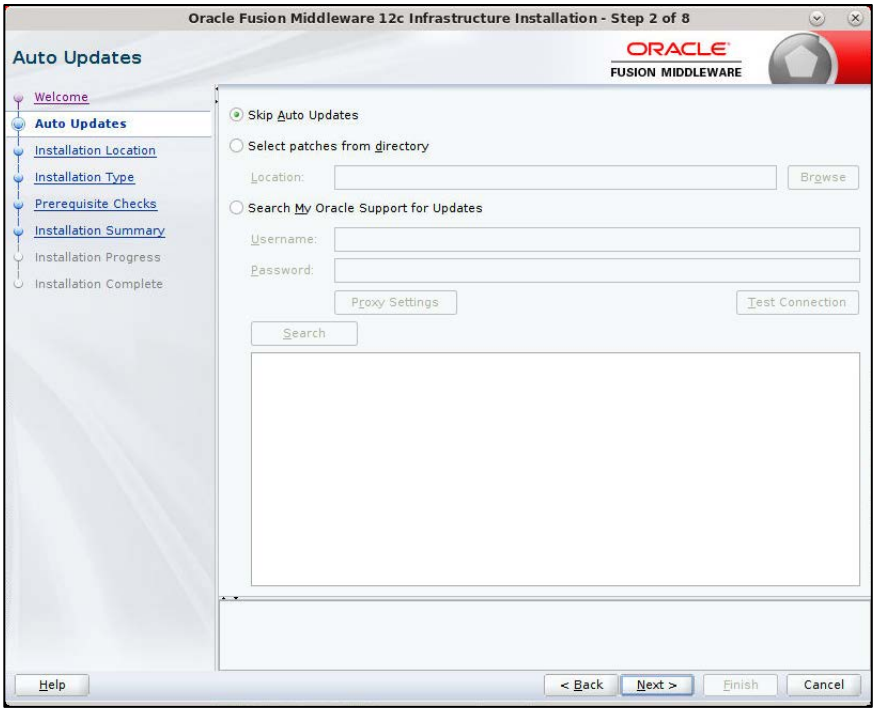
   Going forward we will use the above references for further installations.
2. Go to location where the weblogic jar is downloaded and run the installer using the following command:

   ```
   java -jar ./fmw_12.2.1.4.0_infrastructure.jar
   ```

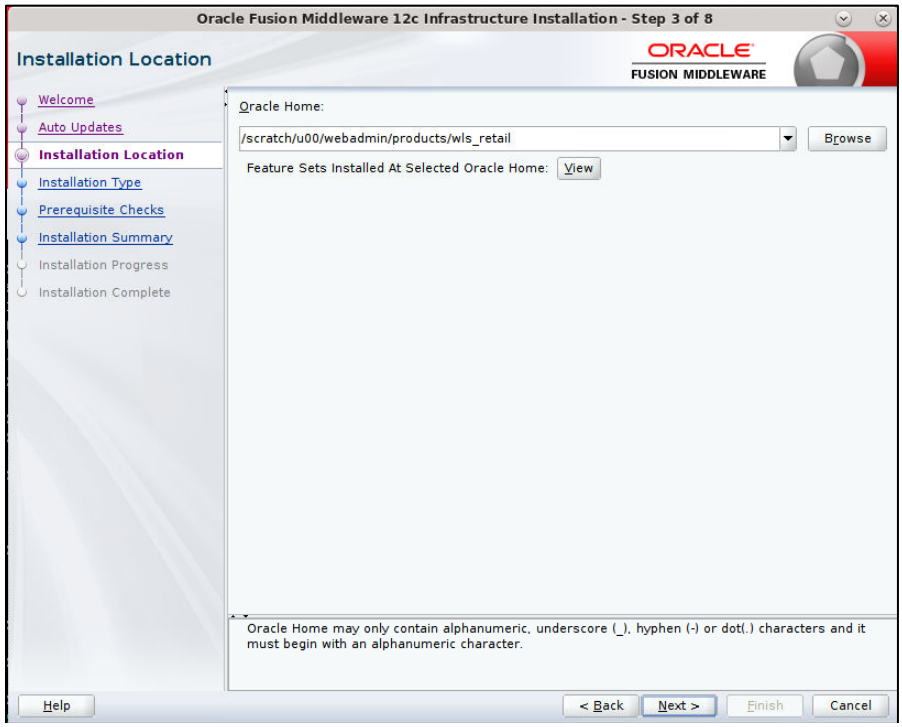
3. Welcome screen appears. Click **Next**.



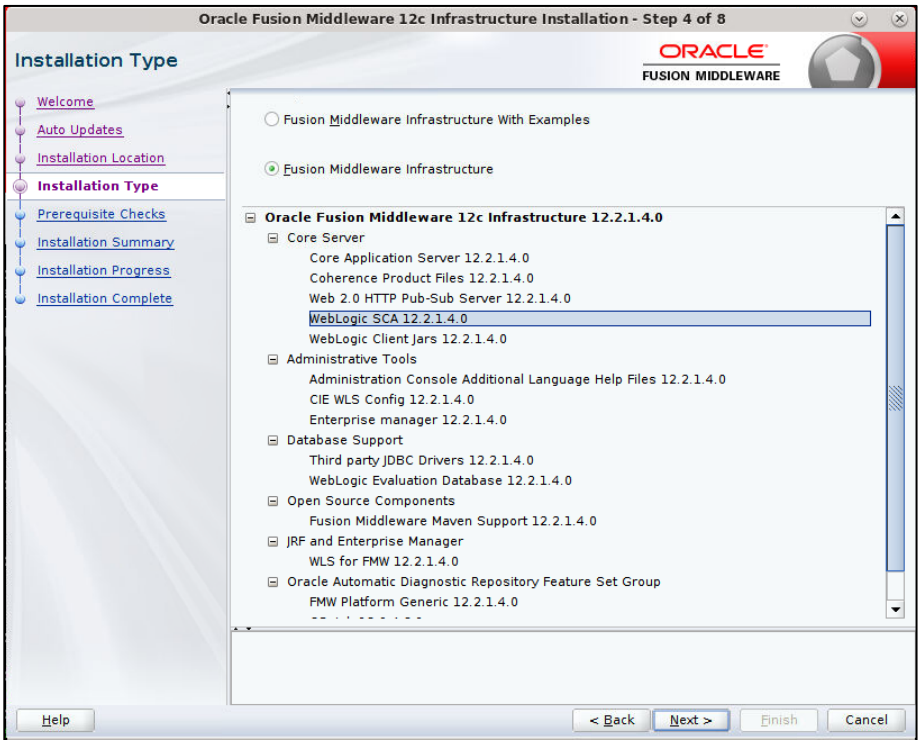
4. Click **Next**.



5. Enter the following and click **Next**.
   `Oracle home =<Path to the ORACLE_HOME>`
   Example:
   `/u00/webadmin/products/wls_retail`

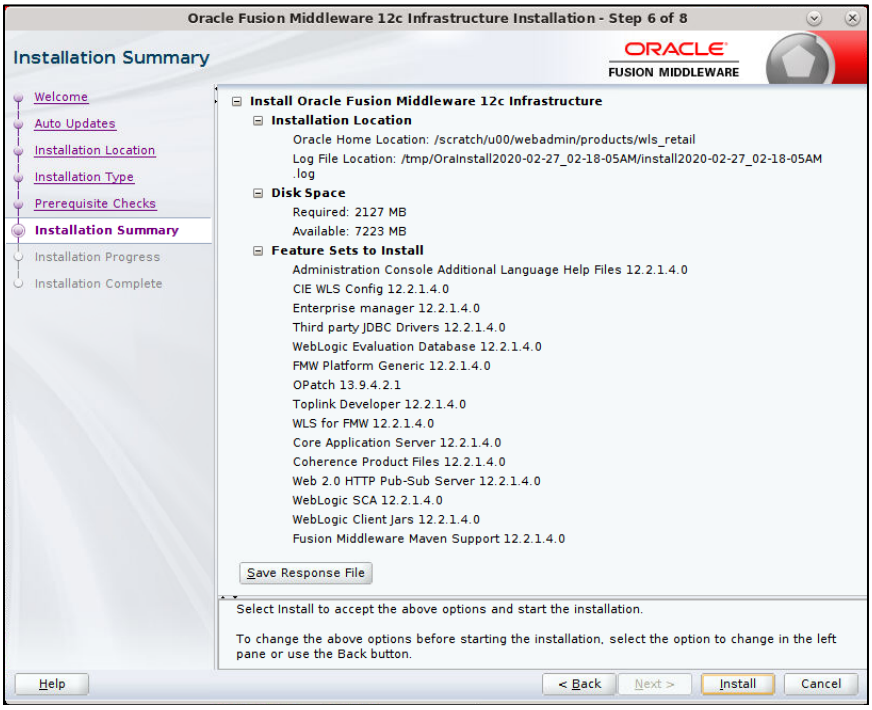


6. Select install type 'Fusion Middleware Infrastructure'. Click **Next**.

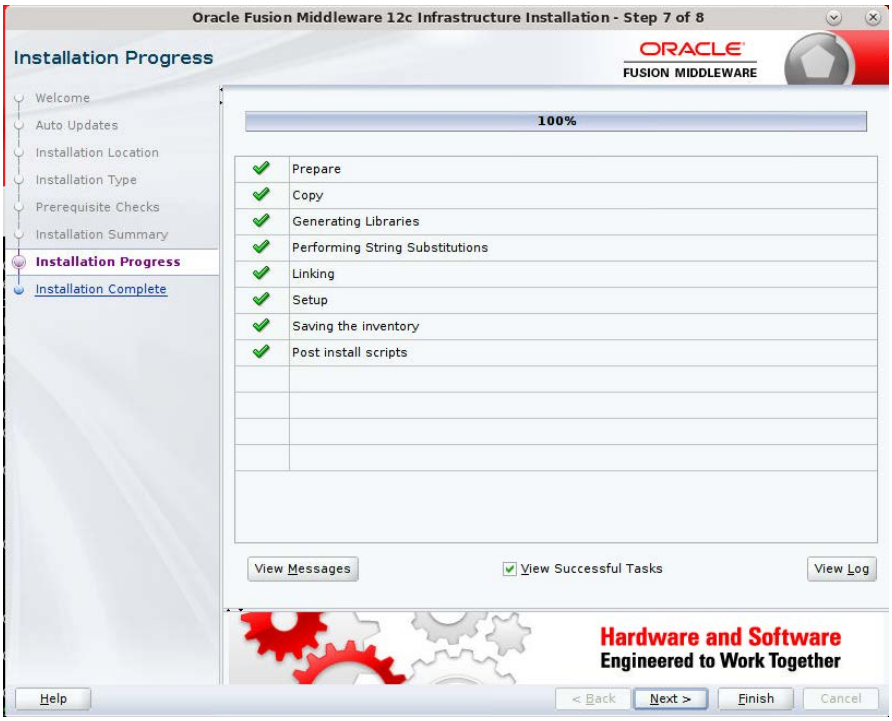


This screen will verify that the system meets the minimum necessary requirements.

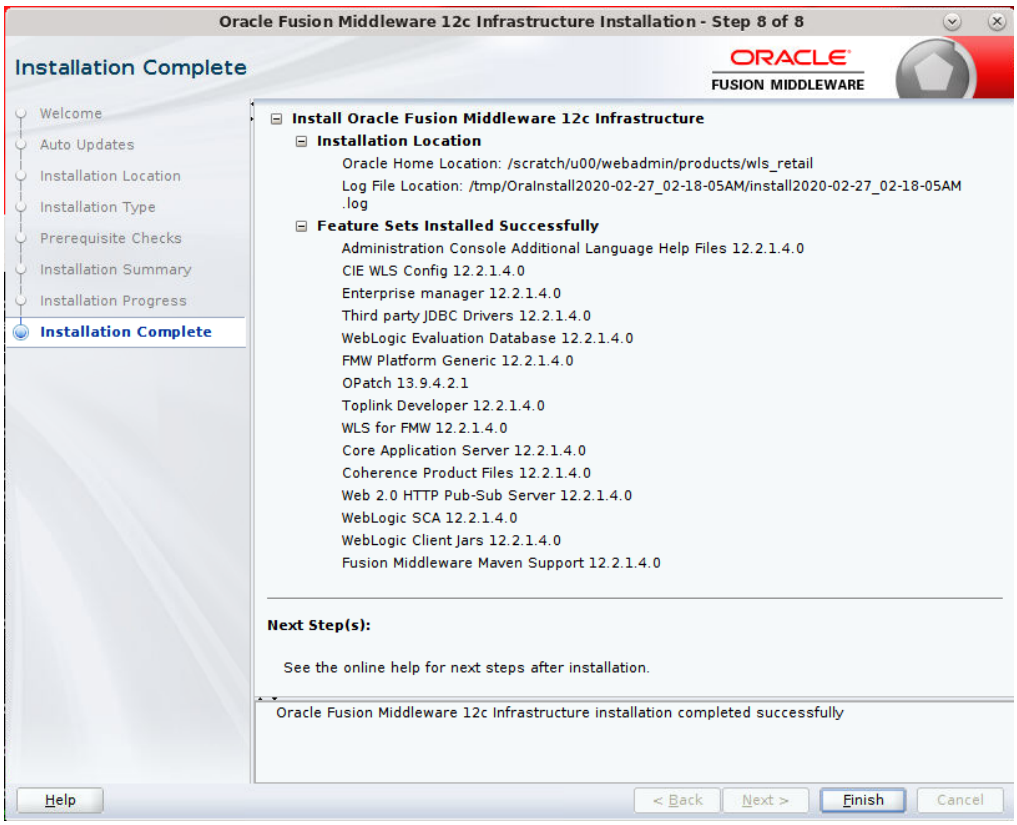
7. Click **Next**
8. Click **Install**.



9. Click Next



10. Click **Finish**.



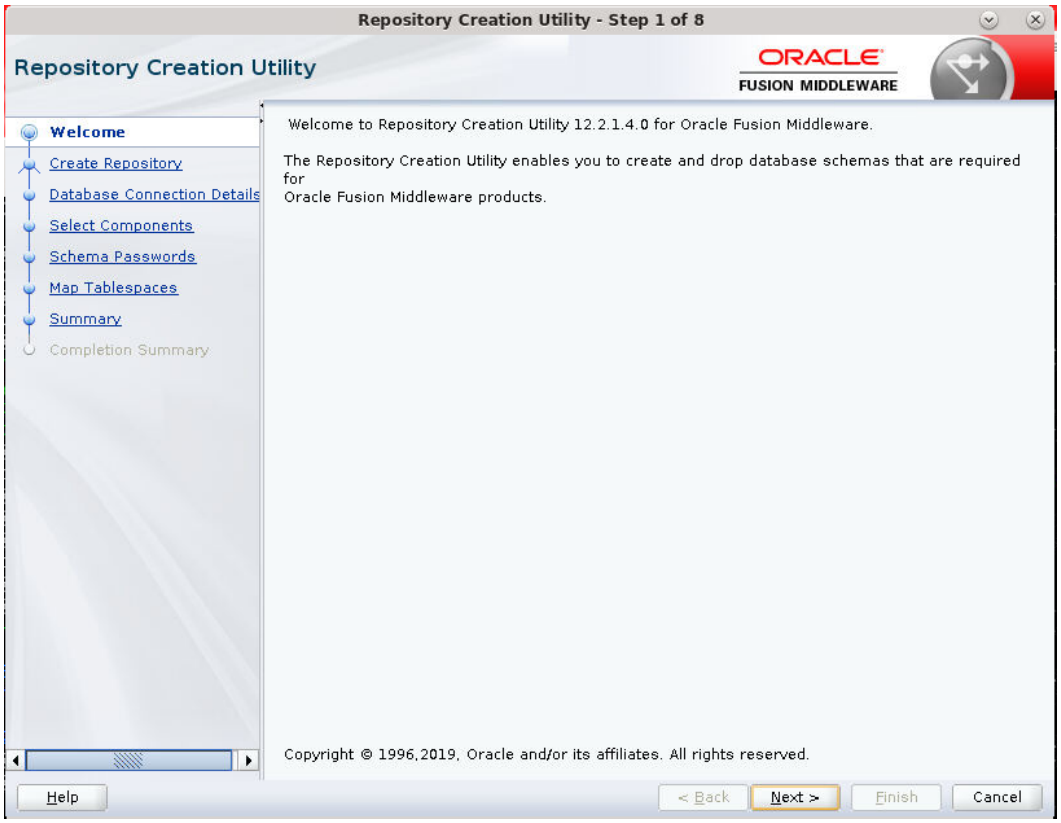
Install RCU Database Schemas

The RCU database schemas are required for the installation of configuration of domain and retail application.

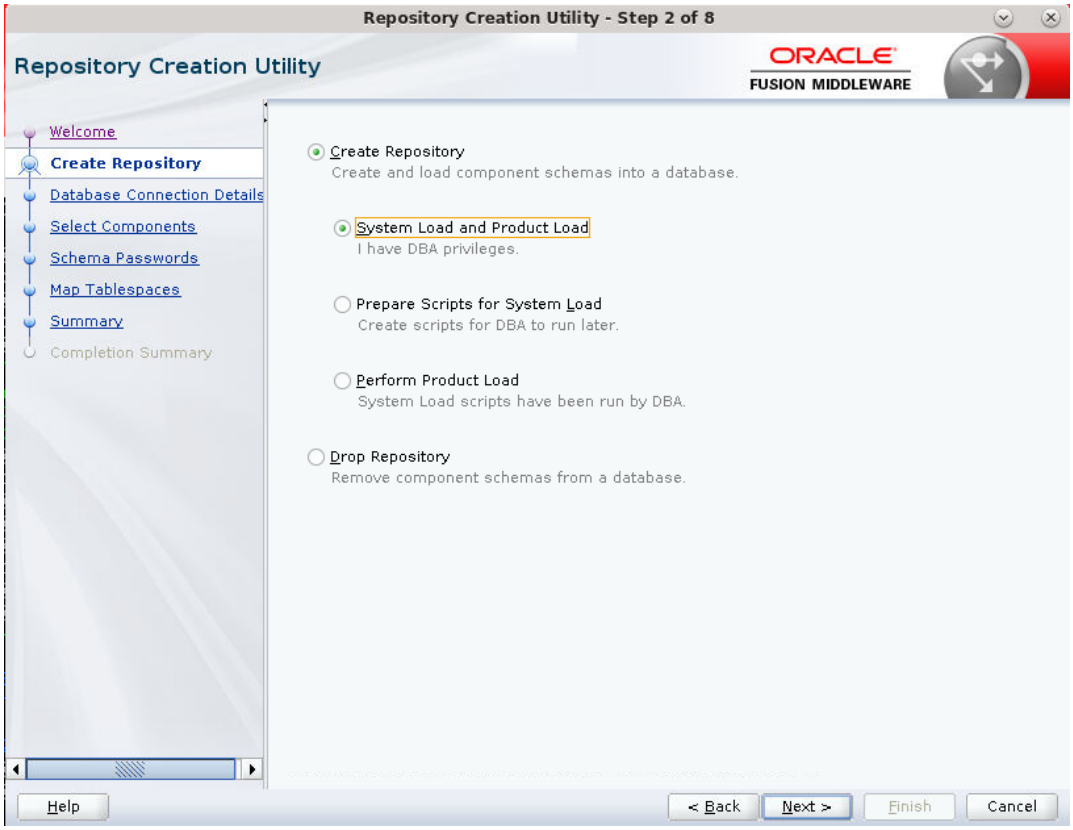
> **Note:** Need user which have sys admin privileges to install the RCU database schemas.

The following steps are provided for the creation of the database schemas:

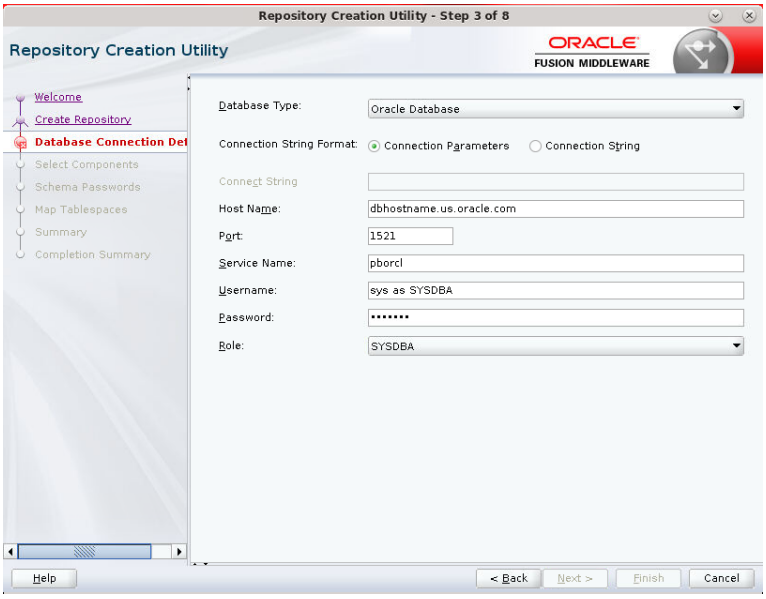
1. Navigate to the directory into which RCU is installed. For example:
   ```
   <ORACLE_HOME>/oracle_common/bin/
   Run "./rcu"
   ```
2. Click **Next**.



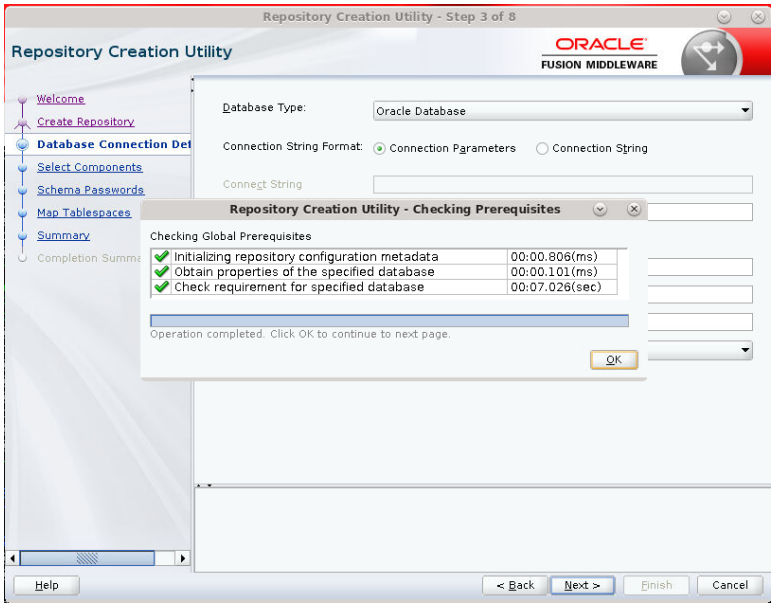
3. Select Create Repository and System Load and Product Load. Click **Next**.



4. Enter database connection details:
   - Database Type: Oracle Database
   - Host Name: dbhostname.us.oracle.com
   - Port: 1521
   - Service Name: dbservicename
   - Username: sys
   - Password: <syspassword>
   - Role: SYSDBA



5. Click **Next**. The Installer checks prerequisites.
6. When the prerequisite checks are complete, click **OK**. Click **Next**.



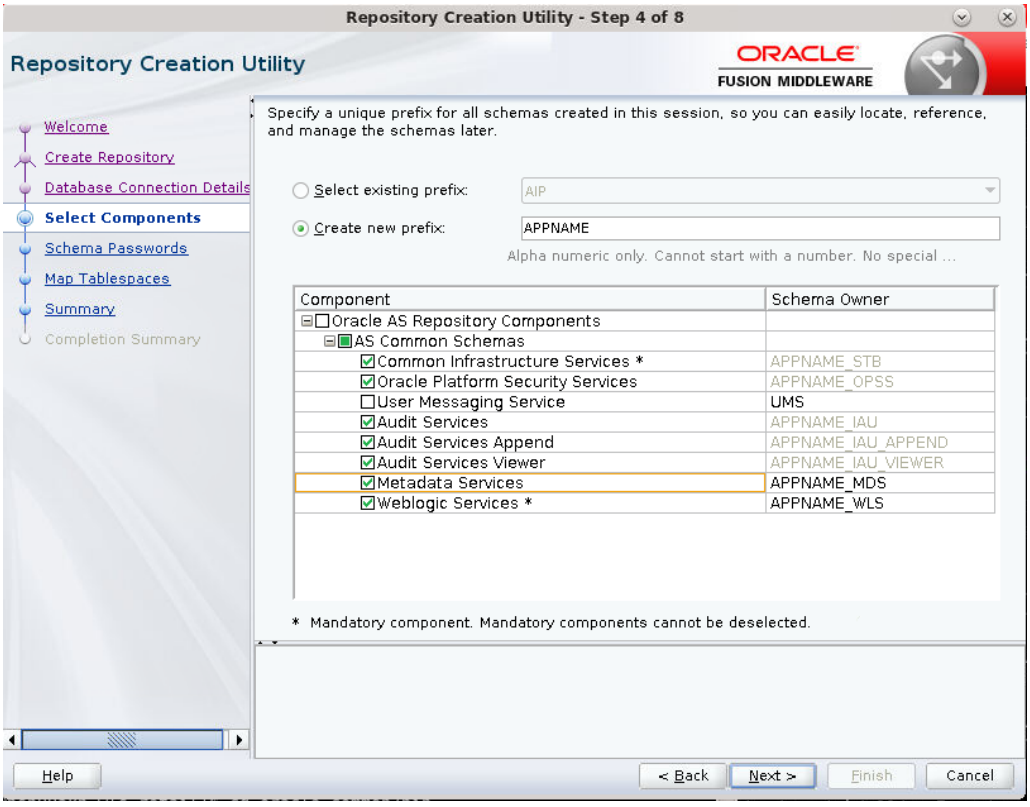
7. Click the **Create a new prefix** option, the prefix name for your schemas should be unique to your application environment.

   Example: ReIM, ALLOC, ReSA, and so on.

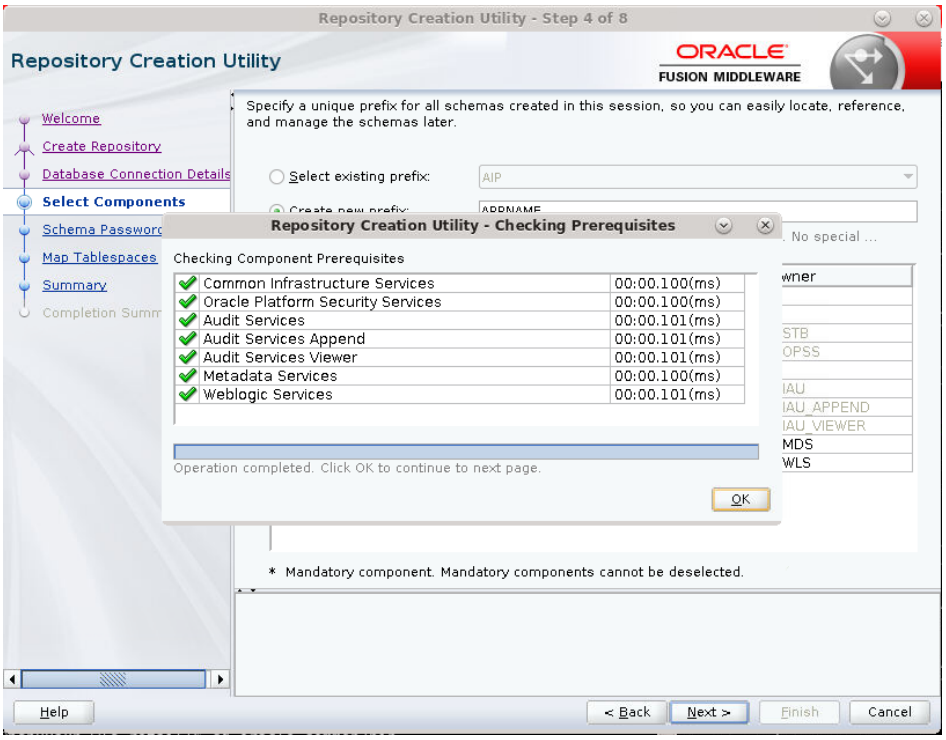
8. Select the components to create:
   - Meta Data Services
   - Oracle Platform Security Services

> **Note:** Once OPSS schema is selected, the following dependent schemas will get selected automatically.
>
> Audit Services
>
> Audit Services Append
>
> Audit Services Viewer

> **Note:** STB schema will be already selected as part of the Common Infrastructure component.

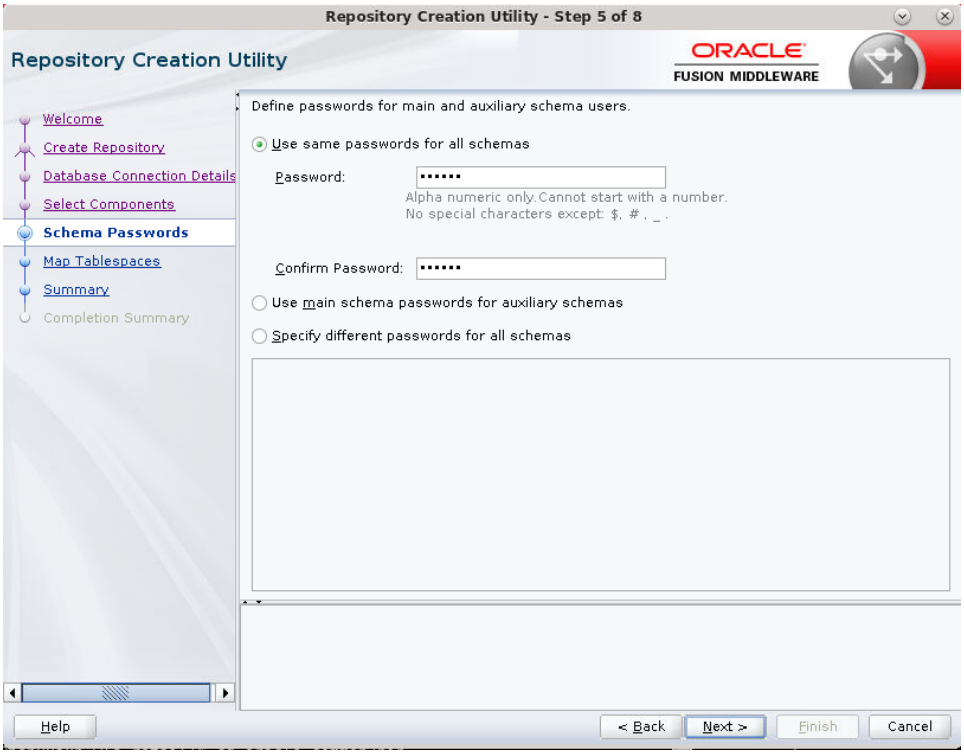


9. Click **Next**.

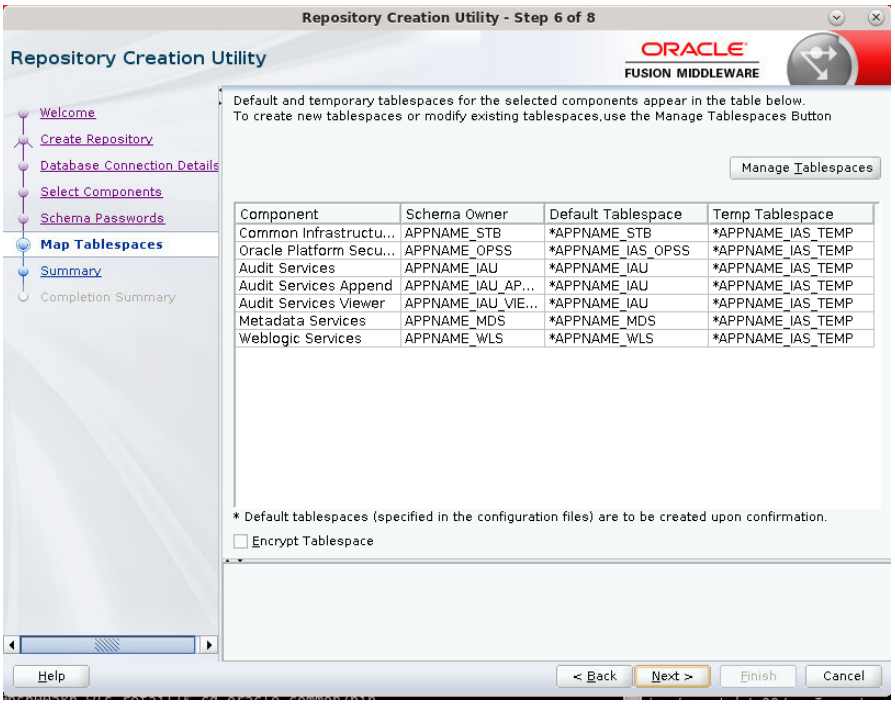


10. Enter password of your choice.

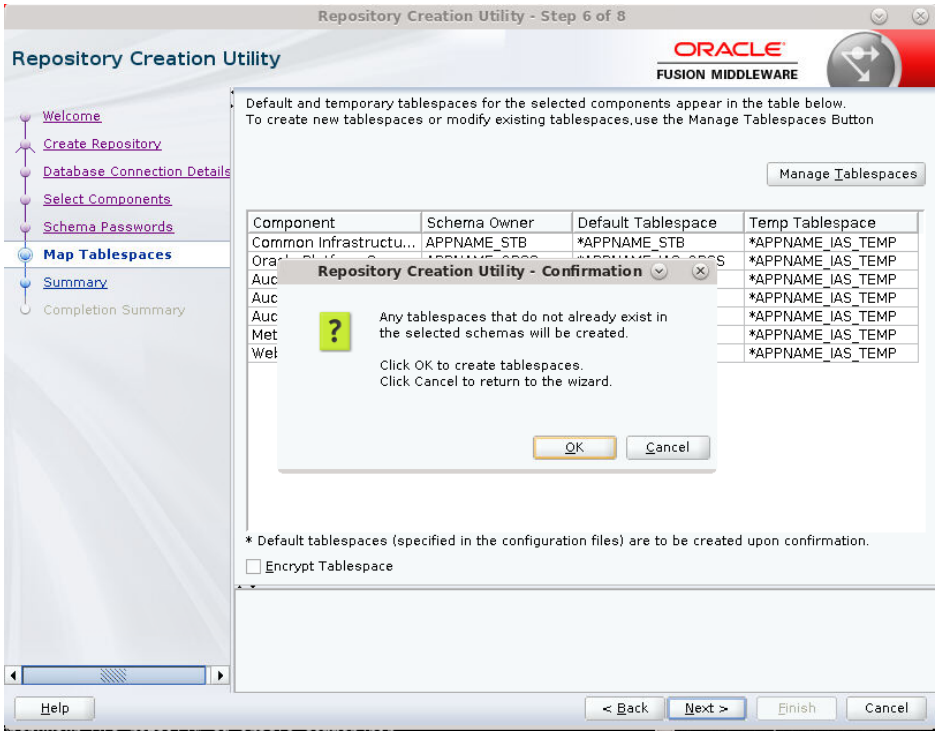
> **Note:** This password is needed at the time of ADF domain creation.



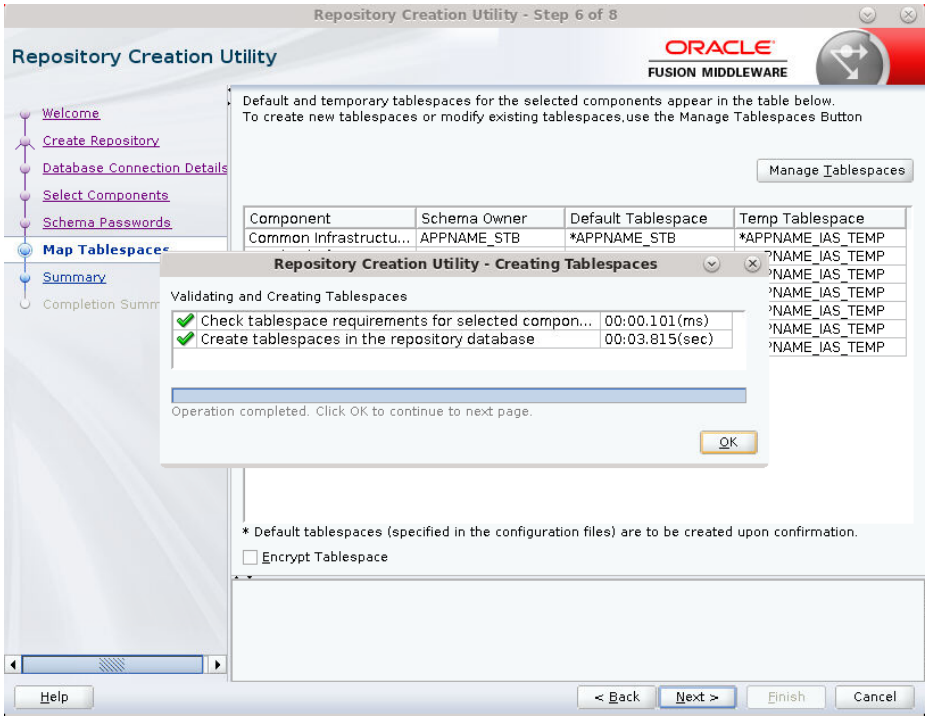
11. Provide the password and Click '**Next**'.



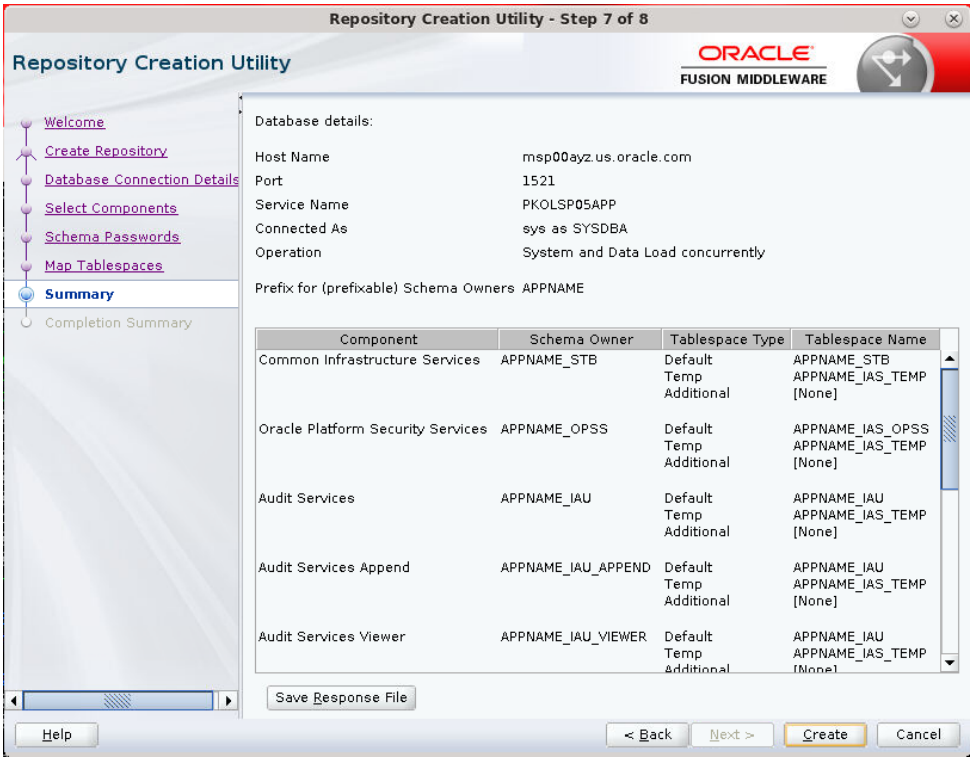
12. Click **Next**. A Repository Creation notification will appear. Click **OK**.



13. Tablespaces are created, and the progress will be displayed in a pop-up notification. When the operation is completed, click **OK**.

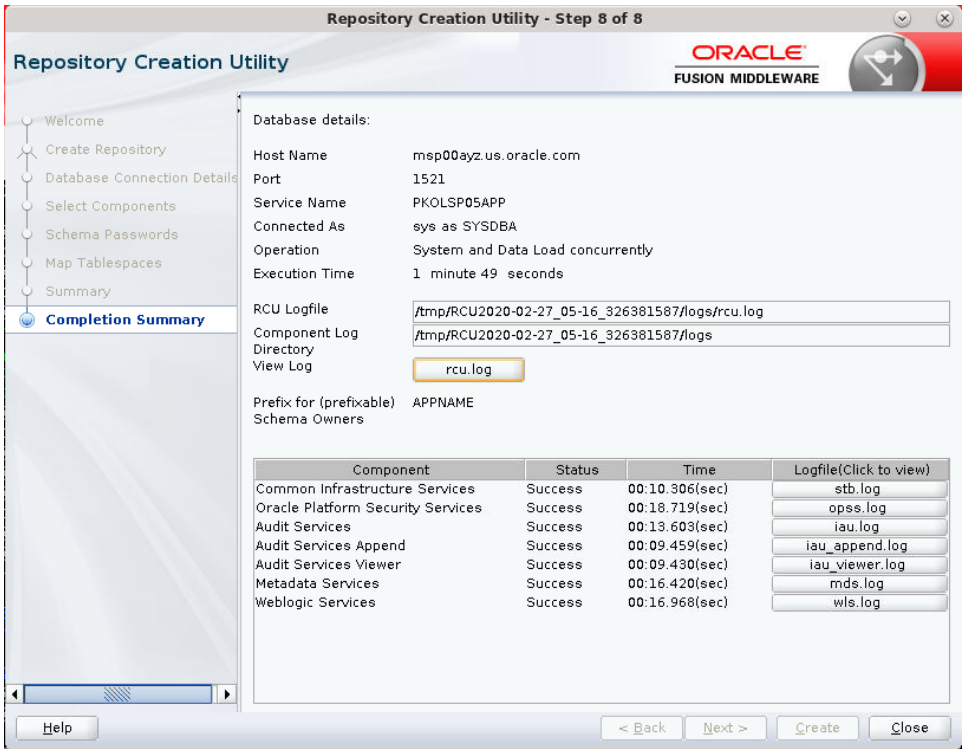


14. Click **Create**. The schema is created.



Upon successful creation of database schemas, a screen will appear with all the schemas created.

15. Click **Close**.



Create a New ADF Domain (with managed server and EM)

To create a new domain and managed server with ADF libraries and EM, follow the below steps:

1. Set the environment variables:

```
export JAVA_HOME=<JDK_HOME>
    (Example:/u00/webadmin/products/jdk_java) [JDK_HOME is the location where
jdk has been installed)
export PATH=$JAVA_HOME/bin:$PATH
export ORACLE_HOME=<ORACLE_HOME>/
 (Example:/u00/webadmin/products/wls_retail)

cd $ORACLE_HOME/oracle_common/common/bin
    (ORACLE_HOMEis the location where Weblogic has been installed.)
```

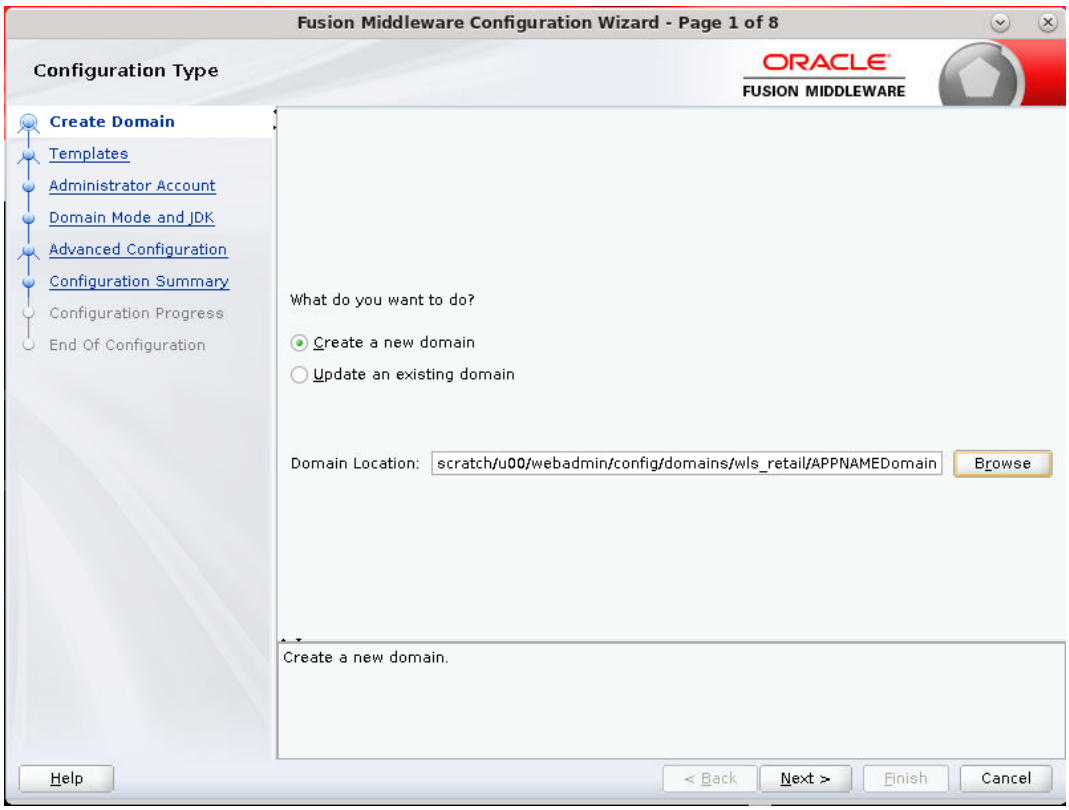
2. Run the following command:

   `./config.sh`

3. Select Create a new domain.

   Domain location: Specify the path to the <DOMAIN_HOME>
   Example:/u00/webadmin/config/domains/wls_retail/APPNAMEDomain

   Click **Next**.



4. Select Create Domain Using Product Templates.

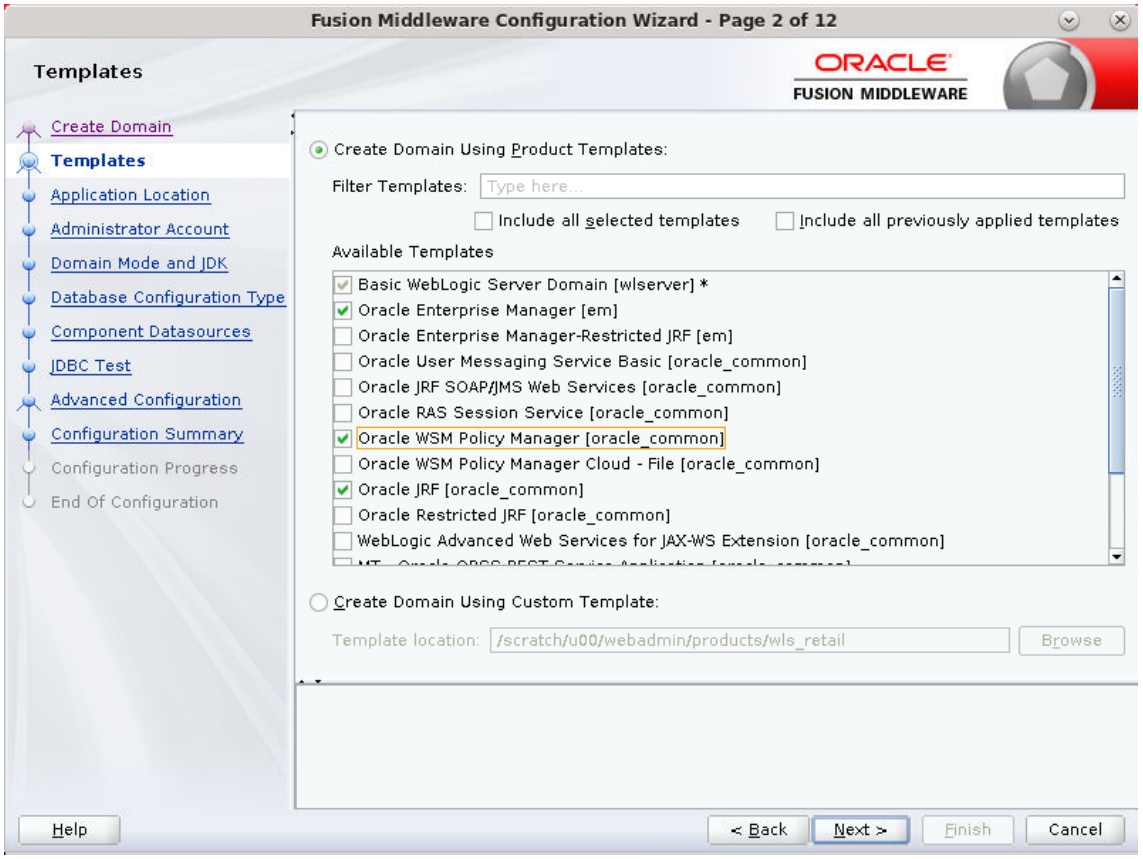
5. Check the following components:
   Oracle Enterprise Manager
   Oracle WSM Policy Manager

   Note: When Oracle Enterprise Manager Component is selected, the following dependent components are selected automatically:

   Oracle JRF

   Weblogic Coherence Cluster Extension

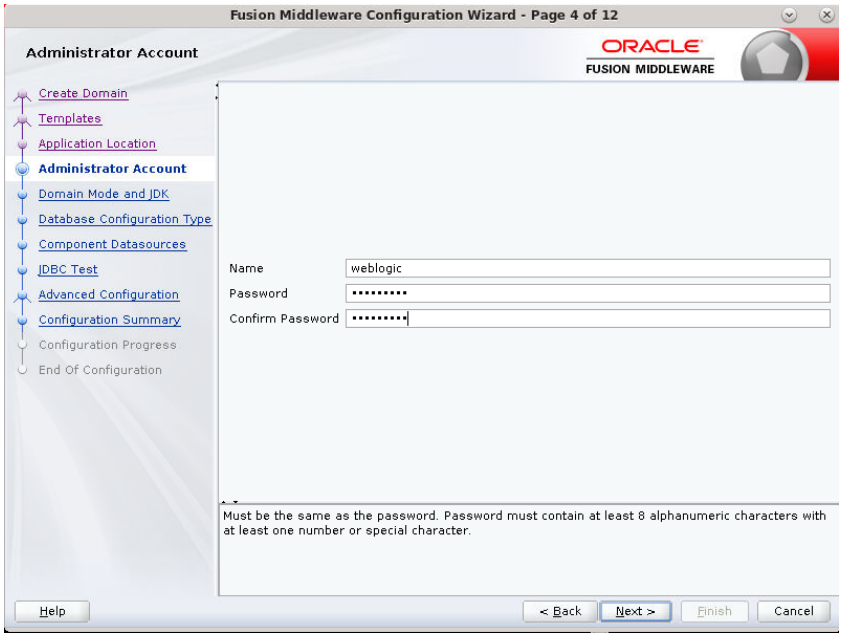
6. Click **Next**.



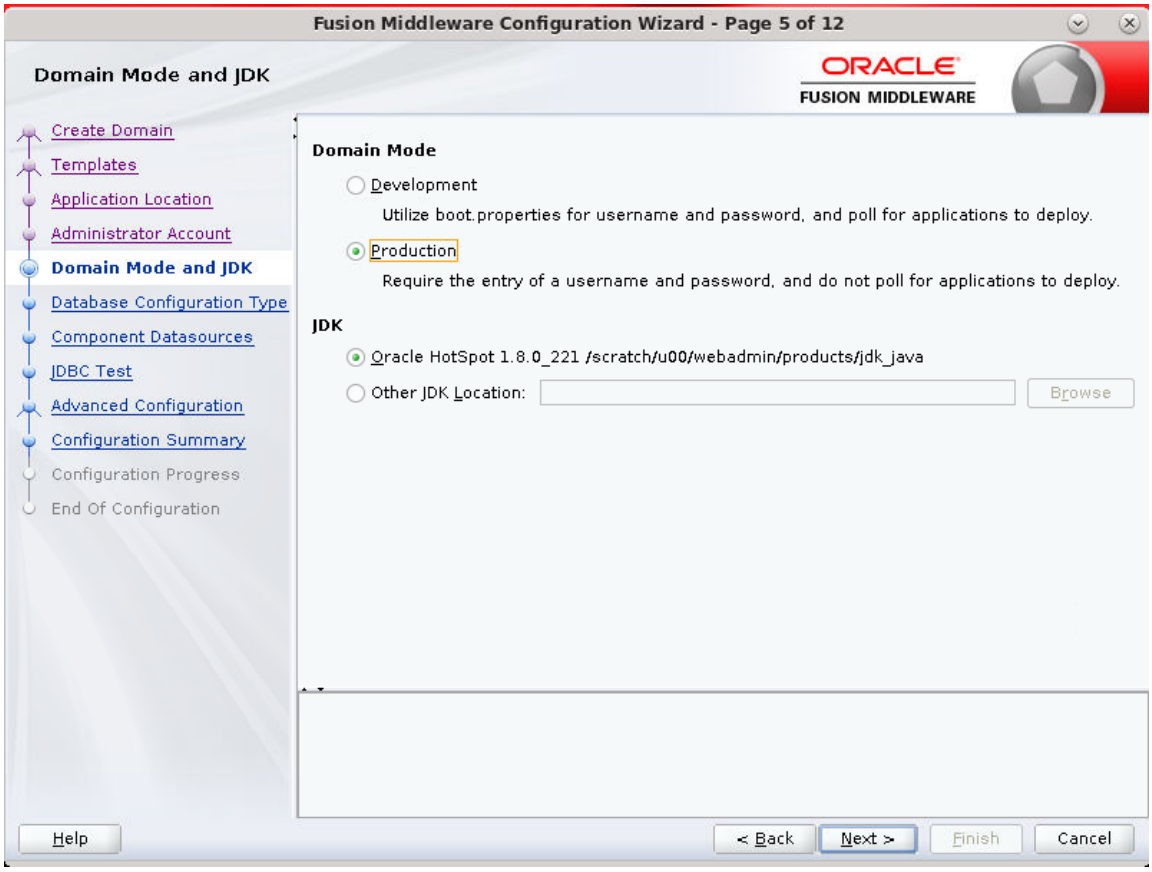
Application location: Application directory location. Example: /u00/webadmin/config/applications/wls_retail/APPNAMEDomain

7. Click **Next**.

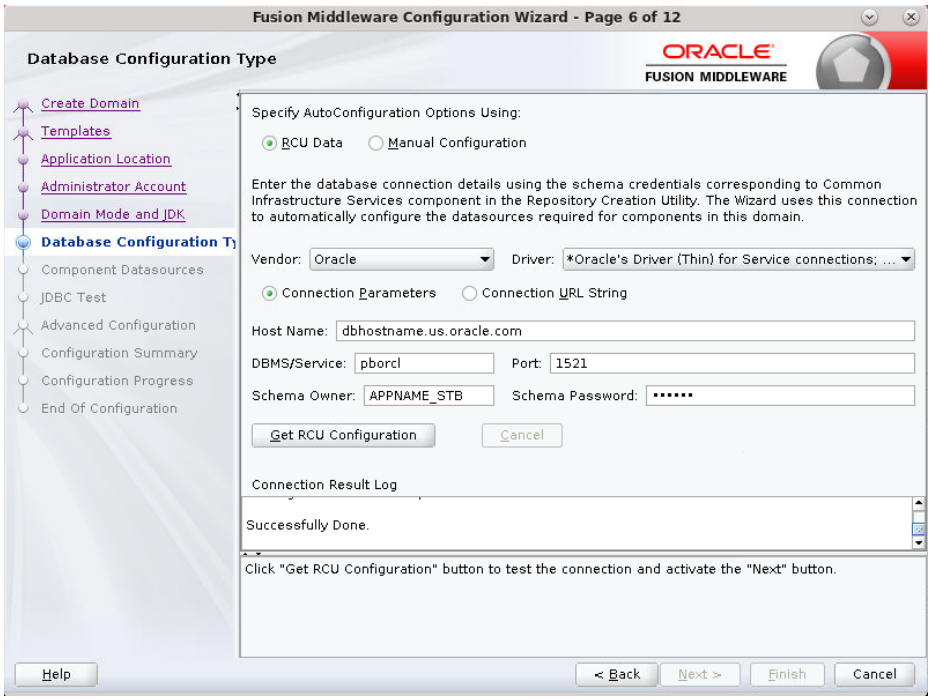
8. Provide the WebLogic administrator credentials and click **Next**:
   - Username: weblogic
   - Password: <Password>



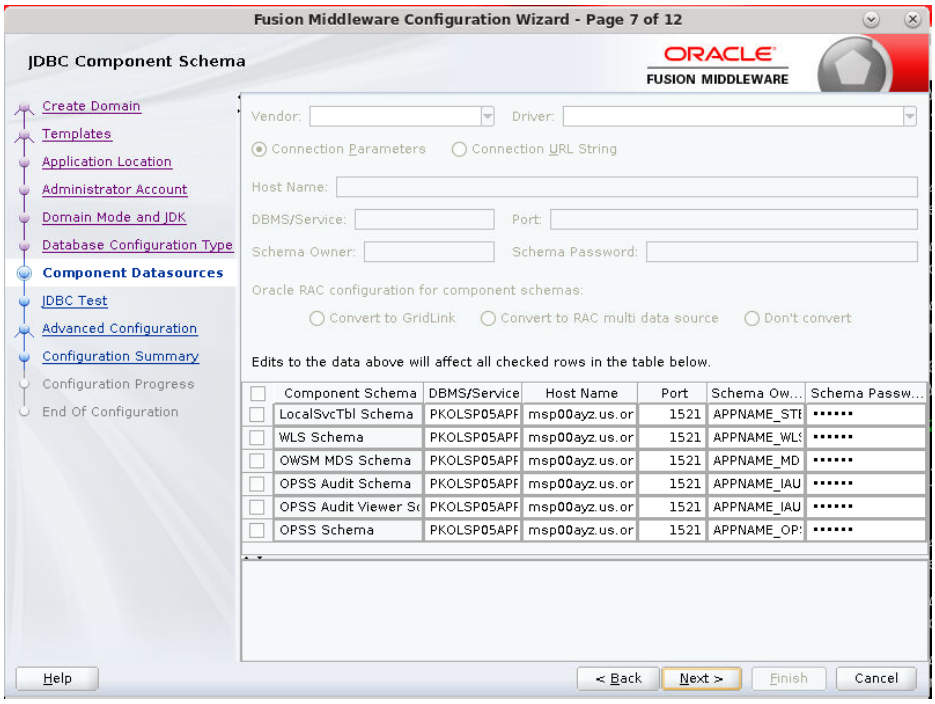
9. Select Domain Mode as Production and the JDK to use (as applicable) and click **Next.**



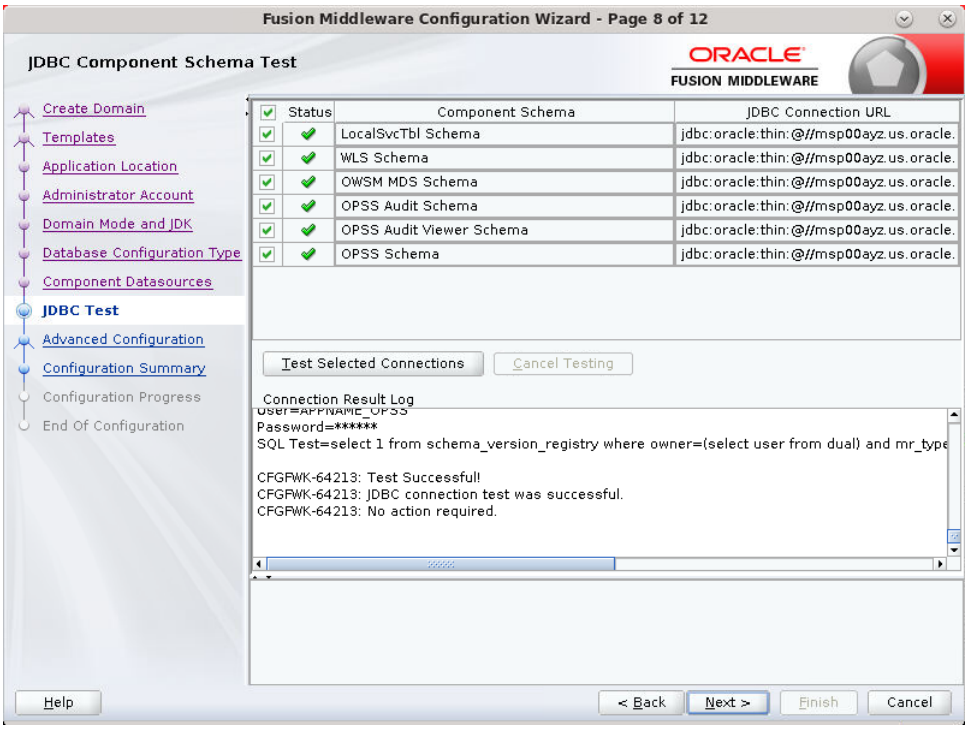
10. Select **RCU Data**.
    - Vendor: Oracle
    - DBMS/Service: dbservicename
    - Host Name: dbhostname.us.oracle.com
    - Port: 1521
    - Schema Owner: APPNAME_STB (Example: ALLOC_STB, ReSA_STB, and so on.)
    - Password: <Password>. This password which was used for RCU schema creation.



11. Click the Get RCU Configuration button.
12. Click Next.

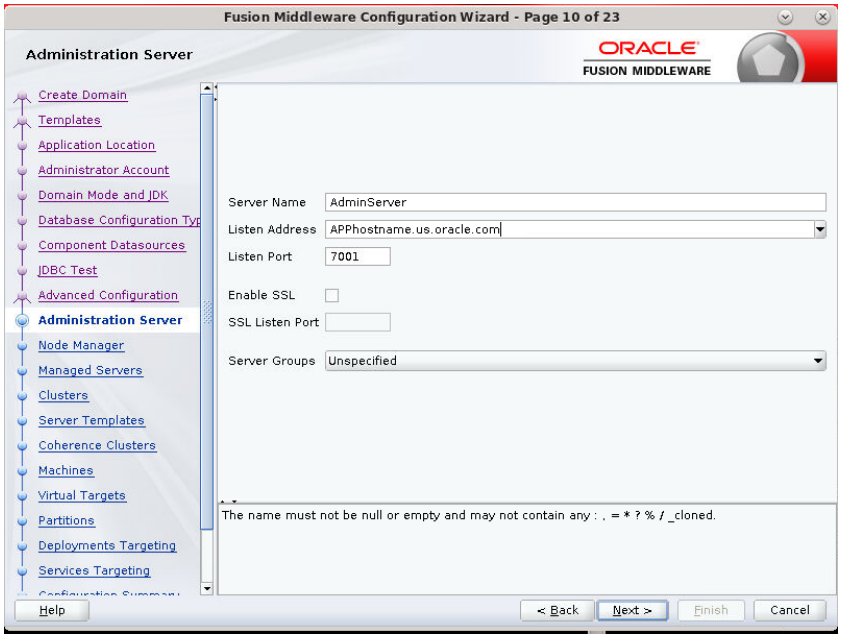


13. Click **Next** and it will test to make sure it can connect to your datasources.

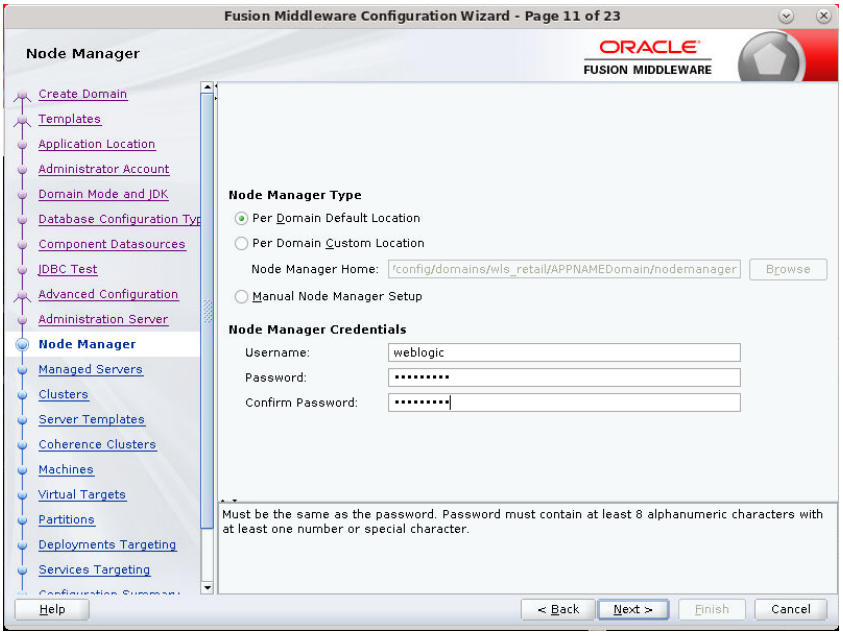


14. Click Next to continue
15. Select advanced configuration for:
    - Administration Server
    - Node manager
    - Managed Servers, Clusters and Coherence
    - Deployments and Services

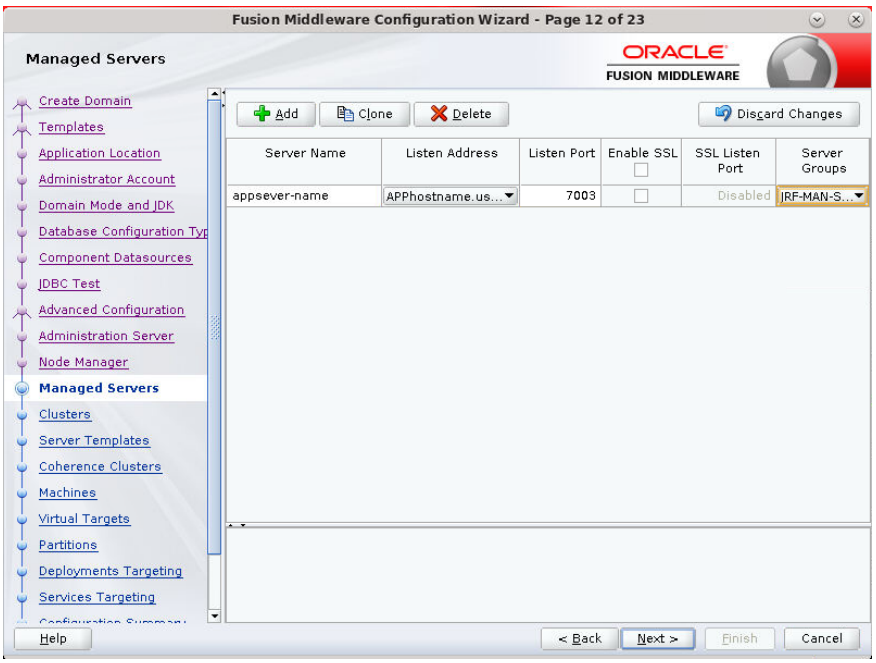
16. Configure the Administration Server:
    - Server Name: <APP name>_AdminServer
    - Listen address: Appserver Hostname or IPAddress of the Appserver Host.
    - Listen port: <Port for Admin Server> Note: The port that is not already used.
    - Server Groups: Unspecified



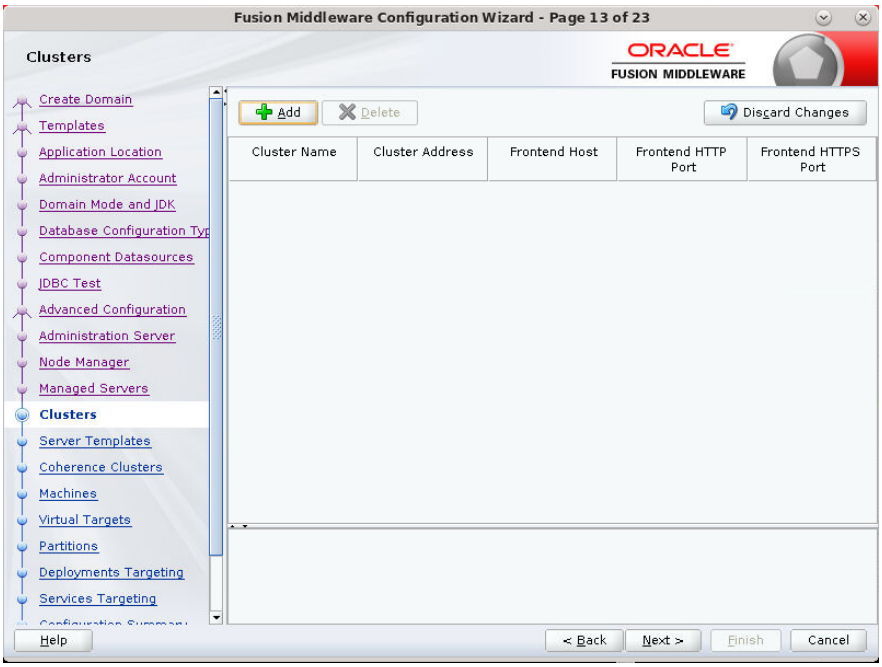
17. Configure Node Manager:
    - Node manager type: Per domain default location
    - Username: weblogic
    - Password: <Password for weblogic>



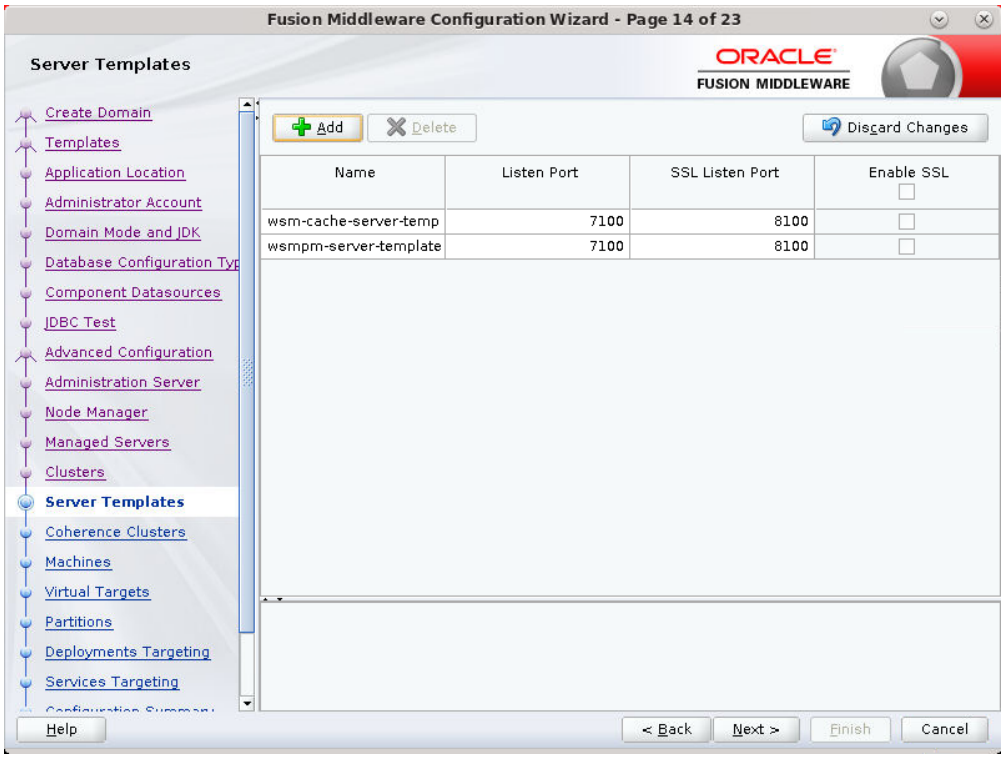
18. Click the **Add** button.
    - Server Name: <appname-server>
    - Listen address: Appserver Hostname or IPAddress of the Appserver Host
    - Listen port: <Port for Managed Server> Note: The port used here must be a free port.
    - Server Groups: JRF-MAN-SVR



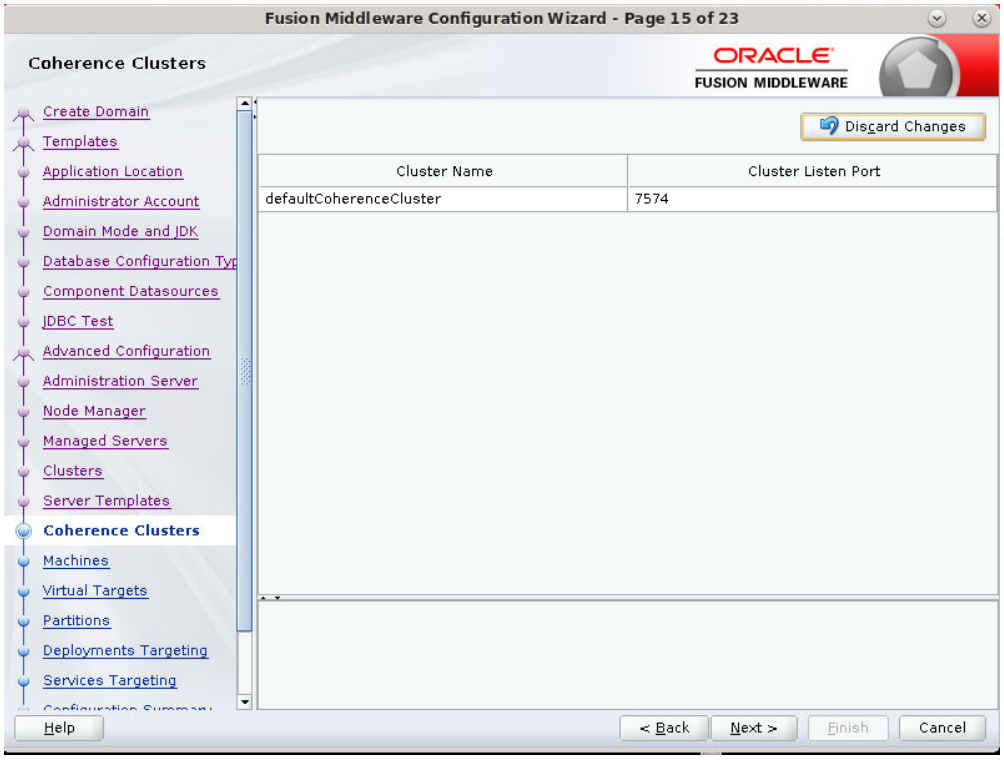
19. Skip Configure Clusters and click **Next**.



20. Do not change anything and click **Next**.



21. Click Next.

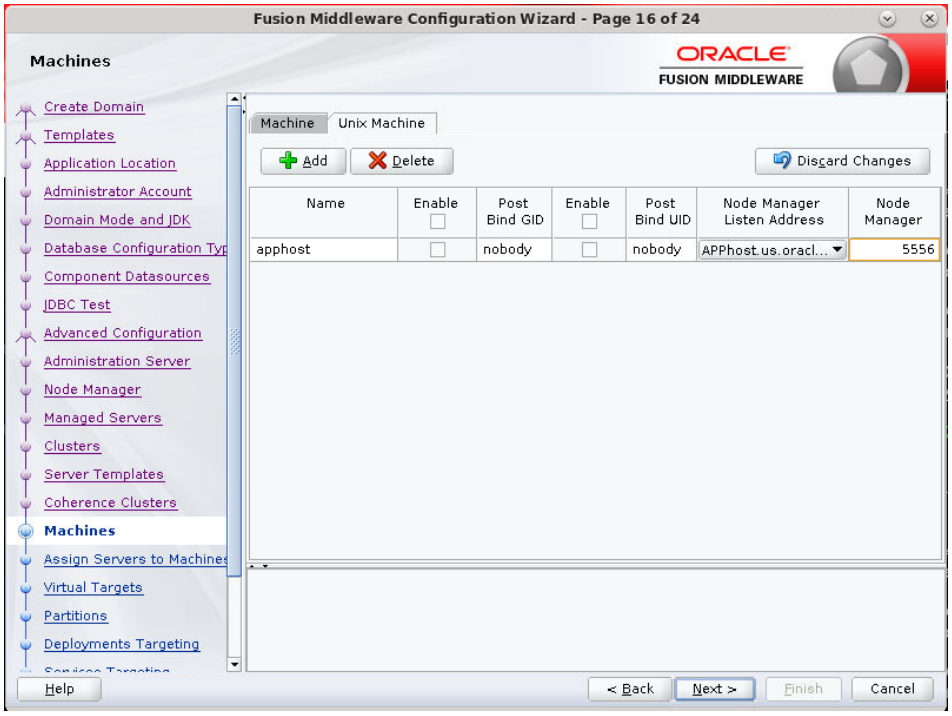


22. Configure Machines

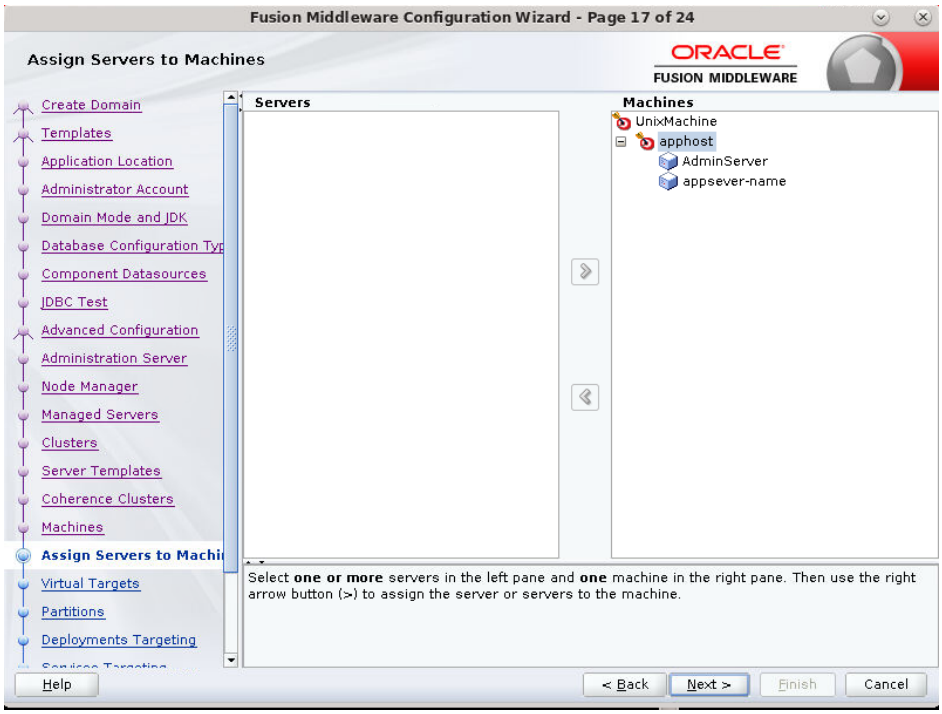
    Select unix Machine :

Click the **Add** button.

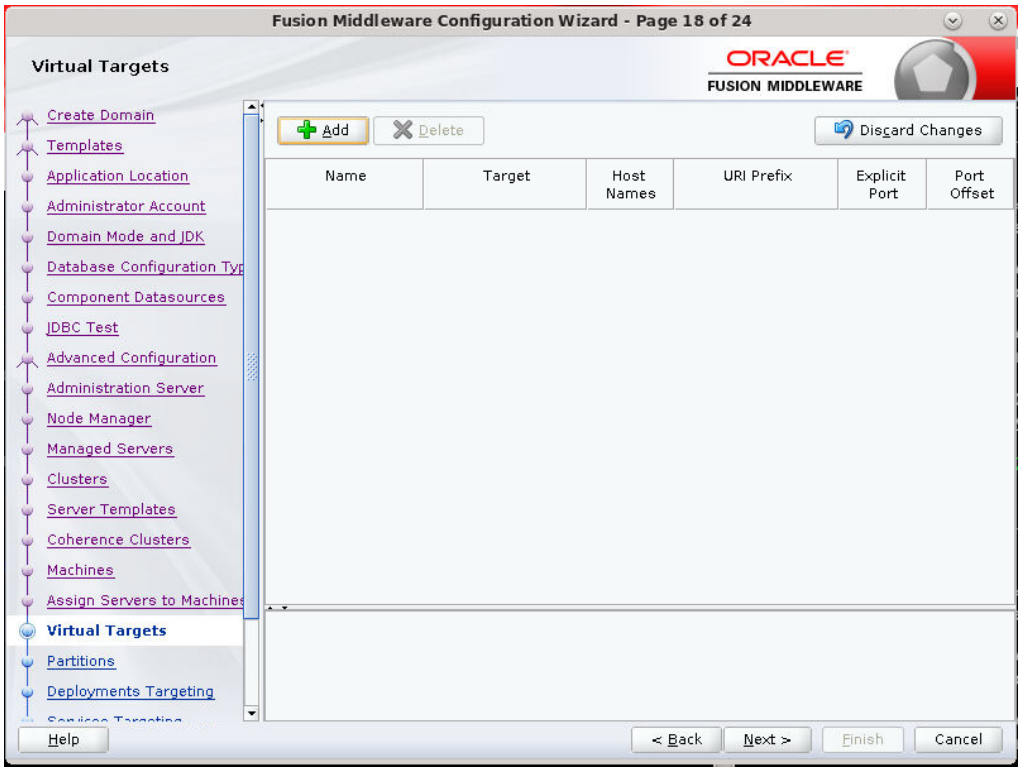
- Name: apphostname_MACHINE
- Listen address: apphostname or IPAddress
- Listen port: <Port for node manager>
- Note: The port used here must be a free port.



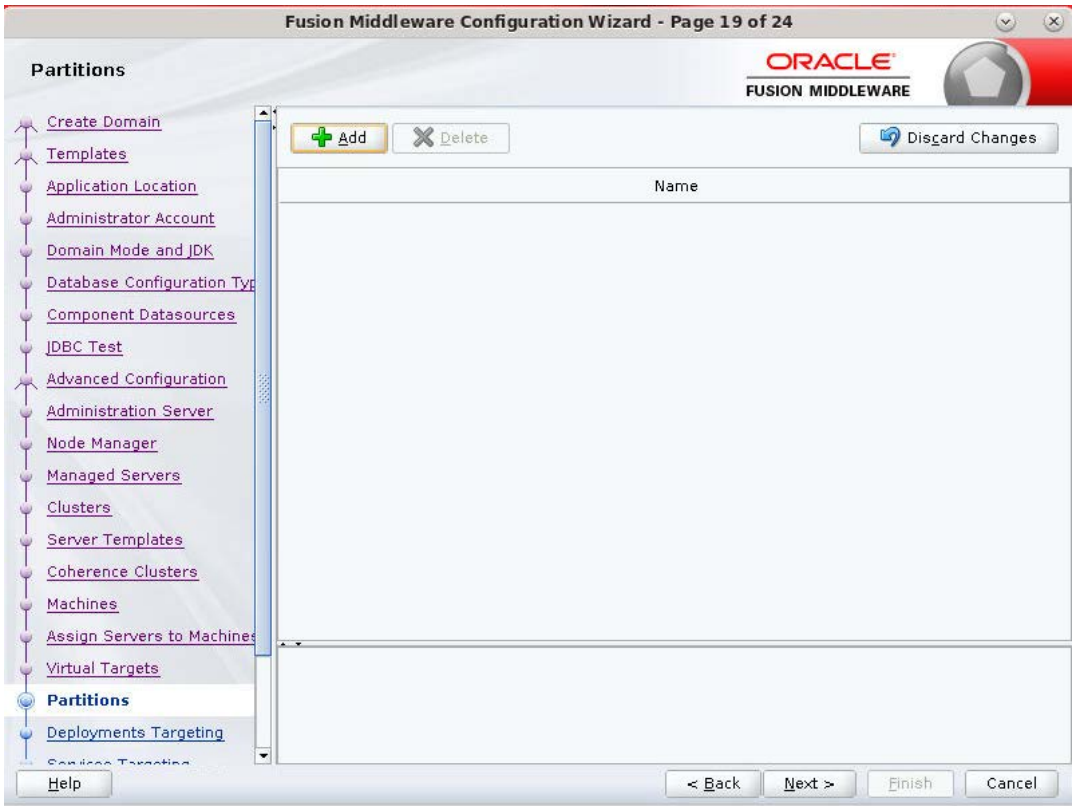
23. Assign the configured Admin server and managed servers to the new machine.



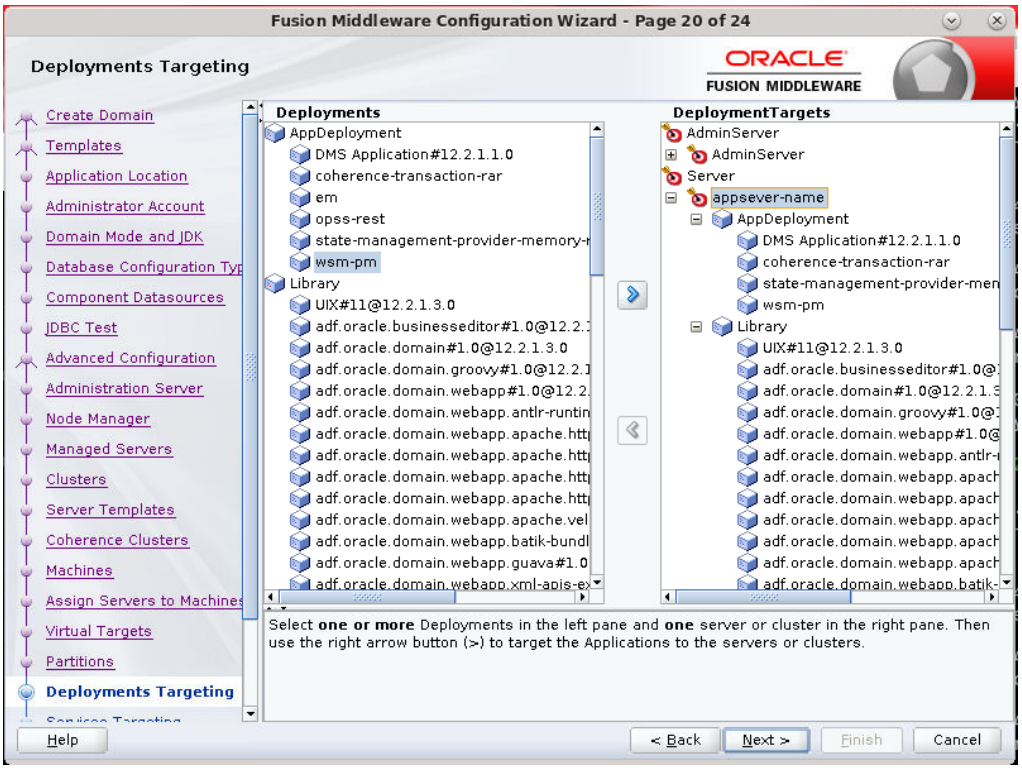
24. Skip Virtual Targets. Click **Next.**



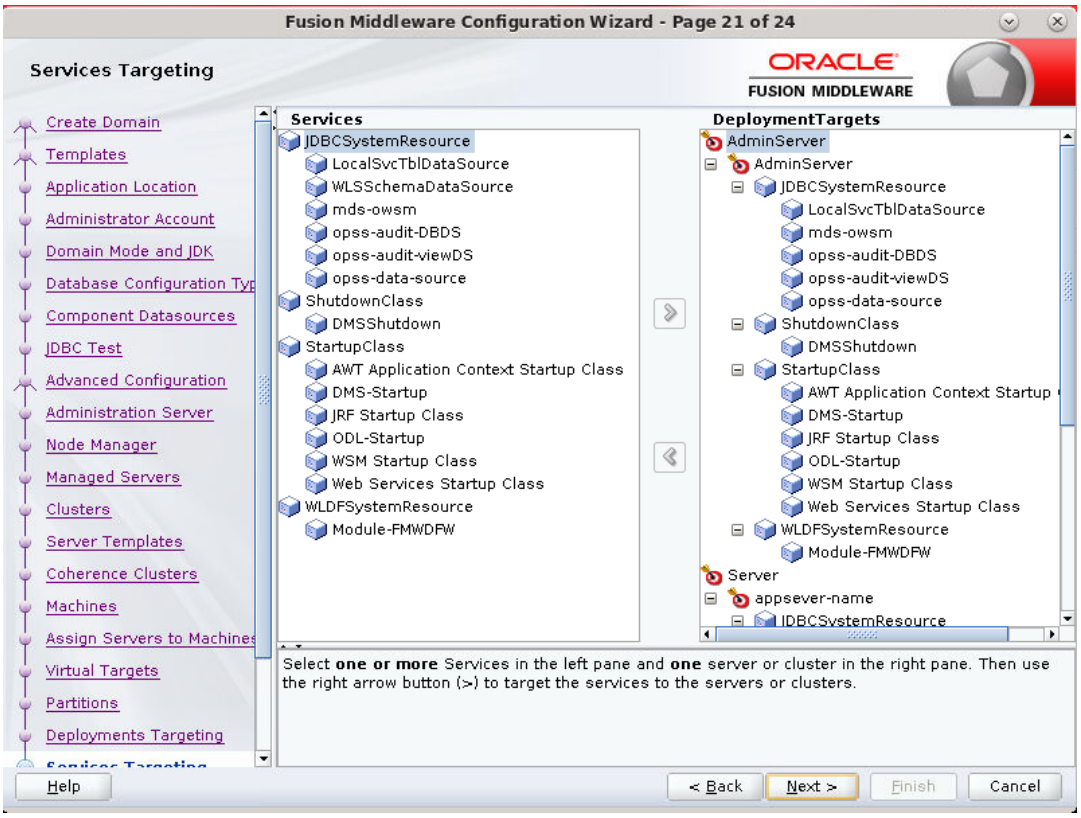
25. Skip Partitions. Click Next.



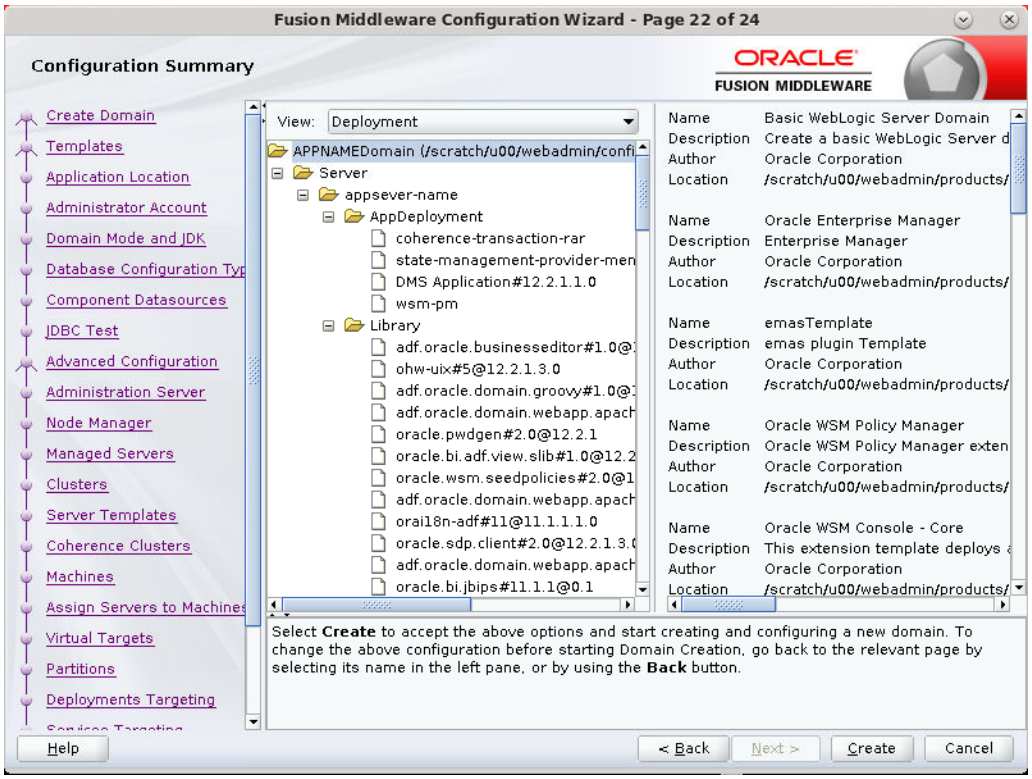
26. Target the "wsm-pm" deployment to APPNAME_AdminServer:



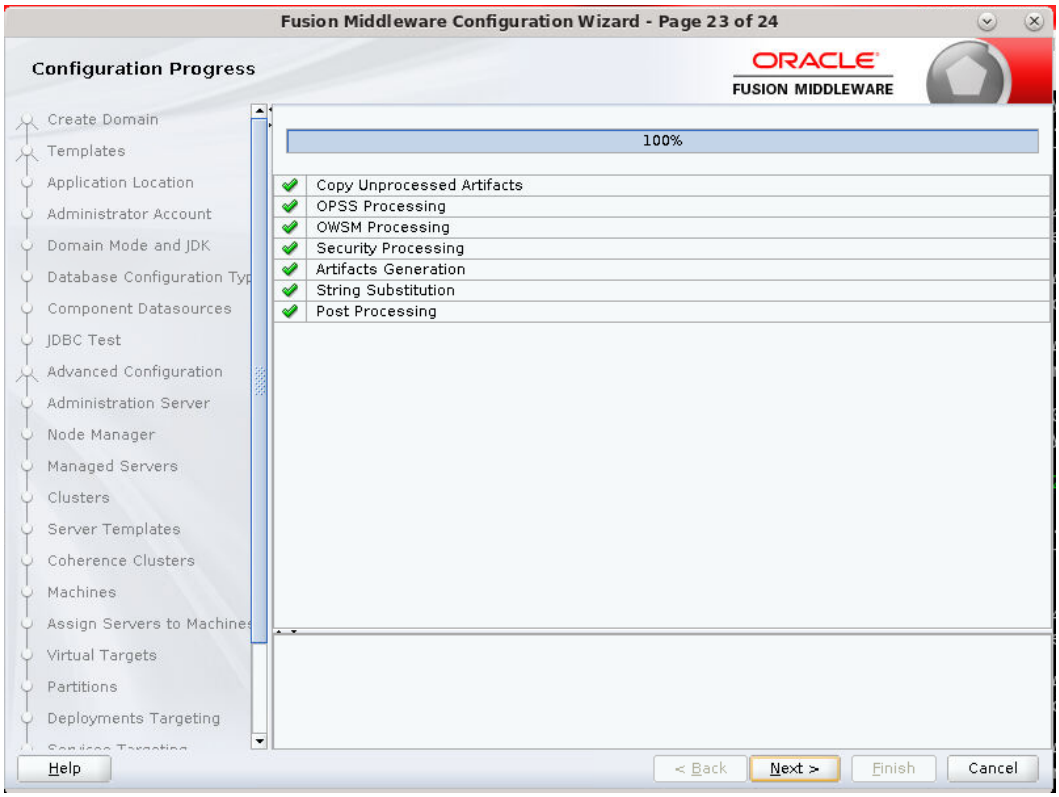
27. .Click Next.



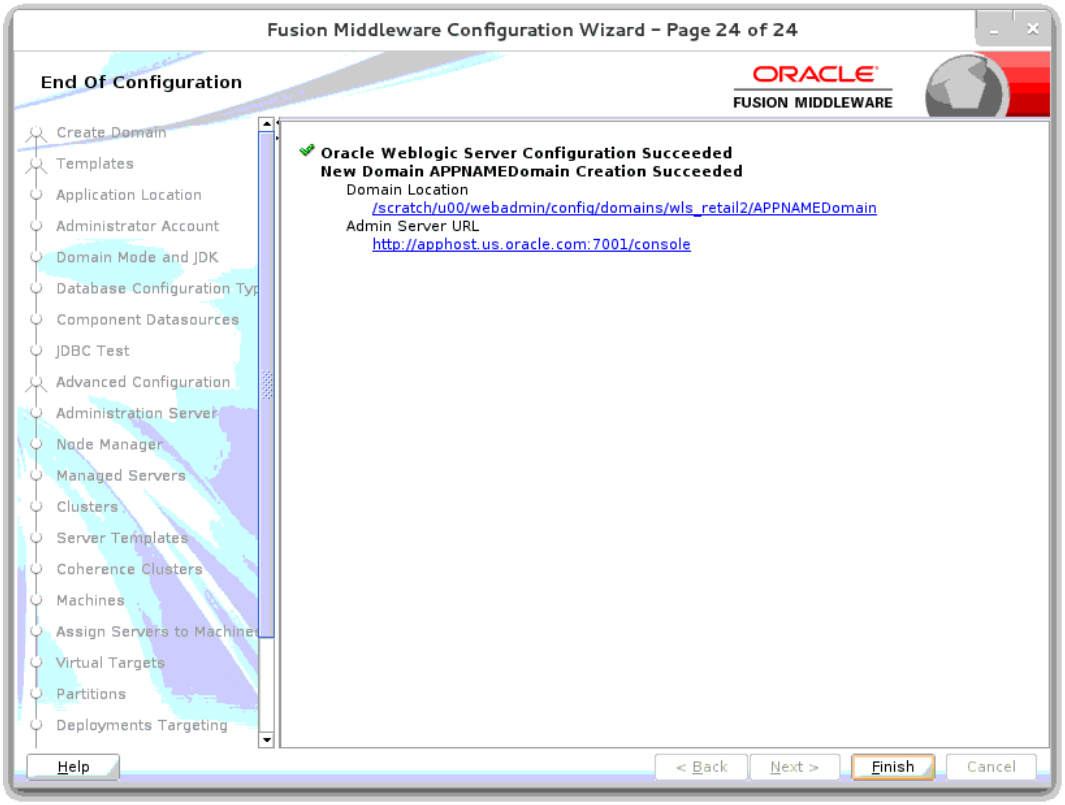
28. .Click **Create**.



29. Click **Next**.



30. When the process completes, click **Finish**.



Start the Node Manager

1. Start the nodemanager from <DOMAIN_HOME>/bin using the following script:

```
nohup ./startNodeManager.sh &
```

Start the AdminServer (admin console)

1. Configure boot.properties for starting the Weblogic domain without prompting to username and password using the following command:
2. Create security folder at <DOMAIN_HOME>/servers/<AdminServer>/ and create boot.properties file under <DOMAIN_HOME>/servers/<AdminServer>/security

   The file 'boot.properties' should have the following:

```
---------------------------------
username=weblogic
password=<password>
---------------------------------
```

   In the above, the password value is the password of WebLogic domain which is given at the time of domain creation.

   Save the boot.properties file and start WebLogic server.

3. Start the WebLogic Domain (Admin Server) from <DOMAIN_HOME> using the following:

```
nohup ./startWebLogic.sh &
```

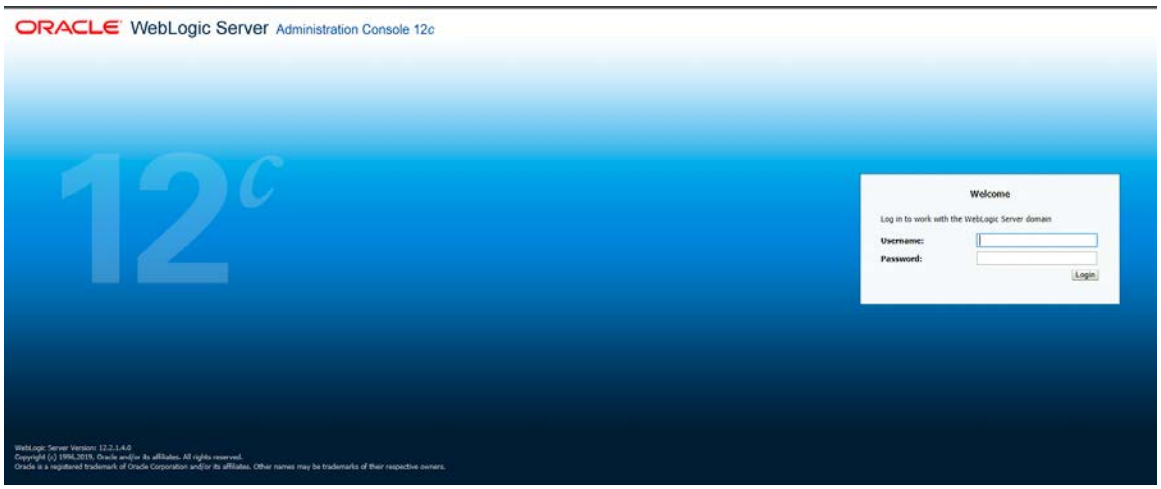
Example:

nohup
/u00/webadmin/config/domains/wls_retail1/APPdomain/startWebLogic.sh &

4. Access the Weblogic Admin console

   Example: http://<HOST_NAME>:<ADMIN_PORT>/console

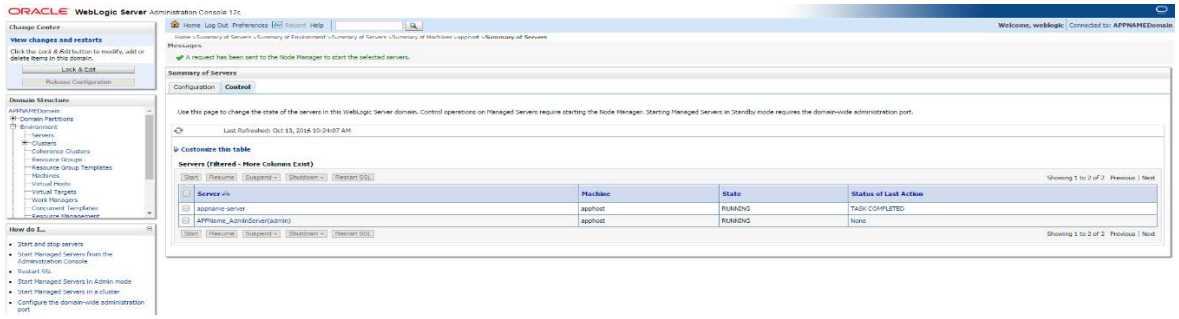
   In the below screen, provide username=weblogic and password=<weblogic password>



Start the Managed Server

After the NodeManager is started, the managed servers can be started via the admin console.

Navigate to Environments -> Servers and click the Control tab. Select appname-server and click **Start**.



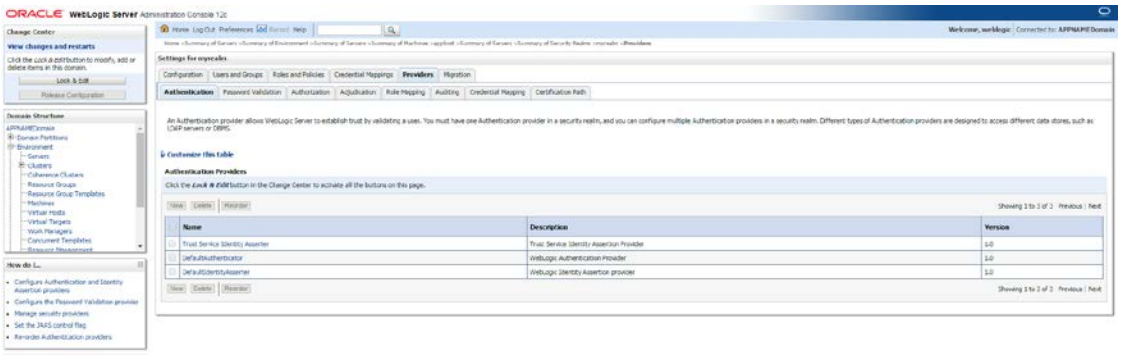
Managed Server should be up and running before configuring further steps

Configuration of OID LDAP Provider in Weblogic Domain:

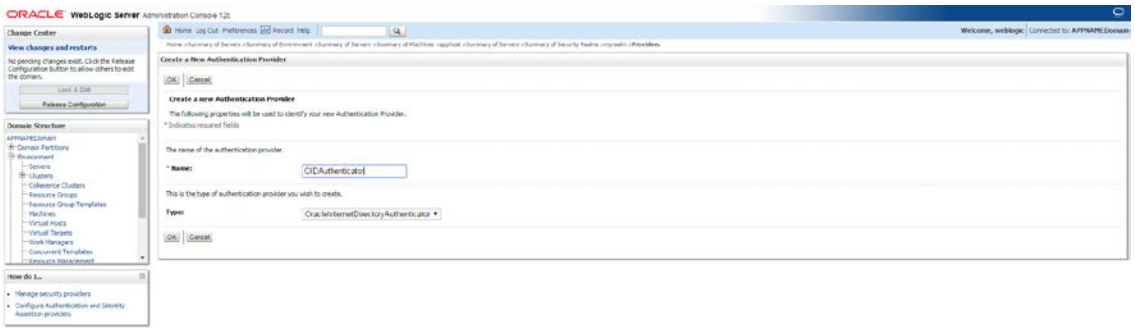
Perform the following procedure to create LDAP providers in the domains created in the previous steps

1. Log in to the Administration Console.
   `http://<HOSTNAME>:<ADMIN_PORT>/console`
2. In the Domain Structure frame, click **Security Realms**.
3. In the Realms table, click **myrealm**. The Settings for myrealm page is displayed.

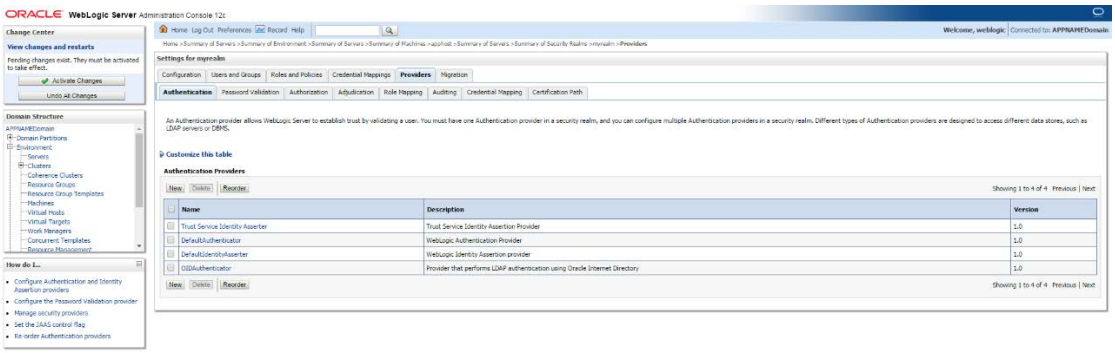
4. Click the Providers tab.



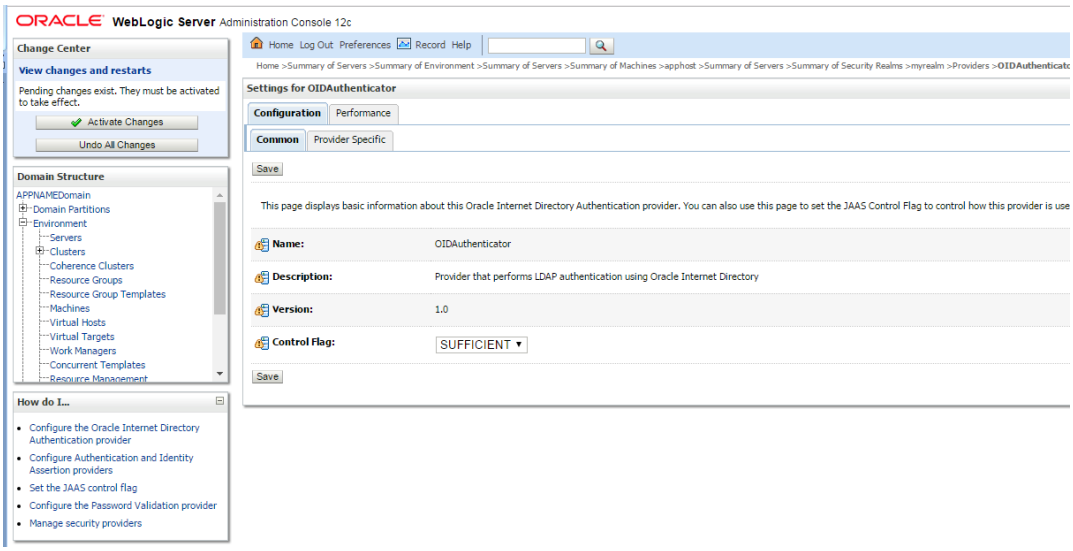
5. Click **Lock & Edit** and then click **New**. The 'Create a New Authentication Provider' page is displayed.



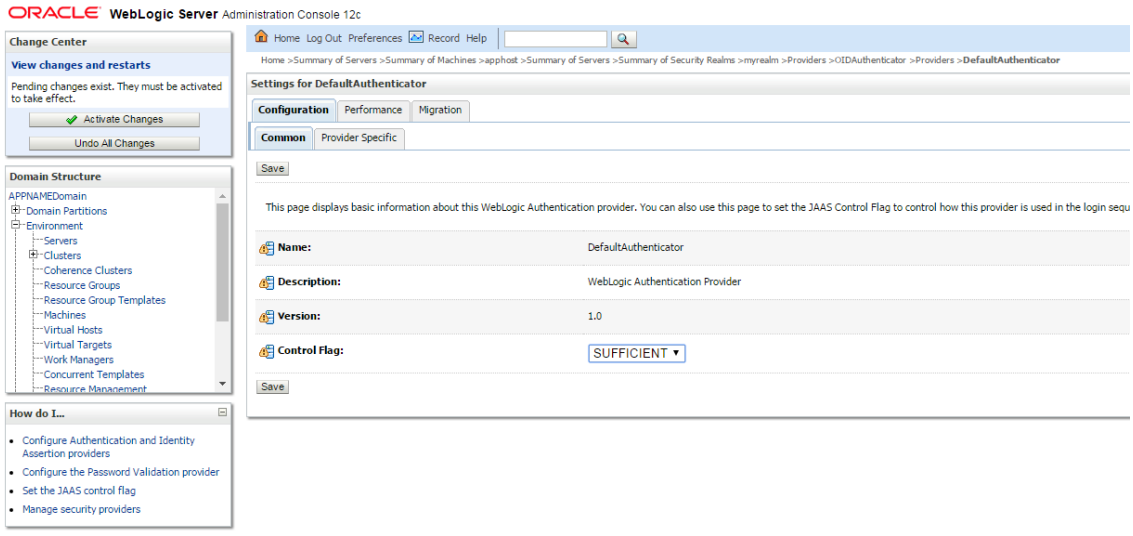
6. Enter OIDAuthenticator in the Name field and select OracleInternetDirectoryAuthenticator as the type. Click **OK**.



7. All the providers are displayed. Click **OID Authenticator**. Settings of OID Authenticator are displayed.



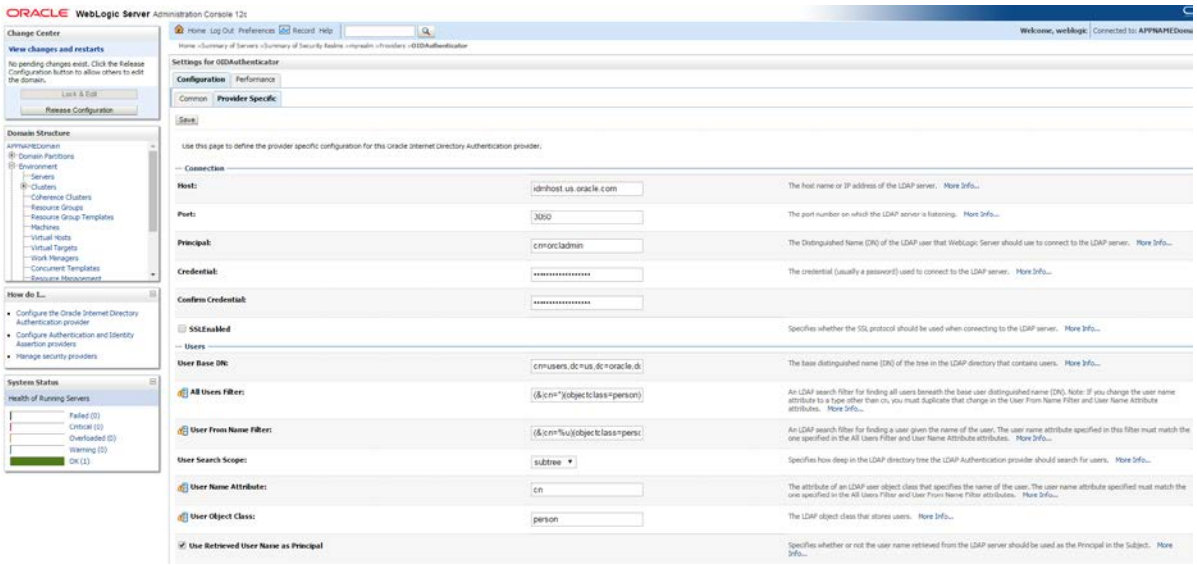
8. Set the Control Flag field to SUFFICIENT and click Save.
9. From the Providers tab, click on DefaultAuthenticator -> Configuration tab -> Common tab. Update the Control Flag to SUFFICIENT.
10. Click Save.



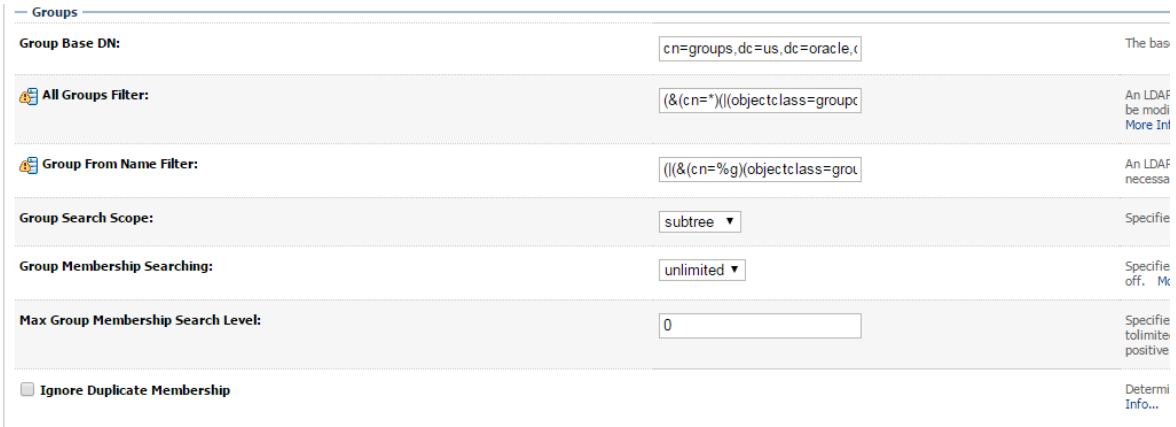
11. From the Providers tab, click the "OIDAuthenticator" (you just created), in the configuration -> Provider Specific tab enter your LDAP connection details:

    The values shown below are examples only. You should match the entries to your OID.

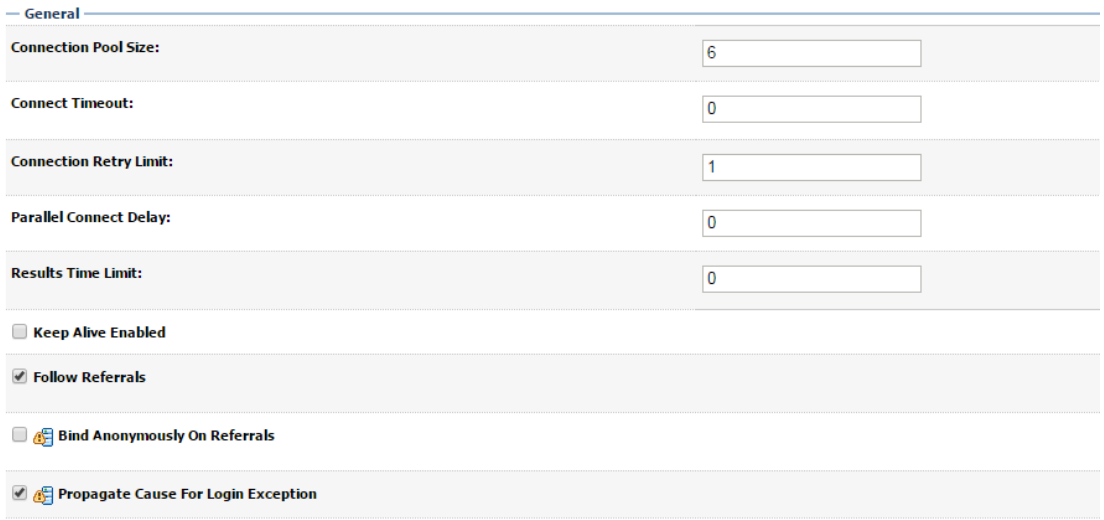
    - Host: <*oidhost*>
    - Port: <*oidport*>
    - Principal: cn=orcladmin
    - Credential: <*password*>
    - Confirm Credential: <*password*>
    - User Base DN: cn=users,dc=us,dc=oracle,dc=com
    - Enable 'Use Retrieved User Name as principal.'



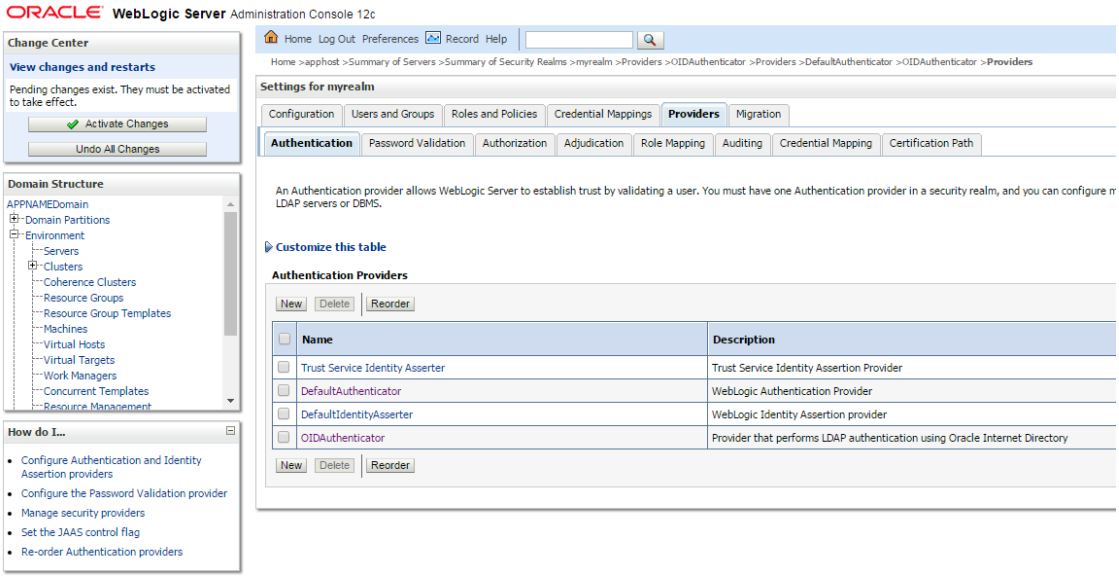
12. Modify the following:
    - Group Base DN: cn=Groups,dc=us,dc=oracle,dc=com



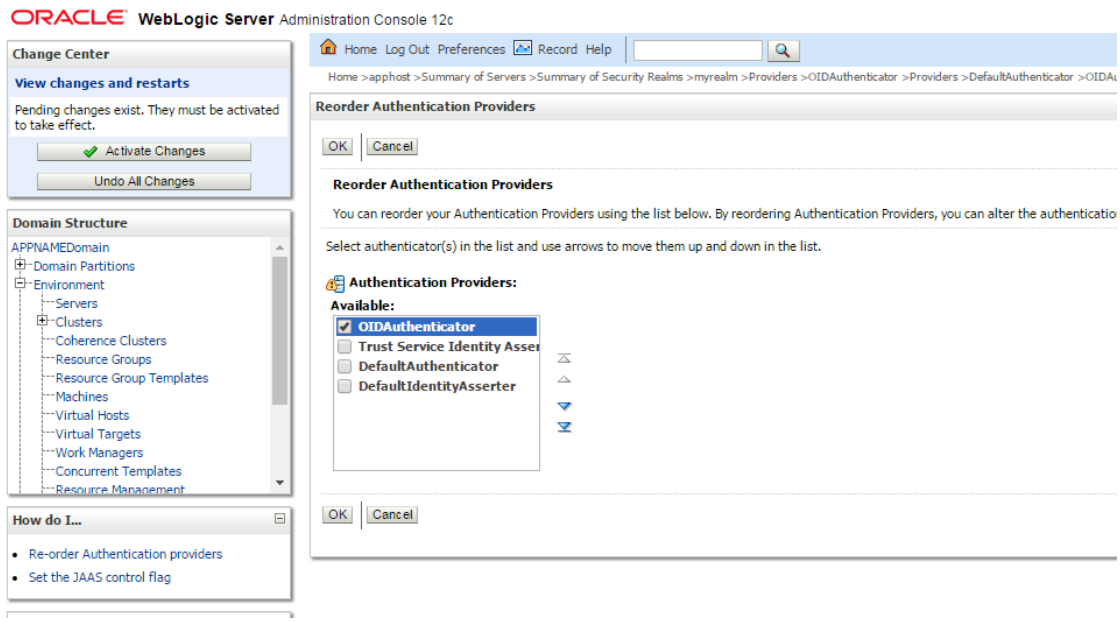
13. Check Propagate Cause For Login Exception



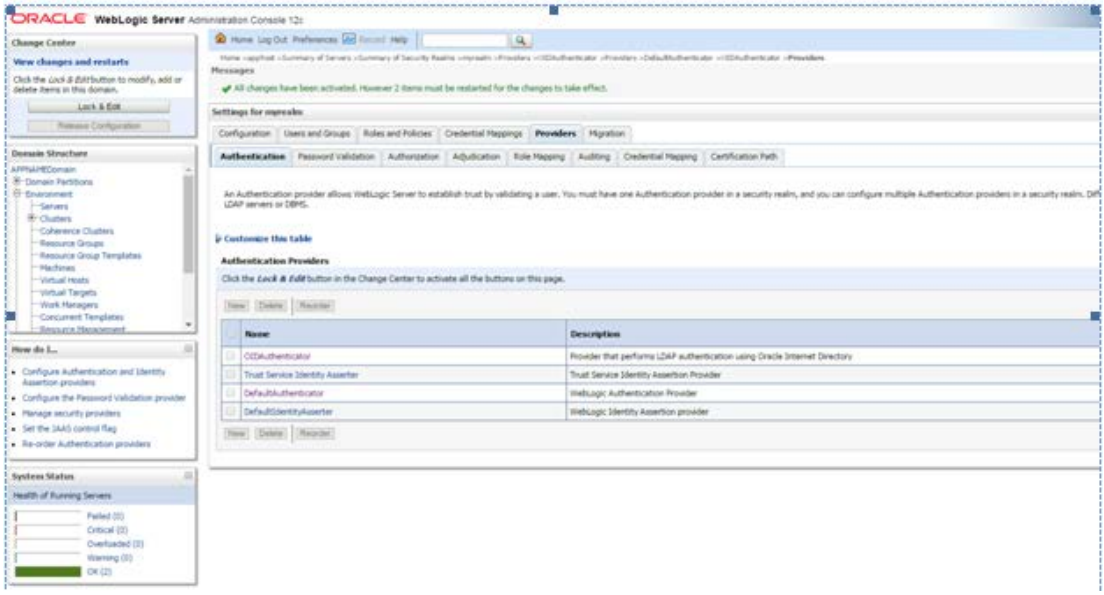
14. Click Save.
15. Click the Providers tab.



16. Click Reorder.
17. Move OIDAuthenticator to the top of the providers list.



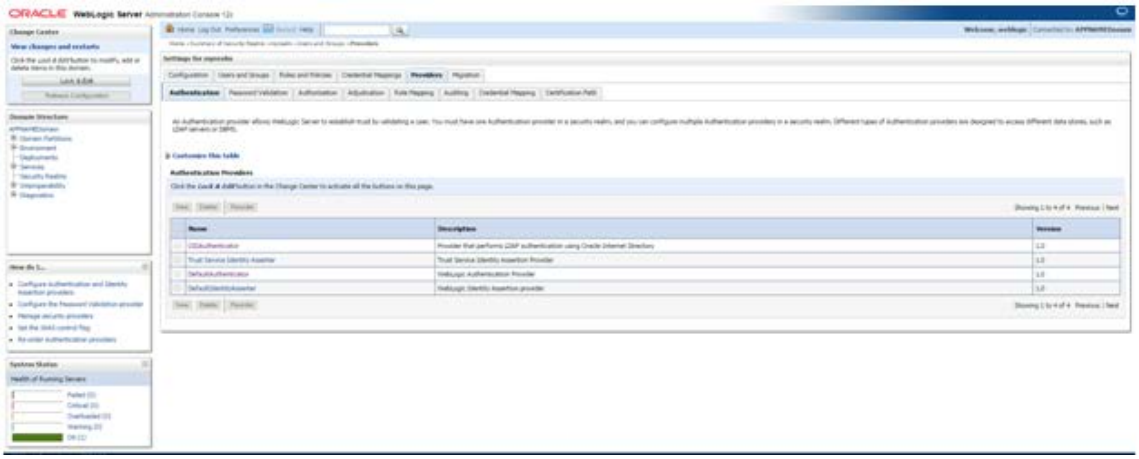
18. Click OK.
19. Once your changes are saved, click Activate Changes.



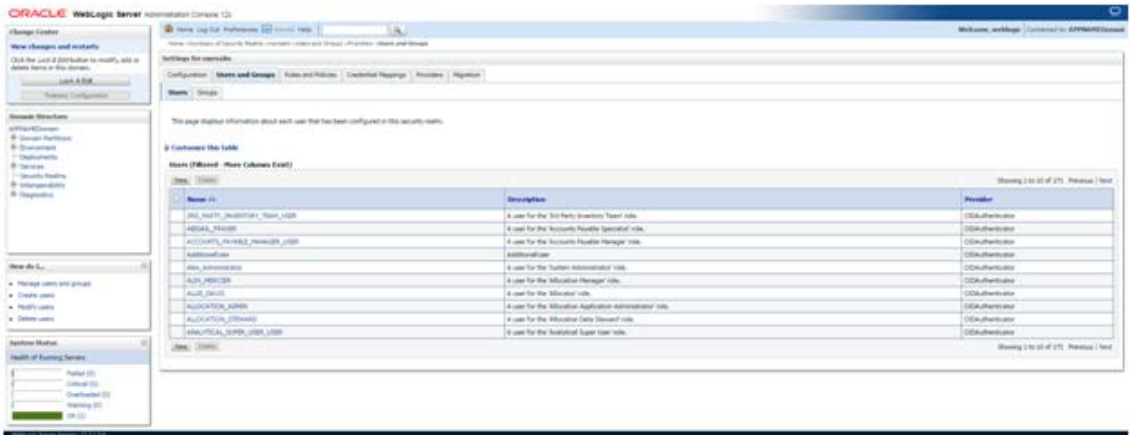
20. Shutdown all servers and restart the admin server using startWebLogic.sh script. Login to Admin Console and restart Managed server.

Verify OID Authenticator

1. Log in to the Administration Console.
   http://<HOST_NAME>:<ADMIN_PORT>/console/
2. In the Domain Structure frame, click Security Realms.
3. In the Realms table, click Default Realm Name. The Settings page is displayed.
4. Click the Providers tab. You must see the OID Provider in that list.



5. Click the Users and Groups tab to see a list of users and groups contained in the configured authentication providers.



Clustered Installations – Pre-Installation Steps

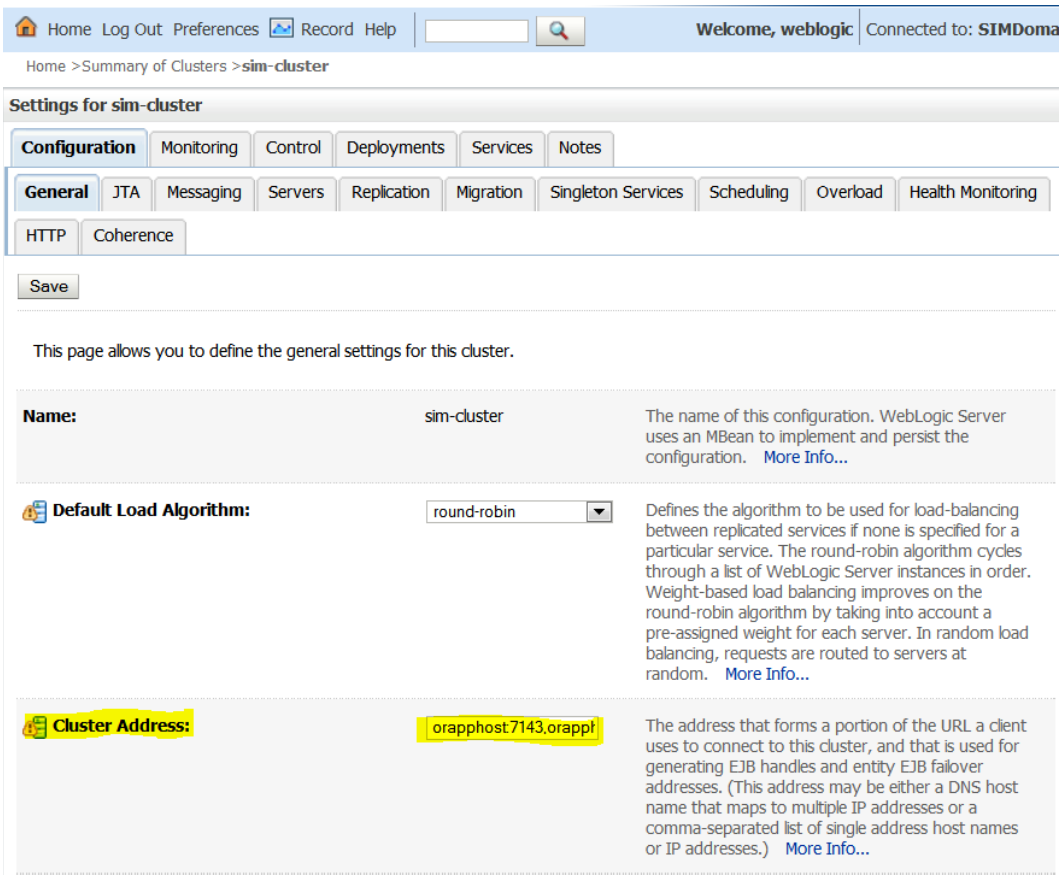
Skip this section if you are not clustering the application server.

If SIM is being installed into a clustered environment, the "Cluster Address" field must be set prior to installation. This is set in:

Clusters -> sim-cluster (or name of your cluster) -> configuration (tab) -> general (tab)

Set the address to your cluster in the "Cluster Address" field, for example,

apphost1:7143,apphost2:7143



Expand the SIM Application Distribution

To expand the SIM application distribution, do the following.

1. Log in to the UNIX server as the user who owns the Web Logic installation. Create a new staging directory for the SIM application distribution (sim16-application.zip). There should be approximately 1 GB of disk space available for the application media and installation files.

   This location is referred to as INSTALL_DIR for the remainder of this chapter.

2. Copy sim16-application.zip to <INSTALL_DIR> and extract its contents.

Loading SIM LDIFs into the OID

The SIM installation media contains a zip file with a group of template LDIF files. They are in the SIM distribution you previously expanded:

<INSTALL_DIR>/sim/application/sim16/ldap/sim-ldap.zip

The LDIF files included are just templates and must be modified to fit the structure and conventions of the OID setup for your environment. Once the LDIFs are updated for your configuration they can be loaded into LDAP using the ldapadd tool that is included in the OID installation.

For example, to load the SIM Object classes (this is done on the OID host):

```
# export ORACLE_HOME=/u00/webadmin/products/wls_idm/Oracle_IDM
# export PATH=$ORACLE_HOME/bin:$PATH
# ldapadd -v -c -h <OID_HOST> -p 3060 -w <ORCLADMIN PASSWORD> -D cn=orcladmin
-f  sim_objectclasses.ldif
```

The order of the LDIF install should be:

1. sim_objectclasses.ldif
2. sim_add_company.ldif
3. sim_add_containers.ldif
4. sim_data_groups.ldif
5. sim_data_roles.ldif
6. sim_data_stores.ldif
7. sim_data_user_groups.ldif
8. sim_data_users.ldif
9. sim_data_users_roles.ldif

> **Note:** Users that are needed for integration with SIM (for example, for XStore or RIB) need to be a member of these groups :
> - SIM_SECURE_USERS
> - SIM_INTEGRATION_USERS

Set the Environment Variables for the SIM Installer

1. Set the following environment variables for the SIM installer (the following are just examples, use values for appropriate for your environment):

   ```
   ORACLE_HOME=/u00/webadmin/products/wls_retail
   WEBLOGIC_DOMAIN_HOME=/u00/webadmin/config/domains/wls_retail/SIMDomain
   JAVA_HOME=/u00/webadmin/product/jdk_java
   PATH=$JAVA_HOME/bin:$PATH
   export ORACLE_HOME WEBLOGIC_DOMAIN_HOME JAVA_HOME PATH
   ```

2. If a secured datasource is going to be configured you also need to set "ANT_OPTS" so the installer can access the key and trust store that is used for the datasource security:

   ```
   export ANT_OPTS="-Djavax.net.ssl.keyStore=<PATH TO KEY STORE> -
   Djavax.net.ssl.keyStoreType=jks -Djavax.net.ssl.keyStorePassword=<KEYSTORE
   PASSWORD> -Djavax.net.ssl.trustStore=<PATH TO TRUST STORE> -
   Djavax.net.ssl.trustStoreType=jks -
   Djavax.net.ssl.trustStorePassword=<TRUSTSTORE PASSWORD>"
   ```

An example of this would be:

```
export ANT_OPTS="-Djavax.net.ssl.keyStore=/u00/webadmin/product/wls_retail
/wlserver/server/lib/apphost.keystore -Djavax.net.ssl.keyStoreType=jks -
Djavax.net.ssl.keyStorePassword=retail123 -Djavax.net.ssl.trustStore=/
u00/webadmin/product/wls_retail /wlserver/server/lib/apphost.keystore -
Djavax.net.ssl.trustStoreType=jks -
Djavax.net.ssl.trustStorePassword=retail123"
Run the SIM Application Installer
```

This installer configures and deploys the SIM application and Java WebStart client files.

1. If you are using an X server set the DISPLAY environment variable so that you can run the installer in GUI mode (recommended). If you are not using an X server, or the GUI is too slow over your network, unset DISPLAY for text mode.
2. Verify that the managed server to which SIM will be installed is currently running.
3. Run the install.sh script. This launches the installer. After installation is completed, a detailed installation log file is created: <INSTALL_DIR>/sim/application/logs/sim-install-app.<timestamp>.log.

Note: The manual install option in the installer is not functional for this release.

Note: See Appendix: SIM Application WebLogic Server Installer Screens for details on every screen and field in the WebLogic application installer.

Note: See Appendix: Common Installation Errors for details on common installation errors.

Clustered Installations – Post-Installation Steps

Skip this section if you are not clustering the application server.

If you are installing the SIM application into a clustered WebLogic server environment the installer will automatically set the cluster to use a consensus migration basis. It is recommended to use database migration basis for clusters with only 2 nodes or if this is to be used in a production system.

The database cluster migration configuration setup is described in the "Whole Server Migration" section of the following Oracle document:

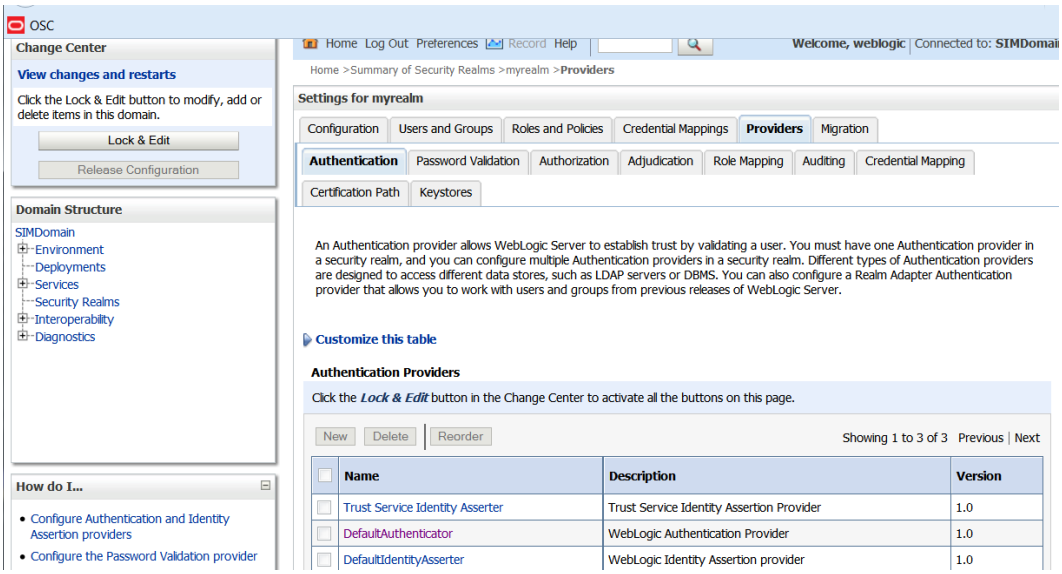
Oracle® Fusion Middleware Administering Clusters for Oracle WebLogic Server (E55168-02)

Please refer to that document on how to perform this procedure. In addition, note that since the installer sets this to consensus, this will need to be done every time that SIM is installed.

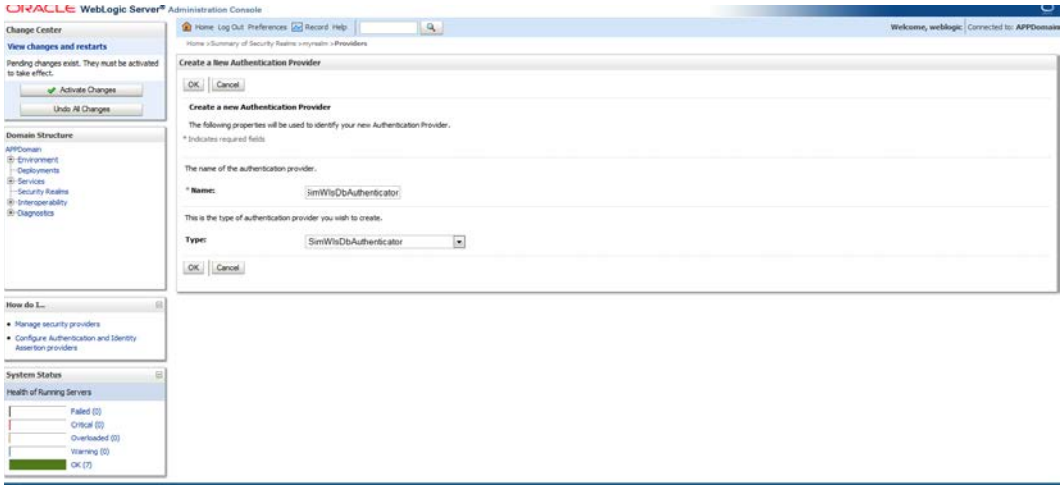
SIM Database Authentication Provider set up (to be done after the application deploy)

> **Note:** This procedure is only needed if you plan on using database authentication for the SIM application. This should be skipped if LDAP is going to be used for authentication.

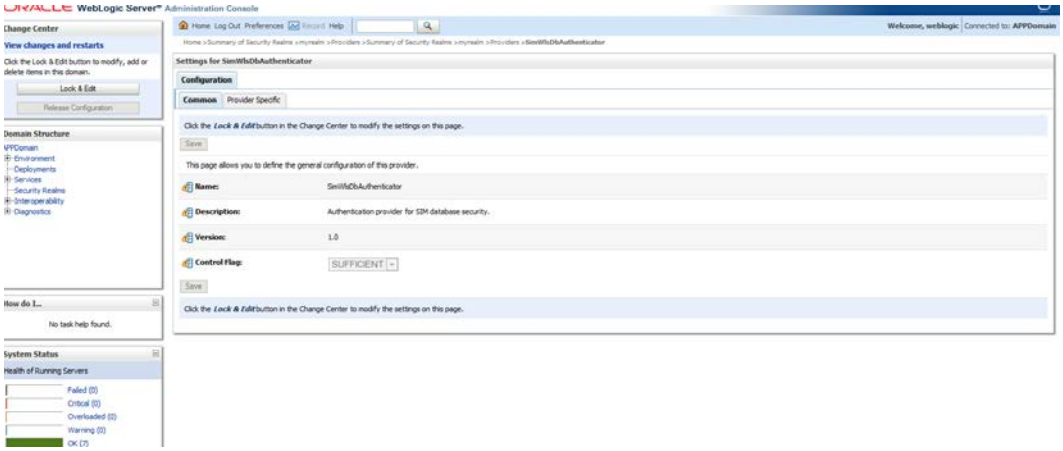
1. Shut down all the servers of the WebLogic Domain created.
2. Once you extract the SIM installer to <INSTALL_DIR> copy the sim-security.zip present in <INSTALL_DIR>/sim/application/sim16 to the <WEBLOGIC_DOMAIN_HOME>/lib and extract it contents in the folder.
3. Start the domain admin server.
4. Log into the WebLogic console.
5. Navigate to: security realms -> myrealm (default realm) -> providers.



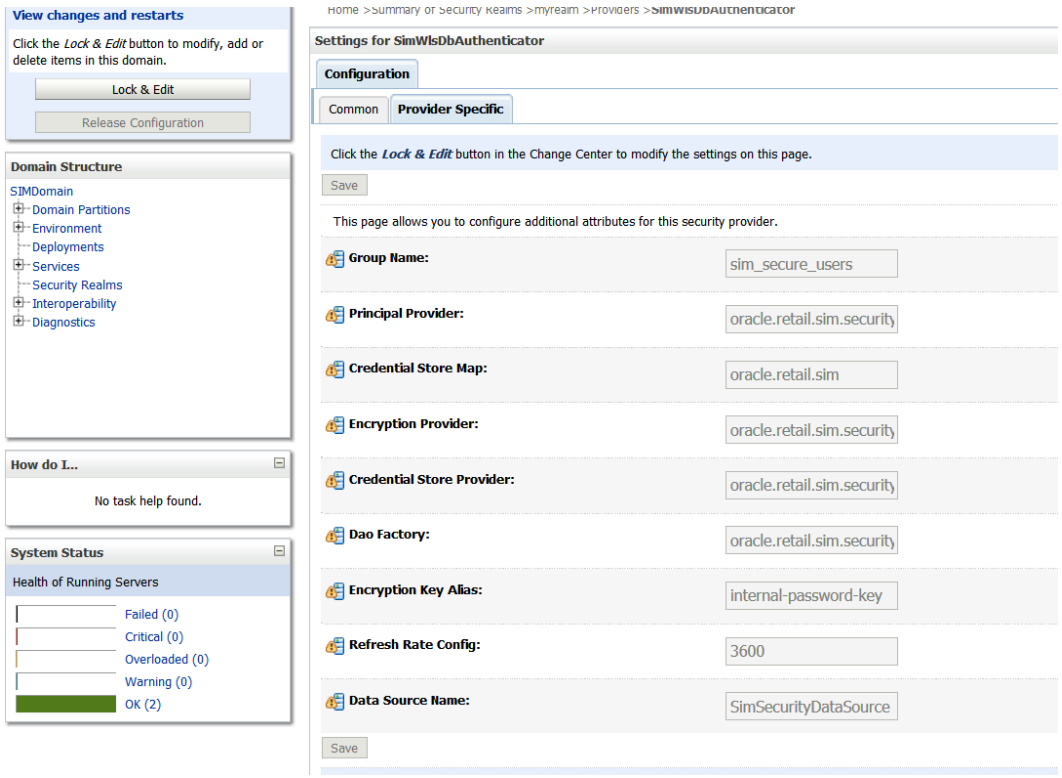
6. Click Lock & Edit in the change center.
7. Click New provider.
8. Select the provider type from the list: SimWlsDbAuthenticator.
9. Set the provider name (Default: SimWlsDbAuthenticator).



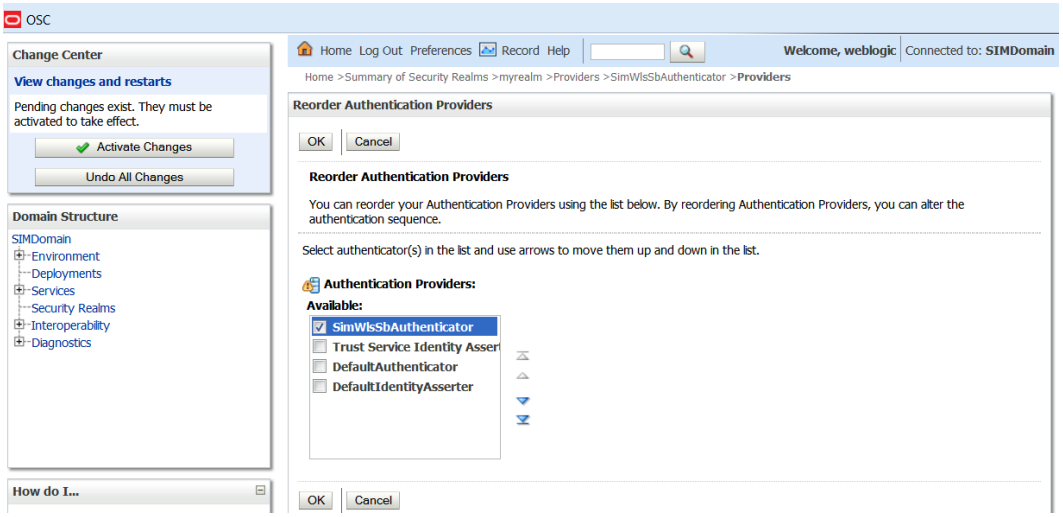
10. Click Ok.
11. Open the new provider configuration.
12. Under Common, set the Control Flag to SUFFICIENT.
13. Click **Save**.



14. Click the Provider Specific tab.
15. The SIM Data Source Name defaults to SimSecurityDataSource which is what the SIM installer creates. It should be left to the default value. The Group Name defaults to 'sim_secure_users'. If this was changed during the SIM installation then it also needs to be changed here.



16. Click Save.
17. Back on the provider tab, click Reorder.
18. Move the SimWlsDbAuthenticator to the top of the list.



19. Click Ok.
20. Click Activate Changes.
21. Shutdown the SIMDomain (The adminserver and the managed servers).
22. Start the admin and managed servers for the domain.

Review and/or Configure Oracle Single Sign-On

Note: This procedure is only needed if you plan on setting up the SIM application using Single Sign On (SSO) authentication. This should be skipped if SSO is not going to be configured for this environment. The Oracle Access Manager must be configured and the Oracle http server (Webtier and webgate) must be registered into the Oracle Access Manager.

Create the SIM SSO provider in the SIMDomain

1. Shut down all the servers of the WebLogic Domain created.
2. Once you copy the contents to <INSTALL_DIR> copy the sim-security.zip present in <INSTALL_DIR>/sim/application/sim16 to the <WEBLOGIC_DOMAIN_HOME>/lib and extract it contents in the folder.
3. Start the domain admin server.
4. Log into the WebLogic console
5. Navigate to: security realms -> myrealm (default realm) -> providers.
6. Click Lock & Edit in the change center.
7. Click New.
8. Select the provider type from the list: SimWlsSsoAuthenticator.
9. Set the provider name (Default: SimSsoAuthenticator).
10. Click OK.
11. Click on the newly created 'SimSsoAuthenticator'.
12. Under Common tab, set the Control Flag to SUFFICIENT and click Save.
13. Click Provider Specific tab.
14. Check that the GroupName is set to the name of the group used for SIM secure users (sim_secure_users by default).
15. All other values under the Provider Specific tab can be left as the default value.
16. Click Save.
17. On the provider list, click Reorder.
18. Move the SimWlsSsoAuthenticator to the top of the list.
19. Click Ok.
20. Click Activate Changes in the control center.
21. Shutdown the domain.
22. Start the admin and managed servers for the domain.

After the SSO provider is created in the SIMDomain, you will also have to set the protection of the SIM application resources correctly in the Application Domain that has been registered in the Oracle Access Manager.

In the Webtier/Webgate http server you need to set the mod_wl_ohs.conf file to redirect the http call to the where the SIM application has been deployed.

For example, in mod_wl_ohs.conf set:

```
<Location /sim-client >
 WebLogicCluster apphost1:apphost2
 SetHandler weblogic-handler
```

```
</Location>
```

Then in Oracle Access Manager, set the protection of the resources in the Application Domain that has been registered for the SIM application. You must protect the /sim-client/launch resource and exclude the rest:

Resource URL: /sim-client/launch
Protection Level: Protected
Authentication Policy: Protected Resource Policy
Authorization Policy: Protected Resource Policy

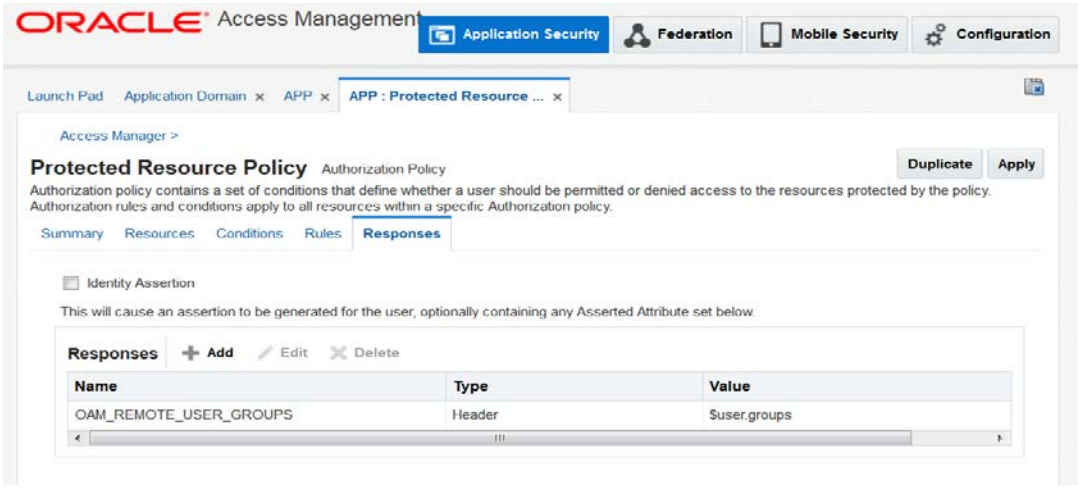
Resource URL: /sim-client/.../*
Protection Level: Excluded

Resource URL: /favicon.ico
Protection Level: Excluded

In the OAM you need to add a response to the Protected Resource Policy:

1. Under Access Manager click Application Domains.
2. Search and click on the domain used for your SIM deployment.
3. Click Authorization Policies.
4. Click Protected Resource Policy.
5. Click Responses.
6. Click Add and enter the following values:
   - Type: Header
   - Name: OAM_REMOTE_USER_GROUPS
   - Value: $user.groups

   It will look similar to the following:



SIM Batch Scripts

The SIM batch programs are installed into the WEBLOGIC_DOMAIN_HOME location that was specified during application installation.

The batch programs can be run from a different location if you cannot run them from under the application server <WEBLOGIC_DOMAIN_HOME>. To install the batch files in a different location just copy the entire batch folder to the appropriate destination.

The batch directory is assumed to be located on the same server as the application server. If you copy the SIM batch directory to a location on a different server, then you need to configure the file path to the sim-batch.log file, which is defined in batch/resources/log4j.xml.

See the "Batch Detail" section of the *Oracle Retail Store Inventory Management Operations Guide* for information about how to run batches.

Resolving Errors Encountered During Application Installation

If the application installer encounters any errors, it halts execution immediately. You can run the installer in silent mode so that you do not have to retype the settings for your environment. See Appendix D of this document for instructions on silent mode.

See "Appendix: Common Installation Errors "for a list of common installation errors.

Since the application installation is a full reinstall every time, any previous partial installs are overwritten by the successful installation.

Web Help Files

The application installer automatically copies the web help files to the proper location. They are accessible from the help links within the application.

Starting and Stopping the Wavelink Server

In order to use handheld wireless devices with SIM, the Wavelink server must be running. The SIM application installer installs, configures, and starts the Wavelink server for you, so once the SIM application install is complete, the Wavelink server is ready to be used.

> **Note:** Even if you use the AdminServer to restart SIM, you will still need to restart the Wavelink server manually.

The Wavelink server scripts are installed into the <sim-wireless-directory>/bin.

The following is an example for stopping and starting the Wavelink server:

```
# cd /u00/webadmin/config/domains/wls_retail/SIMDomain/retail/sim16/wireless/bin
# ./wavelink-shutdown.sh
# ./wavelink-startup.sh
```

> **Note:** The wireless functionality in SIM is dependent on Wavelink and includes a client and server component. Wavelink software ensures that the wireless user interface of SIM can work with various handheld devices.
>
> For the handheld to interact correctly with SIM, it is required to install the appropriate Wavelink studio client. The Wavelink studio client and its installation instructions can be found at http://www.wavelink.com/download/downloads.aspx.

The Oracle Retail Wireless Foundation Server is bundled with the SIM server. It has a single session free license. For multiple sessions additional licenses need to be obtained.

Contact your Oracle sales representative or client partner for Wavelink Studio Client and Oracle Retail Wireless Foundation Server license information.

Note: For configurations of physical handheld devices or wireless network setup, check your hardware manufacturer's manual or Wavelink's studio client information. This information is not covered in this guide.

6

Test the SIM Application

Once SIM database and application are installed, foundation data is imported into SIM, you should have a working SIM application installation. To launch the application client, open a web browser and go to the client URL. You can find the URL in the next steps section of the log file that was produced by the installer.

Example:

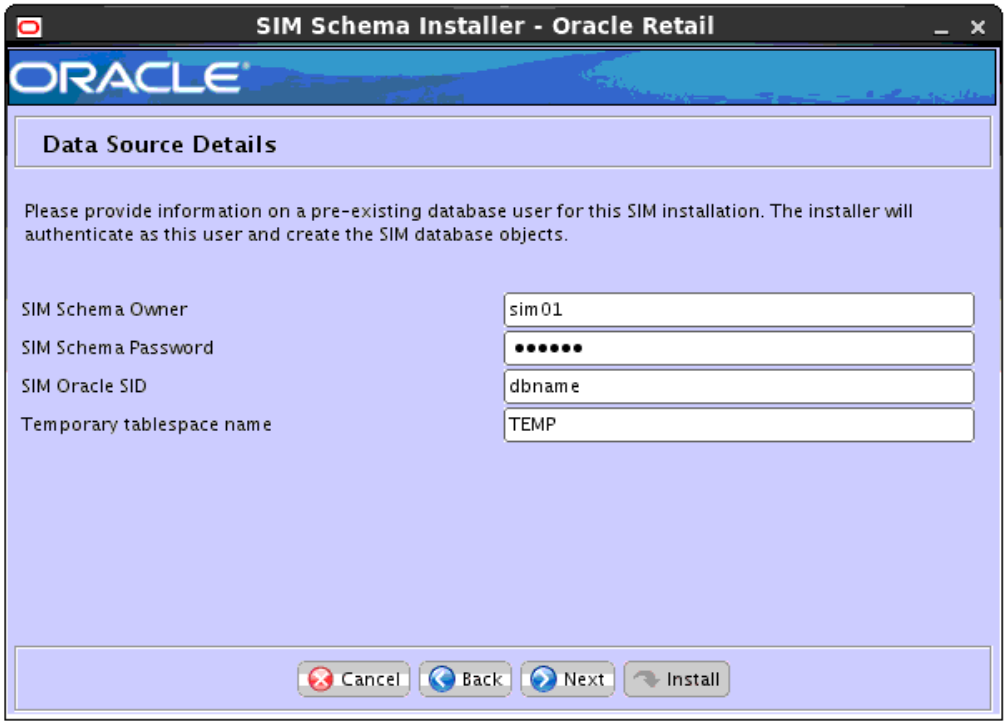
WLS: http://apphost:7143/sim-client/launch

A

Appendix: SIM Database Schema Installer Screens

You need the following details about your environment for the installer to successfully install the SIM database schema. Depending on the options you select, you may not see some screens.

Screen: Data Source Details



Field Title	SIM Schema Owner
Field Description	The pre-existing database user for this installation.
Example	sim01

Field Title	Sim Schema Password
Field Description	The SIM Schema Owner's password.

Field Title	SIM Oracle SID
Field Description	The name of the database or pluggable db service name where the SIM schema will be installed.
Example	dbname

Field Title	Temporary tablespace name
Field Description	Temporary tablespace provided to the create_user_sim_owner.sql script at the time that the SIM database user was created.
Example	TEMP

Screen: Data Source Users Details

Field Title	SIM Database Admin User Name
Field Description	The pre-existing database admin user for this installation.
Example	Sim01_admin

Field Title	SIM Database Admin User Password
Field Description	The SIM database admin user's password.

Field Title	SIM Database Business User Name
Field Description	The pre-existing database business user for this installation.
Example	Sim01_business

Field Title	SIM Database Business User Password
Field Description	The SIM database business user's password.

Field Title	SIM Database Business Viewer User Name
Field Description	The pre-existing database business viewer user for this installation.
Example	Sim01_business_viewer

Field Title	SIM Database Business Viewer User Password
Field Description	The SIM database business viewer user's password.

Field Title	SIM Database MPS User Name
Field Description	The pre-existing database MPS user for this installation.
Example	Sim01_mps

Field Title	SIM Database MPS User Password
Field Description	The SIM database MPS user's password.

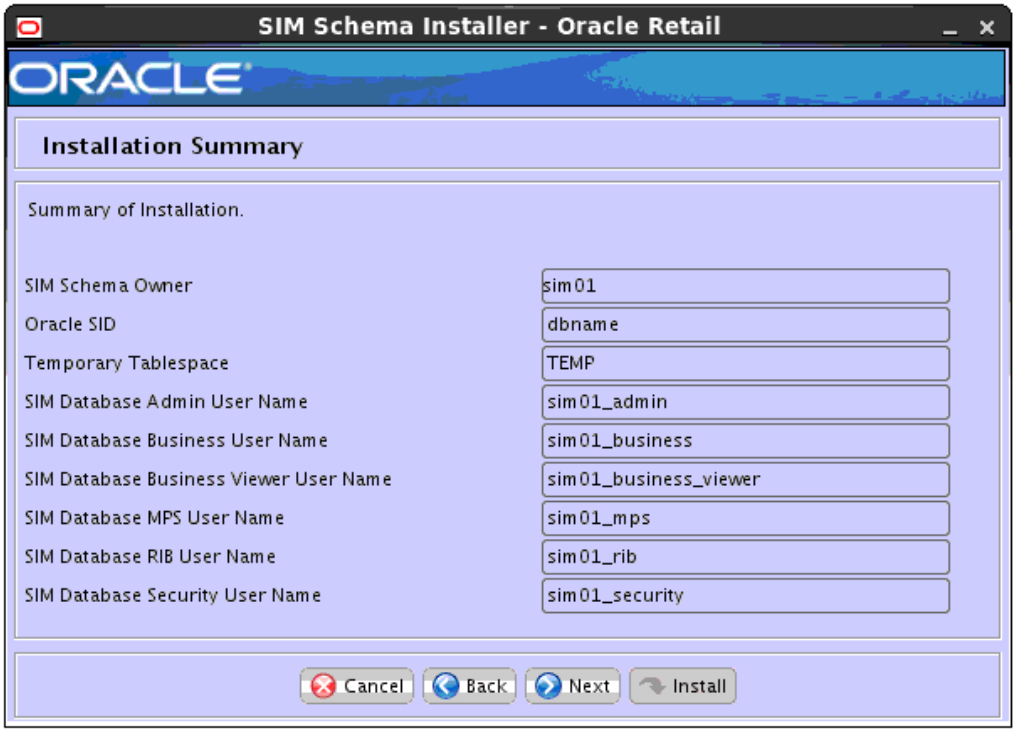
Field Title	SIM Database RIB User Name
Field Description	The pre-existing database RIB user for this installation.
Example	Sim01_rib

Field Title	SIM Database RIB User Password
Field Description	The SIM database RIB user's password.

Field Title	SIM Database Security User Name
Field Description	The pre-existing database security user for this installation.
Example	Sim01_security

Field Title	SIM Database Security User Password
Field Description	The SIM database security user's password.

Screen: Installation Summary

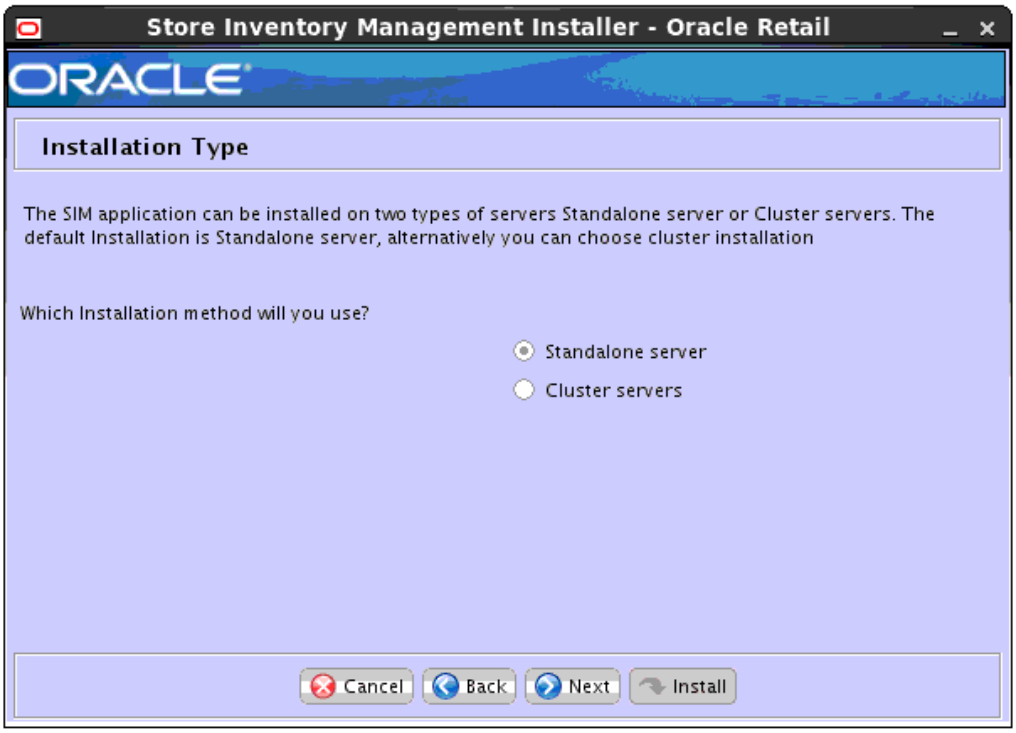


B

Appendix: SIM Application WebLogic Server Installer Screens

You need the following details about your environment for the installer to successfully deploy the SIM application. Depending on the options you select, you may not see some screens.

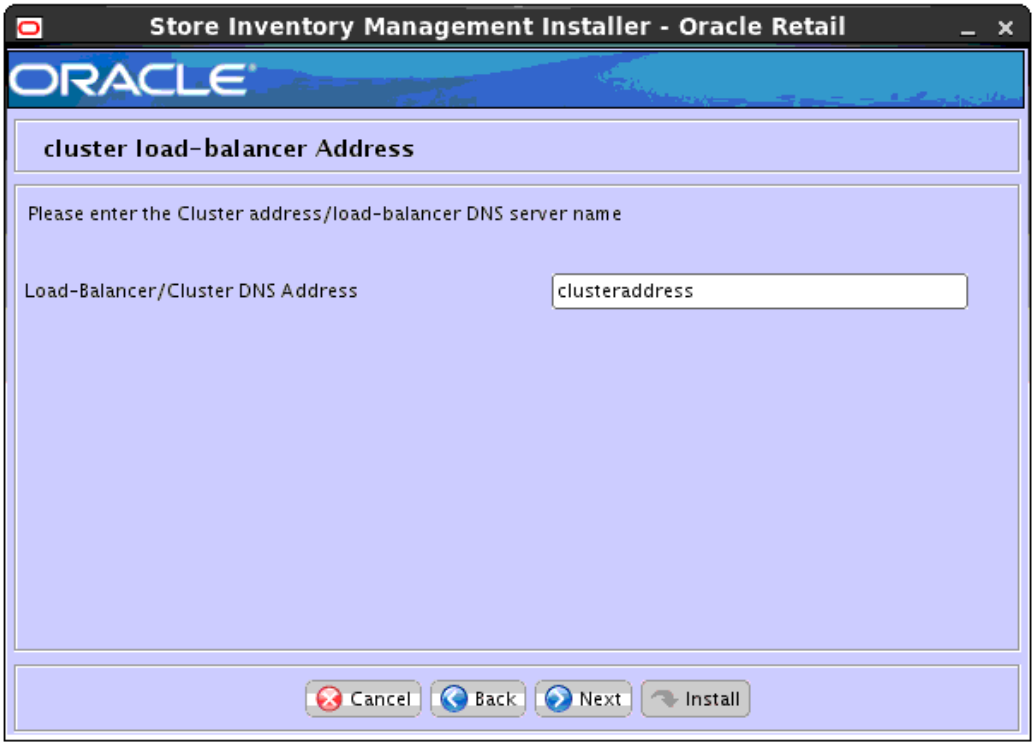
Screen: Installation Type



Field Title	Which Installation Method will you use?
Field Description	Choosing "Standalone server" will deploy SIM to a non-clustered environment, if "Cluster Servers" is chosen then it will deploy SIM to a cluster of servers defined in WebLogic.

Screen: Cluster load-balancer Address

This screen will be displayed, if Cluster Servers option is selected in "Installation Type" screen.



Field Title	Load-Balancer/Cluster DNS Address
Field Description	This contains Virtual Host name of the load balancer that will be used if SIM is to be deployed to a clustered environment. Note, this screen will not appear in case you select Standalone server in previous screen.

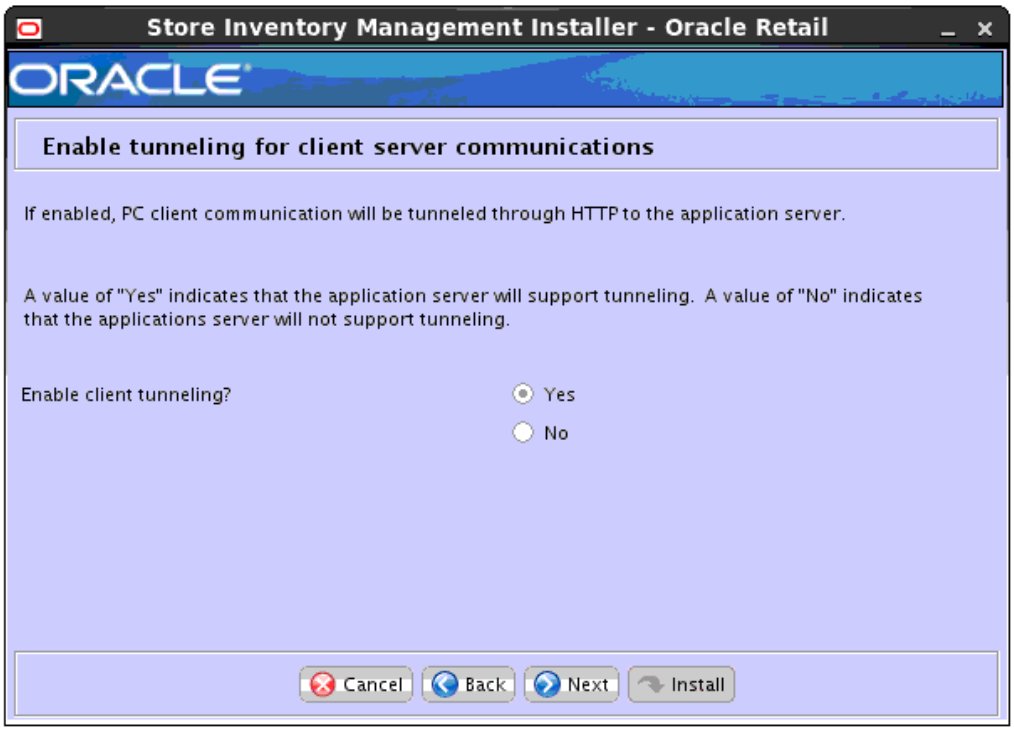
Screen: Security Details

Field Title	Enable SSL for SIM?
Field Description	Choosing yes will deploy SIM using SSL, and will configure SIM to use SSL. In this case, SSL must be configured and enabled for the admin server and SIM managed server or cluster. Choosing no will deploy and configure SIM without SSL.

Screen: Turn off the application server's non-SSL port

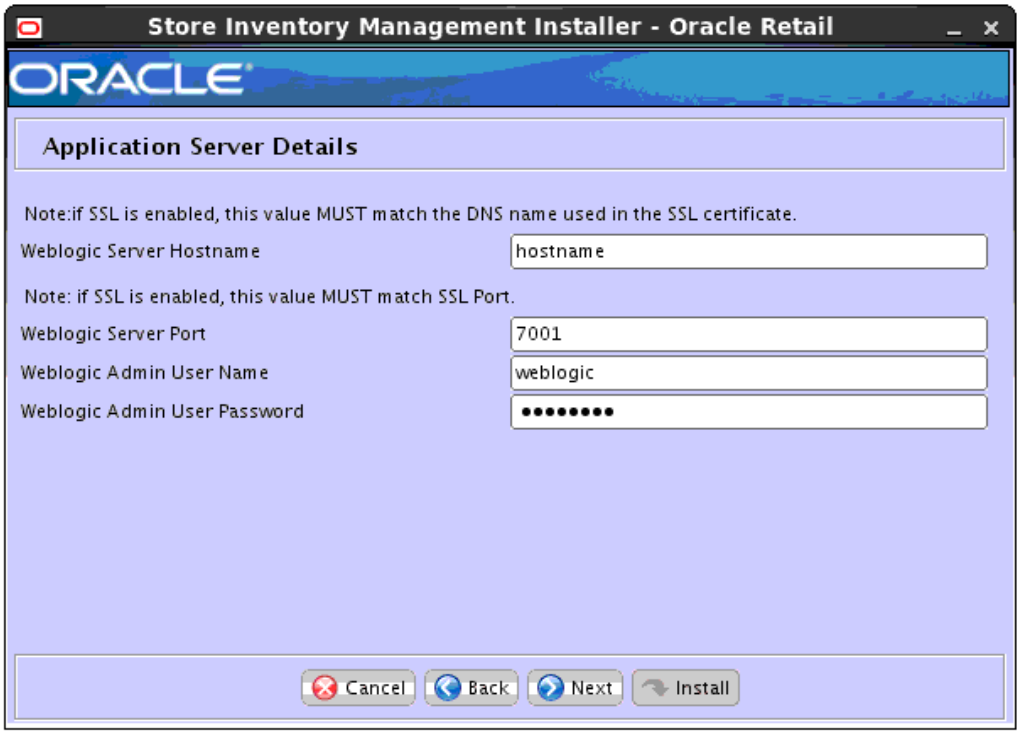
Field Title	Disable non SSL port?
Field Description	Selecting Yes will make that the application server's non – SSL port inactive and a Selecting No will keep application server's non-SSL port active.

Screen: Enable Tunneling for Client Server Communications



Field Title	Enable Tunneling for Client Server Communications?
Field Description	Selecting Yes indicates that the application server will support tunneling. Selecting No indicates that the application server will not support tunneling.

Screen: Application Server Details



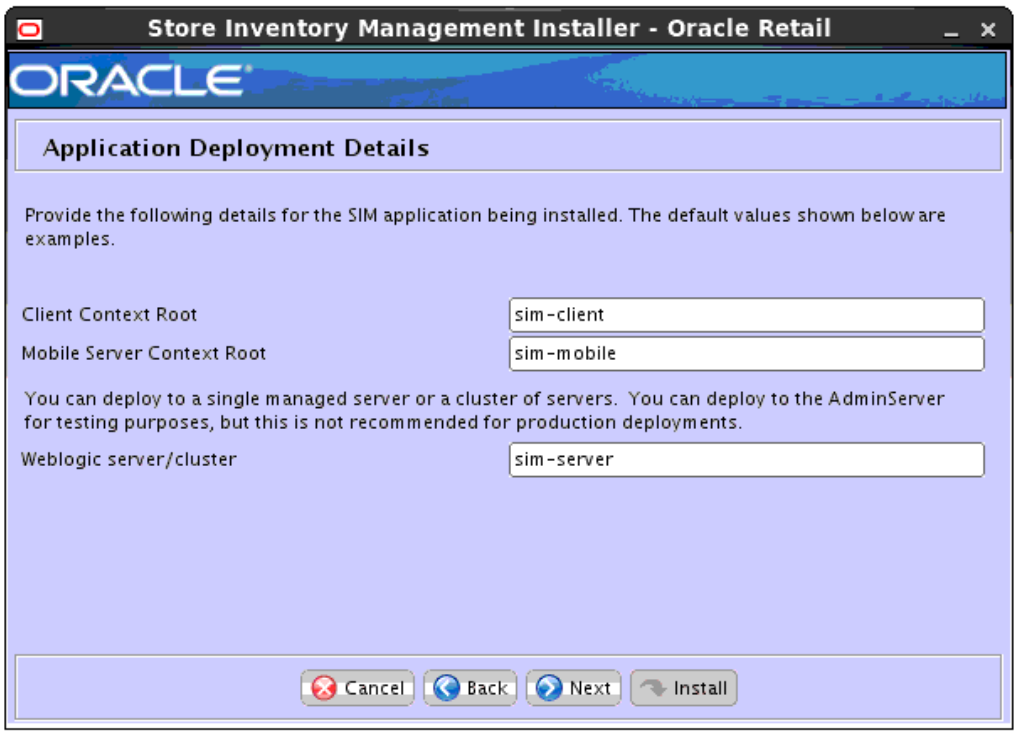
Field Title	WebLogic Server Hostname
Field Description	The hostname of the server where the WebLogic server is installed.
Example	hostname
Notes	Used by installer scripts to install the application and to create default inputs for client codebase and JNDI provider URL.

Field Title	WebLogic Server Port
Field Description	Listen port for the WebLogic Admin server.
Example	7001

Field Title	WebLogic Admin User Name
Field Description	The WebLogic user which will be used to install the SIM application.
Example	weblogic
Notes	Used by installer scripts to install the application

Field Title	WebLogic Admin User Password
Field Description	The password of the WebLogic Admin User used above.
Notes	Used by installer scripts to install the application

Screen: Application Deployment Details



Field Title	Client Context Root
Field Description	Context root for sim client.
Example	sim-client

Field Title	Mobile Server Context Root
Field Description	This is the managed server name for mobile deployment.
Example	sim-mobile

Field Title	WebLogic server/cluster
Field Description	This is the managed server name for standalone deployment and cluster name for deployment to clustered managed servers.
Example	sim-server

Screen: Choose Apps to Integrate with SIM

Field Title	Configure RIB for SIM?
Field Description	Select this option if you will be using RIB with SIM. Note, if you select this option then RIB Integration Details screen will be enabled and appropriate details have to be entered in RIB Integration Details screen.

Field Title	Configure RPM for SIM?
Field Description	Select this option if you will be using RPM with SIM. Note, if you select this option then RPM Integration Details screen will be enabled and appropriate details have to be entered in RPM Integration Details screen.

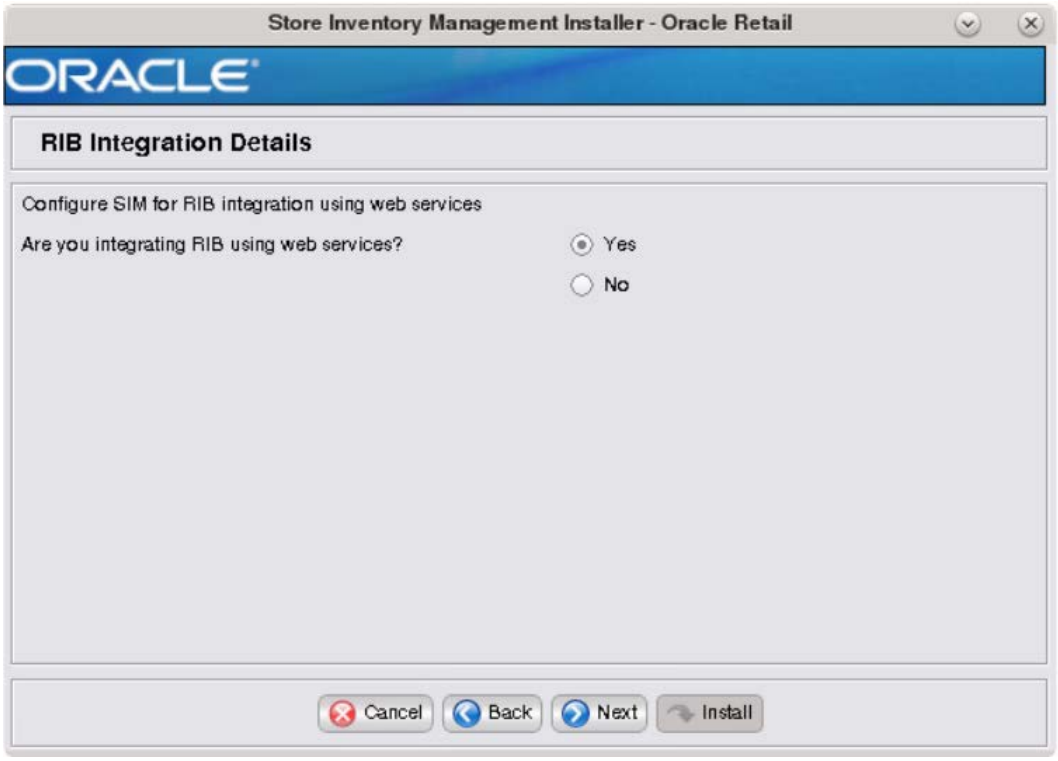
Field Title	Configure RMS for SIM?
Field Description	Select this option if you will be using RMS with SIM. Note, if you select this option then RMS Integration Details screen will be enabled and appropriate details have to be entered in RMS Integration Details screen.

Field Title	Configure Manifest for SIM? **Note**: Refer to the *Oracle Retail Store Inventory Management Operations Guide* for more information.
Field Description	Manifest integration is configured if an external Shipment Management System is to be used in conjunction with SIM. (Optional). Note, if you select this option then the Manifest Integration Details screen will be enabled and appropriate details will be entered in the subsequent Manifest Integration Details screen.

Field Title	Configure OMS for SIM? **Note**: Refer to the *Oracle Retail Store Inventory Management Operations Guide* for more information.
Field Description	OMS integration is configured if an external Shipment Management System is to be used in conjunction with SIM. (Optional). Note, if you select this option then OMS Integration Details screen will be enabled and appropriate details will be entered in the subsequent OMS Integration Details screen.

Screen: RIB Integration Details

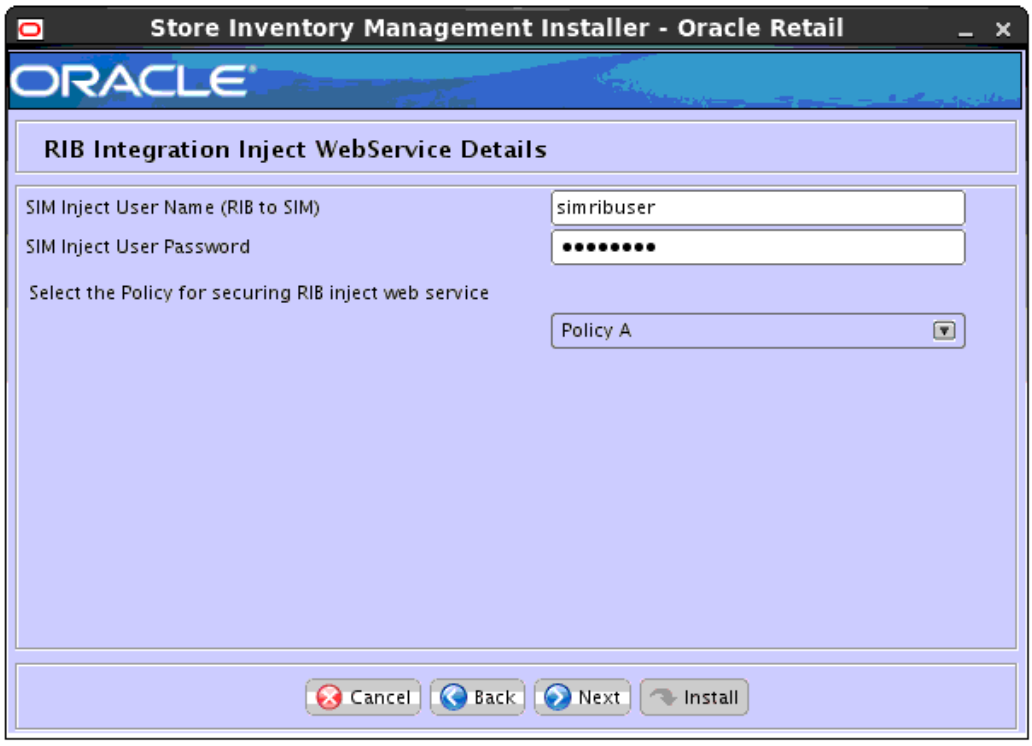
This screen will be displayed if the Configure RIB for SIM option is checked on the Choose Apps to Integrate with SIM screen.



Field Title	Are you integrating RIB using web services?
Field Description	This field provides an option for integrating RIB using web services on integration with SIM. Select "Yes" if you are integrating RIB using web services. Select "No" if you are not integrating RIB using web services

Screen: RIB Integration Inject WebService Details

This page will appear if you selected "Yes" in the previous screen.

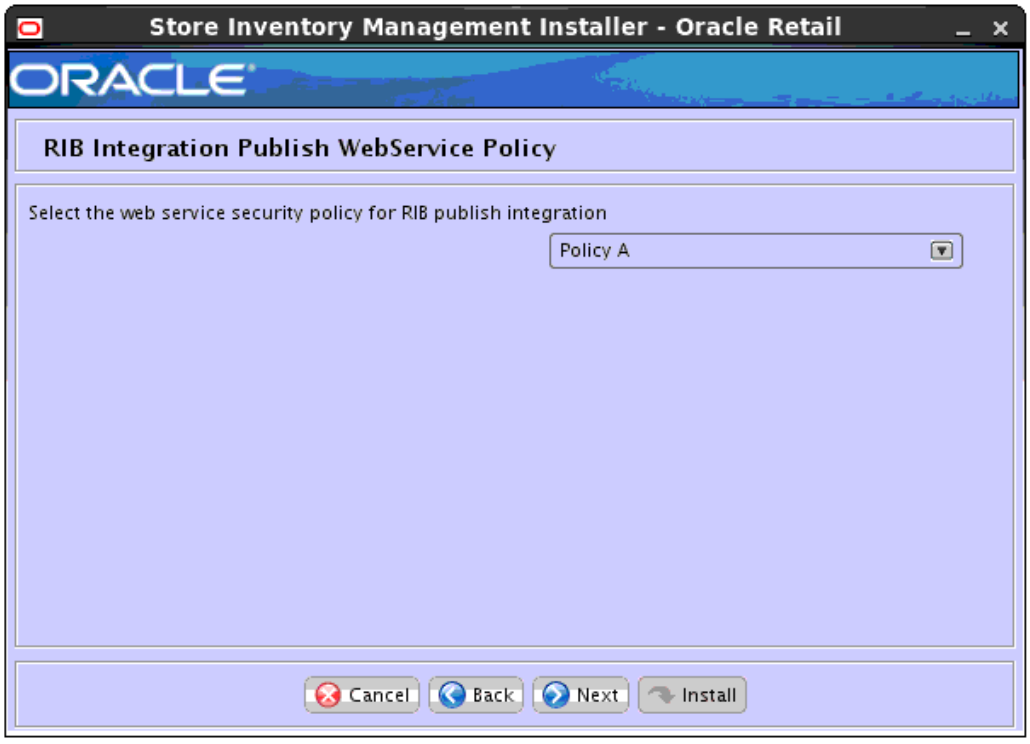


Field Title	SIM Inject User Name
Field Description	The SIM inject user (RIB to SIM) is the user that the installer will setup for RIB to use for injection to SIM. This same user/pass should be used in the RIB-SIM installation for the credentials to call SIM for injection. Note that this user ONLY exists in the SIM server, you do NOT need to create this user in the RIB-SIM server.
Example	simribuser

Field Title	SIM Inject User Password
Field Description	Password for the SIM inject user.

Field Title	Select the Policy for securing RIB inject web service
Field Description	Select the Policy for securing RIB inject web service. Please refer to the *Oracle Retail Store Inventory Management Security Guide* to learn more about Policy A and Policy B.

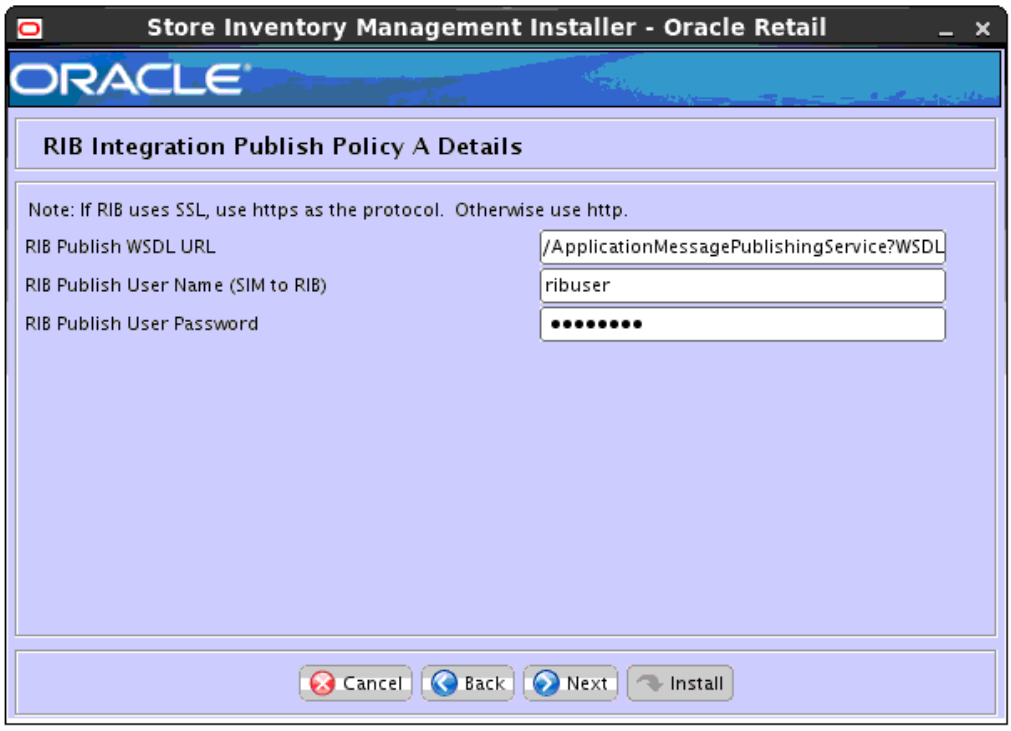
Screen: RIB Integration Publish WebService Policy



Field Title	Select the web service security policy for integration with RIB.
Field Description	Select the web service security policy for integration with RIB. Please refer to the *Oracle Retail Store Inventory Management Security Guide* to learn more about Policy A and Policy B.

Screen: RIB Integration Publish Policy A Details

This screen will appear if you selected "Policy A" in the previous screen.



Field Title	RIB Publish WSDL URL
Field Description	This is the provider URL for RIB Publish WSDL
Example	https://hostname:portno/ApplicationMessagePublishingServiceBean/ApplicationMessagePublishingService?WSDL

Field Title	RIB Publish User Name (SIM to RIB)
Field Description	The SIM publish user (SIM to RIB) is the user that SIM will be configured to use when publishing messages from SIM. This user does NOT exist in the SIM server and will NOT be created by the installer. This user should exist in the RIB-SIM server and should be created as part of the RIB-SIM install.
Example	ribuser

Field Title	RIB Publish User Password
Field Description	This is the password used for the above RIB Publish username
Example	https://hostname:portno/ApplicationMessagePublishingServiceBean/ApplicationMessagePublishingService?WSDL

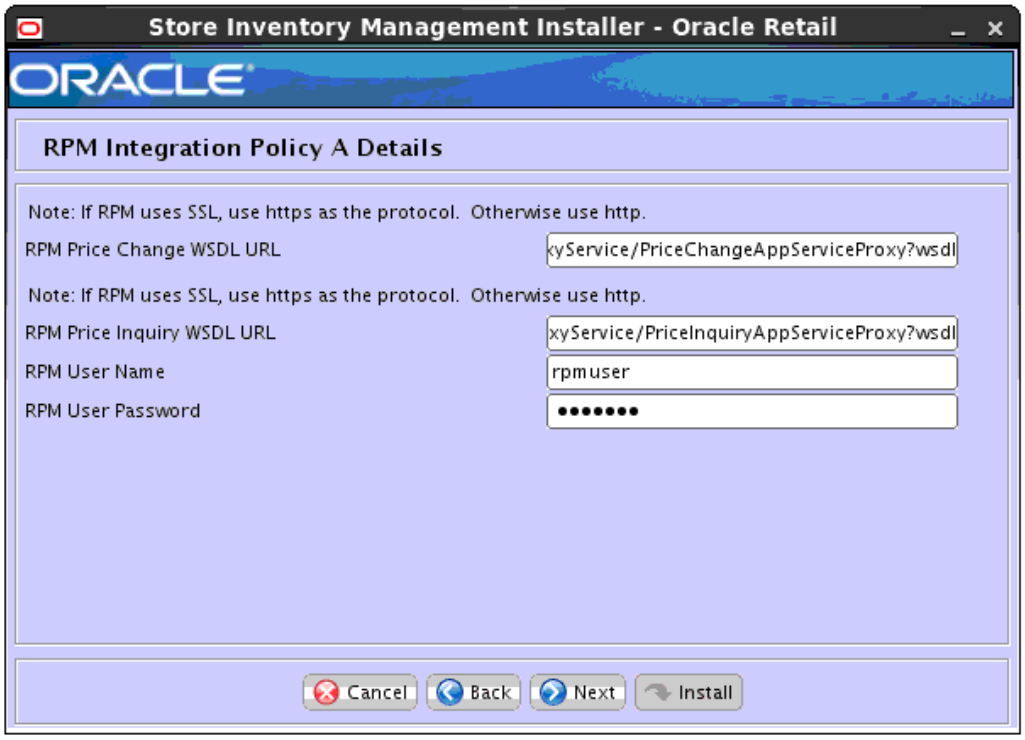
Screen: RPM Web service Policy

This screen will be displayed if the Configure RPM for SIM option is checked on the Choose Apps to Integrate with SIM screen.

Field Title	Select the web service security policy for integration with RPM.
Field Description	Select the web service security policy for integration with RPM. Please refer to the *Oracle Retail Store Inventory Management Security Guide* to learn more about Policy A and Policy B.

Screen: RPM Integration Policy A Details

This screen will be displayed if the Configure RPM for SIM option is checked on the Choose Apps to Integrate with SIM screen and "Policy A" is selected in the previous screen.



Note: If the user chooses to integrate SIM with RPM then RPM installation is a pre-requisite to installing SIM.

Field Title	RPM Price Change WSDL URL
Field Description	This is the provider URL for RPM Price change WSDL.
Example	https://hostname:portno/rpm-PriceChange-AppServiceDecorator/ProxyService/PriceChangeAppServiceProxy?wsdl

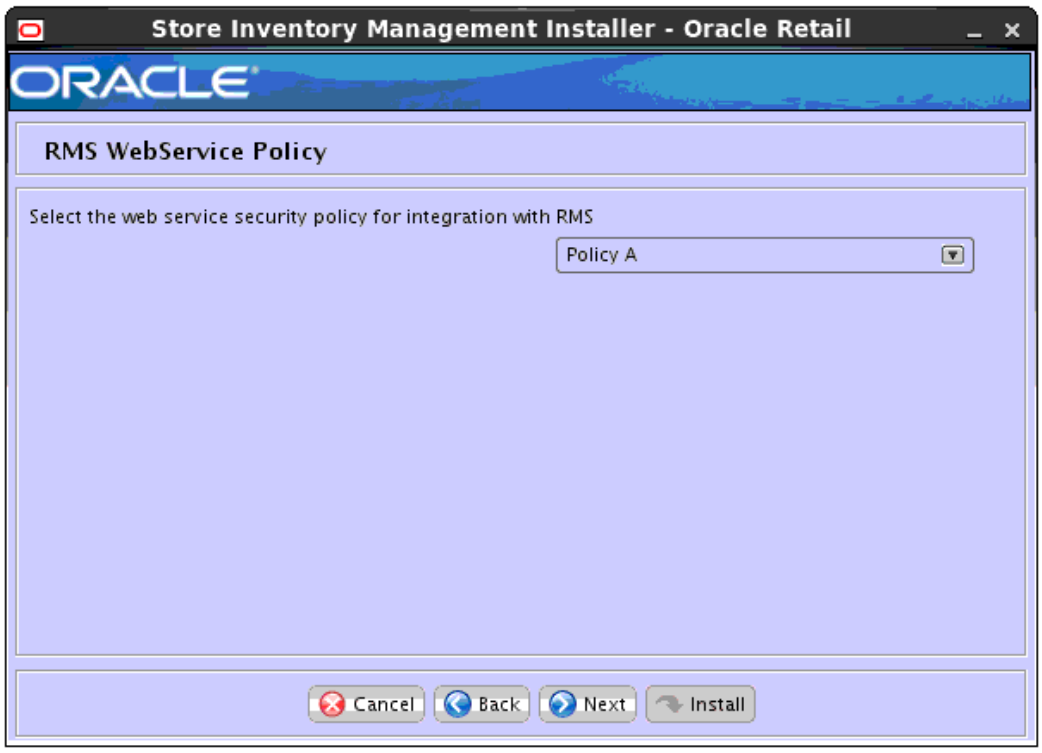
Field Title	RPM Price Inquiry WSDL URL
Field Description	This is the provider URL for RPM Price Inquiry WSDL.
Example	https://hostname:portno/rpm-PriceInquiry-AppServiceDecorator/ProxyService/PriceInquiryAppServiceProxy?wsdl

Field Title	RPM User Name
Field Description	This is the username used while integrating SIM with RPM
Example	rpmuser

Field Title	RPM Password
Field Description	This is the password used while integrating SIM with RPM

Screen: RMS Web service Policy

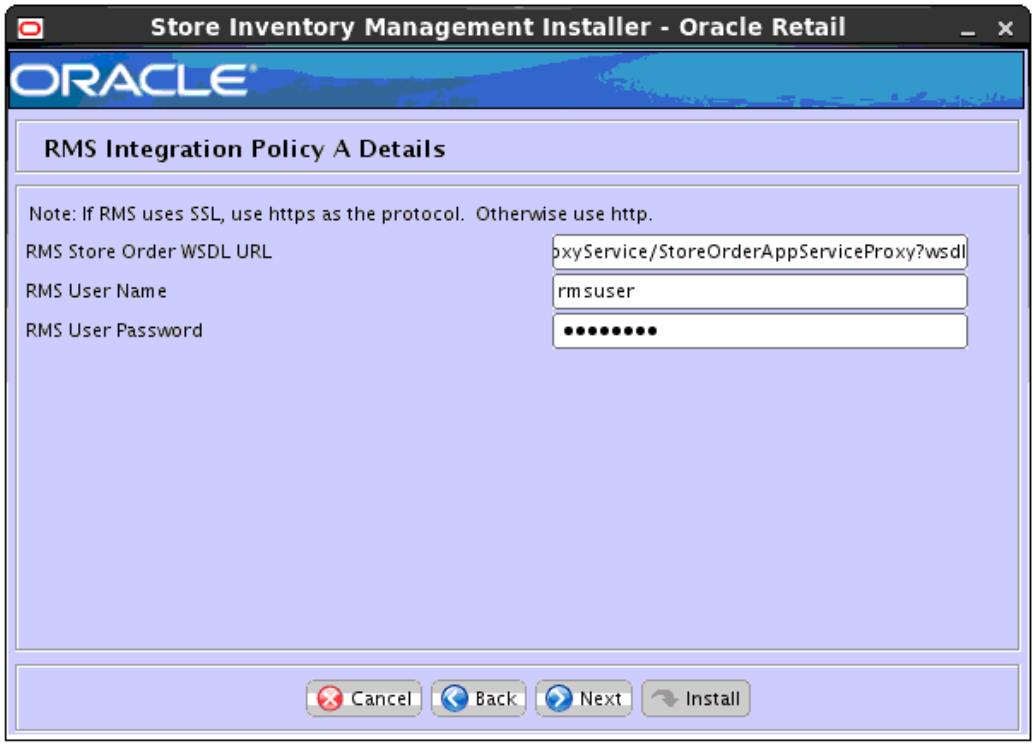
This screen will be displayed if the Configure RMS for SIM option is checked on the Choose Apps to Integrate with SIM screen.



Field Title	Select the web service security policy for Integration with RMS.
Field Description	Select the type of web service security policy for integration with RMS. Please refer to the *Oracle Retail Store Inventory Management Security Guide* to learn more about Policy A and Policy B.
Example	None, PolicyA, PolicyB.

Screen: RMS Integration Policy A Details

This screen will be displayed if the Configure RMS for SIM option is checked on the Choose Apps to Integrate with SIM screen and if you select "Policy A" in the previous screen.



Note: If the user chooses to integrate SIM with RMS then RMS installation is pre-requisite to install SIM.

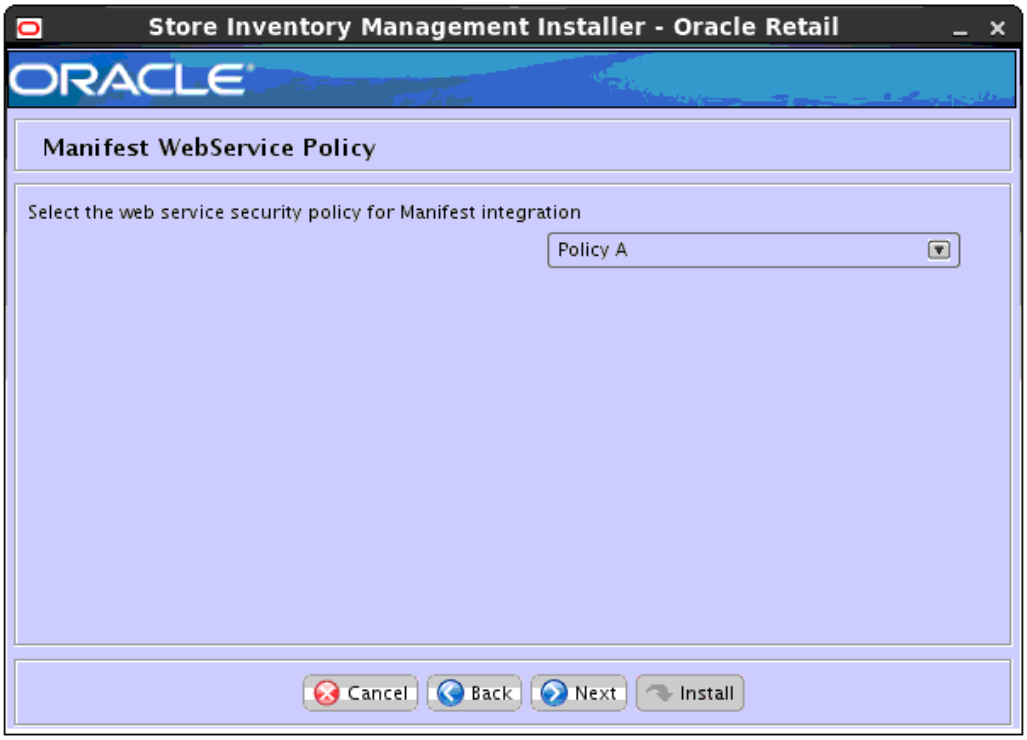
Field Title	RMS Store Order WSDL URL
Field Description	This is the provider URL for RMS Store Order WSDL.
Example	https://hostname:portno/rms-StoreOrder-AppServiceDecorator/ProxyService/StoreOrderAppServiceProxy?wsdl

Field Title	RMS User Name
Field Description	This is the username used while integrating SIM with RMS
Example	rmsuser

Field Title	RMS User Password
Field Description	This is the password used while integrating SIM with RMS

Screen: Manifest Web service Policy

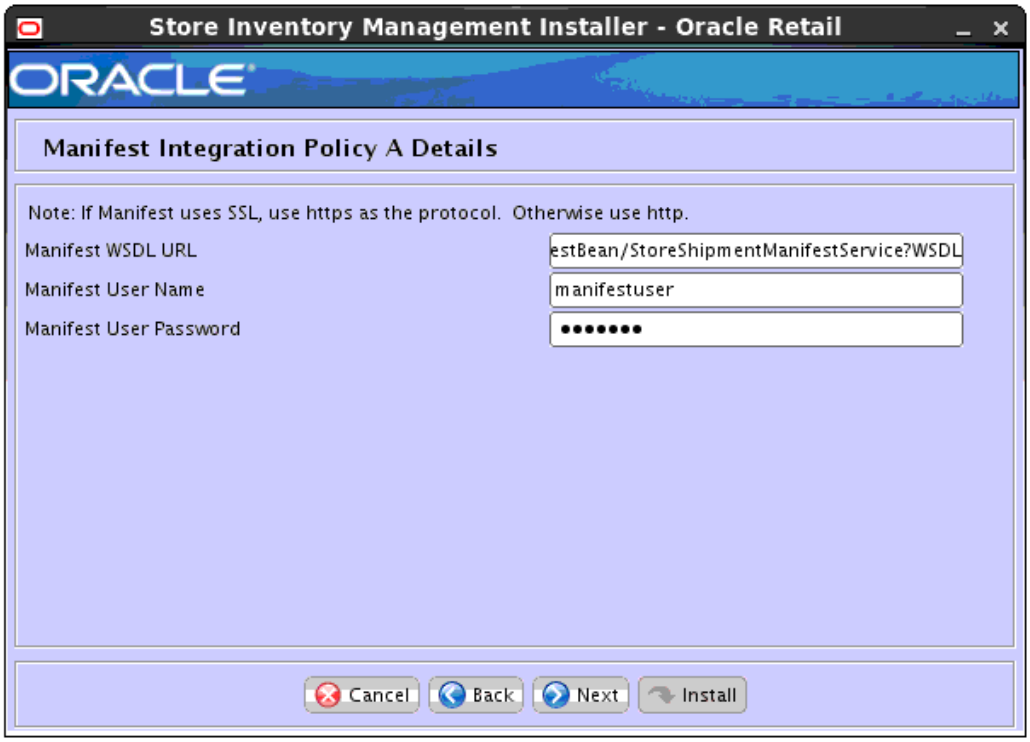
This screen will be displayed if the Configure Manifest for SIM option is checked on the Choose Apps to Integrate with SIM screen.



Field Title	Select the web service security policy for Manifest Integration
Field Description	Select the type of web service security policy for Manifest Integration. Please refer to the *Oracle Retail Store Inventory Management Security Guide* to learn more about Policy A and Policy B.
Example	None, PolicyA, PolicyB

Screen: Manifest Integration Policy A Details

This screen will be displayed if the Configure Manifest for SIM option is checked on the Choose Apps to Integrate with SIM screen and if "Policy A" was selected in the previous screen



Field Title	Manifest WSDL URL
Field Description	This is the provider URL for Manifest WSDL, when SIM is integrated with the manifest system. **Note**: Refer to the *Oracle Retail Store Inventory Management Operations Guide* for more information.
Example	https://hostname:portno/StoreShipmentManifestBean/StoreShipmentManifestService?WSDL

Field Title	Manifest User Name
Field Description	This is the username used while integrating SIM with Manifest
Example	manifestuser

Field Title	Manifest User Password
Field Description	This is the password used while integrating SIM with Manifest

Screen: OMS Web service Policy

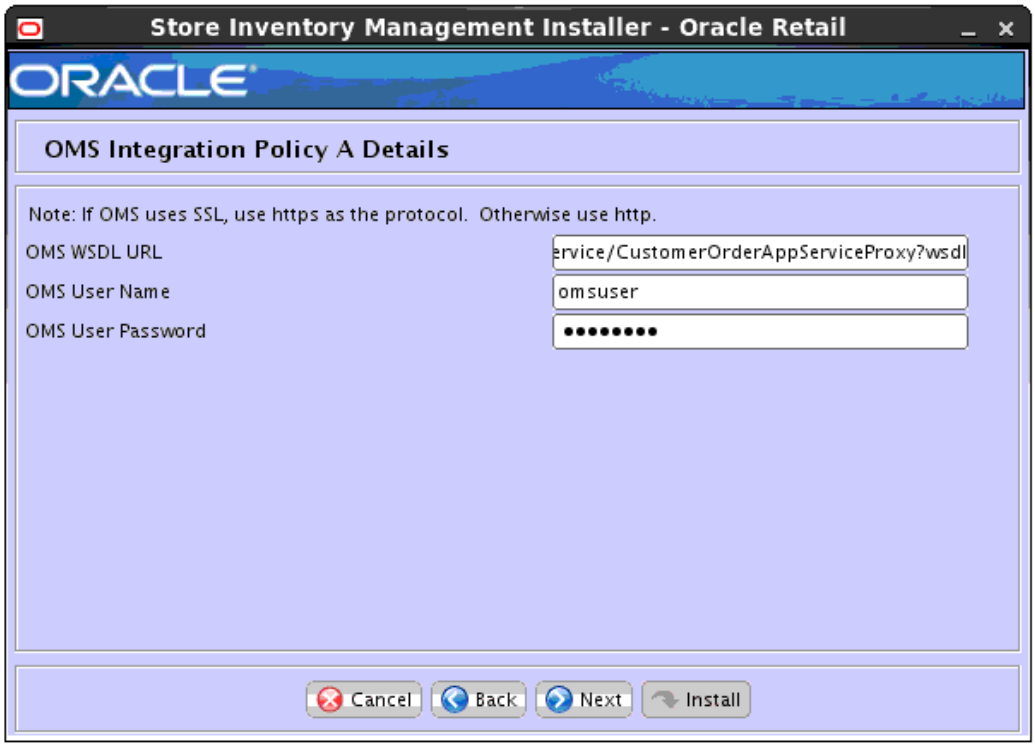
This screen will be displayed, if Configure OMS for SIM option is checked on the Choose Apps to Integrate with SIM screen.

Note: This screen will appear when user chooses to integrate SIM with OMS.

Field Title	Select the web service security policy for integration with OMS
Field Description	Selects the type of web service security policy for integration with OMS. Please refer to the *Oracle Retail Store Inventory Management Security Guide* to learn more about Policy A and Policy B.
Destination	None, PolicyA, PolicyB

Screen: OMS Integration Policy A Details

This screen will be displayed if the Configure OMS for SIM option is checked on the Choose Apps to Integrate with SIM screen and if "Policy A" is selected in the previous screen.



Field Title	OMS WSDL URL
Field Description	This is the provider URL for the OMS WSDL. **Note**: Refer to the *Oracle Retail Store Inventory Management Operations Guide* for more information.
Example	https://hostname:portno/oms-CustomerOrder-AppServiceDecorator/ProxyService/CustomerOrderAppServiceProxy?wsdl

Field Title	OMS User Name
Field Description	This is the username used while integrating SIM with OMS
Example	omsuser

Field Title	OMS User Password
Field Description	This is the password used while integrating SIM with OMS

Screen: JDBC Security Details

Field Title	Enable Secure JDBC connection
Field Description	Select Yes if you have a secured database already set up, otherwise select No.

Screen: Data Source Details

Field Title	SIM JDBC URL
Field Description	URL used by the SIM application to access the SIM database schema.
Destination	WebLogic admin server
Example	Standard Thin Connection: jdbc:oracle:thin:@myhost:1521:mysimsid If it is a pluggable db then use the URL as shown belowjdbc:oracle:thin:@myhost:1521/<service name> RAC connection: jdbc:oracle:thin:@(DESCRIPTION =(ADDRESS_LIST =(ADDRESS = (PROTOCOL = TCP)(HOST = myhost1)(PORT = 1521))(ADDRESS = (PROTOCOL = TCP)(HOST = myhost2)(PORT = 1521))(LOAD_BALANCE = yes))(CONNECT_DATA =(SERVICE_NAME = mysimsid)))

Field Title	SIM Database Schema Owner User Name
Field Description	The schema owner name.
Destination	WebLogic admin server
Notes	The schema owner name should match the name you provided when you ran the SIM database schema installer.

Field Title	SIM Database Schema Owner User Password
Field Description	The password for the SIM schema owner.

Field Title	SIM Database Admin User Name
Field Description	The database admin user name.

Field Title	SIM Database Admin User Password
Field Description	The password for the database admin user.

Field Title	SIM Database Business User Name
Field Description	The database business user name.

Field Title	SIM Database Business User Password
Field Description	The password for the database business user.

Field Title	SIM Database MPS User Name
Field Description	The database MPS user name.

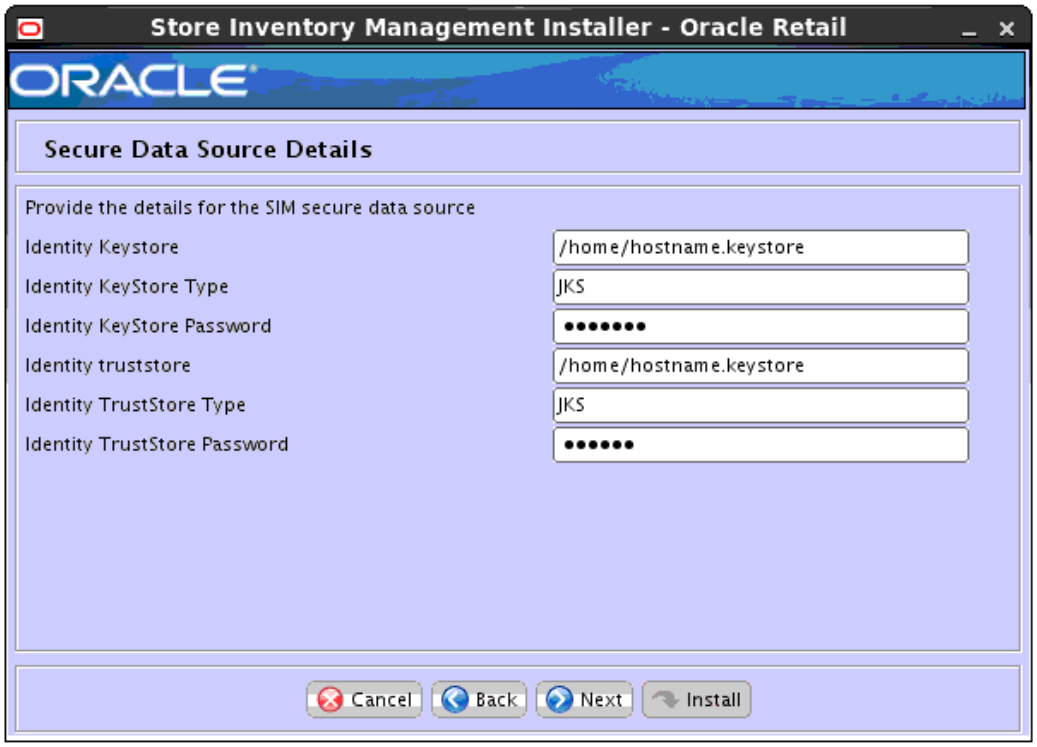
Field Title	SIM Database MPS User Password
Field Description	The password for the database MPS user.

Field Title	SIM Database Security User Name
Field Description	The database security user name.

Field Title	SIM Database Security User Password
Field Description	The password for the database Security user.

Screen: Secure Data Source Details

This screen is displayed if a Secure JDBC connection is enabled.



Field Title	Identity Keystore
Field Description	Path to the identity keystore, for example, /home/hostname.keystore

Field Title	Identity Keystore Type
Field Description	Keystore type i.e: JKS

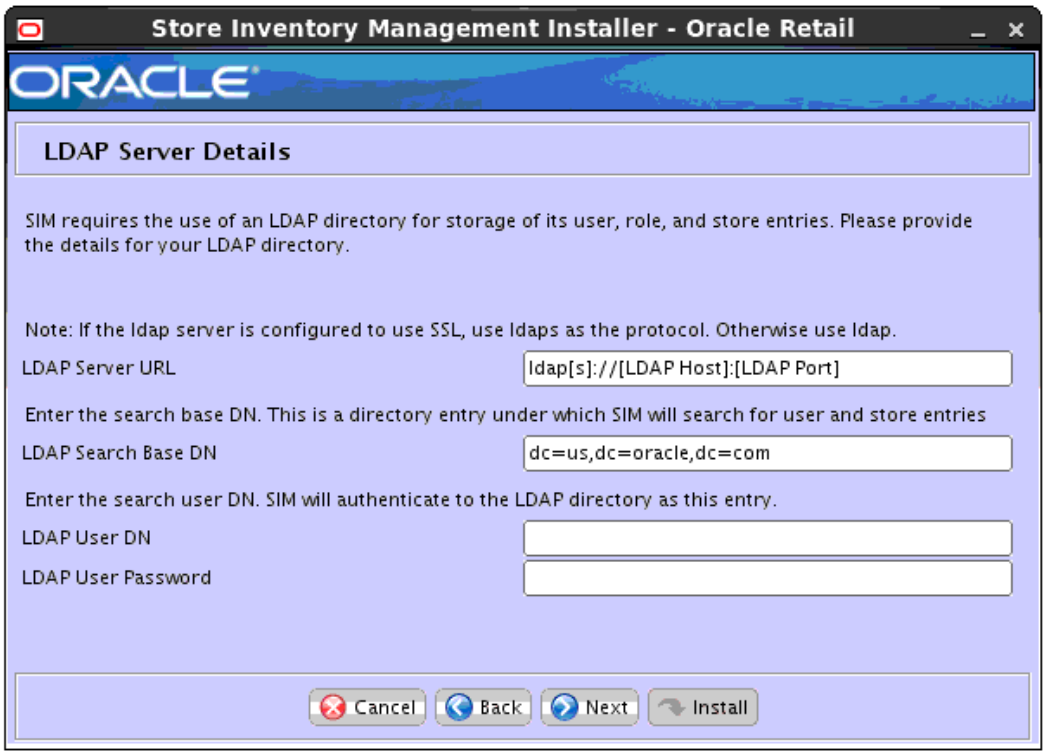
Field Title	Identity Keystore Password
Field Description	Password used to access the identity keystore defined above.

Field Title	Identity TrustStore
Field Description	Path to the identity truststore, for example, /home/hostname.keystore

Field Title	Identity TrustStore Type
Field Description	Keystore type for example, JKS

Field Title	Identity TrustStore Password
Field Description	Password used to access the identity truststore defined above.

Screen: LDAP Server Details



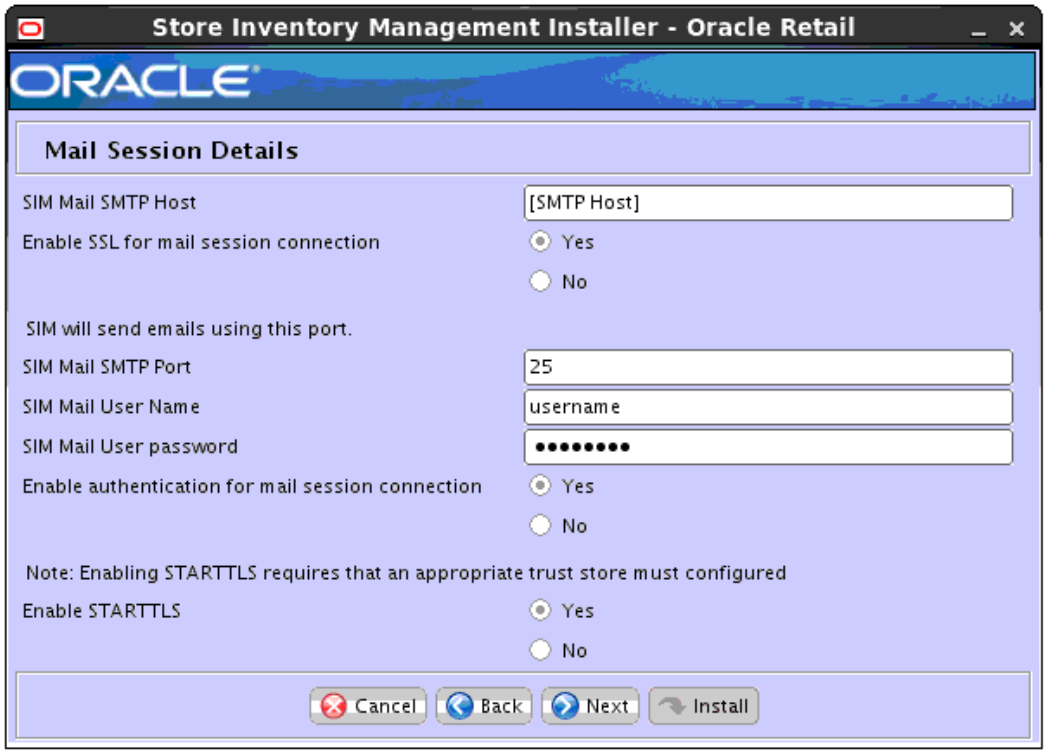
Field Title	LDAP server URL
Field Description	URL for your LDAP directory server.
Example	Non-secured ldap: ldap://myhost:3060/ Secured ldap: ldaps://myhost:2484/

Field Title	LDAP Search Base DN
Field Description	The directory entry under which SIM will search for user and store entries.
Example	dc=us,dc=oracle,dc=com

Field Title	LDAP User DN
Field Description	Distinguished name of the user that SIM uses to authenticate to the LDAP directory.

Field Title	LDAP User Password
Field Description	Password for the search user DN.

Screen: Mail Session Details



Field Title	SIM Mail SMTP Host
Field Description	The SMTP server that will be used to send notification emails from SIM.
Example	mail.oracle.com

Field Title	Enable SSL for Mail session connection
Field Description	Select Yes for secure connection. Select No for plain connection.

Field Title	SIM Mail SMTP Port
Field Description	Port that the mail client is configured to use.

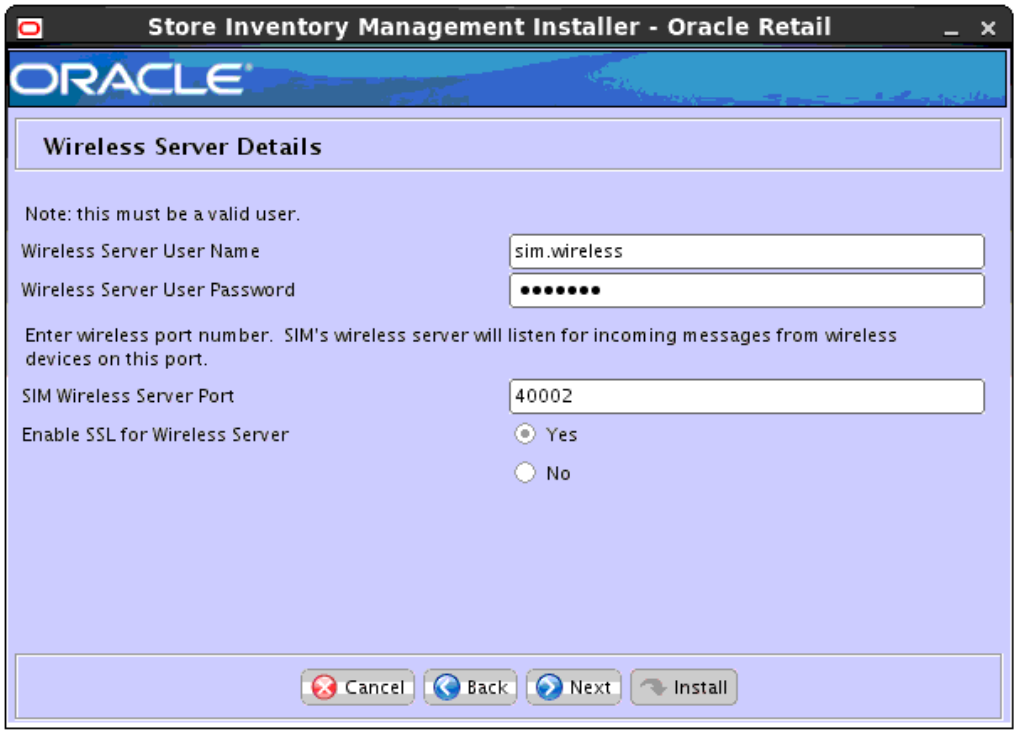
Field Title	SIM Mail User Name
Field Description	Username used to access the mail client.

Field Title	SIM Mail User Password
Field Description	Password for the above user.

Field Title	Enable authentication for mail session connection
Field Description	Yes or no depending on mail client configuration.

Field Title	Enable STARTTLS
Field Description	StartTLS is an email command used to upgrade security of unprotected connections. This is a technical parameter and should be configured according to how the email server is setup.

Screen: Wireless Server Details



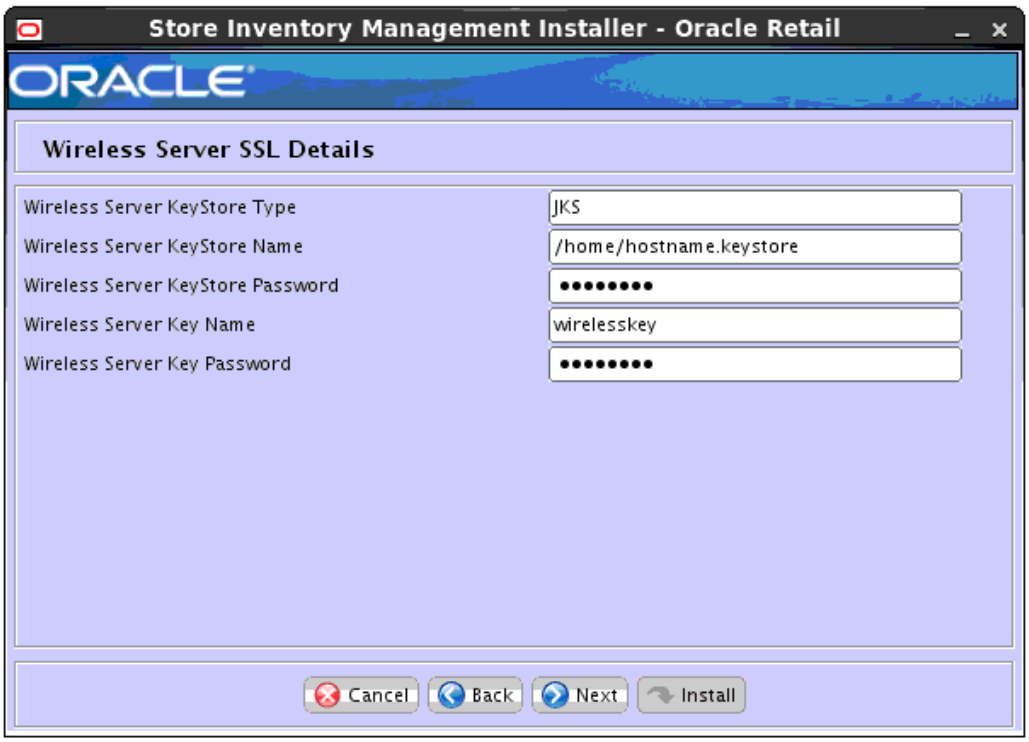
Field Title	Wireless Server User Name
Field Description	User name for wireless server
Destination	Retail config wallet and installer creates WebLogic user with the given name above.

Field Title	Wireless Server User Password
Field Description	Password for wireless server user, the password must follow WebLogic password requirements (at least 8 characters in length and one non-alphabetic character).
Destination	Retail config wallet.

Field Title	SIM Wireless Server Port
Field Description	Choose an available port that the Wavelink server will use to listen for incoming messages from wireless devices.
Destination	wireless.cfg, wavelink-startup.sh
Example	40002

Field Title	Enable SSL for Wireless Server
Field Description	Yes or No depending on SSL configuration.

Screen: Wireless Server SSL Details



Field Title	Wireless Server Keystore Type
Field Description	Keystore type i.e: JKS

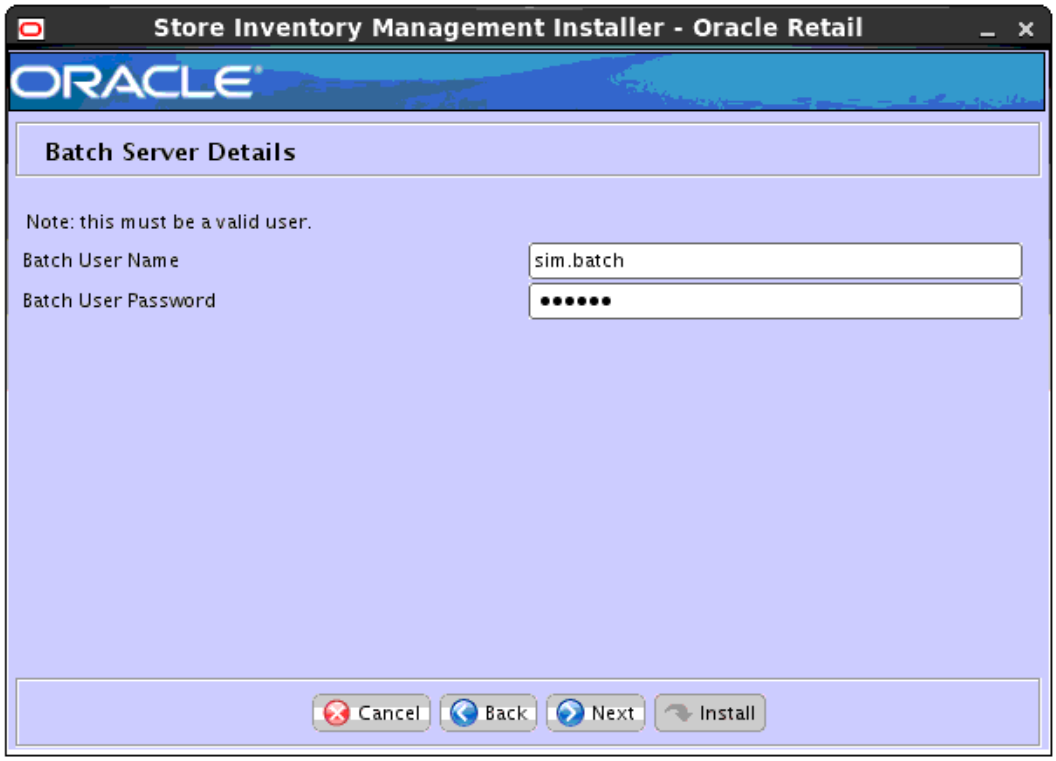
Field Title	Wireless Server Keystore Name
Field Description	Path to the wireless server keystore, for example, /home/hostname.keystore

Field Title	Wireless Server Keystore Password
Field Description	Password used to access the wireless server keystore defined above.

Field Title	Wireless Server Key Name
Field Description	The wireless server key alias name

Field Title	Wireless Server Key Password
Field Description	Password used to access the wireless server key alias name defined above.

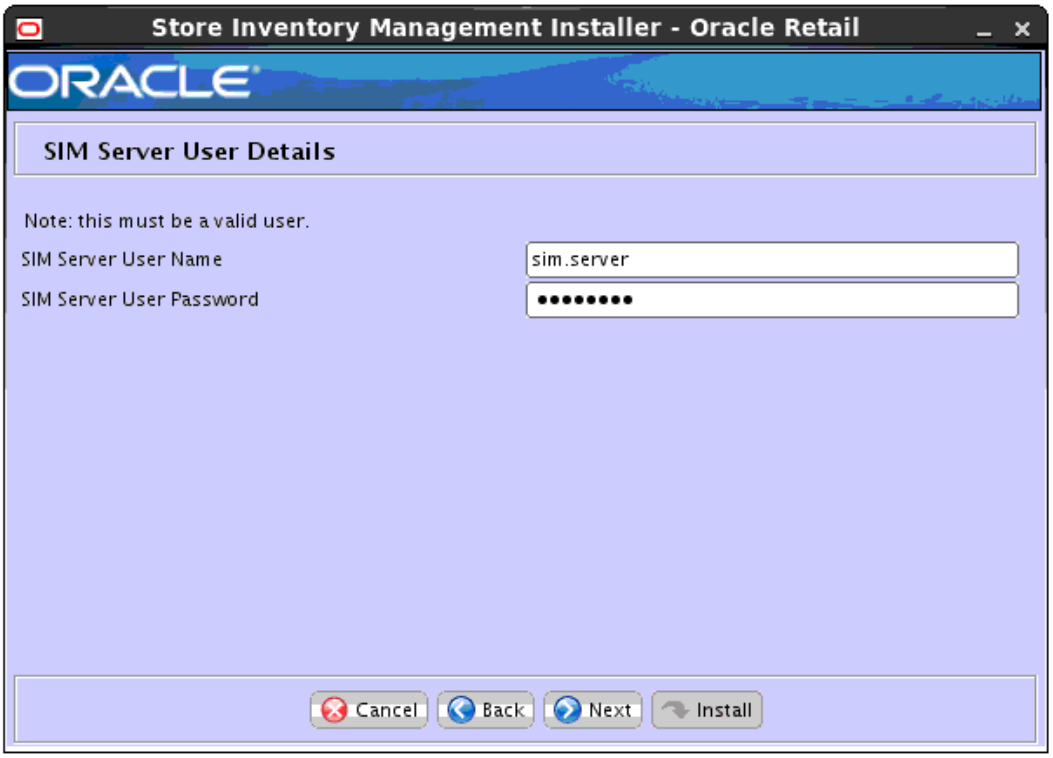
Screen: Batch Server Details



Field Title	Batch User Name
Field Description	User name for Batch.
Destination	Retail config wallet and installer creates WebLogic user with the given name above.

Field Title	Batch User Password
Field Description	Password for batch user, the password must follow weblogic password requirements (at least 8 characters in length and one non-alphabetic character).
Destination	Retail config wallet.

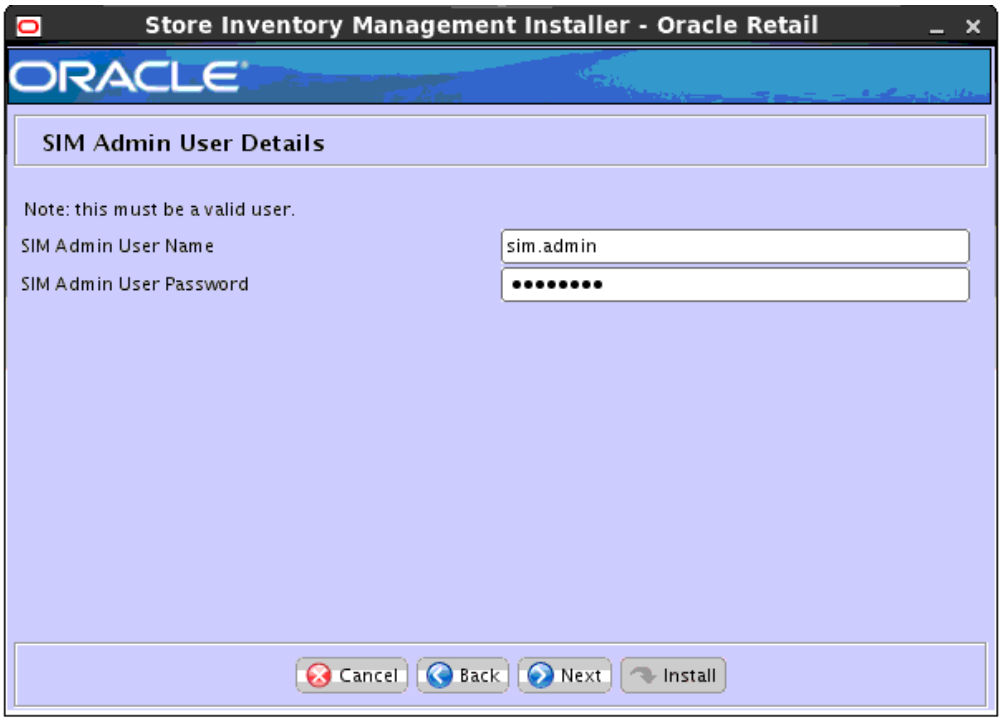
Screen: Server User Details



Field Title	SIM Server User Name
Field Description	User name for SIM Server
Destination	Domain wallet and installer creates WebLogic user with the given name above.

Field Title	SIM Server User Password
Field Description	Password for SIM Server User, the password must follow WebLogic password requirements (at least 8 characters in length and one non-alphabetic character).
Destination	Weblogic Domain wallet/ weblogic default

Screen: SIM Admin User Details



Field Title	SIM Admin User
Field Description	User name for SIM Internal Administrator
Example	sim.admin

Field Title	SIM Admin User Password
Field Description	Password for SIM Internal Administrator

Screen: Internal Security Installation User Details

Field Title	SIM Internal Security Installation User Name
Field Description	User name for SIM Internal Security Installation.
Destination	SIM database user for the SIM application and WebLogic user in database provider authentication. SIM stores are tied to this user. Example: simsecuser

Field Title	SIM Internal Security Installation User Password
Field Description	Password for SIM Internal Security Installation User, the password must follow WebLogic password requirements (at least 8 characters in length and one non-alphabetic character).
Destination	SIM database user for the SIM application and WebLogic user in database provider authentication.

Screen: SIM Webservice Provider Policy

Field Title	Select the policy for securing SIM web service providers
Field Description	Select the type of web service policy for SIM. Please refer to the *Oracle Retail Store Inventory Management Security Guide* to learn more about Policy A and Policy B.
Example	None, PolicyA, PolicyB

Screen: Printing Details

Field Title	Configure SIM reporting for BI publisher
Field Description	Select this option if you will be using BI Publisher for SIM reporting. Note, if you select this option then the "Report BIP Details" screen will be enabled and appropriate details will have to be entered in the subsequent "Report BIP Details" Details screen.

Field Title	Configure SIM ticket Printing
Field Description	Choose the ticket printing option.

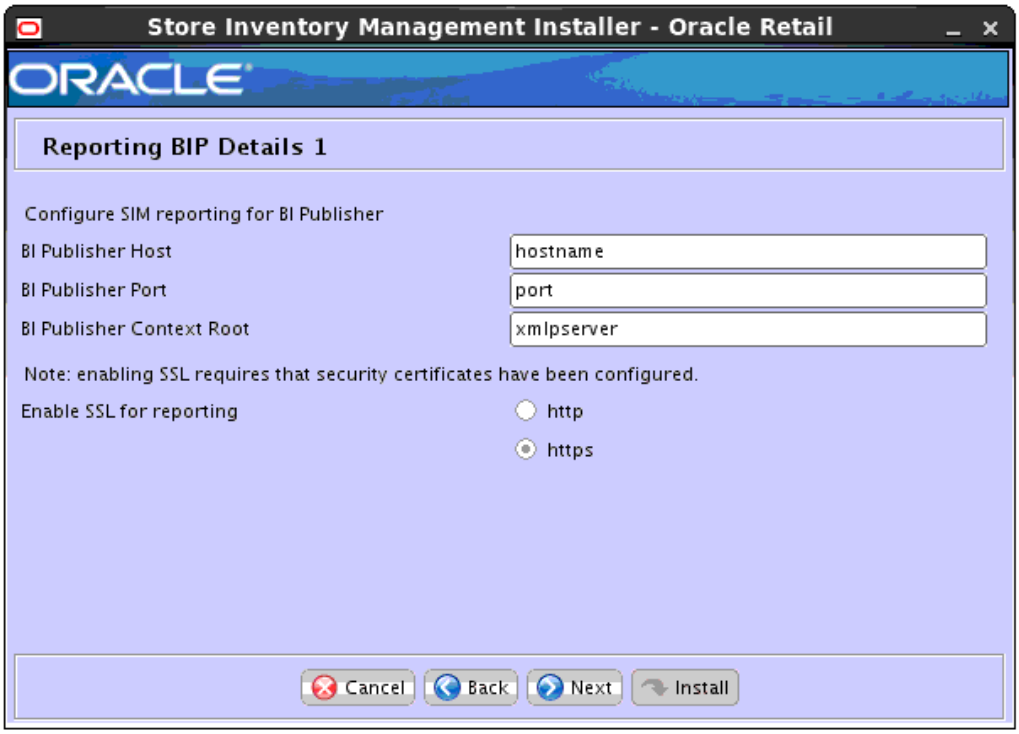
Field Title	Configure SIM ticket Printing "None"
Field Description	Select this option if you will not be using ticket printing feature.

Field Title	Configure SIM ticket Printing "BI Publisher"
Field Description	Select this option if you will be using an out of box BI Publisher ticketing implementation. Note, if you select this option then the "Ticket Printing BIP Details" screen will be enabled and appropriate details will have to be entered in the subsequent "Ticket Printing BIP Details" Details screen.

Field Title	Configure SIM ticket Printing "External Web Service"
Field Description	Select this option if you will provide web service provider. See the "Item Ticket Printing" Section in SIM Implementation Guide for details. Note, if you select this option then the "External Ticket Printing Service Details" screen will be enabled and appropriate details will have to be entered in the subsequent "External Ticket Printing Service Details" screen.

Screen: Reporting BIP Details 1

This screen will be displayed if you select the Configure SIM reporting for BI Publisher option on the Printing Details screen.



Field Title	BI Publisher Host
Field Description	Host name where BI Publisher is installed.
Destination	Updates the BI Publisher related default values in SIM database.
Example	redevlv0074.us.example.com

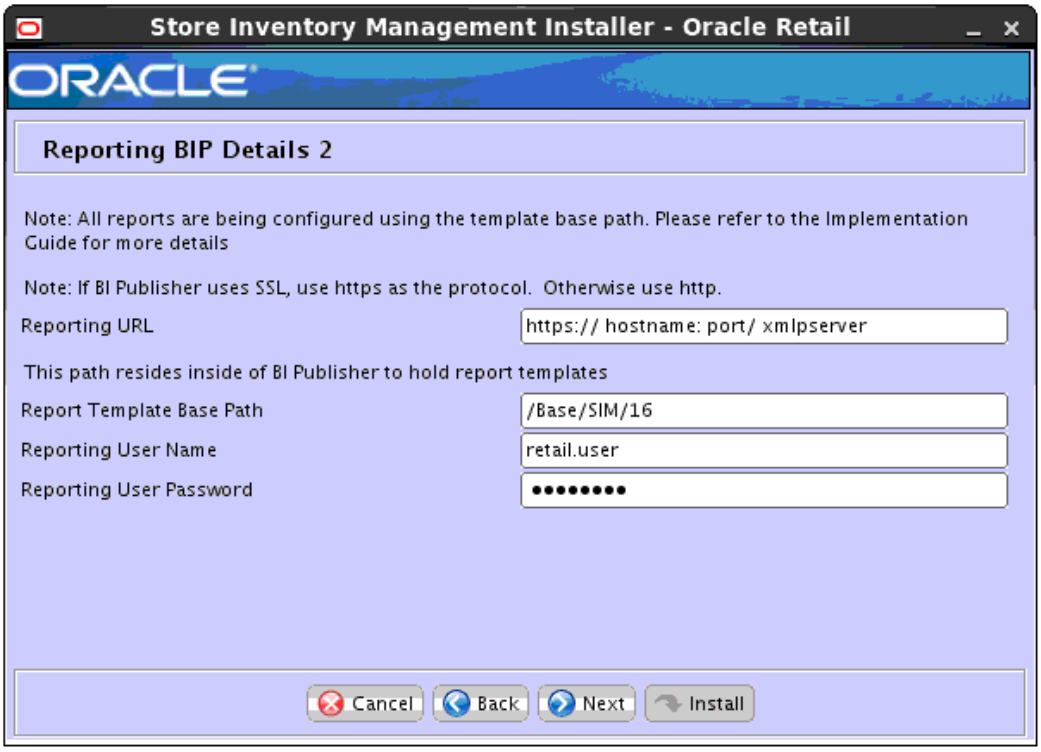
Field Title	BI Publisher Port
Field Description	Port where BI Publisher is configured.
Destination	Updates the BI Publisher related default values in SIM database.
Example	7003

Field Title	BI Publisher Context Root
Field Description	Context root where BI Publisher is installed.
Destination	Updates the BI Publisher related default values in SIM database.
Example	Xmlpserver

Field Title	Enable SSL for reporting
Field Description	The Protocol to be used for configuring reporting.

Screen: Reporting BIP Details 2

This screen will be displayed if you select the Configure SIM reporting for BI Publisher option on the Printing Details screen.



Field Title	Reporting URL
Field Description	Confirmation field of address configured from values provided on previous screen.
Destination	Updates the reporting tool related default values in SIM database.
Example	http://hostname:18005/xmlpserver/

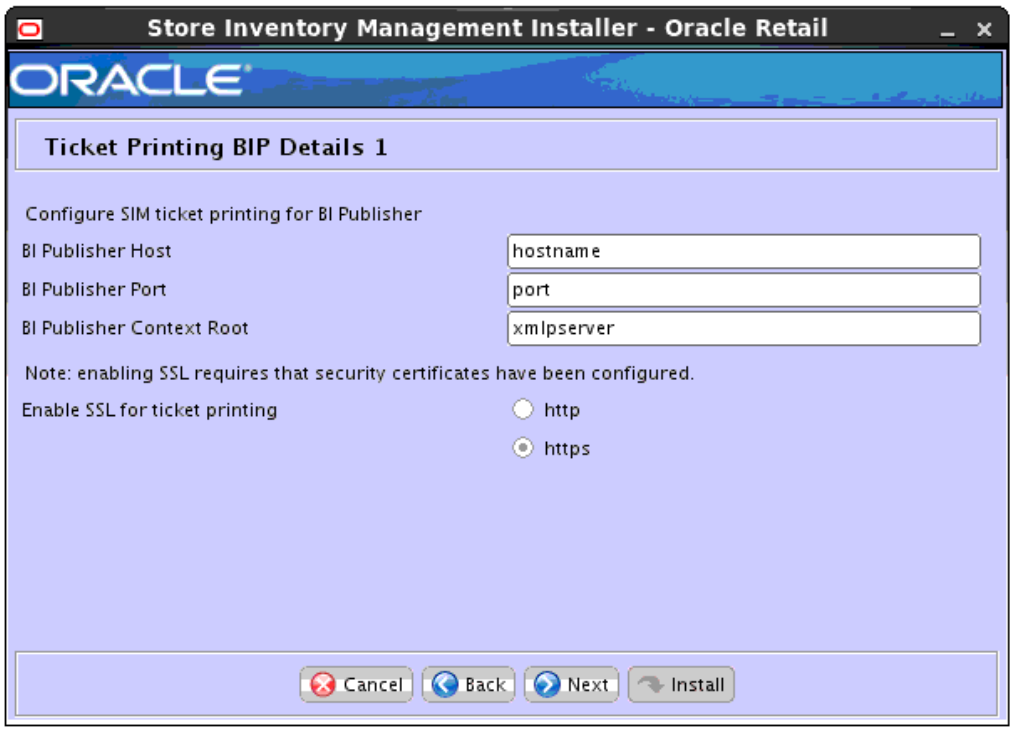
Field Title	Report Template Base Path
Field Description	The root directory in which your SIM report templates are located.
Example	/Base/SIM /u00/webadmin/product/10.3.X/WLS/user_projects/domains/bifoundation_domain/config/bipublisher/repository/Reports/Guest/SIM

Field Title	Reporting Username
Field Description	From the Oracle Retail Store Inventory Management Implementation Guide: <*BIP_REPORTS_USER*> or <*SSO_USER*>
Destination	This user MUST exist as a BI Publisher user.
Example	retail.user

Field Title	Reporting user Password
Field Description	From the Oracle Retail Store Inventory Management Implementation Guide: <*BIP_REPORTS_USER_PASSWORD*> or <*SSO_PASSWORD*>
Destination	Updates security wallet info

Screen: Ticket Printing BIP Details 1

This screen will be displayed if you select the Configure SIM ticket printing option on the Printing Details screen.



Field Title	BI Publisher Host
Field Description	Host name where BI Publisher is installed.
Destination	Updates the BI Publisher related default values in SIM database.
Example	hostname

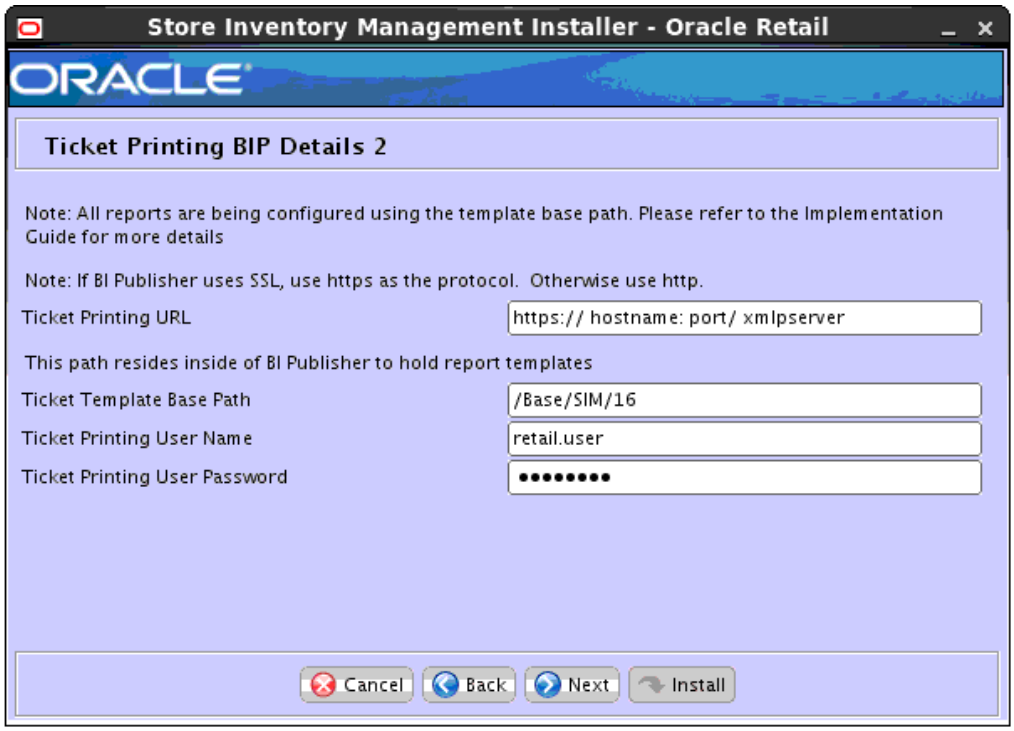
Field Title	BI Publisher Port
Field Description	Port where BI Publisher is configured.
Destination	Updates the BI Publisher related default values in SIM database.
Example	7003

Field Title	BI Publisher Context Root
Field Description	Context root where BI Publisher is installed.
Destination	Updates the BI Publisher related default values in SIM database.
Example	Xmlpserver

Field Title	Enable SSL for ticket printing
Field Description	The Protocol to be used for ticket printing.

Screen: Ticket Printing BIP Details 2

This screen will be displayed if you select the Configure SIM ticket printing option on the Printing Details screen.



Field Title	Ticket Printing URL
Field Description	Confirmation field of address configured from values provided on previous screen.
Destination	Updates the ticket printing BIP related default values in SIM database.
Example	http://hostname:18006/xmlpserver

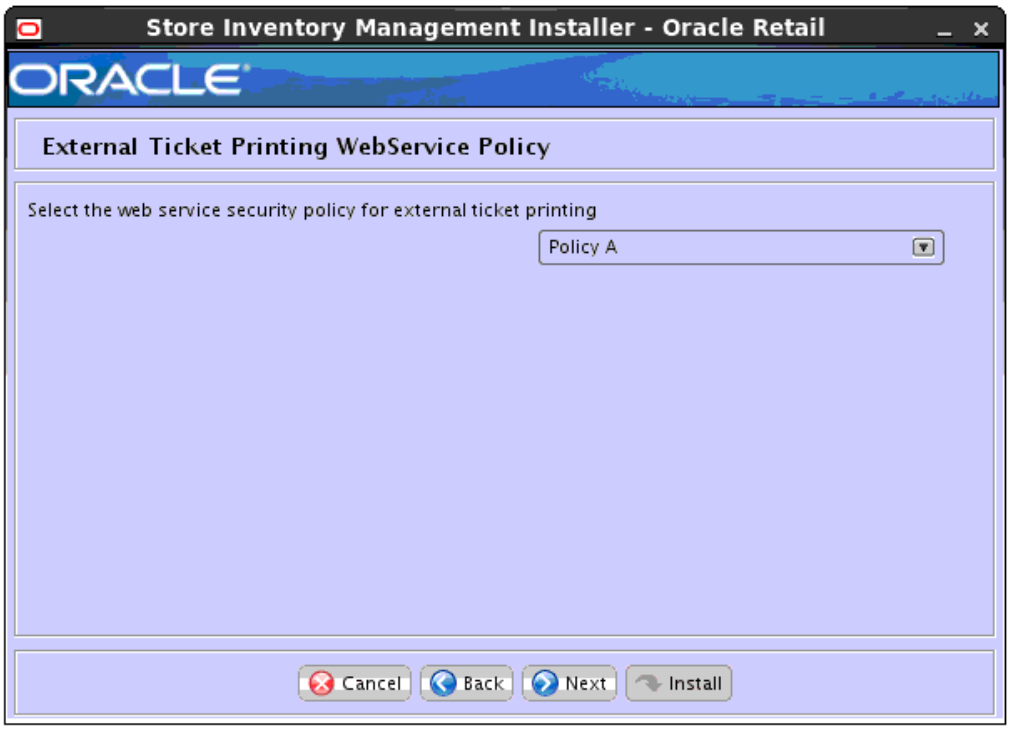
Field Title	Ticket Template Base Path
Field Description	The root directory in which your SIM ticket templates are located. **Note:** See Appendix: Setting up SIM Reports/Tickets in BI Publisher for instructions for migrating SIM reports/tickets to BI Publisher
Example	/Base/SIM An example from this install guide is: /u00/webadmin/product/10.3.X/WLS/user_projects/domains/bifoundation_domain/config/bipublisher/repository/Reports/Guest/SIM

Field Title	Ticket Printing Username
Field Description	From the Oracle Retail Store Inventory Management Implementation Guide: <*BIP_TICKETPRINTING_USER*> or <*SSO_USER*>
Destination	This user MUST exist as a BI Publisher user.
Example	retail.user

Field Title	Ticket Printing user Password
Field Description	From the Oracle Retail Store Inventory Management Implementation Guide: <*BIP_TICKETPRINTING_USER_PASSWORD*> or <*SSO_PASSWORD*>
Destination	Updates security wallet info

Screen: External Ticket Printing WebService Policy

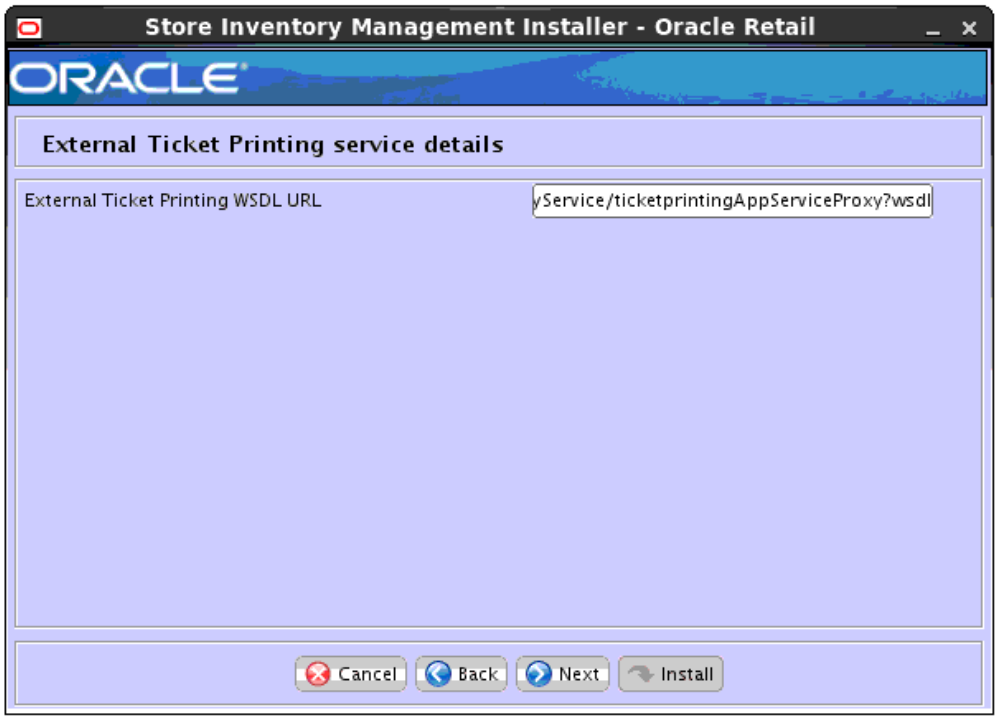
This screen will be displayed if you select the External Webservice option on the Printing Details screen.



Field Title	Select the web service security policy for external ticket printing **Note**: The user should refer the *Oracle Retail Store Inventory Management Operations Guide* to know what OMS to choose.
Field Description	Selects the type of web service security policy for external ticket printing. Please refer to the *Oracle Retail Store Inventory Management Security Guide* to learn more about Policy A and Policy B.
Destination	None, PolicyA, PolicyB **Note**: If web services are to be secured using either Policy A or Policy B, then user should have some basic knowledge about the same. A user can refer to security guide to know more about Policy A and Policy B

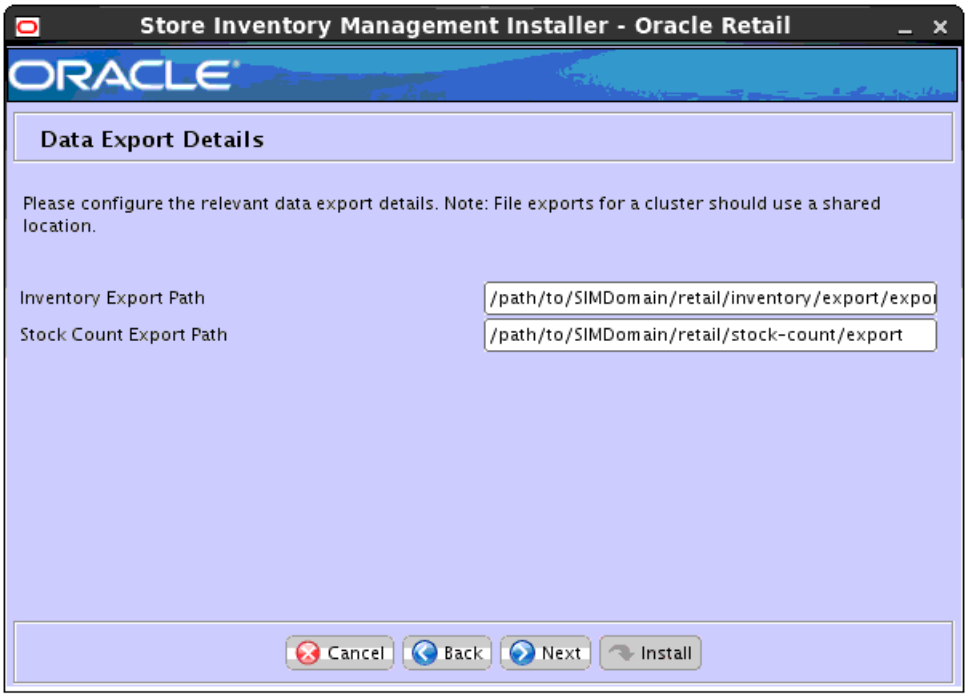
Screen: External Ticket Printing Service Details

This screen will be displayed if you select the External Webservice option on the Printing Details screen.



Field Title	External Ticket Printing WSDL URL
Field Description	This is the External Ticket Printing WSDL URL.
Example	http://hostname:18007/ticket-printing-AppServiceDecorator/ProxyService/ticketprintingAppServiceProxy?wsdl

Screen: Data Export Details



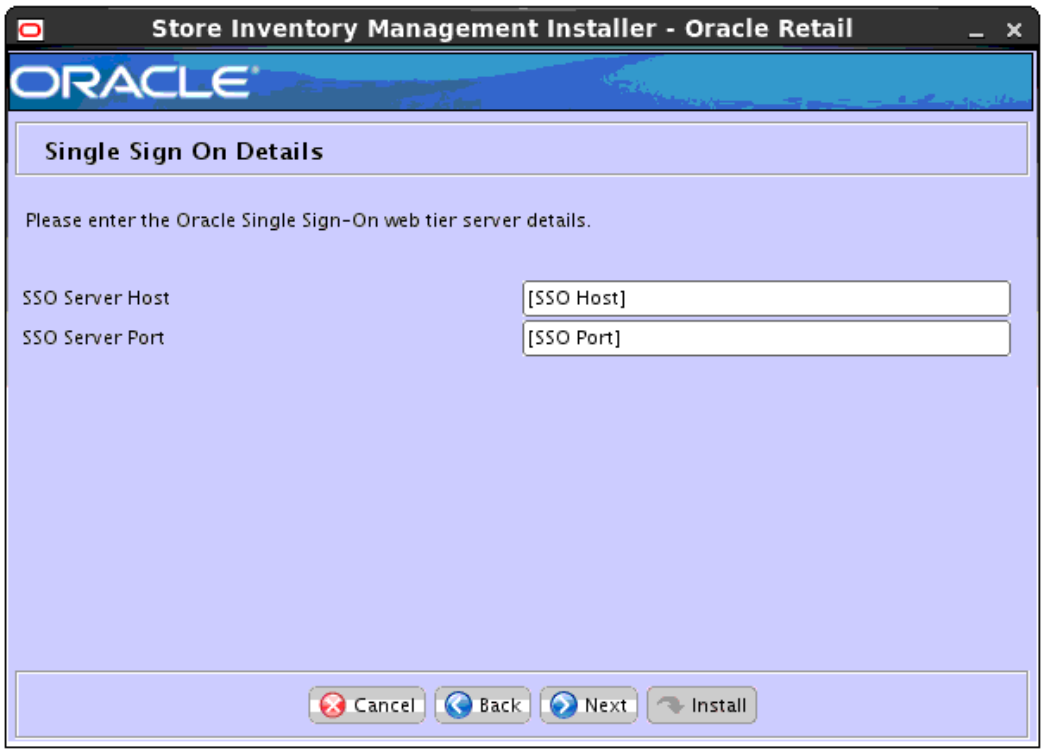
Field Title	Inventory Export Path
Field Description	Directory in which export the store inventory data in the export file on application server. The directory must exist. If directory is not specified, the installation will create a default location at <SIM application Server Domain>/<retail_dir>/inventory/export The directory and file into which you export the data should be protected with operating system security. Read, Write, and Create New File permissions to the export data directory for SIM application server user. Note: Ftp user permission for moving data to secure ftp site: Read, Write permissions to the export data directory for user who is responsible to move export data files to secure ftp sites for importer applications.
Example	<SIM application Server Domain>/retail/inventory/export

Field Title	Stock Count Export Path
Field Description	Directory in which export the store inventory data in the export file on application server. The directory must exist. If directory is not specified, the installation will create a default location at <SIM application Server Domain>/<retail_dir>/inventory/export The directory and file into which you export the data should be protected with operating system security. Read, Write, and Create New File permissions to the export data directory for SIM application server user. Note: Ftp user permission for moving data to secure ftp site: Read, Write permissions to the export data directory for user who is responsible to move export data files to secure ftp sites for importer applications.
Example	<SIM application Server Domain>/retail/inventory/export

Screen: Enable SSO in SIM

Field Title	Use Single Sign-On for user identification and authentication?
Field Description	This version of SIM has the option to use Single Sign-On (SSO) technology to authenticate users. If SSO is being used in your environment then click the check box. Leaving the box unchecked will configure SIM to use its own LDAP directory settings for authentication.

Screen: Single Sign-On Details



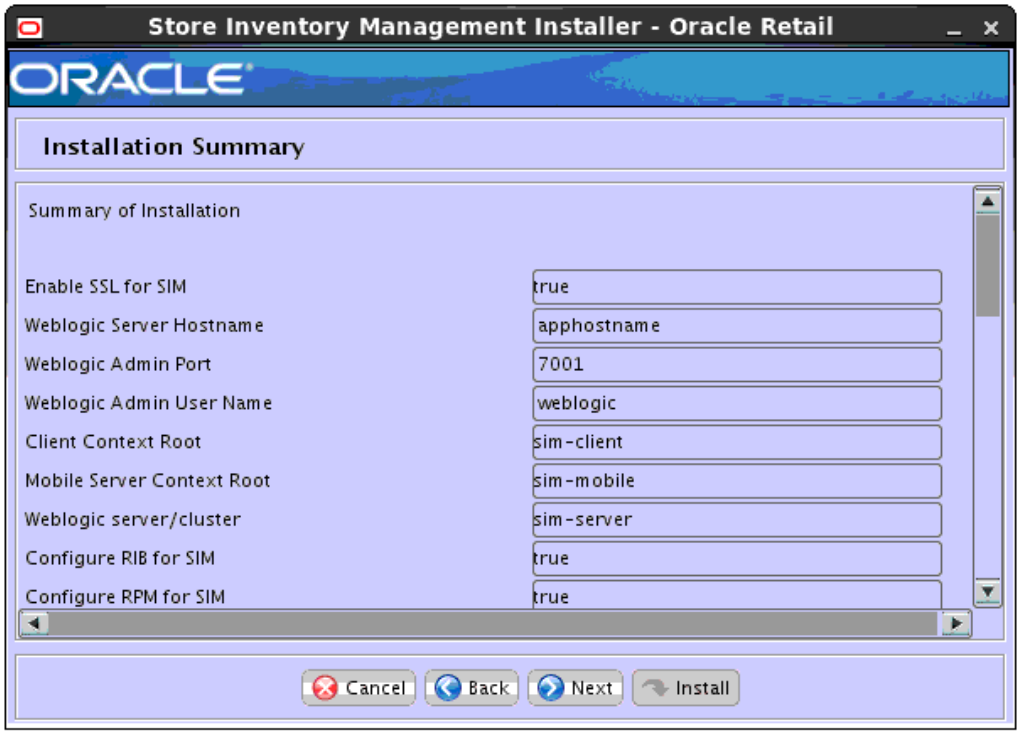
Field Title	SSO Server Host
Field Description	This is the host used to access the Single Sign-On web tier.
Example	WEBTIERSERVER.us.com

Field Title	SSO Server Port
Field Description	This is the HTTP port used to access the Single Sign-On web tier.
Example	18888

Screen: Manual Deployment Option

Field Title	Install files to app server?
Field Description	The installer will configure the application and application server files. Then, it can proceed with installing the application into the server. If a user does not have filesystem access to application server, or wishes to deploy using a different method, he can choose to have the installer skip the final installation phase.
Example	Select Yes, I have write access to the application server.

Screen: Installation Summary



Field Title	Summary of Installation
Field Description	Lists the values entered in the previous install screens.
Example	Verify each value and if they are correct click next, else use back button to go back and change the value. Selecting Cancel will Cancel the installation.

C

Appendix: Common Installation Errors

This section provides some common errors encountered during installation.

EJB Deployment Errors during Installation to WebLogic

Symptom

On servers that are encountering high memory usage, deployment of sim-server.ear will occasionally fail due to WebLogic's inability to start the EJB polling timer service.

```
     [java] .....Failed to deploy the application with status failed
     [java] Current Status of your Deployment:
     [java] Deployment command type: deploy
     [java] Deployment State       : failed
     [java] Deployment Message     : weblogic.application.ModuleException:
Exception activating module: EJBModule(
sim-ejb3.jar)
     [java]
     [java]
     [java] weblogic.management.scripting.ScriptException: Error occured while
performing deploy : Deployment Fail
ed.
     [java] Unable to deploy EJB: PollingCoordinatorThreadBean from sim-ejb3.jar:
     [java]
     [java] Error starting Timer service
```

Solution

Delete the WebLogic managed server/cluster where sim was targeted in the Admin Console, and activate the changes. Manually delete the managed server directory <DOMAIN HOME>/servers/<SIM SERVER NAME>. Bounce the WebLogic admin server. Re-create the managed server in the Admin Console, Finally, re-run the installer. If the error persists after re-installation, consider reducing the cpu, disk, and memory load on the server.

Output Freezes during Text Mode Installation to WebLogic

Symptom

The standard output of the installer in text mode will sometimes freeze partway through the installation.

Solution

Open a new terminal to the server and tail the log file located in sim/application/logs.

Database Installer Hangs on Startup

Symptom

When the database schema installer is run, the following is written to the console and the installer hangs indefinitely:

```
Running pre-install checks
Running tnsping to get listener port
```

Solution

The installer startup script is waiting for control to return from the **tnsping** command, but tnsping is hanging. Type Control+C to cancel the installer, and investigate and solve the problem that is causing the **tnsping <sid>** command to hang. This can be caused by duplicate database listeners running.

Warning: Could not create system preferences directory

Symptom

The following text appears in the installer Errors tab:

```
May 22, 2006 11:16:39 AM java.util.prefs.FileSystemPreferences$3 run
WARNING: Could not create system preferences directory. System preferences are
unusable.
May 22, 2006 11:17:09 AM java.util.prefs.FileSystemPreferences
checkLockFile0ErrorCode
WARNING: Could not lock System prefs. Unix error code -264946424.
```

Solution

This is related to Java bug 4838770. The /etc/.java/.systemPrefs directory may not have been created on your system. See http://bugs.sun.com for details.

This is an issue with your installation of Java and does not affect the Oracle Retail product installation.

Warning: Couldn't find X Input Context

Symptom

The following text appears in the console window during execution of the installer in GUI mode:

```
Couldn't find X Input Context
```

Solution

This message is harmless and can be ignored.

ConcurrentModificationException in Installer GUI

Symptom

In GUI mode, the errors tab shows the following error:

```
java.util.ConcurrentModificationException
            at
java.util.AbstractList$Itr.checkForComodification(AbstractList.java:448)
            at java.util.AbstractList$Itr.next(AbstractList.java:419)
… etc
```

Solution

You can ignore this error. It is related to third-party Java Swing code for rendering of the installer GUI and does not affect the retail product installation.

A Second Login Screen Appears After Single Sign-On Login

If you are using Single Sign-On, you should not need to enter a SIM user name and password once SIM is launched. If the SIM login screen pops up, it means something went wrong with the SSO login. This could be caused by any of the following problems:

- There is no SIM user in LDAP for the SSO user name you are using.
- Permissions are not set up correctly for the SSO user in SIM.
- SSO is configured incorrectly on the server.
- SSO timed out. (This can happen especially the first time you launch SIM. Try launching SIM again.)

Symptom

A second login screen appears after you have already logged in to Single Sign-On.

Solution

See the *Oracle Retail Store Inventory Management Configuration Guide* for more information on setting up SIM users using LDAP with SIM.

Error Connecting to Database URL

Symptom

After entering database credentials in the installer screens and hitting next, a message pops up with an error like this:

```
Error connecting to database URL <url> as user <user> details...
```

The message prevents you from moving on to the next screen to continue the installation.

Solution

This error occurs when the installer fails to validate the user credentials you have entered on the screen. Make sure that you have entered the credentials properly. If you receive a message similar to this:

```
Error connecting to database URL <url> as user <user> java.lang.Exception:
UnsatisfiedLinkError encountered when using the Oracle driver.
Please check that the library path is set up properly or switch to the JDBC thin
client.
```

It may mean that the installer is using the incorrect library path variables for the platform you are installing on. Open the file <STAGING_DIR>/rms/dbschema/common/preinstall.sh and toggle the variable, use32bit, to True if it is set to False or vice versa. This setting is dependent on the JRE that is being used.

GUI screens fail to open when running Installer

Symptom

When running the installer in GUI mode, the screens fail to open and the installer ends, returning to the console without an error message. The ant.install.log file contains this error:

```
Fatal exception: Width (0) and height (0) cannot be <= 0
java.lang.IllegalArgumentException: Width (0) and height (0) cannot be <= 0
```

Solution

This error is encountered when Antinstaller is used in GUI mode with certain X Servers. To work around this issue, copy ant.install.properties.sample to ant.install.properties and rerun the installer.

Log in fails with invalid username/password or user unauthorized errors

Symptom

The SIM application log in fails with the following messages: “Invalid username/password” or “User unauthorized or Not authenticated.”

Solution

In SIM Database, in the CONFIG_SYSTEM table, the value for SECURITY_AUTHENTICATION_METHOD should be set to 1 for LDAP authentication.

Check in LDAP to be sure the password is set to the correct value.

D

Appendix: Setting up SIM Reports/Tickets in BI Publisher

SIM 16.0.3 reports and ticket printing supports BiPublisher 12.2.1.3 for new installs or 12.2.1.4 for upgrades. At the time of creating this install guide BiPublisher is not supporting a BIP only install for new installs of 12.2.1.4 you must install BiPublisher 12.2.1.3.

BiPublisher 12c – BI Server Component Installation Tasks

Oracle BI Publisher is used as the main RMS, RWMS, REIM, and SIM reporting engine and can be used in conjunction with external printing solutions like label printing. This section describes the installation of Oracle BI Publisher as a server application within WebLogic 12c. One deployment of BI Publisher can be used for any of the RMS, RWMS, REIM, and SIM reports.

BiPublisher 12c only - Installation Process Overview

Oracle BiPublisher must be installed in a standalone setup, it cannot be incorporated with OBIEE Analytics as this would prevent Guest access to the BiPublisher reports.

The BiPublisher install steps are documented here:
http://docs.oracle.com/middleware/12213/bip/index.html

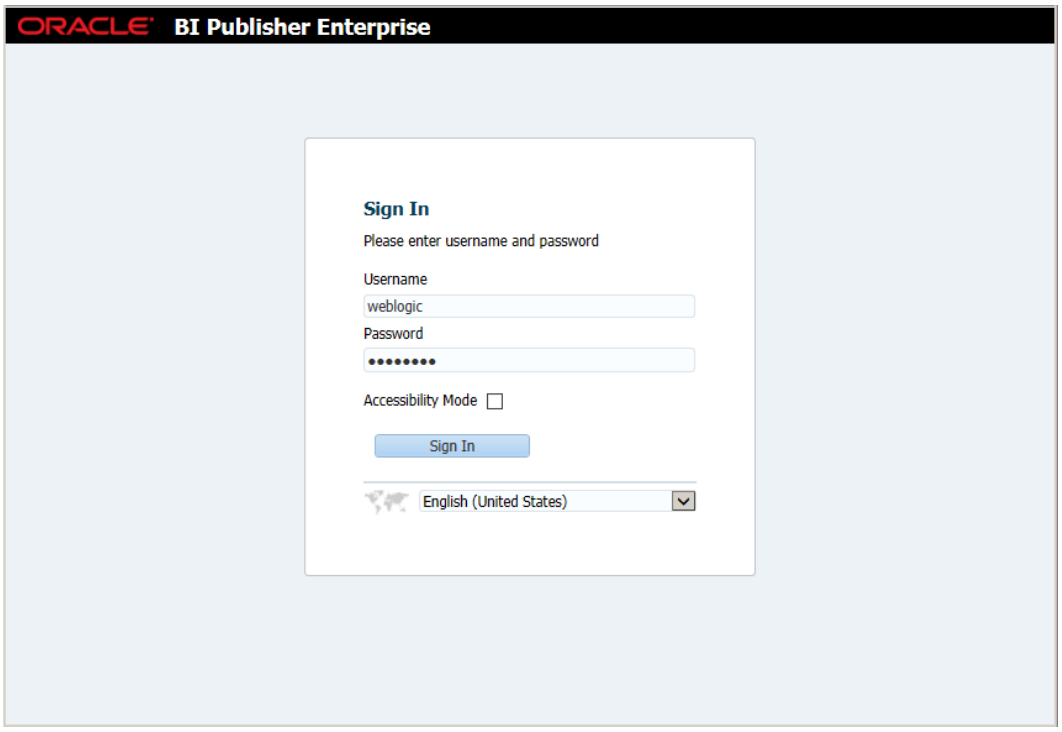
Note: BIP standalone was not part of the featureset in 12.2.1.4.0. It is currently scheduled for a future release (Doc ID 2472158.1). BIP standalone is supported only when it is upgraded to 12.2.1.4 version.

Note: To make report rendering work in BIP 12.2.1.4 using Guest access, below WebLogic patches must to be applied in the same order after shutting down all the domains running on that Oracle home- 28186730 (Update OPatch to 13.9.4.2.x), 30499026 (OBI Bundle patch for 12.2.1.4) and 27545920 (Guest report access error).

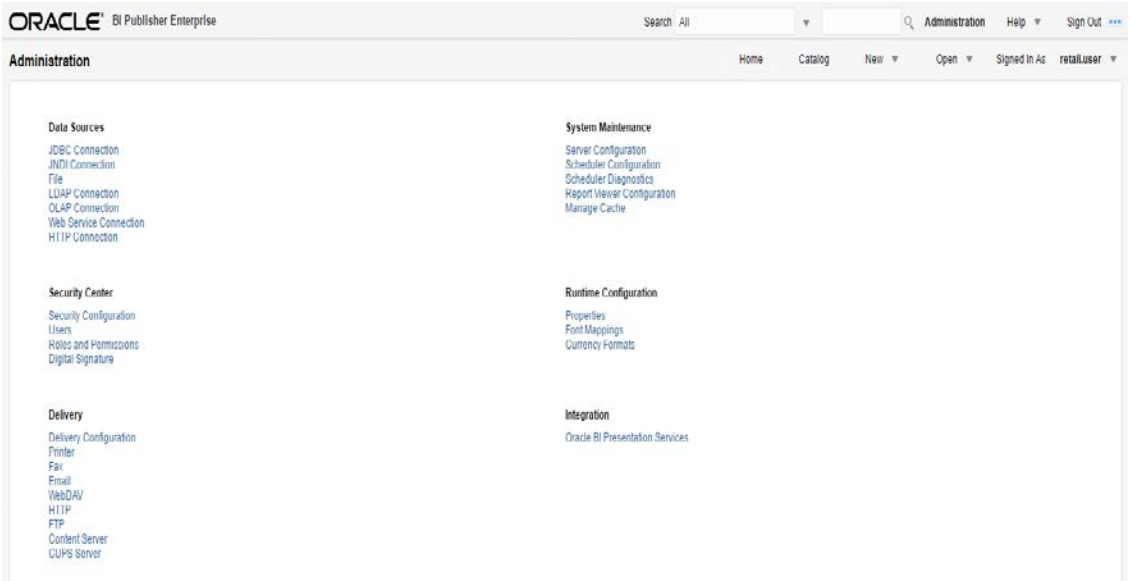
Once BiPublisher is installed follow the post install steps below to configure the reports.

Post install steps for BiPublisher 12C

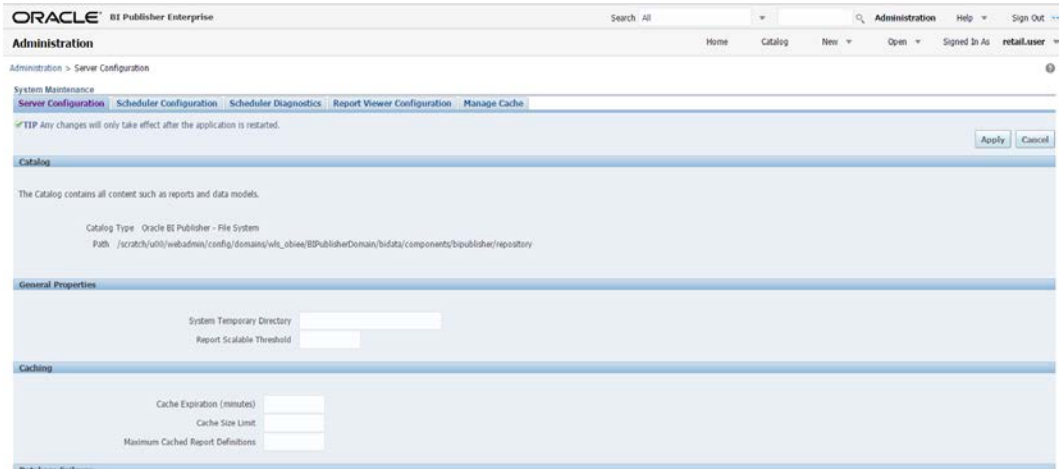
1. Test your BIPublisher installation, Get the xmlpserver url from your Installation Screen and launch xmlpserver. Login with the credentials you entered in your Oracle BI EE configuration (weblogic / password). Example URL:http://[obiee_host]:[obiee server_port]/xmlpserver



2. After sign on, select "Administration".

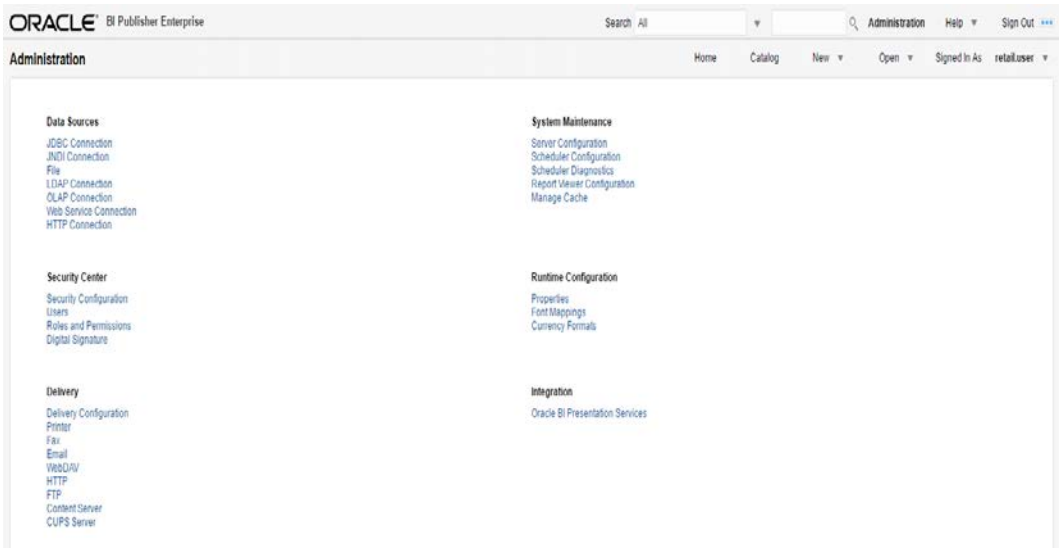


3. On the System Maintenance Section, click **Server Configuration**.

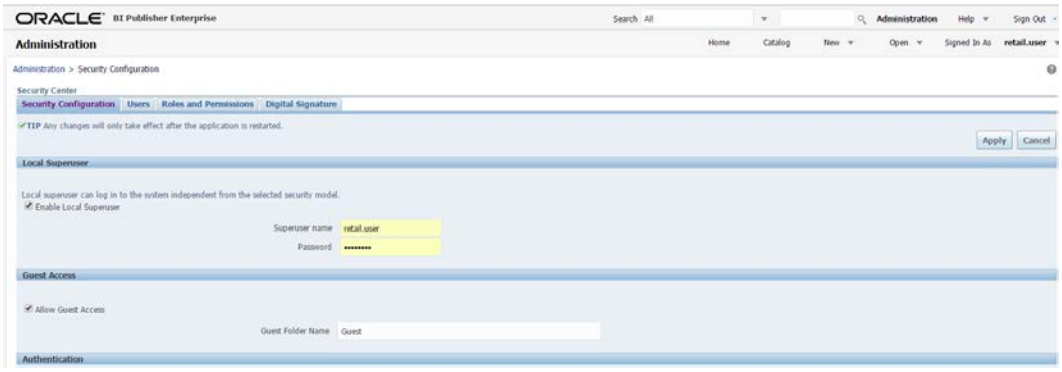


4. On this screen - In the Server Configuration Folder section, enter the path to your repository.
   - This is the path you entered in the Configuration Section and Catalog Section:

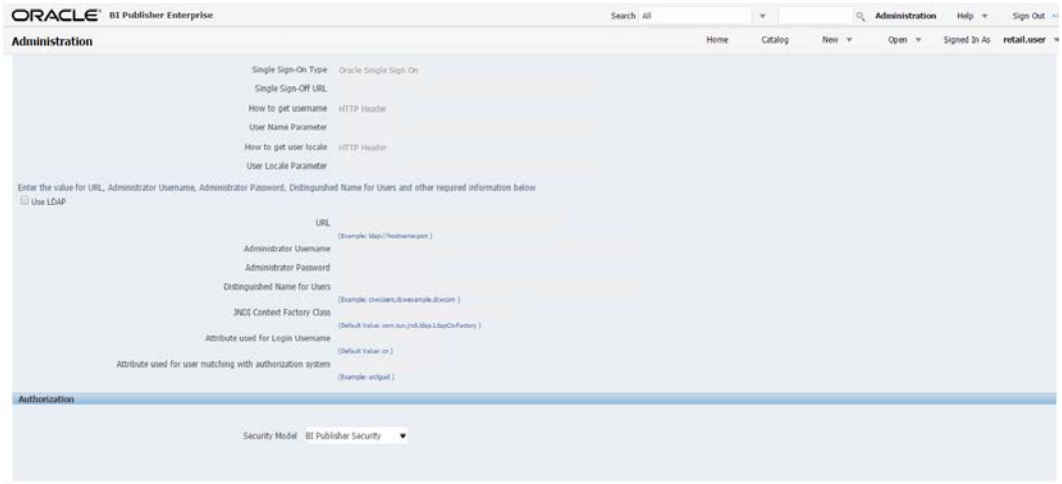
   `Example: $<OBIEE_DOMAIN_HOME>/bidata/components/bipublisher/repository`
5. Click **Apply**.
6. Click Administration link at top of screen.



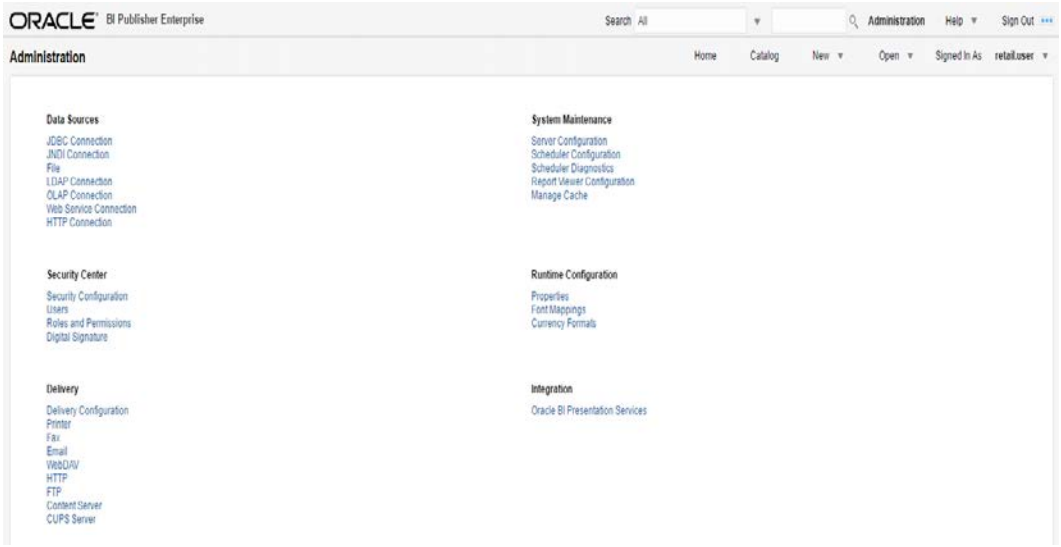
7. Click on the Security Configuration link under the Security Center to setup a super user and apply the BI Publisher security model.



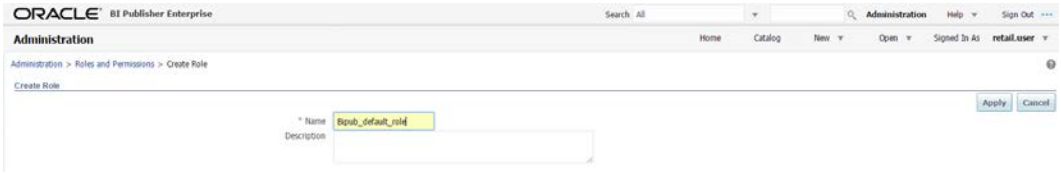
8. Enable a Superuser by checking the "Enable Local Superuser" box and by entering name and password on the corresponding fields on this screen.
9. Mark "Allow Guest Access" check box. Enter "Guest" as Guest Folder Name.
10. Click **Apply**.
11. Scroll down the screen and locate the Authorization section:



12. Select BI Publisher Security from the Security Model list.
13. The default user name for the BI Publisher Security Model is Administrator.
14. On the password text field, enter a value that you can remember. It is going to be the password for Login to xmlpserver.
15. Click **Apply**.
    - Leave BI Publisher up while completing the next section.
16. Post install step: Create role Bipub_default_role.
    a. From the xmlpserver Administration screen, scroll down to Security Center and click Roles and Permissions.



b. On the Roles and Permissions screen, click the Create Role button.

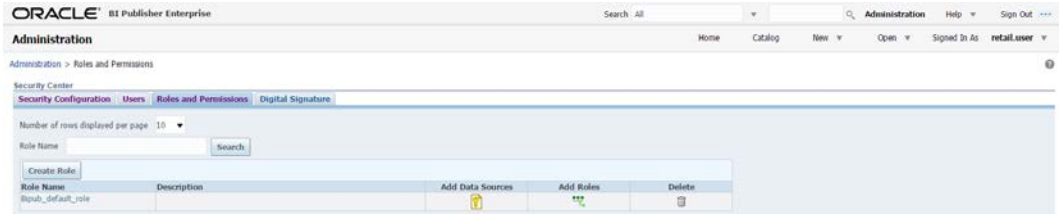


c. Create the Bipub_default_role. Enter in Create Role Section name of the role.

d. When the information has been entered press Apply changes.

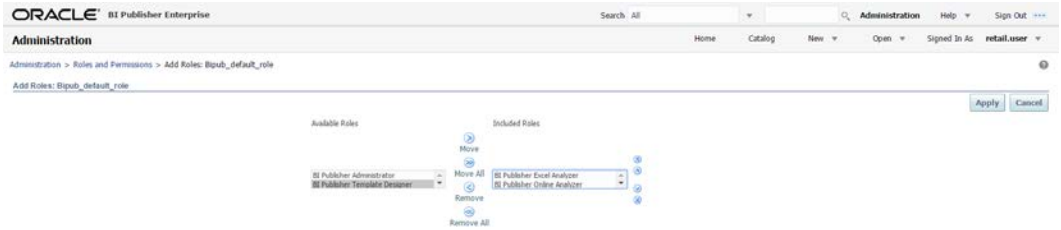
17. Post install step: Assign BiPub system roles to the newly created Bipub_default_role.

a. To assign BiPub system roles to the newly create Bipub_default_role, go to Security Center section and navigate to the Roles and Permissions screen:



b. On the Roles and Permissions screen you should see the new role created: "Bipub_default_role" . Add multiple roles to the Bipub_Default_Role by pressing the corresponding green icon on the Add Roles column.

BI Publisher Excel Analyzer, BI Publisher Online Analyzer, BI Publisher Scheduler.

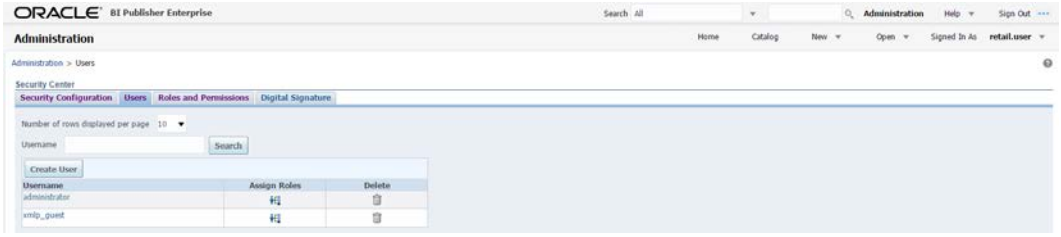


c. From the "Available Roles" panel, select the ones needed for your reports and move them to the "Included Roles" panel

d. Press the Apply button to save your changes.

18. Post install step: create Guest (XMLP_GUEST) user.

a. From the xmlpserver Administration screen scroll down to Security Center section and press Users to navigate to the next screen



b. Select the "Create User" button to create the "xmlp_guest" user and save the changes

19. Post install step: Adding the Bipub_default_role to XMLP_GUEST user.

a. Open the Users section:

b. For xmlp_guest user, press on the "Assign Roles" icon to navigate to the next screen:

c. On the Assign Roles screen, select the BiPub_default_role from the Available Roles panel to the "Assigned Roles" panel and press the Apply button to save your changes.

Installing the SIM BI Publisher Templates

In this section we will outline how the SIM report templates are installed into the appropriate BI server repositories.

```
Example: $<OBIEE_DOMAIN_HOME>/bidata/components/bipublisher/repository
```

Report files are included in the SIM installation media and have to be copied into a newly created directory within BI Publisher repository Guest Reports directory.

1. Create the directory to hold the reports under <BI_REPOSITORY>

   ```
   mkdir <BI_REPOSITORY>/Reports/Guest/SIM
   ```

2. Change directory to the <INSTALL_DIR>/sim16/reports/ in the SIM installation media extracted previously. This directory contains a sim-reports.zip file which contains all the SIM reports.
3. Copy the sim-reports.zip above to your repository and extract them

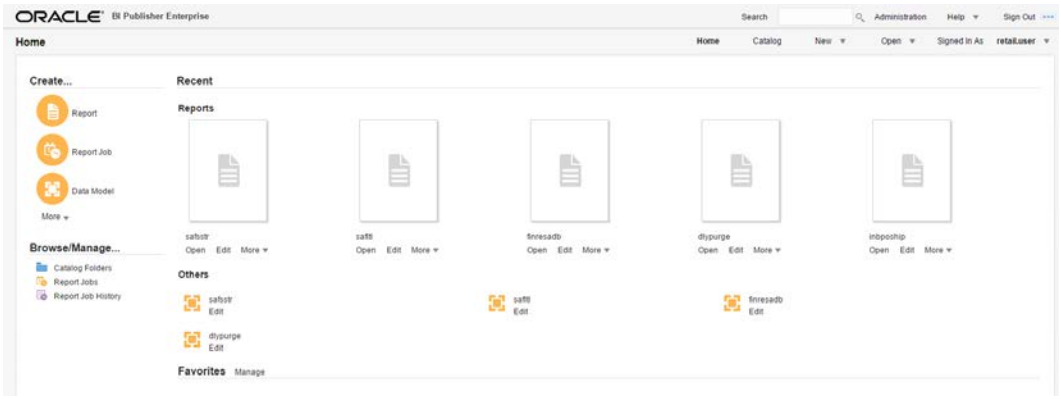
   ```
   cp sim-reports.xip  <BI_REPOSITORY>/Reports/Guest/SIM
   cd <BI_REPOSITORY>/Reports/Guest/SIM
   unzip sim-reports.zip
   ```

Configuring the SIM JDBC connection

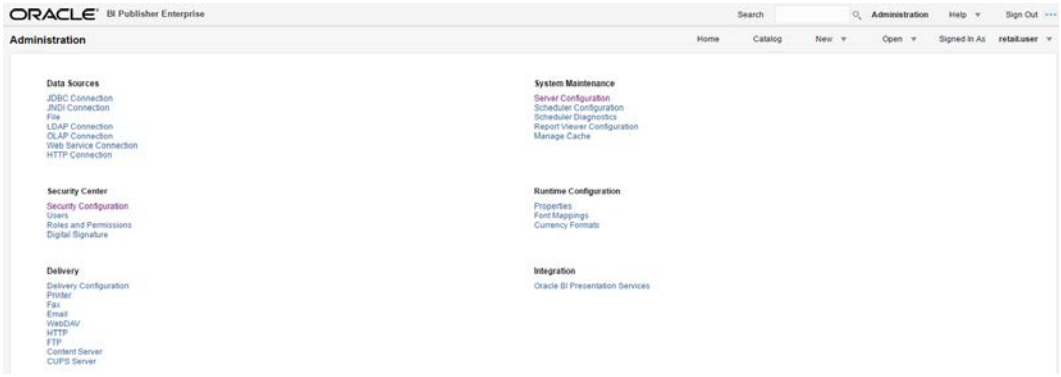
Follow the below steps to configure a JDBC connection for the SIM Data Source, which is required for SIM reports.

1. If not still logged into BIPublisher:
   - Login with the credentials you entered in your Oracle BI EE configuration. (weblogic / password)
2. If the server was restarted:
   - Login as the super user that was created in prior security setup steps.

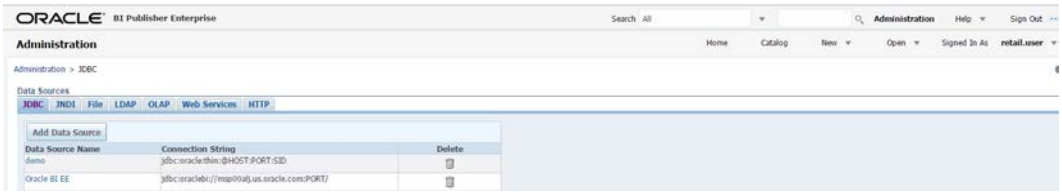
> **Note:** You will not be able to login to xmlpserver as weblogic any more because we have already changed the Security Model.



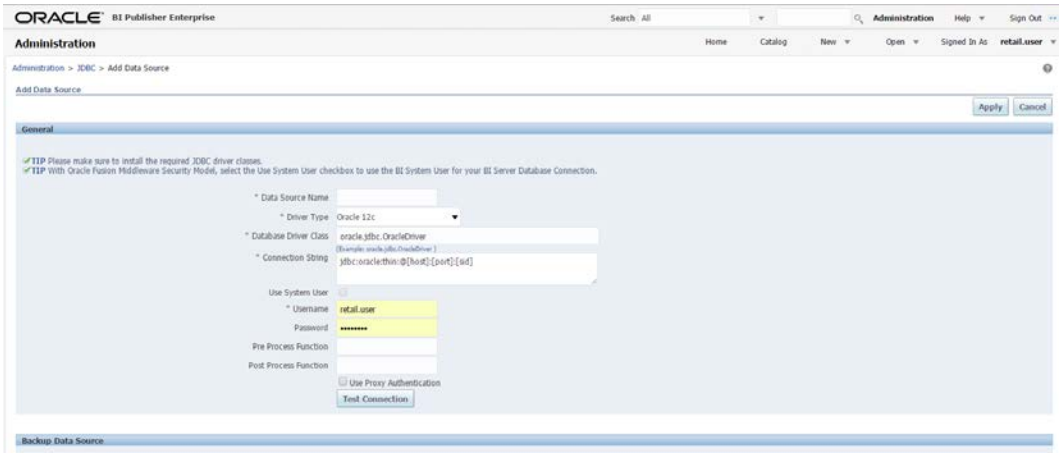
3. Click the **Administration** link at top of screen



4. Select the JDBC Connection hyperlink in the Data Sources lists.



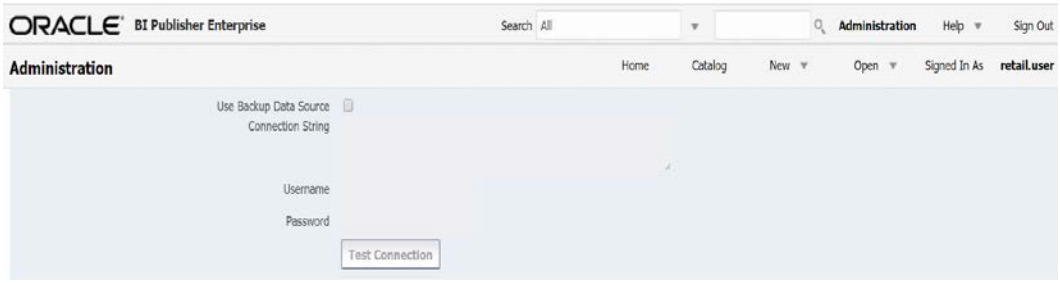
5. Click the **Add Data Source** button.

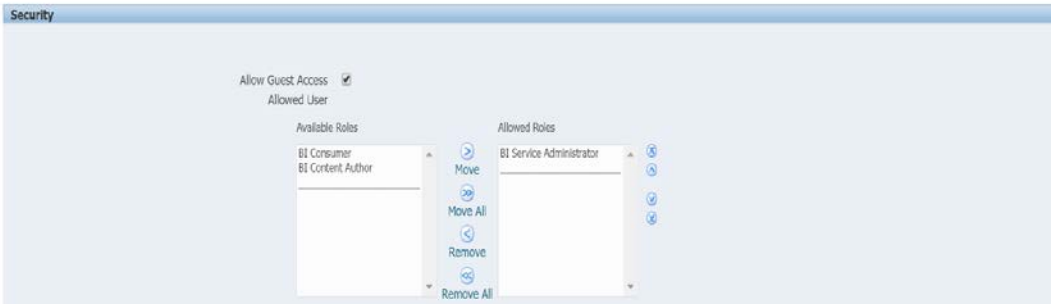


6. Enter the appropriate details for the SIM data source. Click Test Connection to test the connection on the screen once the data is entered.
   - Data Source Name: BIP-SIM-DATASOURCE
     - Must be this name due to code dependencies.
   - Driver type is ORACLE 12C
   - Database driver class should be oracle.jdbc.OracleDriver.
   - Connection string is similar to this example:
     - Pluggable: jdbc:oracle:thin:@dbhostname:1521/servicename
     - Non- Pluggable dbc:oracle:thin:@dbhostname:1521:SID

- Enter the username and password for the SIM application user's data source. Click Test Connection to test the connection on the screen once the data is entered.

7. Scroll to the bottom of the screen and check the Allow Guest Access check box. Click **Apply**.



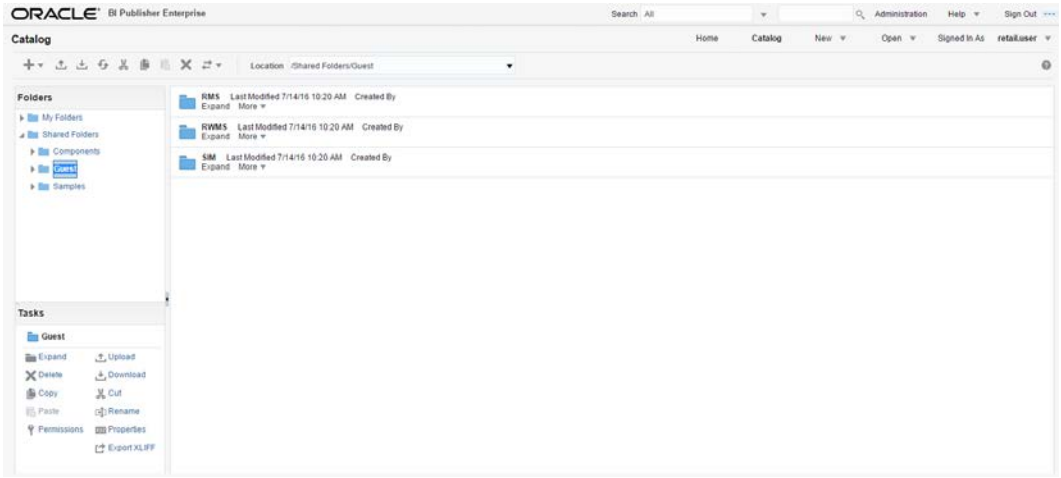


8. Restart WebLogic Server.

Verify Oracle BI Publisher Set Up for SIM Reports

Verify that Oracle BI Publisher has been set up correctly as follows:

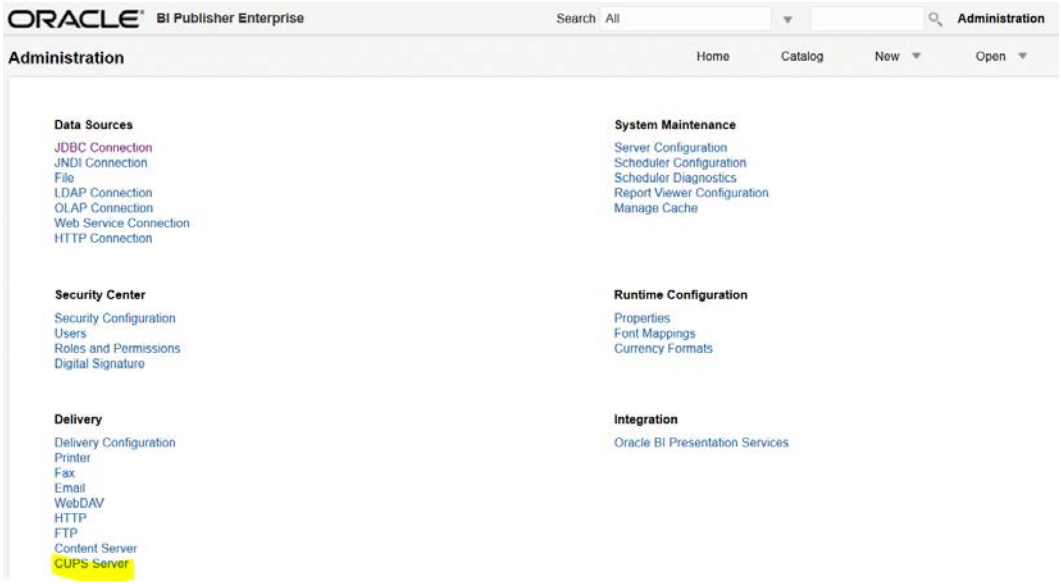
1. Click the **Administration** tab. Click **Server Configuration** under System Maintenance. The Catalog path variable should be set as part of the BI Publisher install, REPORTS_DIR.
2. Click Catalog link at the top of the screen – and then click the Guest folder on the left so that it is highlighted. You should see the SIM reports are now in the catalog:



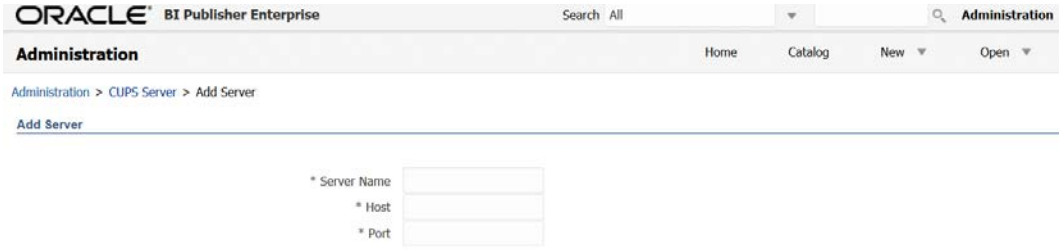
Configuring SIM for CUPS printers using BIPublisher 12c

Prerequisite: CUPS printer has to be set up on the host that the BIPublisher application is installed on.

1. Login to BI Publisher using the Super user that was created earlier and Click the Administration link at the top of the screen. Click on the CUPS Server under the Delivery section.



2. Click Add Server.



3. Enter in values and click **Apply**:
   - Server Name: SIMCUP
     - Can be any name
   - Host: localhost
   - Port: 631
     - 631 is default port that is used as an example – This may be different on the host.

4. After adding, refresh the servers and printers.

E

Appendix: Single Sign-On for WebLogic

Single Sign-On (SSO) is a term for the ability to sign onto multiple Web applications via a single user ID/Password. There are many implementations of SSO. Oracle provides an implementation with Oracle Access Manager.

Most, if not all, SSO technologies use a session cookie to hold encrypted data passed to each application. The SSO infrastructure has the responsibility to validate these cookies and, possibly, update this information. The user is directed to log on only if the cookie is not present or has become invalid. These session cookies are restricted to a single browser session and are never written to a file.

Another facet of SSO is how these technologies redirect a user's Web browser to various servlets. The SSO implementation determines when and where these redirects occur and what the final screen shown to the user is.

Most SSO implementations are performed in an application's infrastructure and not in the application logic itself. Applications that leverage infrastructure managed authentication (such as deployment specifying Basic or Form authentication) typically have little or no code changes when adapted to work in an SSO environment.

What Do I Need for Single Sign-On?

A Single Sign-On system involves the integration of several components, including Oracle Identity Management and Oracle Access Management. This includes the following components:

- An Oracle Internet Directory (OID) LDAP server, used to store user, role, security, and other information. OID uses an Oracle database as the back-end storage of this information.
- An Oracle Access Manager (OAM) 12c Release server and administrative console for implementing and configuring policies for single sign-on.
- A Policy Enforcement Agent such as Oracle Access Manager 12c Agent (WebGate), used to authenticate the user and create the Single Sign-On cookies.
- Oracle Directory Services Manager (ODSM) application in Oracle Identity Management (12.2.1.4), used to administer users and group information. This information may also be loaded or modified via standard LDAP Data Interchange Format (LDIF) scripts.
- Additional administrative scripts for configuring the OAM system and registering HTTP servers.

Additional WebLogic managed servers are needed to deploy the business applications leveraging the Single Sign-On technology.

Can Oracle Access Manager Work with Other SSO Implementations?

Yes, Oracle Access Manager has the ability to interoperate with many other SSO implementations, but some restrictions exist.

Oracle Single Sign-on Terms and Definitions

The following terms apply to single sign-on.

Authentication

Authentication is the process of establishing a user's identity. There are many types of authentication. The most common authentication process involves a user ID and password.

Dynamically Protected URLs

A Dynamically Protected URL is a URL whose implementing application is aware of the Oracle Access Manager environment. The application may allow a user limited access when the user has not been authenticated. Applications that implement dynamic protection typically display a Login link to provide user authentication and gain greater access to the application's resources.

Oracle Identity Management (OIM) and Oracle Access Manager (OAM) for 12c

Oracle Identity Management (OIM) 12c includes Oracle Internet Directory and ODSM. Oracle Access Manager (OAM) 12c should be used for SSO using WebGate. Oracle Forms 12c contains Oracle HTTP server and other Retail Applications will use Oracle WebTier for HTTP Server.

MOD_WEBLOGIC

mod_WebLogic operates as a module within the HTTP server that allows requests to be proxied from the OracleHTTP server to the Oracle WebLogic server.

Oracle Access Manager 12c Agent (WebGate)

Oracle WebGates are policy enforcement agents that reside with relying parties and delegate authentication and authorization tasks to OAM servers.

Oracle Internet Directory

Oracle Internet Directory (OID) is an LDAP-compliant directory service. It contains user ids, passwords, group membership, privileges, and other attributes for users who are authenticated using Oracle Access Manager.

Partner Application

A partner application is an application that delegates authentication to the Oracle Identity Management Infrastructure. One such partner application is the Oracle HTTP Server (OHS) supplied with Oracle Forms Server or WebTier Server if using other Retail Applications other than Oracle Forms Applications.

All partner applications must be registered with Oracle Access Manager (OAM) 12c. An output product of this registration is a configuration file the partner application uses to verify a user has been previously authenticated.

Statically Protected URLs

A URL is considered to be Statically Protected when an Oracle HTTP server is configured to limit access to this URL to only SSO authenticated users. Any unauthenticated attempt to access a Statically Protected URL results in the display of a login page or an error page to the user.

Servlets, static HTML pages, and JSP pages may be statically protected.

What Single Sign-On is not

Single Sign-On is NOT a user ID/password mapping technology.

However, some applications can store and retrieve user IDs and passwords for non-SSO applications within an OID LDAP server. An example of this is the Oracle Forms Web Application framework, which maps Single Sign-On user IDs to a database logins on a per-application basis.

How Oracle Single Sign-On Works

Oracle Access Manager involves several different components. These are:

- The Oracle Access Manager (OAM) server, which is responsible for the back-end authentication of the user.
- The Oracle Internet Directory LDAP server, which stores user IDs, passwords, and group (role) membership.
- The Oracle Access Manager Agent associated with the Web application, which verifies and controls browser redirection to the Oracle Access Manager server.
- If the Web application implements dynamic protection, then the Web application itself is involved with the OAM system.

About SSO Login Processing with OAM Agents

1. The user requests a resource.
2. Webgate forwards the request to OAM for policy evaluation
3. OAM:
   a. Checks for the existence of an SSO cookie.
   b. Checks policies to determine if the resource is protected and if so, how?
4. OAM Server logs and returns the decision
5. Webgate responds as follows:
   - **Unprotected Resource:** Resource is served to the user
   - **Protected Resource:**
     Resource is redirected to the credential collector.
     The login form is served based on the authentication policy.
     Authentication processing begins
6. User sends credentials
7. OAM verifies credentials
8. OAM starts the session and creates the following host-based cookies:
   - **One per partner:** OAMAuthnCookie set by 12c WebGates using authentication token received from the OAM Server after successful authentication.
     Note: A valid cookie is required for a session.
   - **One for OAM Server:** OAM_ID
9. OAM logs Success of Failure.
10. Credential collector redirects to WebGate and authorization processing begins.
11. WebGate prompts OAM to look up policies, compare them to the user's identity, and determine the user's level of authorization.
12. OAM logs policy decision and checks the session cookie.
13. OAM Server evaluates authorization policies and cache the result.

14. OAM Server logs and returns decisions
15. WebGate responds as follows:
    - If the authorization policy allows access, the desired content or applications are served to the user.
    - If the authorization policy denies access, the user is redirected to another URL determined by the administrator.

Installation Overview

Installing an Oracle Retail supported Single Sign-On installation using OAM 11g requires installation of the following:

1. Oracle Internet Directory (OID) LDAP server and the Oracle Directory Services Manager. They are typically installed using the Installer of Oracle Identity Management . The ODSM application can be used for user and realm management within OID.
2. Oracle Access Manager 12C must be installed and configured.
3. Additional application servers to deploy other Oracle Retail applications and performing application specific initialization and deployment activities must be registered with OAM installed in step 2.

Infrastructure Installation and Configuration

The Infrastructure installation for Oracle Access Manager (OAM) is dependent on the environment and requirements for its use. Deploying Oracle Access Manager (OAM) to be used in a test environment does not have the same availability requirements as for a production environment. Similarly, the Oracle Internet Directory (OID) LDAP server can be deployed in a variety of different configurations. See the *Oracle Identity Management Installation Guide 12c.*

OID User Data

Oracle Internet Directory is an LDAP v3 compliant directory server. It provides standards-based user definitions out of the box.

Customers with existing corporate LDAP implementations may need to synchronize user information between their existing LDAP directory servers and OID. OID supports standard LDIF file formats and provides a JNDI compliant set of Java classes as well. Moreover, OID provides additional synchronization and replication facilities to integrate with other corporate LDAP implementations.

Each user ID stored in OID has a specific record containing user specific information. For role-based access, groups of users can be defined and managed within OID. Applications can thus grant access based on group (role) membership saving administration time and providing a more secure implementation.

User Management

User Management consists of displaying, creating, updating or removing user information. There are many methods of managing an LDAP directory including LDIF scripts or Oracle Directory Services Manager (ODSM) available for OID12c.

ODSM

Oracle Directory Services Manager (ODSM) is a Web-based application used in OID12c and is designed for both administrators and users and enables you to configure the structure of the directory, define objects in the directory, add and configure users,

groups, and other entries. ODSM is the interface you use to manage entries, schema, security, adapters, extensions, and other directory features.

LDIF Scripts

Script based user management can be used to synchronize data between multiple LDAP servers. The standard format for these scripts is the LDAP Data Interchange Format (LDIF). OID supports LDIF script for importing and exporting user information. LDIF scripts may also be used for bulk user load operations.

User Data Synchronization

The user store for Oracle Access Manager resides within the Oracle Internet Directory (OID) LDAP server. Oracle Retail applications may require additional information attached to a user name for application-specific purposes and may be stored in an application-specific database. Currently, there are no Oracle Retail tools for synchronizing changes in OID stored information with application-specific user stores. Implementers should plan appropriate time and resources for this process. Oracle Retail strongly suggests that you configure any Oracle Retail application using an LDAP for its user store to point to the same OID server used with Oracle Access Manager.

F

Appendix: Setting Up Password Stores with wallets/credential stores

As part of an application installation, administrators must set up password stores for user accounts using wallets/credential stores. Some password stores must be installed on the application database side. While the installer handles much of this process, the administrators must perform some additional steps.

Password stores for the application and application server user accounts must also be installed; however, the installer takes care of this entire process.

ORACLE Retail Merchandising applications now have 3 different types of password stores. They are database wallets, java wallets, and database credential stores. Background and how to administer them below are explained in this appendix

About Database Password Stores and Oracle Wallet

Oracle databases have allowed other users on the server to see passwords in case database connect strings (username/password@db) were passed to programs. In the past, users could navigate to `ps –ef|grep <username>` to see the password if the password was supplied in the command line when calling a program.

To make passwords more secure, Oracle Retail has implemented the Oracle Software Security Assurance (OSSA) program. Sensitive information such as user credentials now must be encrypted and stored in a secure location. This location is called password stores or wallets. These password stores are secure software containers that store the encrypted user credentials.

Users can retrieve the credentials using aliases that were set up when encrypting and storing the user credentials in the password store. For example, if `username/password@db` is entered in the command line argument and the alias is called `db_username`, the argument to a program is as follows:

```
sqlplus /@db_username
```

This would connect to the database as it did previously, but it would hide the password from any system user.

After this is configured, as in the example above, the application installation and the other relevant scripts are no longer needed to use embedded usernames and passwords. This reduces any security risks that may exist because usernames and passwords are no longer exposed.

When the installation starts, all the necessary user credentials are retrieved from the Oracle Wallet based on the alias name associated with the user credentials.

There are three different types of password stores. One type explain in the next section is for database connect strings used in program arguments (such as `sqlplus /@db_username`). The others are for Java application installation and application use.

Setting Up Password Stores for Database User Accounts

After the database is installed and the default database user accounts are set up, administrators must set up a password store using the Oracle wallet. This involves assigning an alias for the username and associated password for each database user account. The alias is used later during the application installation. This password store must be created on the system where the application server and database client are installed.

This section describes the steps you must take to set up a wallet and the aliases for the database user accounts. For more information on configuring authentication and password stores, see the *Oracle Database Security Guide*.

> **Note:** In this section, `<wallet_location>` is a placeholder text for illustration purposes. Before running the command, ensure that you specify the path to the location where you want to create and store the wallet.

To set up a password store for the database user accounts, perform the following steps:

1. Create a wallet using the following command:

   ```
   mkstore -wrl <wallet_location> -create
   ```

   After you run the command, a prompt appears. Enter a password for the Oracle Wallet in the prompt.

   > **Note:** The `mkstore` utility is included in the Oracle Database Client installation.

   The wallet is created with the auto-login feature enabled. This feature enables the database client to access the wallet contents without using the password. For more information, refer to the *Oracle Database Advanced Security Administrator's Guide.*

2. Create the database connection credentials in the wallet using the following command:

   ```
   mkstore -wrl <wallet_location> -createCredential <alias-name> <database-user-name>
   ```

   After you run the command, a prompt appears. Enter the password associated with the database user account in the prompt.

3. Repeat Step 2 for all the database user accounts.

4. Update the sqlnet.ora file to include the following statements:

   ```
   WALLET_LOCATION = (SOURCE = (METHOD = FILE) (METHOD_DATA = (DIRECTORY = <wallet_location>)))
   SQLNET.WALLET_OVERRIDE = TRUE
   SSL_CLIENT_AUTHENTICATION = FALSE
   ```

5. Update the tnsnames.ora file to include the following entry for each alias name to be set up.

   ```
   <alias-name> =
       (DESCRIPTION =
        (ADDRESS_LIST =
               (ADDRESS = (PROTOCOL = TCP) (HOST = <host>) (PORT = <port>))
           )
           (CONNECT_DATA =
               (SERVICE_NAME = <service>)
            )
        )
   ```

In the previous example, <alias-name>, <host>, <port>, and <service> are placeholder text for illustration purposes. Ensure that you replace these with the relevant values.

Setting up Wallets for Database User Accounts

The following examples show how to set up wallets for database user accounts for the following applications:

- For RMS, RWMS, RPM Batch using sqlplus or sqlldr, RETL, RMS and RWMS

For RMS, RWMS, RPM Batch using sqlplus or sqlldr, RETL, RMS, RWMS, and ARI

To set up wallets for database user accounts, do the following.

1. Create a new directory called wallet under your folder structure.

```
cd /projects/rms16/dev/
mkdir .wallet
```

> **Note:** The default permissions of the wallet allow only the owner to use it, ensuring the connection information is protected. If you want other users to be able to use the connection, you must adjust permissions appropriately to ensure only authorized users have access to the wallet.

2. Create a sqlnet.ora in the wallet directory with the following content.

```
WALLET_LOCATION =   (SOURCE =     (METHOD = FILE)     (METHOD_DATA =
(DIRECTORY =  /projects/rms16/dev/.wallet)) )
SQLNET.WALLET_OVERRIDE=TRUE
SSL_CLIENT_AUTHENTICATION=FALSE
```

> **Note**: WALLET_LOCATION must be on line 1 in the file.

3. Setup a tnsnames.ora in the wallet directory. This tnsnames.ora includes the standard tnsnames.ora file. Then, add two custom tns_alias entries that are only for use with the wallet. For example, `sqlplus /@dvols29_rms01user`.

```
ifile = /u00/oracle/product/19.3.0.0/network/admin/tnsnames.ora

Examples for a NON pluggable db:
dvols29_rms01user =
  (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = tcp)
  (host = xxxxxx.us.oracle.com) (Port = 1521)))
    (CONNECT_DATA = (SID = <sid_name> (GLOBAL_NAME = <sid_name>)))

dvols29_rms01user.world =
  (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = tcp)
  (host = xxxxxx.us.oracle.com) (Port = 1521)))
    (CONNECT_DATA = (SID = <sid_name>) (GLOBAL_NAME = <sid_name>)))

Examples for a pluggable db:
dvols29_rms01user =
  (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = tcp)
  (host = xxxxxx.us.oracle.com) (Port = 1521)))
    (CONNECT_DATA = (SERVICE_NAME = <pluggable db name>)))

dvols29_rms01user.world =
  (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = tcp)
  (host = xxxxxx.us.oracle.com) (Port = 1521)))
    (CONNECT_DATA = (SERVICE_NAME = <pluggable db name>)))
```

Note: It is important to not just copy the tnsnames.ora file because it can quickly become out of date. The ifile clause (shown above) is key.

4. Create the wallet files. These are empty initially.
   a. Ensure you are in the intended location.
      ```
      $ pwd
      /projects/rms16/dev/.wallet
      ```
   b. Create the wallet files.
      ```
      $ mkstore -wrl . –create
      ```
   c. Enter the wallet password you want to use. It is recommended that you use the same password as the UNIX user you are creating the wallet on.
   d. Enter the password again.

      Two wallet files are created from the above command:
      - ewallet.p12
      - cwallet.sso
5. Create the wallet entry that associates the user name and password to the custom tns alias that was setup in the wallet's tnsnames.ora file.
   ```
   mkstore –wrl . –createCredential <tns_alias> <username> <password>
   ```

Example: `mkstore –wrl . –createCredential dvols29_rms01user rms01user passwd`

6. Test the connectivity. The ORACLE_HOME used with the wallet must be the same version or higher than what the wallet was created with.
   ```
   $ export TNS_ADMIN=/projects/rms16/dev/.wallet /* This is very import to use
   wallet to point at the alternate tnsnames.ora created in this example */

   $ sqlplus /@dvols29_rms01user

   SQL*Plus: Release 19.0.0.0.0 - Production on Fri Feb 28 09:40:53 2020
   Version 19.3.0.0.0

   Copyright (c) 1982, 2019, Oracle.  All rights reserved.

   Last Successful login time: Fri Feb 28 2020 09:21:25 -08:00

   Connected to:
   Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
   Version 19.3.0.0.0

   SQL> show user
   USER is “rms01user”
   ```

   Running batch programs or shell scripts would be similar:

   ```
   Ex: dtesys /@dvols29_rms01user
   script.sh /@dvols29_rms01user

   Set the UP unix variable to help with some compiles :

   export UP=/@dvols29_rms01user
   for use in RMS batch compiles, and RMS, RWMS, and ARI forms compiles.
   ```

As shown in the example above, users can ensure that passwords remain invisible.

Additional Database Wallet Commands

The following is a list of additional database wallet commands.

- Delete a credential on wallet

  ```
  mkstore –wrl . –deleteCredential dvols29_rms01user
  ```

- Change the password for a credential on wallet

  ```
  mkstore –wrl . –modifyCredential dvols29_rms01user rms01user passwd
  ```

- List the wallet credential entries

  ```
  mkstore –wrl . –list
  ```

  This command returns values such as the following.

  ```
  oracle.security.client.connect_string1
  oracle.security.client.user1
  oracle.security.client.password1
  ```

- View the details of a wallet entry

  ```
  mkstore –wrl . –viewEntry oracle.security.client.connect_string1
  ```

  Returns the value of the entry:

  ```
  dvols29_rms01user
  mkstore –wrl . –viewEntry oracle.security.client.user1
  ```

  Returns the value of the entry:

  ```
  rms01user
  ```

  ```
  mkstore –wrl . –viewEntry oracle.security.client.password1
  ```

  Returns the value of the entry:

  ```
  Passwd
  ```

Setting up RETL Wallets

RETL creates a wallet under $RFX_HOME/etc/security, with the following files:

- cwallet.sso
- jazn-data.xml
- jps-config.xml
- README.txt

To set up RETL wallets, perform the following steps:

1. Set the following environment variables:
   - `ORACLE_SID=<retaildb>`
   - `RFX_HOME=/u00/rfx/rfx-13`
   - `RFX_TMP=/u00/rfx/rfx-13/tmp`
   - `JAVA_HOME=/usr/jdk1.6.0_12.64bit`
   - `LD_LIBRARY_PATH=$ORACLE_HOME`
   - `PATH=$RFX_HOME/bin:$JAVA_HOME/bin:$PATH`
2. Change directory to $RFX_HOME/bin.
3. Run setup-security-credential.sh.
   - Enter 1 to add a new database credential.
   - Enter the dbuseralias. For example, `retl_java_rms01user`.
   - Enter the database user name. For example, `rms01user`.

- Enter the database password.
- Re-enter the database password.
- Enter D to exit the setup script.

4. Update your RETL environment variable script to reflect the names of both the Oracle Networking wallet and the Java wallet.

   For example, to configure RETLforRPAS, modify the following entries in `$RETAIL_HOME/RETLforRPAS/rfx/etc/rmse_rpas_config.env.`

   - The RETL_WALLET_ALIAS should point to the Java wallet entry:
     - `export RETL_WALLET_ALIAS="retl_java_rms01user"`
   - The ORACLE_WALLET_ALIAS should point to the Oracle network wallet entry:
     - `export ORACLE_WALLET_ALIAS="dvols29_rms01user"`
   - The SQLPLUS_LOGON should use the ORACLE_WALLET_ALIAS:
     - `export SQLPLUS_LOGON="/@${ORACLE_WALLET_ALIAS}"`

5. To change a password later, run `setup-security-credential.sh`.
   - Enter 2 to update a database credential.
   - Select the credential to update.
   - Enter the database user to update or change.
   - Enter the password of the database user.
   - Re-enter the password.

For Java Applications (SIM, ReIM, RPM, RIB, AIP, Alloc, ReSA, RETL)

For Java applications, consider the following:

- For database user accounts, ensure that you set up the same alias names between the password stores (database wallet and Java wallet). You can provide the alias name during the installer process.
- Document all aliases that you have set up. During the application installation, you must enter the alias names for the application installer to connect to the database and application server.
- Passwords are not used to update entries in Java wallets. Entries in Java wallets are stored in partitions, or application-level keys. In each retail application that has been installed, the wallet is located in <WEBLOGIC_DOMAIN_HOME>/retail/<appname>/config Example: /u00/webadmin/config/domains/wls_retail/RPMDomain/retail/rpm/config
- Application installers should create the Java wallets for you, but it is good to know how this works for future use and understanding.
- Scripts are located in <WEBLOGIC_DOMAIN_HOME>/retail/<appname>/retail-public-security-api/bin for administering wallet entries.
- Example:
- /u00/webadmin/config/domains/wls_retail/RPMDomain/retail/rpm/retail-public-security-api/bin
- In this directory is a script to help you update each alias entry without having to remember the wallet details. For example, if you set the RPM database alias to rms01user, you will find a script called update-RMS01USER.sh.

> **Note:** These scripts are available only with applications installed by way of an installer.

- Two main scripts are related to this script in the folder for more generic wallet operations: dump_credentials.sh and save_credential.sh.
- If you have not installed the application yet, you can unzip the application zip file and view these scripts in <app>/application/retail-public-security-api/bin.
- Example:
- /u00/webadmin/rpm/application/rpm/Build/orpatch/deploy/retail-public-security-api/bin

update-<ALIAS>.sh

update-<ALIAS>.sh updates the wallet entry for this alias. You can use this script to change the user name and password for this alias. Because the application refers only to the alias, no changes are needed in application properties files.

Usage:

```
update-<username>.sh <myuser>
```

Example:

```
/u00/webadmin/config/domains/wls_retail/RPMDomain/retail/rpm/retail-public-
security-api/bin> ./update-RMS01USER.sh
usage: update-RMS01USER.sh <username>
<username>: the username to update into this alias.
Example: update-RMS01USER.sh myuser
Note: this script will ask you for the password for the username that you pass in.
/u00/webadmin/config/domains/wls_retail/RPMDomain/retail/rpm/retail-public-
security-api/bin>
```

dump_credentials.sh

dump_credentials.sh is used to retrieve information from wallet. For each entry found in the wallet, the wallet partition, the alias, and the user name are displayed. Note that the password is not displayed. If the value of an entry is uncertain, run save_credential.sh to resave the entry with a known password.

```
dump_credentials.sh <wallet location>
```

Example:

```
dump_credentials.sh location:
/u00/webadmin/config/domains/wls_retail/RPMDomain/retail/rpm/config

Retail Public Security API Utility
==========================================
```

Below are the credentials found in the wallet at the location/u00/webadmin/config/domains/wls_retail/RPMDomain/retail/rpm/config

```
==========================================

Application level key partition name:rpm
User Name Alias:WLS-ALIAS User Name:weblogic
User Name Alias:RETAIL-ALIAS User Name:retail.user
User Name Alias:LDAP-ALIAS User Name:RETAIL.USER
User Name Alias:RMS-ALIAS User Name:rms16mock
User Name Alias:REIMBAT-ALIAS User Name:rpmbat
```

save_credential.sh

save_credential.sh is used to update the information in wallet. If you are unsure about the information that is currently in the wallet, use dump_credentials.sh as indicated above.

```
save_credential.sh -a <alias> -u <user> -p <partition name>  –l <path of the
wallet file location where credentials are stored>
```

Example:

```
/u00/webadmin/mock16_testing/rpm16/application/retail-public-security-api/bin>
save_credential.sh -l wallet_test -a myalias -p mypartition -u myuser

==========================================
Retail Public Security API Utility
==========================================

Enter password:
Verify password:
```

Note: -p in the above command is for partition name. You must specify the proper partition name used in application code for each Java application.

save_credential.sh and dump_credentials.sh scripts are the same for all applications. If using save_credential.sh to add a wallet entry or to update a wallet entry, bounce the application/managed server so that your changes are visible to the application. Also, save a backup copy of your cwallet.sso file in a location outside of the deployment path, because redeployment or reinstallation of the application will wipe the wallet entries you made after installation of the application. To restore your wallet entries after a redeployment/reinstallation, copy the backed up cwallet.sso file over the cwallet.sso file. Then bounce the application/managed server.

Usage

```
==========================================
Retail Public Security API Utility
==========================================
usage: save_credential.sh -au[plh]
E.g. save_credential.sh -a rms-alias -u rms_user -p rib-rms -l ./
 -a,--userNameAlias <arg>              alias for which the credentials
needs to be stored
 -h,--help                             usage information
 -l,--locationofWalletDir <arg>        location where the wallet file is
created.If not specified, it creates the wallet under secure-credential-wallet
directory which is already present under the retail-public-security-api/
directory.
 -p,--appLevelKeyPartitionName <arg>   application level key partition name
 -u,--userName <arg>                   username to be stored in secure
credential wallet for specified alias*
```

How does the Wallet Relate to the Application?

The ORACLE Retail Java applications have the wallet alias information you create in an <app-name>.properties file. Below is the reim.properties file. Note the database information and the user are presented as well. The property called datasource.credential.alias=RMS-ALIAS uses the ORACLE wallet with the argument of RMS-ALIAS at the csm.wallet.path and csm.wallet.partition.name = rpm to retrieve the password for application use.

Reim.properties code sample:

```
datasource.url=jdbc:oracle:thin:@xxxxxxx.us.oracle.com:1521:pkols07
datasource.schema.owner=rms16mock
datasource.credential.alias=RMS-ALIAS
# ================================================================
# ossa related Configuration
#
# These settings are for ossa configuration to store credentials.
# ================================================================

csm.wallet.path=/u00/webadmin/config/domains/wls_retail/RPMDomain/retail/rpm/confi
g
csm.wallet.partition.name=rpm
```

How does the Wallet Relate to Java Batch Program use?

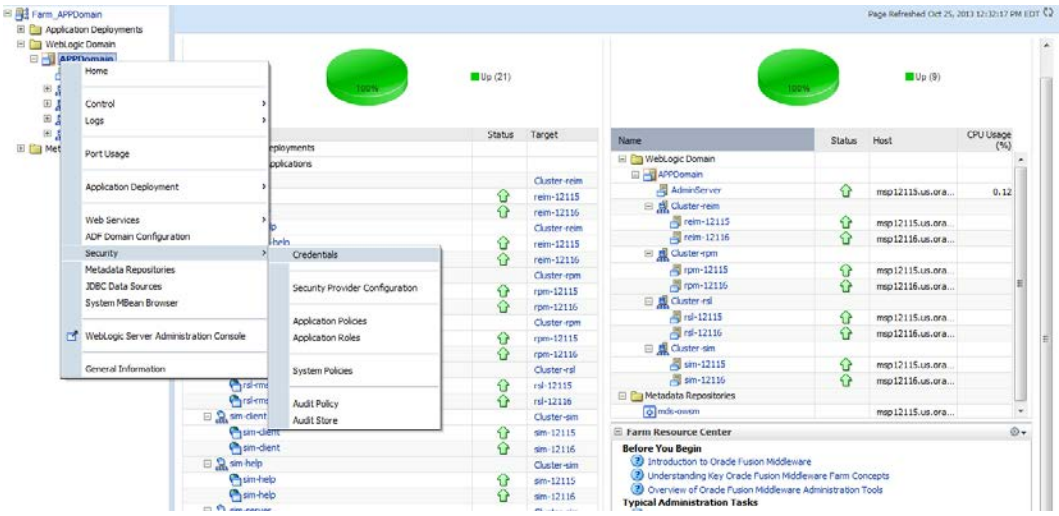
Some of the ORACLE Retail Java batch applications have an alias to use when running Java batch programs. For example, alias REIMBAT-ALIAS maps through the wallet to dbuser RMS01APP, already on the database. To run a ReIM batch program the format would be: reimbatchpgmname REIMBAT-ALIAS <other arguments as needed by the program in question>

Database Credential Store Administration

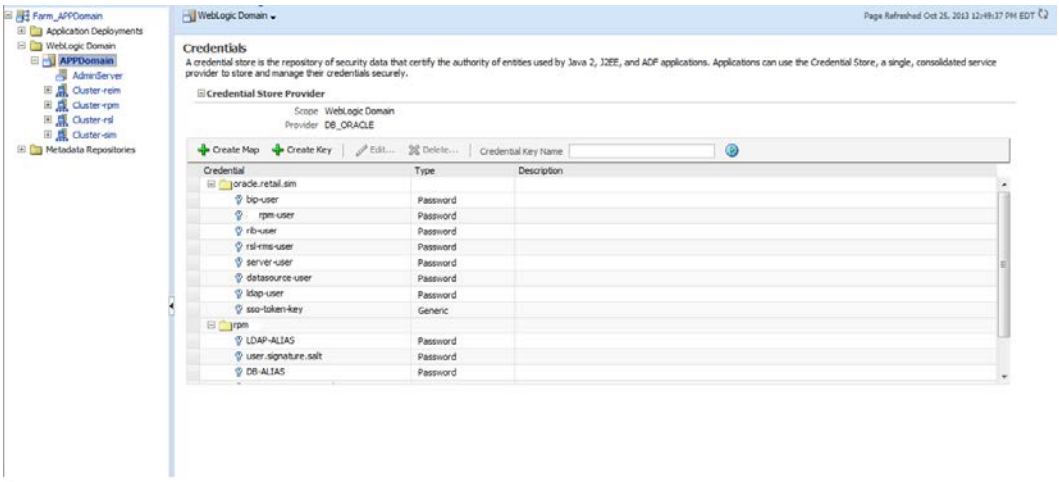
The following section describes a domain level database credential store. This is used in RPM login processing, SIM login processing, RWMS login processing, RESA login processing and Allocation login processing and policy information for application permission. Setting up the database credential store is addressed in the RPM, SIM, RESA, RWMS, and Alloc install guides.

The following sections show an example of how to administer the password stores thru ORACLE Enterprise Manger Fusion Middleware Control, a later section will show how to do this thru WLST scripts.

1. The first step is to use your link to Oracle Enterprise Manager Fusion Middleware Control for the domain in question. Locate your domain on the left side of the screen and do a right mouse click on the domain and select **Security > Credentials**

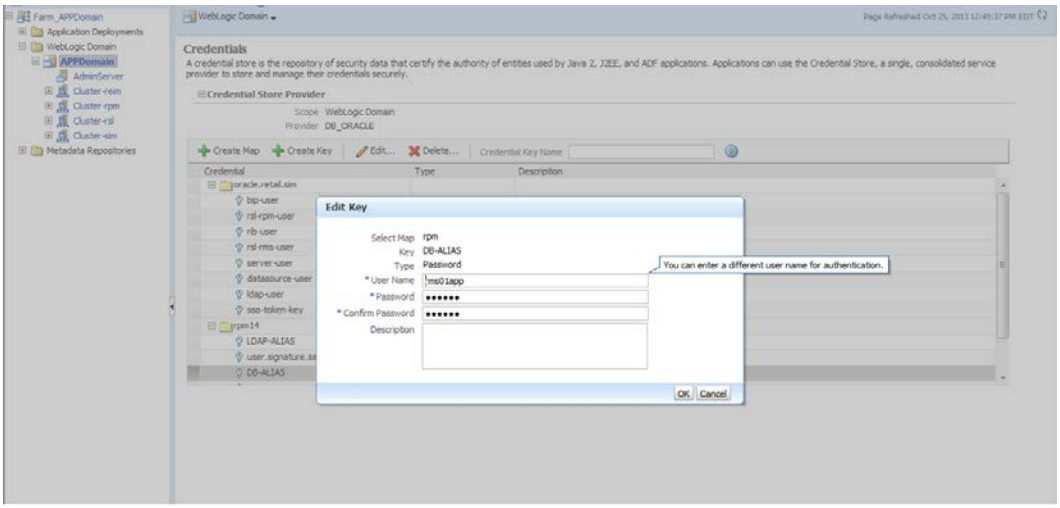


2. Click on Credentials and you will get a screen similar to the following. The following screen is expanded to make it make more sense. From here you can administer credentials.



The Create Map add above is to create a new map with keys under it. A map would usually be an application such as rpm. The keys will usually represent alias to various users (database user, WebLogic user, LDAP user, and so on). The application installer should add the maps so you should not often have to add a map.

Creation of the main keys for an application will also be built by the application installer. You will not be adding keys often as the installer puts the keys out and the keys talk to the application. You may be using EDIT on a key to see what user the key/alias points to and possibly change/reset its password. To edit a key/alias, highlight the key/alias in question and push the edit icon nearer the top of the page. You will then get a screen as follows:



The screen above shows the map (rpm) that came from the application installer, the key (DB-ALIAS) that came from the application installer (some of the keys/alias are selected by the person who did the application install, some are hard coded by the application installer in question), the type (in this case password), and the user name and password. This is where you would check to see that the user name is correct and reset the password if needed. REMEMBER, a change to an item like a database password WILL make you come into this and also change the password. Otherwise your application will NOT work correctly.

Managing Credentials with WSLT/OPSS Scripts

This procedure is optional as you can administer the credential store through the Oracle enterprise manager associated with the domain of your application install for ReIM, RPM, SIM, RESA, or Allocation.

An Oracle Platform Security Scripts (OPSS) script is a WLST script, in the context of the Oracle WebLogic Server. An online script is a script that requires a connection to a running server. Unless otherwise stated, scripts listed in this section are online scripts and operate on a database credential store. There are a few scripts that are offline, that is, they do not require a server to be running to operate.

Read-only scripts can be performed only by users in the following WebLogic groups: Monitor, Operator, Configurator, or Admin. Read-write scripts can be performed only by users in the following WebLogic groups: Admin or Configurator. All WLST scripts are available out-of-the-box with the installation of the Oracle WebLogic Server.

WLST scripts can be run in interactive mode or in script mode. In interactive mode, you enter the script at a command-line prompt and view the response immediately after. In

script mode, you write scripts in a text file (with a py file name extension) and run it without requiring input, much like the directives in a shell script.

The weakness with the WLST/OPSS scripts is that you have to already know your map name and key name. In many cases, you do not know or remember that. The database credential store way through enterprise manager is a better way to find your map and key names easily when you do not already know them. A way in a command line mode to find the map name and alias is to run orapki. An example of orapki is as follows:

/u00/webadmin/product/wls_apps/oracle_common/bin> ./orapki wallet display –wallet
/u00/webadmin/product/wls_apps/user_projects/domains/APPDomain/config/fmwconfig

(where the path above is the domain location of the wallet)

Output of orapki is below. This shows map name of rpm and each alias in the wallet:

Requested Certificates:

User Certificates:

Oracle Secret Store entries:

rpm@#3#@DB-ALIAS

rpm@#3#@LDAP-ALIAS

rpm@#3#@RETAIL.USER

rpm@#3#@user.signature.salt

rpm@#3#@user.signature.secretkey

rpm@#3#@WEBLOGIC-ALIAS

rpm@#3#@WLS-ALIAS

Trusted Certificates:

Subject: OU=Class 1 Public Primary Certification Authority,O=VeriSign\, Inc.,C=US

OPSS provides the following scripts on all supported platforms to administer credentials (all scripts are online, unless otherwise stated. You need the map name and the key name to run the scripts below

- listCred
- updateCred
- createCred
- deleteCred
- modifyBootStrapCredential
- addBootStrapCredential

listCred

The script `listCred` returns the list of attribute values of a credential in the credential store with given map name and key name. This script lists the data encapsulated in credentials of type password only.

Script Mode Syntax

```
listCred.py -map mapName -key keyName
```

Interactive Mode Syntax

```
listCred(map="mapName", key="keyName")
```

The meanings of the arguments (all required) are as follows:

- map specifies a map name (folder).
- key specifies a key name.

Examples of Use:

The following invocation returns all the information (such as user name, password, and description) in the credential with map name myMap and key name myKey:

```
listCred.py -map myMap -key myKey
```

The following example shows how to run this command and similar credential commands with WLST:

```
/u00/webadmin/product/wls_apps/oracle_common/common/bin>
sh wlst.sh

Initializing WebLogic Scripting Tool (WLST)...

Welcome to WebLogic Server Administration Scripting Shell


wls:/offline> connect('weblogic','password123','xxxxxx.us.oracle.com:17001')
Connecting to t3://xxxxxx.us.oracle.com:17001 with userid weblogic ...
Successfully connected to Admin Server 'AdminServer' that belongs to domain
'APPDomain'.

wls:/APPDomain/serverConfig> listCred(map="rpm",key="DB-ALIAS")
Already in Domain Runtime Tree

[Name : rms01app, Description : null, expiry Date : null]
PASSWORD:retail
*The above means for map rpm in APPDomain, alias DB-ALIAS points to database user
rms01app with a password of retail
```

updateCred

The script updateCred modifies the type, user name, and password of a credential in the credential store with given map name and key name. This script updates the data encapsulated in credentials of type password only. Only the interactive mode is supported.

Interactive Mode Syntax

```
updateCred(map="mapName", key="keyName", user="userName", password="passW",
[desc="description"])
```

The meanings of the arguments (optional arguments are enclosed by square brackets) are as follows:

- map specifies a map name (folder) in the credential store.
- key specifies a key name.
- user specifies the credential user name.
- password specifies the credential password.
- desc specifies a string describing the credential.

Example of Use:

The following invocation updates the user name, password, and description of the password credential with map name myMap and key name myKey:

```
updateCred(map="myMap", key="myKey", user="myUsr", password="myPassw")
```

createCred

The script createCred creates a credential in the credential store with a given map name, key name, user name and password. This script can create a credential of type password only. Only the interactive mode is supported.

Interactive Mode Syntax

```
createCred(map="mapName", key="keyName", user="userName", password="passW",
[desc="description"])
```

The meanings of the arguments (optional arguments are enclosed by square brackets) are as follows:

- map specifies the map name (folder) of the credential.
- key specifies the key name of the credential.
- user specifies the credential user name.
- password specifies the credential password.
- desc specifies a string describing the credential.

Example of Use:

The following invocation creates a password credential with the specified data:

```
createCred(map="myMap", key="myKey", user="myUsr", password="myPassw")
```

deleteCred

The script deleteCred removes a credential with given map name and key name from the credential store.

Script Mode Syntax

```
deleteCred.py -map mapName -key keyName
```

Interactive Mode Syntax

```
deleteCred(map="mapName",key="keyName")
```

The meanings of the arguments (all required) are as follows:

- map specifies a map name (folder).
- key specifies a key name.

Example of Use:

The following invocation removes the credential with map name myMap and key name myKey:

```
deleteCred.py -map myMap -key myKey
```

modifyBootStrapCredential

The offline script modifyBootStrapCredential modifies the bootstrap credentials configured in the default jps context, and it is typically used in the following scenario: suppose that the policy and credential stores are LDAP-based, and the credentials to access the LDAP store (stored in the LDAP server) are changed. Then this script can be used to seed those changes into the bootstrap credential store.

This script is available in interactive mode only.

Interactive Mode Syntax

```
modifyBootStrapCredential(jpsConfigFile="pathName", username="usrName",
password="usrPass")
```

The meanings of the arguments (all required) are as follows:

- `jpsConfigFile` specifies the location of the file `jps-config.xml` relative to the location where the script is run. Example location: /u00/webadmin/product/wls_apps/user_projects/domains/APPDomain/config/fmwconfig. Example location of the bootstrap wallet is /u00/webadmin/product/wls_apps/user_projects/domains/APPDomain/config/fmwconfig/bootstrap
- `username` specifies the distinguished name of the user in the LDAP store.
- `password` specifies the password of the user.

Example of Use:

Suppose that in the LDAP store, the password of the user with distinguished name `cn=orcladmin` has been changed to `<password>`, and that the configuration file `jps-config.xml` is located in the current directory.Then the following invocation changes the password in the bootstrap credential store to `<password>`:

```
modifyBootStrapCredential(jpsConfigFile='./jps-config.xml',
username='cn=orcladmin', password='<password>')
```

Any output regarding the audit service can be disregarded.

addBootStrapCredential

The offline script `addBootStrapCredential` adds a password credential with given map, key, user name, and user password to the bootstrap credentials configured in the default jps context of a jps configuration file.

Classloaders contain a hierarchy with parent classloaders and child classloaders. The relationship between parent and child classloaders is analogous to the object relationship of super classes and subclasses. The bootstrap classloader is the root of the Java classloader hierarchy. The Java virtual machine (JVM) creates the bootstrap classloader, which loads the Java development kit (JDK) internal classes and `java.*` packages included in the JVM. (For example, the bootstrap classloader loads `java.lang.String`.)

This script is available in interactive mode only.

Interactive Mode Syntax

```
addBootStrapCredential(jpsConfigFile="pathName", map="mapName", key="keyName",
username="usrName", password="usrPass")
```

The meanings of the arguments (all required) are as follows:

- `jpsConfigFile` specifies the location of the file `jps-config.xml` relative to the location where the script is run. Example location: /u00/webadmin/product/wls_apps/user_projects/domains/APPDomain/config/fmwconfig
- `map` specifies the map of the credential to add.
- `key` specifies the key of the credential to add.
- `username` specifies the name of the user in the credential to add.
- `password` specifies the password of the user in the credential to add.

Example of Use:

The following invocation adds a credential to the bootstrap credential store:

```
addBootStrapCredential(jpsConfigFile='./jps-config.xml', map='myMapName',
key='myKeyName', username='myUser', password ='myPass')
```

Quick Guide for Retail Password Stores (db wallet, java wallet, DB credential stores)

Retail app	Wallet type	Wallet loc	Wallet partition	Alias name	User name	Use	Create by	Alias Example	Notes
RMS batch	DB	<RMS batch install dir (RETAIL_HOME)>/.wallet	n/a	<Database SID>_<Database schema owner>	<rms schema owner>	Compile, execution	Installer	n/a	Alias hard-coded by installer
RMWS forms	DB	<forms install dir>/base/.wallet	n/a	<Database SID>_<Database schema owner>	<rwms schema owner>	Compile forms, execute batch	Installer	n/a	Alias hard-coded by installer
RPM batch plsql and sqlldr	DB	<RPM batch install dir>/.wallet	n/a	<rms schema owner alias>	<rms schema owner>	Execute batch	Manual	rms-alias	RPM plsql and sqlldr batches
RWMS auto-login	JAVA	<forms install dir>/base/.javawallet							
			<RWMS Installation name>	<RWMS database user alias>	<RWMS schema owner>	RWMS forms app to avoid dblogin screen	Installer	rwms16inst	
			<RWMS Installation name>	BI_ALIAS	<BI Publisher administrative user>	RWMS forms app to connect to BI Publisher	Installer	n/a	Alias hard-coded by installer
AIP app	JAVA	<weblogic domain home>/retail/<deployed aip app name>/config							Each alias must be unique

Retail app	Wallet type	Wallet loc	Wallet partition	Alias name	User name	Use	Create by	Alias Example	Notes
			aip	<AIP weblogic user alias>	<AIP weblogic user name>	App use	Installer	aip-weblogic-alias	
			aip	<AIP database schema user alias>	<AIP database schema user name>	App use	Installer	aip01user-alias	
			aip	<rib-aip weblogic user alias>	<rib-aip weblogic user name>	App use	Installer	rib-aip-weblogic-alias	
RPM app	DB credential store		Map=rpm or what you called the app at install time.	Many for app use					<weblogic domain home>/config/fmwconfig/jps-config.xml has info on the credential store. This directory also has the domain cwallet.sso file.
RPM app	JAVA	<weblogic domain home>/retail/<deployed rpm app name>/config							Each alias must be unique
			rpm	<rpm weblogic user alias>	<rpm weblogic user name>	App use	Installer	rpm-weblogic-alias	

Retail app	Wallet type	Wallet loc	Wallet partition	Alias name	User name	Use	Create by	Alias Example	Notes
			rpm	<rpm batch user name> is the alias. Yes, here alias name = user name	<rpm batch user name>	App, batch use	Installer	RETAIL.USER	
	JAVA	<retail_home>/orpatch/config/javaapp_rpm							Each alias must be unique
			retail_installer	<rpm weblogic user alias>	<rpm weblogic user name>	App use	Installer	weblogic-alias	
			retail_installer	<rms shema user alias>	<rms shema user name>	App, batch use	Installer	rms01user-alias	
			retail_installer	<reim batch user alias>	<reim batch user name>	App, batch use	Installer	reimbat-alias	
			retail_installer	<LDAP-ALIAS>	cn=rpm.admin,cn=Users,dc=us,dc=oracle,dc=com	LDAP user use	Installer	LDAP_ALIAS	
ReIM app	JAVA	<weblogic domain home>/retail/<deployed reim app name>/config							Each alias must be unique
			<installed app name, ex: reim>	<reim weblogic user alias>	<reim weblogic user name>	App use	Installer	weblogic-alias	
			<installed app name, ex: reim>	<rms shema user alias>	<rms shema user name>	App, batch use	Installer	rms01user-alias	

Retail app	Wallet type	Wallet loc	Wallet partition	Alias name	User name	Use	Create by	Alias Example	Notes
			<installed app name, ex: reim>	<reim webservice validation user alias>	<reim webservice validation user name>	App use	Installer	reimwebservice-alias	
			<installed app name, ex: reim>	<reim batch user alias>	<reim batch user name>	App, batch use	Installer	reimbat-alias	
			<installed app name, ex: reim>	<LDAP-ALIAS>	cn=REIM.ADMIN,cn=Users,dc=us,dc=oracle,dc=com	LDAP user use	Installer	LDAP_ALIAS	
	JAVA	<retail_home>/orpatch/config/javaapp_reim							Each alias must be unique
			retail_installer	<reim weblogic user alias>	<reim weblogic user name>	App use	Installer	weblogic-alias	
			retail_installer	<rms shema user alias>	<rms shema user name>	App, batch use	Installer	rms01user-alias	
			retail_installer	<reim webservice validation user alias>	<reim webservice validation user name>	App use	Installer	reimwebservice-alias	
			retail_installer	<reim batch user alias>	<reim batch user name>	App, batch use	Installer	reimbat-alias	
			retail_installer	<LDAP-ALIAS>	cn=REIM.ADMIN,cn=Users,dc=us,dc=oracle,dc=com	LDAP user use	Installer	LDAP_ALIAS	

Retail app	Wallet type	Wallet loc	Wallet partition	Alias name	User name	Use	Create by	Alias Example	Notes
RESA app	DB credential store		Map=resaor what you called the app at install time	Many for login and policies					<weblogic domain home>/config/fmwconfig/jps-config.xml has info on the credential store. This directory also has the domain cwallet.sso file. The bootstrap directory under this directory has bootstrap cwallet.sso file.
RESA app	JAVA	<weblogic domain home>/retail/<deployed resa app name>/config							Each alias must be unique
			<installed app name>	<resa weblogic user alias>	<resa weblogic user name>	App use	Installer	wlsalias	
			<installed app name>	<resa schema db user alias>	<rmsdb shema user name>	App use	Installer	Resadb-alias	
			<installed app name>	<resa schema user alias>	<rmsdb shema user name>>	App use	Installer	resa-alias	
	JAVA	<retail_home>/orpatch/config/javaapp_resa							Each alias must be unique

Retail app	Wallet type	Wallet loc	Wallet partition	Alias name	User name	Use	Create by	Alias Example	Notes
			retail_installer	<resa weblogic user alias>	<resa weblogic user name>	App use	Installer	wlsalias	
			retail_installer	<resa schema db user alias>	<rmsdb shema user name>	App use	Installer	Resadb-alias	
	JAVA	<retail_ home>/orpatch/config/javaapp_rasrm							Each alias must be unique
			retail_installer	<alloc weblogic user alias>	<alloc weblogic user name>	App use	Installer	weblogic-alias	
Alloc app	DB credential store		Map=alloc or what you called the app at install time	Many for login and policies					<weblogic domain home>/config/fmwconfig/jps-config.xml has info on the credential store. This directory also has the domain cwallet.sso file. The bootstrap directory under this directory has bootstrap cwallet.sso file.
Alloc app	JAVA	<weblogic domain home>/retail/config							Each alias must be unique

Retail app	Wallet type	Wallet loc	Wallet partition	Alias name	User name	Use	Create by	Alias Example	Notes
			<installed app name>	<alloc weblogic user alias>	<alloc weblogic user name>	App use	Installer	weblogic-alias	
			<installed app name>	<rms schema user alias>	<rms schema user name>	App use	Installer	dsallocAlias	
			<installed app name>	<alloc batch user alias>	<SYSTEM_ ADMINIST RATOR>	Batch use	Installer	alloc14	
	JAVA	<retail_ home>/orpatch/config/javaapp_alloc							Each alias must be unique
			retail_installer	<alloc weblogic user alias>	<alloc weblogic user name>	App use	Installer	weblogic-alias	
			retail_installer	<rms schema user alias>	<rms schema user name>	App use	Installer	dsallocAlias	
			retail_installer	<alloc batch user alias>	<SYSTEM_ ADMINIST RATOR>	Batch use	Installer	alloc14	
	JAVA	<retail_ home>/orpatch/config/javaapp_rasrm							Each alias must be unique
			retail_installer	<alloc weblogic user alias>	<alloc weblogic user name>	App use	Installer	weblogic-alias	

Retail app	Wallet type	Wallet loc	Wallet partition	Alias name	User name	Use	Create by	Alias Example	Notes
SIM app	DB credential store		Map=oracle.retail.sim	Aliases required for SIM app use					<weblogic domain home>/config/fmwconfig/jps-config.xml has info on the credential store. This directory also has the domain cwallet.sso file.
	JAVA	<weblogic domain home>/retail/<deployed sim app name>/batch/resources/conf	oracle.retail.sim	<sim batch user alias>	<sim batch user name>	App use	Installer	BATCH-ALIAS	
	JAVA	<weblogic domain home>/retail/<deployed sim app name>/wireless/resources/conf	oracle.retail.sim	<sim wireless user alias>	<sim wireless user name>	App use	Installer	WIRELESS-ALIAS	
RETL	JAVA	<RETL home>/etc/security	n/a	<target application user alias>	<target application db userid>	App use	Manual	retl_java_rms01user	User may vary depending on RETL flow's target application
RETL	DB	<RETL home>/.wallet	n/a	<target application user alias>	<target application db userid>	App use	Manual	<db>_<user>	User may vary depending on RETL flow's target application
RIB	JAVA	<RIBHOME DIR>/deployment-home/conf/security							<app> is one of aip, rfm, rms, rpm, sim, rwms, tafr
JMS			jms<1-5>	<jms user alias> for jms<1-5>	<jms user name> for jms<1-5>	Integration use	Installer	jms-alias	

Retail app	Wallet type	Wallet loc	Wallet partition	Alias name	User name	Use	Create by	Alias Example	Notes
WebLogic			rib-<app>-app-server-instance	<rib-app weblogic user alias>	<rib-app weblogic user name>	Integration use	Installer	weblogic-alias	
Admin GUI			rib-<app>#web-app-user-alias	<rib-app admin gui user alias>	<rib-app admin gui user name>	Integration use	Installer	admin-gui-alias	
Application			rib-<app>#user-alias	<app weblogic user alias>	<app weblogic user name>	Integration use	Installer	app-user-alias	Valid only for aip, rpm, sim
DB			rib-<app>#app-db-user-alias	<rib-app database schema user alias>	<rib-app database schema user name>	Integration use	Installer	db-user-alias	Valid only for rfm, rms, rwms, tafr
Error Hospital			rib-<app>#hosp-user-alias	<rib-app error hospital database schema user alias>	<rib-app error hospital database schema user name>	Integration use	Installer	hosp-user-alias	
RFI	Java	<RFI-HOME>/retail-financial-integration-solution/service-based-integration/conf/security							
			<installed app name>	rfiAppServerAdminServerUserAlias	<rfi weblogic user name>	App use	Installer	rfiAppServerAdminServerUserAlias	
			<installed app name>	rfiAdminUiUserAlias	<ORFI admin user>	App use	Installer	rfiAdminUiUserAlias	

Retail app	Wallet type	Wallet loc	Wallet partition	Alias name	User name	Use	Create by	Alias Example	Notes
			<installed app name>	rfiDataSourceUserAlias	<ORFI schema user name>	App use	Installer	rfiDataSourceUserAlias	
			<installed app name>	ebsDataSourceUserAlias	<EBS schema user name>	App use	Installer	ebsDataSourceUserAlias	
			<installed app name>	smtpMailFromAddressAlias	<From email address>	App use	Installer	smtpMailFromAddressAlias	

G

Appendix: Tablespace Creation

Non-Encrypted Tablespace Creation

Standard SIM tablespaces are created using the create_tablespaces.sql script located in <INSTALL_DIR>/sim/dbschema/dbutils /.

1. Update the paths of the script in <INSTALL_DIR>/sim/dbschema/dbutils /create_tablespaces.sql as pertain to your environment.
2. The table below shows the default initial sizes.

TABLESPACE_NAME	Size
SIM_ENCRYPTED_INDEX	12G
SIM_ENCRYPTED_DATA	10G
SIM_INDEX	10G
SIM_DATA	8G
SIM_LOB_DATA	2G
SIM_LOB_INDEX	2G
USERS	2G

3. Once the paths of script has been modified, execute it in SQL*Plus as sys.
4. Review create_tablespaces.log for errors and correct as needed.

Encrypted Tablespace Creation

If you do not have an Advanced Security Option license, create the sim_encrypted_data and sim_encrypted_index tablespaces as normal tablespaces but without the encryption.

1. Modify the paths of the script <INSTALL_DIR>/sim/dbschema/dbutils /create_encrypted_tablespaces_no_TDE.sql
2. Run the script using SQL*Plus as sys
3. Review create_encrypted_tablespaces_no_TDE.log for errors and correct as needed

With an Advanced Security license, tablespaces can be created in an encrypted format. The steps are:

Configure a Wallet

1. Create a sqlnet.ora in $TNS_ADMIN directory of the database server similar to the below entry:

```
ENCRYPTION_WALLET_LOCATION =
  (SOURCE = (METHOD = FILE)
    (METHOD_DATA =
      (DIRECTORY = /u00/oracle/admin/ORACLE_SID/wallet)))
```

2. Create the wallet directory:

```
mkdir –p /u00/oracle/admin/<ORACLE_SID>/wallet
```

3. As a user with the 'alter system' privilege, create the wallet as follows:

```
Non-container databases:
```

   a. ADMINISTER KEY MANAGEMENT CREATE KEYSTORE '/u00/oracle/admin/dbName/wallet' IDENTIFIED BY "pwd#";
   b. ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN IDENTIFIED BY "pwd#";
   c. ADMINISTER KEY MANAGEMENT SET KEY IDENTIFIED BY "pwd#" WITH BACKUP;
   d. ADMINISTER KEY MANAGEMENT CREATE AUTO_LOGIN KEYSTORE FROM KEYSTORE '/u00/oracle/admin/dbName/wallet' identified by pwd#;
   a. Container databases:
   b. ADMINISTER KEY MANAGEMENT CREATE KEYSTORE '/u00/oracle/admin/dbName/wallet' IDENTIFIED BY "pwd#";
   c. ADMINISTER KEY MANAGEMENT CREATE AUTO_LOGIN KEYSTORE FROM KEYSTORE '/u00/oracle/admin/dbName/wallet' identified by "pwd#";
   d. ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN IDENTIFIED BY "pwd#" Container=ALL;
   e. ADMINISTER KEY MANAGEMENT SET KEY IDENTIFIED BY "pwd#" WITH BACKUP USING 'TDE_ENCRYPTION' Container=all;

4. Confirm if the wallet is created and open (the TDE master encryption key has been created and inserted automatically):

```
SQL>
select substr(wrl_type, 1, 10) wrl_type, substr(wrl_parameter, 1, 45) param,
substr(status, 1, 10) status, substr(wallet_type, 1, 15) w_type
from v$encryption_wallet;

WRL_TYPE   PARAM                                    STATUS     W_TYPE
---------- ---------------------------------------- ---------- ---------------
FILE       /u00/oracle/admin/ORACLE_SID/wallet  OPEN       AUTOLOGIN


An auto-open wallet is created.  You are ready to create the encrypted
tablespaces as shown in the following section.
```

Encryption at Tablespace Level

Once the wallet is configured, determine an encryption algorithm to be used for the encrypted tablespace and then create them. The sample scripts use the default algorithm AES128:

1. Modify the paths of the script <INSTALL_DIR>/sim/dbschema/dbutils /create_encrypted_ tablespaces_TDE.sql.
2. Run the script using SQL*Plus as sys.
3. Review Create_encrypted_tablespaces_TDE.log for errors and correct as needed.

   Once the tablespaces have been created, the SIM schema installation can be run.

Note: After encryption at the tablespace level, it is absolutely crucial to backup the contents in the wallet directory; otherwise, if they are lost you will not be able to access the tablespaces.

H

Appendix: Database Parameter File

```
###########################################################################
# Copyright (c) 2014 by Oracle Corporation
# Oracle 19.x.x Parameter file
# NOTES: Before using this script:
#        1.  Change <datafile_path>, <admin_path>, <utl_file_path>, <diag_path>
and <hostname>
#            values as appropriate.
#        2.  Replace the word SID with the database name.
#        3.  Size parameters as necessary for development, test, and production
environments.
# ---------------------------------------------------------------------------
*.audit_file_dest=full_path_of_audit_dir
*.audit_trail='db'
*.compatible='19.0.0.0'
*.control_files='full_path_of_controlfile_1','full_path_of_controlfile_2'
##############################################
# Memory Settings:
# xxxM = Some reasonable starting value for your environment.
##############################################
*.db_block_size=xxxM
*.db_cache_size=xxxM
*.java_pool_size=xxxM
*.memory_target=xxxM
*.pga_aggregate_target=xxxM
*.shared_pool_size=xxxM
*.streams_pool_size=xxxM

##############################################

*.db_block_size=8192
*.db_domain=''
*.db_name='dbName'
*.diagnostic_dest='full_path_of_diag_dir'
*.enable_pluggable_database=true|false
*.fast_start_mttr_target=900
*.nls_calendar='GREGORIAN'
*.nls_date_format='DD-MON-RR'
*.nls_language='AMERICAN'
*.nls_numeric_characters='.,'
*.nls_sort=BINARY
*.open_cursors=900
*.os_authent_prefix=''
*.plsql_optimize_level=2
*.processes=2000
*.query_rewrite_enabled='true'
*.remote_dependencies_mode='SIGNATURE'
*.remote_login_passwordfile='EXCLUSIVE'
*.remote_os_authent=true
*.sec_case_sensitive_logon=false
*.undo_tablespace='UNDOTBS1'
```

I

Appendix: Installation Order

This section provides a guideline as to the order in which the Oracle Retail applications should be installed. If a retailer has chosen to use some, but not all, of the applications the order is still valid less the applications not being installed.

> **Note:** The installation order is not meant to imply integration between products.

Enterprise Installation Order

1. Oracle Retail Merchandising System (RMS), Oracle Retail Trade Management (RTM)
2. Oracle Retail Sales Audit (ReSA)
3. Oracle Retail Extract, Transform, Load (RETL)
4. Oracle Retail Warehouse Management System (RWMS)
5. Oracle Retail Invoice Matching (ReIM)
6. Oracle Retail Price Management (RPM)
7. Oracle Retail Allocation
8. Oracle Retail Mobile Merchandising (ORMM)
9. Oracle Retail Customer Engagement (ORCE)
10. Oracle Retail Xstore Office
11. Oracle Retail Xstore Point-of-Service, including Xstore Point-of-Service for Grocery, and including Xstore Mobile
12. Oracle Retail Xstore Environment
13. Oracle Retail EFTLink
14. Oracle Retail Store Inventory Management (SIM), including Mobile SIM
15. Oracle Retail Predictive Application Server (RPAS)
16. Oracle Retail Predictive Application Server Batch Script Architecture (RPAS BSA)
17. Oracle Retail Demand Forecasting (RDF)
18. Oracle Retail Category Management Planning and Optimization/Macro Space Optimization (CMPO/MSO)
19. Oracle Retail Replenishment Optimization (RO)
20. Oracle Retail Regular Price Optimization (RPO)
21. Oracle Retail Merchandise Financial Planning (MFP)
22. Oracle Retail Size Profile Optimization (SPO)
23. Oracle Retail Assortment Planning (AP)
24. Oracle Retail Item Planning (IP)
25. Oracle Retail Item Planning Configured for COE (IP COE)
26. Oracle Retail Advanced Inventory Planning (AIP)
27. Oracle Retail Integration Bus (RIB)
28. Oracle Retail Service Backbone (RSB)

29. Oracle Retail Financial Integration (ORFI)
30. Oracle Retail Bulk Data Integration (BDI)
31. Oracle Retail Integration Console (RIC)
32. Oracle Commerce Retail Extension Module (ORXM)
33. Oracle Retail Data Extractor for Merchandising
34. Oracle Retail Clearance Optimization Engine (COE)
35. Oracle Retail Analytic Parameter Calculator for Regular Price Optimization (APC-RPO)
36. Oracle Retail Insights, including Retail Merchandising Insights (previously Retail Merchandising Analytics) and Retail Customer Insights (previously Retail Customer Analytics)
37. Oracle Retail Order Broker