Oracle® Fusion Middleware

Enterprise Deployment Guide for Oracle Identity and Access Management

14c (14.1.2.0.0)
G24625-01
August 2025

ORACLE

Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity and Access Management, 14c (14.1.2.0.0)

G24625-01

Copyright © 2018, 2025, Oracle and/or its affiliates.

Primary Author: Michael Rhys, Russell Hodgson

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

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

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

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

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

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

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

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

Contents

Preface

Audience i
Documentation Accessibility i
Conventions i
Diversity and Inclusion ii

Part I Understanding an Enterprise Deployment

1 Enterprise Deployment Overview

About the Enterprise Deployment Guide 1
When to Use the Enterprise Deployment Guide 2

2 About a Typical Enterprise Deployment

Diagram of a Typical Enterprise Deployment 1
About the Typical Enterprise Deployment Topology Diagram 2
- Understanding the Firewalls and Zones of a Typical Enterprise Deployment 3
- About the Elements of a Typical Enterprise Deployment Topology 3
- Receiving Requests Through External Load Balancer 4
  - Purpose of the External Load Balancer (LBR) 4
  - Summary of the Typical Load Balancer Virtual Server Names 6
  - HTTPS Versus HTTP Requests to the External Virtual Server Name 6
- Understanding the Web Tier 6
  - Benefits of Using a Web Tier to Route Requests 7
  - Alternatives to Using a Web Tier 7
  - Configuration of Oracle HTTP Server in the Web Tier 8
  - About Mod_WL_OHS 8
- About the Application Tier 8
  - Configuration of the Administration Server and Managed Servers Domain Directories 8
  - Best Practices and Variations on the Configuration of the Clusters and Hosts on the Application Tier 9
  - About the Node Manager Configuration in a Typical Enterprise Deployment 10

Copyright © 2018, 2025, Oracle and/or its affiliates.

About Using Unicast for Communications within the Application Tier 11
About OPSS and Requests to the Authentication and Authorization Stores 12
About the Data Tier 12

3 About the IAM Enterprise Deployment

About the Primary and Build-Your-Own Enterprise Deployment Topologies 2
Diagram of Oracle Identity and Access Management on Distributed Hardware 2
About the Primary Oracle Identity and Access Management Topology Diagrams 5
Product Separation 5
Understanding the Directory Tier 6
About Oracle Unified Directory Assured Replication 6
Summary of Oracle Identity and Access Management Load Balancer Virtual Server Names 7
About the igdinternal Virtual Server Name 8
Summary of the Managed Servers and Clusters on the Application Tier Hosts 9
About the Forgotten Password Functionality 9
Integrating Oracle LDAP, Oracle Access Manager, and Oracle Identity Governance 10
Roadmap for Implementing the Primary IAM Suite Topologies 11
Building Your Own Oracle Identity and Access Management Topology 12
About Using SSL Certificates in the Oracle Identity Management Enterprise Topology 13
About Using JDBC Persistent Stores 13
About Using Service or Server Migration to Enable High Availability of the Enterprise Topology 14

Part II Preparing for an Enterprise Deployment

4 Procuring Resources for an Oracle Cloud Infrastructure Deployment

Procuring Resources for OCI 1
Load Balancer Requirements 1
Network 2
Compute Instances 2
Gateway 2
Database 2
Sizing 3

5 Procuring Resources for an Enterprise Deployment

Hardware and Software Requirements for the Enterprise Deployment Topology 1
External Load Balancer Requirements 2
Host Computer Hardware Requirements 3

Copyright © 2018, 2025, Oracle and/or its affiliates.

General Considerations for Enterprise Deployment Host Computers 3
Reviewing the Oracle Fusion Middleware System Requirements 3
Typical Memory, File Descriptors, and Processes Required for an Enterprise Deployment 4
Typical Disk Space Requirements for an Enterprise Deployment 5
Operating System Requirements for an Enterprise Deployment Topology 5
About Private Networks 5
About Virtual Server Templates 6
Reserving the Required IP Addresses for an Enterprise Deployment 6
What is a Virtual IP (VIP) Address? 7
Why Use Virtual Host Names and Virtual IP Addresses? 7
Physical and Virtual IP Addresses Required by the Enterprise Topology 7
Identifying and Obtaining Software Distributions for an Enterprise Deployment 8
Obtaining SSL Certificates 12
Understanding SSL 13
Types of Certificate 14
Keystores and Truststores 15
Using Certificates in WebLogic Servers 15
Creating a Certificate Authority (CA) 16
Creating Certificates 16
Creating a Host Based Certificate 16
Creating a SAN Based Certificate 17
Creating a Wildcard Based Certificate 19
Creating a PKCS12 Keystore 20
Creating a PKCS12 Truststore 21
Certificates Used in this Guide 22

6 Download and Configure the WebLogic Remote Console

7 Preparing the Load Balancer and Firewalls for an Enterprise Deployment

Configuring Virtual Hosts on the External Load Balancer 1
Overview of the External Load Balancer Configuration 1
Typical Procedure for Configuring the External Load Balancer 2
Load Balancer Health Monitoring 3
Summary of the Virtual Servers Required for an Enterprise Deployment 3
Configuring the Firewalls and Ports for an Enterprise Deployment 5

8 Preparing the File System for an Enterprise Deployment

Overview of Preparing the File System for an Enterprise Deployment 1

Copyright © 2018, 2025, Oracle and/or its affiliates.

Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment 2
About the Recommended Directory Structure for an Enterprise Deployment 3
File System and Directory Variables Used in This Guide 7
About Creating and Mounting the Directories for an Enterprise Deployment 11
Summary of the Shared Storage Volumes in an Enterprise Deployment 12

9 Preparing the Host Computers for an Enterprise Deployment

Verifying the Minimum Hardware Requirements for Each Host 2
Verifying Linux Operating System Requirements 2
Setting Linux Kernel Parameters 2
Setting the Open File Limit and Number of Processes Settings on UNIX Systems 3
Viewing the Number of Currently Open Files 3
Setting the Operating System Open File and Processes Limits 3
Verifying IP Addresses and Host Names in DNS or Hosts File 4
Enabling Unicode Support 5
Setting the DNS Settings 5
Configuring Users and Groups 5
Configuring a Host to Use an NTP (time) Server 6
Configuring a Host to Use an NIS/YP Host 7
Creating Local Directories 8
Creating Mount Points for Shared Storage 8
Mounting the Required Shared File Systems on Each Host 8
Validating the Shared Storage Configuration 9
Enabling the Required Virtual IP Addresses on Each Host 10

10 Preparing the Oracle Cloud Infrastructure for an Enterprise Deployment

About the OCI Deployment 2
Creating an SSH Key Pair 5
Creating an OCI Compartment 6
Creating OCI Networking 6
Creating an Oracle Virtual Cloud Network 6
Creating Gateways 7
Creating Security Lists 7
Required Security Lists 8
Creating the Required Security Lists 19
Creating Route Tables 19
Required Route Tables 20
Creating the Required Route Tables 20
Creating Subnets 20

Copyright © 2018, 2025, Oracle and/or its affiliates.

Required Subnets 21
Creating the Required Subnets 21
Creating Compute Instances 22
Required Compute Instances 22
Creating the Required Compute Instances 22
Assigning a Secondary IP Address to a Compute Instance 23
Connecting to a Compute Instance 24
Configuring the Compute Instances (General) 24
Installing X11 and Additional Packages 24
Enabling X11 Forwarding 25
Editing Compute Instance Fault Domain 25
Creating a Software Owner Account 25
Creating Local Directories 26
Creating Mount Points for Shared Storage 26
Preparing the Compute Instance 26
Using the Firewall 26
Configuring the Bastion Node 27
Installing Git 27
Setting Up the Hosts File 27
Configuring the OHS Nodes 28
Preparing the Compute Instance for Use by Oracle HTTP Server 28
Preparing the Hosts File 28
Creating File Systems and Mount Targets 28
Overview of Preparing the File System for an Enterprise Deployment 29
Summary of File Systems 29
Creating a File System 29
Setting the Mount Target Storage Reporting 30
Mounting File Systems on Hosts 30
Creating Load Balancers 31
Creating a Public Load Balancer 31
Creating a Self-Signed Certificate 32
Creating a Load Balancer 35
Uploading Load Balancer Certificates 37
Creating Host Names 37
Creating Backend Sets 38
Creating Listeners 39
Updating the Default Listener 40
Creating a Private Load Balancer 41
Creating a Load Balancer 41
Creating Host Names 42
Updating the Default Listener 43
Uploading Load Balancer Certificates 43

Copyright © 2018, 2025, Oracle and/or its affiliates.

Creating Backend Sets 44
Creating Listeners 45
Creating an LDAP Load Balancer 46
Creating a Load Balancer 46
Creating Host Names 48
Updating the Default Listener 48
Creating a Database 48
Creating the Database 49
Creating a Secondary Pluggable Database 50
Connecting to the Database Node 50
Configuring the Database 51
Creating a DNS Server 51
Creating a DNS Zone 51
Creating DNS Records 51
Validating Your Environment 53

11 Preparing the Database for an Enterprise Deployment

About Preparing the Database for an Enterprise Deployment 1
About Database Requirements 2
Supported Database Versions 2
Additional Database Software Requirements 2
Databases Required 3
Minimum Initialization Parameters 4
Applying the Database Patches 5
Creating Database Services 5
Creating a PDB Using an Existing PDB as a Template 7
Using SecureFiles for Large Objects (LOBs) in an Oracle Database 8
About Database Backup Strategies 9

Part III Configuring the Enterprise Deployment

12 Configuring Oracle LDAP for an Enterprise Deployment

Configuring Oracle Unified Directory 1
Variables Used When Configuring Oracle Unified Directory 1
Setting Environment Variables 4
Installing a Supported JDK 5
Locating and Downloading the JDK Software 5
Installing the JDK Software 5
Installing Oracle Unified Directory 6
Starting the Oracle Unified Directory Installer 6

Copyright © 2018, 2025, Oracle and/or its affiliates.

Navigating the Oracle Unified Directory Installation Screens 6
Installing the Stack Bundle Patch 8
Installing the Software on Other Host Computers 8
Verifying the Installation 8
Configuring the Oracle Unified Directory Instances 10
Configuring Oracle Unified Directory on LDAPHOST1 10
Configuring Oracle Unified Directory Instance on LDAPHOST2 14
Preparing an Existing LDAP Directory 17
About the Enterprise Deployment Users and Groups 18
About Using Unique Administration Users for Each Domain 18
Creating a Configuration File 18
Oracle Unified Directory Example 19
Explanation of Property Values 20
Preparing OUD as the Identity Store 23
Setting Environment Variables 23
Directory Pre-Configuration 23
Seeding Users and Groups 24
Granting OUD changelog Access 25
Creating Access Control Lists in Non-Oracle Directories 26

13 Configuring Oracle Unified Directory Service Manager

Variables Used When Configuring Oracle Unified Directory 1
Setting Environment Variables 2
Installing the Oracle Fusion Middleware Infrastructure 2
Installing a Supported JDK 2
Locating and Downloading the JDK Software 2
Installing the JDK Software 3
Starting the Oracle Identity Management Quick Installer 3
Navigating the Installation Screens 4
Installing the Stack Bundle Patch 5
Checking the Directory Structure 5
Installing Oracle Fusion Middleware Infrastructure on the Other Host Computers 6
Creating a Domain for Oracle Unified Directory Service Manager 6
Starting the Oracle Unified Directory Service Manager Domain 7
Configuring Oracle HTTP Server for Oracle Unified Directory Services Manager 8

14 Configuring Oracle Access Management

About the Initial Infrastructure Domain 3
About the Infrastructure Distribution 3
Characteristics of the Domain 3

Copyright © 2018, 2025, Oracle and/or its affiliates.

Variables Used When Creating Infrastructure for Oracle Access Management 4
Setting Environment Variables 10
URLs Used in This Chapter 11
Installing the Oracle Fusion Middleware 11
Installing a Supported JDK 12
Locating and Downloading the JDK Software 12
Installing the JDK Software 12
Installing the Oracle Fusion Middleware Infrastructure 13
Starting the Oracle Identity Management Quick Installer 13
Navigating the Installation Screens 13
Installing Oracle Fusion Middleware Infrastructure on the Other Host Computers 14
Checking the Directory Structure 15
Installing Oracle Access Management for an Enterprise Deployment 15
Starting the Oracle Identity and Access Management Installation Program 16
Navigating the Installation Screens 16
Installing the Stack Bundle Patch 17
Installing Oracle Access Management on the Other Host Computers 17
Verifying the Installation 17
Configuring LDAP 19
Creating the Database Schemas for Access Manager 19
Installing and Configuring a Certified Database 19
Creating the Database Schemas Using GUI 19
Starting the Repository Creation Utility (RCU) 19
Navigating the RCU Screens to Create the Schemas 20
Creating the Database Schemas Using CLI 24
Verifying Schema Access 24
Configuring the Oracle Access Management Domain 25
Starting the Configuration Wizard 25
Navigating the Configuration Wizard Screens to Configure Oracle Access Management Domain 25
Enabling SSL 38
Adding Certificate Stores Location to the WebLogic Servers Start Scripts 39
Update Server's Security Settings Using the Remote Console 39
Connecting to the Remote Console Using the Administration Server’s Virtual Hostname as Provider 39
Updating the WebLogic Servers Security Settings 41
Configuring a Per Host Node Manager for an Enterprise Deployment 42
Creating a Per Host Node Manager Configuration 42
Starting the Node Manager on OAMHOST1 45
Configuring the Node Manager Credentials 46
Enrolling the Domain with NM 46
Adding Truststore Configuration to Node Manager 48

Copyright © 2018, 2025, Oracle and/or its affiliates.

Configuring the Domain Directories and Starting the Servers 48
Create a ServerOverrides File 48
Starting the Administration Server Using the Node Manager 49
Validating the Administration Server 50
Creating a Separate Domain Directory for Managed Servers 50
Removing OAM Server from WebLogic Server defaultCoherenceCluster 52
Tuning the WebLogic Server 53
Adding a Load Balancer Certificate to the Oracle Keystore Service 54
Tuning the oamDS Data Source 56
Configuring the WebLogic Proxy Plug-In 57
Configuring and Integrating with LDAP 57
Configuring Access Manager to Use the LDAP Directory 57
Creating a Configuration File 57
Integrating Access Manager and LDAP Using the idmConfigTool 59
Validating the OAM LDAP Configuration 60
Updating WebGate Agents 61
Updating Host Identifiers 62
Updating Idle Timeout Value 62
Validating the Authentication Providers 63
Configuring Oracle ADF and OPSS Security with Oracle Access Manager 63
Propagating the Domain to OAMHOST2 66
Unpacking the Domain Configuration on OAMHOST2 66
Starting the Node Manager on OAMHOST2 67
Starting the Managed Servers in the Domain 67
Starting the oam_server1 Managed Server 67
Validating Access Manager 68
Enabling Forgotten Password 69
Prerequisites for Enabling Forgotten Password 69
Add Permissions to oamLDAP user 69
Create an OTP Administrative Group in LDAP 70
Enabling Adaptive Authentication Service 70
Configuring Adaptive Authentication Plug-in 71
Enabling Password Management in the Directory 71
Storing User Messaging Credentials in CSF 72
Setting Up the Forgot Password Link on the Login Page 73
Restarting the Domain 74
Validating the Forgotten Password Functionality 75
Replacing Connect Strings with the Appropriate TNS Alias 75
Backing Up the Configuration 75

Copyright © 2018, 2025, Oracle and/or its affiliates.

15 Configuring Oracle Identity Governance

Synchronizing the System Clocks 4
About the Initial Infrastructure Domain 4
About the Infrastructure Distribution 4
Characteristics of the Domain 4
Variables Used When Creating the Infrastructure Domain 5
Setting Environment Variables 8
URLs Used in This Chapter 9
Installing the Oracle Fusion Middleware 9
Installing the Oracle Fusion Middleware Infrastructure, SOA Suite, and Oracle Identity Governance on OIGHOST1 9
Installing a Supported JDK 10
Starting the Oracle Identity Management Quick Installer 11
Navigating the Installation Screens 11
Installing the Stack Bundle Patch 12
Installing Oracle Fusion Middleware Infrastructure on the Other Host Computers 13
Verifying the Installation 13
Reviewing the Installation Log Files 13
Checking the Directory Structure 13
Viewing the Contents of Your Oracle Home 14
Downloading the Oracle Connector Bundle 14
Installing the Oracle Identity Governance Connector 15
Creating the Database Schemas for Oracle Identity Governance 15
Installing and Configuring a Certified Database 16
Verifying Schema Access 16
Creating the Database Schemas Using GUI 16
Starting the Repository Creation Utility (RCU) 16
Navigating the RCU Screens to Create the Schemas 17
Creating the Database Schemas Using CLI 21
Configuring the Oracle Identity Governance Domain 21
Starting the Configuration Wizard 21
Navigating the Configuration Wizard Screens to Configure the Oracle Identity Governance Domain 21
Creating Oracle Identity Manager Authenticator 36
Enabling SSL 37
Adding Certificate Stores Location to the WebLogic Servers Start Scripts 37
Update Server's Security Settings Using the Remote Console 37
Connecting to the Remote Console Using the Administration Server’s Virtual Hostname as Provider 38
Updating the WebLogic Servers Security Settings 39
Configuring a Per Host Node Manager for an Enterprise Deployment 40
Creating a Per Host Node Manager Configuration 41

Copyright © 2018, 2025, Oracle and/or its affiliates.

Starting the Node Manager on OIGHOST1 43
Configuring the Node Manager Credentials 44
Enrolling the Domain with NM 45
Adding Truststore Configuration to Node Manager 46
Tuning JMS Queues 46
Configuring the Domain Directories and Starting the Servers 47
Create a ServerOverrides File 47
Starting the Administration Server Using the Node Manager 47
Validating the Administration Server 48
Starting and Validating the soa_server1 Managed Server on OIGHOST1 49
Creating the New Authentication Provider 49
Adding a Load Balancer Certificate to Oracle Keystore Service 50
Configuring the WebLogic Proxy Plug-In 52
Modifying the Upload and Stage Directories to an Absolute Path 53
Starting and Validating the Oracle Identity Governance Managed Servers 53
Starting the Oracle Identity Governance Managed Servers and Bootstrapping the Domain 53
Starting the soa_server1 and oim_server1 Managed Servers 54
Validating the Managed Server by Logging in to the Identity Console 54
Starting and Validating soa_server2 and oim_server2 Managed Servers 54
Analyzing the Bootstrap Report 55
Creating a Separate Domain Directory for Managed Servers 55
Stopping the Managed Servers 55
Packing the Domain 56
Unpacking the Domain 57
Validating the Fusion Middleware Control Application 58
Configuring the Web Tier for the Domain 59
Validating the Oracle SOA Suite URLs Through the Load Balancer 59
Integrating Oracle Identity Governance with Oracle SOA Suite 59
Managing the Notification Service 60
Using SMTP for Notification 61
Adding a CSF Key 61
Configuring the Messaging Drivers 62
Configuring the Email Driver 62
Forcing Oracle Identity Governance to use Correct Multicast Address 63
Integrating Oracle Identity Governance with LDAP 64
Variables Used in OIG Integration with LDAP 64
Installing the Connector Bundle 67
Configuring the Oracle Connector for LDAP 68
Restarting the Domains 69
Integrating Oracle Identity Governance and Oracle Access Manager 69
Variables Used in OIG and OAM Integration 69

Copyright © 2018, 2025, Oracle and/or its affiliates.

Configuring SSO Integration in the IAMGovernanceDomain 73
Restarting the Domains 74
Enable OAM Notifications 74
Propagating the Domain and Starting the Servers on OIGHOST2 75
Unpacking the Domain Configuration on OIGHOST2 75
Starting the Node Manager in the Managed Server Domain Directory OIGHOST2 76
Starting and Validating the soa_server2 and oim_server2 Managed Servers on OIGHOST2 77
Configuring OIG Workflow Notifications to be Sent by Email 77
Adding the wsm-pm Role to the Administrators Group 78
Adding the Oracle Access Manager Load Balancer Certificate to the Oracle Keystore Service 78
Restarting the IAMGovernanceDomain 80
Setting Challenge Questions 80
Replacing Connect Strings with the Appropriate TNS Alias 80
Integrating Oracle Identity Manager with Oracle Business Intelligence Publisher 80
Creating a User to Run BI Reports 81
Configuring Oracle Identity Manager to Use BI Publisher 82
Assigning the BIServiceAdministrator Role to idm_report 82
Storing the BI Credentials in Oracle Identity Governance 83
Creating OIM and BPEL Data Sources in BIP 83
Deploying Oracle Identity Governance Reports to BI 85
Enable Certification Reports 86
Validating the Reports 86
Generating Reports Against the Sample Data Source 86
Generating Reports Against the Oracle Identity Manager JDBC Data Source 87
Generating Reports Against the BPEL-Based JDBC Data Source 87
Verification of Manual Failover of the Administration Server 89
Backing Up the Configuration 89

16 Configuring Oracle HTTP Server for an Enterprise Deployment

About the Oracle HTTP Server Domains 2
Variables Used When Configuring the Oracle HTTP Server 2
Setting Environment Variables 3
Installing Oracle HTTP Server on WEBHOST1 3
Starting the Installer on WEBHOST1 4
Navigating the Oracle HTTP Server Installation Screens 4
Verifying the Oracle HTTP Server Installation 5
Installing a Supported JDK 7
Locating and Downloading the JDK Software 7
Installing the JDK Software 7
Creating an Oracle HTTP Server Domain on WEBHOST1 8

Copyright © 2018, 2025, Oracle and/or its affiliates.

Starting the Configuration Wizard on WEBHOST1 8
Navigating the Configuration Wizard Screens for an Oracle HTTP Server Domain 8
Installing and Configuring an Oracle HTTP Server Domain on WEBHOST2 11
Starting the Node Manager and Oracle HTTP Server Instances on WEBHOST1 and WEBHOST2 11
Starting the Node Manager on WEBHOST1 and WEBHOST2 12
Starting the Oracle HTTP Server Instances 12
Backing Up the Configuration 13
Generate Required Certificates for OHS SSL Listeners 13
Configuring Oracle HTTP Server to Route Requests to the Application Tier 14
About the Oracle HTTP Server Configuration for an Enterprise Deployment 14
Purpose of the Oracle HTTP Server Virtual Hosts 14
About the WebLogicCluster Parameter of the <VirtualHost> Directive 15
Recommended Structure of the Oracle HTTP Server Configuration Files 15
Modifying the httpd.conf File to Include Virtual Host Configuration Files 15
Modifying the httpd.conf File to Set Server Runtime Parameters 18
Creating the Virtual Host Configuration Files 19
SSL Terminated Deployments 19
End to End SSL Deployments 20
Routing Requests to WebLogic Servers 22
Copying Configuration Files to WEBHOST2 28
Restarting the OHS Instances on WEBHOST1 and WEBHOST2 29
Validating Access Through the Load Balancer 29
Verifying the URLs 29
Validating the Virtual Server Configuration and Access to the Consoles 30
Sample Virtual Host Files 30
SSL Terminated Deployments 30
End to End SSL Deployments 39

Part IV Common Configuration and Management Procedures for an Enterprise Deployment

17 Common Configuration and Management Tasks for an Enterprise Deployment

Configuration and Management Tasks for All Enterprise Deployments 1
Verifying Appropriate Sizing and Configuration for the WLSSchemaDataSource 2
Verifying Manual Failover of the Administration Server 2
Failing Over the Administration Server When Using a Per Host Node Manager 3
Validating Access to the Administration Server on OIGHOST2 Through Oracle HTTP Server 5

Copyright © 2018, 2025, Oracle and/or its affiliates.

Failing the Administration Server Back to OIGHOST1 When Using a Per Host Node Manager 5
Modifying the Upload and Stage Directories to an Absolute Path in an Enterprise Deployment 6
Setting the Front End Host and Port for a WebLogic Cluster 7
Using Persistent Stores for TLOGs and JMS in an Enterprise Deployment 8
About JDBC Persistent Stores for Web Services 8
Best Configuration Practices When Using RAC and Gridlink Datasources 9
Using TNS Alias in Connect Strings 10
Performing Backups and Recoveries for an Enterprise Deployment 13

18 Scaling Procedures for an Enterprise Deployment

Scaling Out the Topology 1
Prerequisites for Scaling Out 1
Scaling Out a Cluster 1
Verifying the Scale Out 11
Scaling in the Topology 12

19 Configuring Single Sign-On for an Enterprise Deployment

About Oracle Webgate 1
General Prerequisites for Configuring Oracle HTTP Server WebGate 2
Configuring Oracle HTTP Server WebGate for an Enterprise Deployment 2
Adding a Load Balancer Certificate to WebGate 4
Copying the LoadBalancer Certificates to WebGate Config 4
Ensuring that the REST Endpoints are Set Correctly 5
Copying WebGates Artifacts to Web Tier 5
Copying Generated Artifacts to the Oracle HTTP Server WebGate Instance Location 5
Restarting the Oracle HTTP Server Instance 6
Setting Up the WebLogic Server Authentication Providers 7
Backing Up the Configuration Files 7
Configuring a New Authenticator for Oracle WebLogic Server 7
Setting Up the Oracle Access Manager Identity Assertion Provider 10
Updating the Default Authenticator and Setting the Order of Providers 11
Setting the Default Authenticator 12
Ordering the Authentication Providers 12
Adding the Administration Role to the New Administration Group 12
Configuring Oracle ADF and OPSS Security with Oracle Access Manager 13

Copyright © 2018, 2025, Oracle and/or its affiliates.

20 Troubleshooting

Troubleshooting Oracle Access Management Access Manager 1
- Access Manager Runs out of Memory 1
- User Reaches the Maximum Allowed Number of Sessions 2
- Policies Do Not Get Created When Oracle Access Management Access Manager is First Installed 3
- You Are Not Prompted for Credentials After Accessing a Protected Resource 3
- Cannot Log In to Access Management Console 4
- Oracle Coherence Cluster Startup Errors in oam_policy_mgr Server Logs 4
- Errors in log File when Starting OAM Servers 5
- Too Many Redirects Error in Browser 7

Troubleshooting Oracle Identity Governance 7
- OIM Bootstrap Process Fails 8
- java.io.FileNotFoundException When Running Oracle Identity Governance Configuration 8
- ResourceConnectionValidationxception When Creating User in Oracle Identity Governance 9
- Oracle Identity Manager Reconciliation Jobs Fail 9
- OIM Reconciliation Jobs Fail When Running Against Oracle Unified Directory 11
- Cannot Open Reports from OIM Self Service Console 11
- Pending Violations Not Displaying the Correct List 12

Troubleshooting Oracle SOA Suite 12
- Transaction Timeout Error 12

Troubleshooting Integration OIGOAMIntegration.sh-configureLDAPConnector 12

General Troubleshooting 13
- Cannot Start Managed Server from WebLogic Console 13

Copyright © 2018, 2025, Oracle and/or its affiliates.

Preface

This guide explains how to install, configure, and manage a highly available Oracle Fusion Middleware enterprise deployment.

- Audience
- Documentation Accessibility
- Conventions
- Diversity and Inclusion

Audience

In general, this document is intended for administrators of Oracle Fusion Middleware, who are assigned the task of installing and configuring Oracle Fusion Middleware software for production deployments.

Specific tasks can also be assigned to specialized administrators, such as database administrators (DBAs) and network administrators, where applicable.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at `https://www.oracle.com/corporate/accessibility/`.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit `https://support.oracle.com/portal/` or visit `Oracle Accessibility Learning and Support` if you are hearing impaired.

Conventions

The following text conventions are used in this document:

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

Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> This guide focuses on the implementation of the enterprise deployment reference topology on Oracle Linux systems.
>
> The topology can be implemented on any certified, supported operating system, but the examples in this guide typically show the commands and configuration steps as they should be performed using the bash shell on Oracle Linux.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Part I

Understanding an Enterprise Deployment

It is important to understand the concept and general characteristics of a typical enterprise deployment, before you configure the Oracle Identity and Access Management enterprise deployment topology.

This part of the Enterprise Deployment Guide contains the following topics.

- Enterprise Deployment Overview
  The Enterprise Deployment Guide provides detailed, validated instructions that help you plan, prepare, install, and configure a multi-host, secure, highly available, production topology for selected Oracle Fusion Middleware products.
- About a Typical Enterprise Deployment
  It is essential to understand the components of a typical enterprise deployment topology.
- About the IAM Enterprise Deployment

Copyright © 2018, 2025, Oracle and/or its affiliates.

1

Enterprise Deployment Overview

The Enterprise Deployment Guide provides detailed, validated instructions that help you plan, prepare, install, and configure a multi-host, secure, highly available, production topology for selected Oracle Fusion Middleware products.

This chapter introduces the concept of an Oracle Fusion Middleware enterprise deployment. It also provides information on when to use the Enterprise Deployment guide.

- About the Enterprise Deployment Guide
  The Enterprise Deployment Guide provides a comprehensive, scalable example for installing, configuring, and maintaining a secure, highly available, production-quality deployment of selected Oracle Fusion Middleware products. The resulting environment is known as an enterprise deployment topology. Enterprise Deployment Guides are the foundation to Maximum Availability Architectures for Oracle FMW products. Oracle's Maximum Availability Architecture provides a set of architectures, configurations and operational best practices that provide High Availability and Disaster Recovery solutions for the entire Oracle Stack. An Enterprise Deployment is the incarnation of these best practices in the scope of a single data center. When combined with the appropriate configuration and operational models for Disaster Protection, the Enterprise Deployment will achieve optimal high availability, data protection and disaster recovery at the lowest cost and complexity, while providing the best Recovery Time Objective (RTO) and Recovery Point Objective (RPO).
- When to Use the Enterprise Deployment Guide
  This guide describes the installation and configuration options for Oracle Fusion Middleware. Use this guide to help you plan, prepare, install, and configure a multi-host, secure, highly available, production topology for selected Oracle Fusion Middleware products.

About the Enterprise Deployment Guide

The Enterprise Deployment Guide provides a comprehensive, scalable example for installing, configuring, and maintaining a secure, highly available, production-quality deployment of selected Oracle Fusion Middleware products. The resulting environment is known as an enterprise deployment topology. Enterprise Deployment Guides are the foundation to Maximum Availability Architectures for Oracle FMW products. Oracle's Maximum Availability Architecture provides a set of architectures, configurations and operational best practices that provide High Availability and Disaster Recovery solutions for the entire Oracle Stack. An Enterprise Deployment is the incarnation of these best practices in the scope of a single data center. When combined with the appropriate configuration and operational models for Disaster Protection, the Enterprise Deployment will achieve optimal high availability, data protection and disaster recovery at the lowest cost and complexity, while providing the best Recovery Time Objective (RTO) and Recovery Point Objective (RPO).

For example, the enterprise deployment topology introduces key concepts and best practices that you can use to implement a similar Oracle Fusion Middleware environment for your organization.

Each Enterprise Deployment Guide provides detailed, validated instructions for implementing the reference topology. Along the way, the guide also offers links to supporting documentation

Copyright © 2018, 2025, Oracle and/or its affiliates.

that explains concepts, reference material, and additional options for an Oracle Fusion Middleware enterprise deployment.

Note that the enterprise deployment topologies described in the enterprise deployment guides cannot meet the exact requirements of all Oracle customers. In some cases, you can consider alternatives to specific procedures in this guide, depending on whether the variations to the topology are documented and supported by Oracle.

Oracle recommends customers use the Enterprise Deployment Guides as a first option for deployment. If variations are required, then those variations should be verified by reviewing the related Oracle documentation or by working with Oracle Support.

When to Use the Enterprise Deployment Guide

This guide describes the installation and configuration options for Oracle Fusion Middleware. Use this guide to help you plan, prepare, install, and configure a multi-host, secure, highly available, production topology for selected Oracle Fusion Middleware products.

Alternatively, you can use the other primary installation and configuration options.

Use the instructions in one of the product-specific installation guides to install and configure a **standard installation topology** for a selected set of Oracle Fusion Middleware products.

A standard installation topology can be installed on a single host for evaluation purposes, but it can also serve as a starting point for scaling out to a more complex production environment.

For Oracle Identity and Access Management, see Installing and Configuring Oracle Identity and Access Management

Review *Planning an Installation of Oracle Fusion Middleware*, which provides additional information to help you prepare for any Oracle Fusion Middleware installation.

Copyright © 2018, 2025, Oracle and/or its affiliates.

2

About a Typical Enterprise Deployment

It is essential to understand the components of a typical enterprise deployment topology.

This chapter provides information on the Enterprise Deployment Topology diagram.

- Diagram of a Typical Enterprise Deployment
  This diagram shows all the components of a typical enterprise deployment, including the Web tier, Application tier, and Data tier. All enterprise deployments are based on these basic principles.
- About the Typical Enterprise Deployment Topology Diagram
  A typical enterprise deployment topology consists of an External Load Balancer (LBR), web tier, an application tier, and data tier. This section provides detailed information on these components.

Diagram of a Typical Enterprise Deployment

This diagram shows all the components of a typical enterprise deployment, including the Web tier, Application tier, and Data tier. All enterprise deployments are based on these basic principles.

All Oracle Fusion Middleware enterprise deployments are designed to demonstrate the best practices for installing and configuring an Oracle Fusion Middleware production environment.

A best practices approach starts with the basic concept of a multi-tiered deployment and standard communications between the different software tiers.

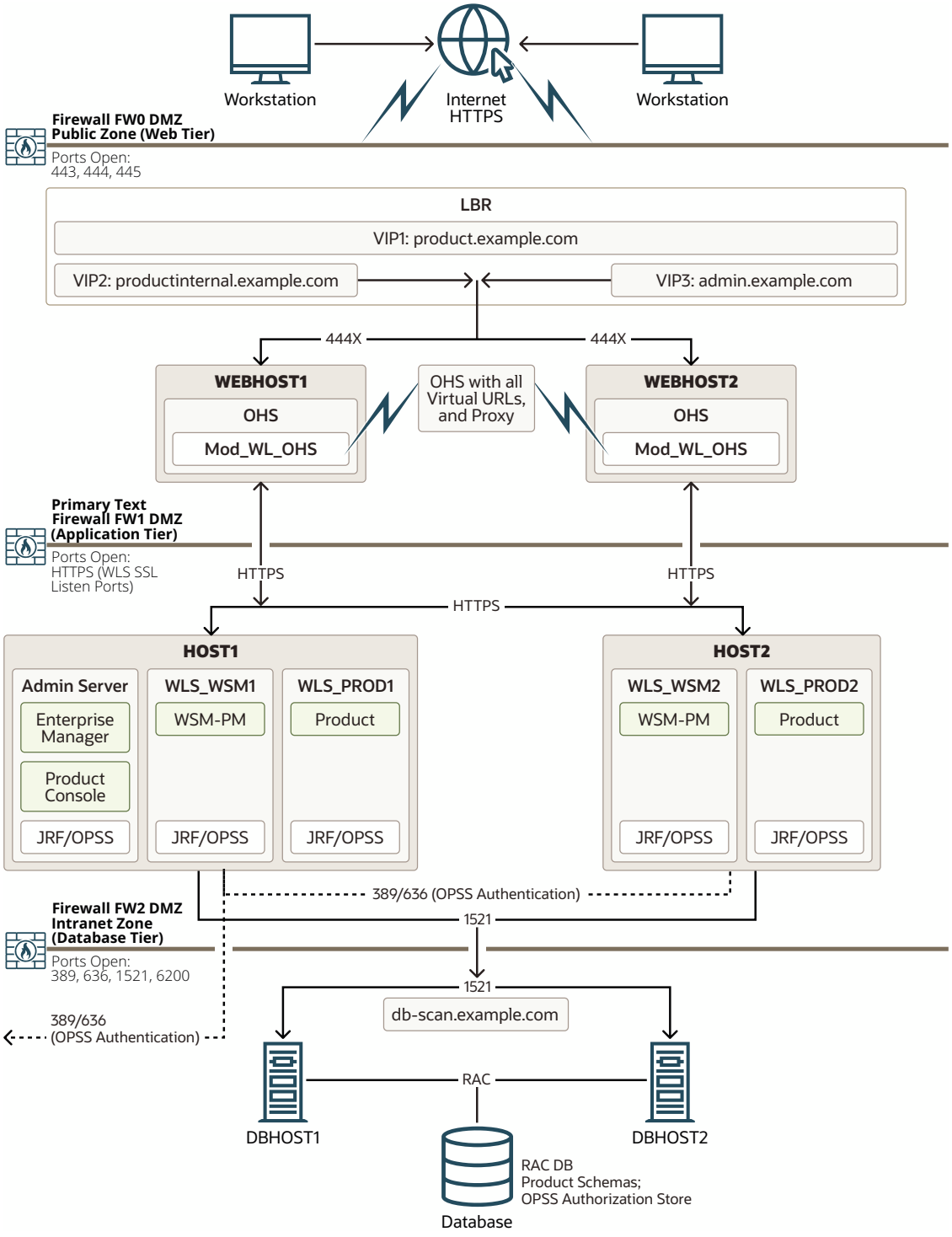
This Enterprise Deployment uses secured communications using SSL all the way from the external clients to the backend WebLogic Servers. Although this eliminates numerous security vulnerabilities and increases the resiliency against different types of attacks, it has implications on the overall performance of the system. These implications vary depending on the applications deployed and the workloads in each invocation. For more information, see Configuring SSL in *Administering Security for Oracle WebLogic Server*. An alternative approach, is to terminate SSL at the load balancer. This approach also ensures the communication between the client and the load balancer is secured, whilst maximizing performance within the system components by avoiding SSL overhead in the rest of the tiers. The downside is that this may not offer sufficient security in cloud based applications.

Typical Enterprise Deployment Topology Diagram shows a typical enterprise deployment, including the Web tier, Application tier, and Data tier. All enterprise deployments are based on these basic principles.

For a description of each tier and the standard protocols used for communications within a typical Oracle Fusion Middleware enterprise deployment, see About the typical Enterprise Deployment Topology Diagram.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Figure 2-1 Typical Enterprise Deployment Topology Diagram



About the Typical Enterprise Deployment Topology Diagram

A typical enterprise deployment topology consists of an External Load Balancer (LBR), web tier, an application tier, and data tier. This section provides detailed information on these components.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Understanding the Firewalls and Zones of a Typical Enterprise Deployment
- About the Elements of a Typical Enterprise Deployment Topology
- Receiving Requests Through External Load Balancer
- Understanding the Web Tier
- About the Application Tier
- About the Data Tier

Understanding the Firewalls and Zones of a Typical Enterprise Deployment

The topology is divided into several security zones, which are separated by firewalls:

- The web tier (or DMZ), which is used for the external load balancer and Web servers (in this case, Oracle HTTP Server instances) that receive the initial requests from users. This zone is accessible only through the virtual server names that are defined on the load balancer.
- The application tier, which is where the business and application logic resides.
- The data tier, which is not accessible from the Internet and reserved in this topology for the highly available database instances.

The firewalls are configured to allow data to be transferred only through specific communication ports. Those ports (or in some cases, the protocols that need open ports in the firewall) are shown on each firewall line in the diagram.

For example:

- On the firewall protecting the web tier, only the HTTP(s) ports are open: 443 for HTTPS.
- On the firewall protecting an application tier, HTTP(s) ports, and MBean proxy port are open.
- Applications that require external HTTP(s) access can use the Oracle HTTP Server instances as a proxy. Note that this port is used for outbound communications only and the proxy capabilities on the Oracle HTTP Server must be enabled.
- On the firewall protecting the data tier, the database listener port (typically, 1521) must be open.
- The LDAP ports (typically, 389 and 636) are also required to be open for communication between the authorization provider and the LDAP-based identity store.
- The ONS port (typically, 6200) is also required so that the application tier can receive notifications about workload and events in the Oracle RAC Database. These events are used by the Oracle WebLogic Server connection pools to adjust quickly (creating or destroying connections), depending on the availability and workload on the Oracle RAC database instances.

For a complete list of the ports that you must open for a specific Oracle Fusion Middleware enterprise deployment topology, see the chapter that describes the topology that you want to implement, or refer to the *Enterprise Deployment Workbook* for the topology that you want to implement. See Using the Enterprise Deployment Workbook.

About the Elements of a Typical Enterprise Deployment Topology

The enterprise deployment topology consists of the following high-level elements:

Copyright © 2018, 2025, Oracle and/or its affiliates.

- An external load balancer that routes requests from the Internet to the web servers in the web tier. It also routes requests from internal clients or other components that perform internal invocations within the corporate network.
- A web tier, consisting of an external load balancer and two or more physical computers that host the web server instances (for high availability).

  The web server instances are configured to authenticate users (through an external identity store and a single sign-on server) and then route the HTTP requests to the Oracle Fusion Middleware products and components that are running in the Application tier.

  The web server instances also host static web content that does not require the application logic to be delivered. Placing such content in the web tier reduces the overhead on the application servers and eliminates unnecessary network activity.
- An application tier, consisting of two or more physical computers that are hosting a cluster of Oracle WebLogic Managed Servers, and the Administration Server for the domain. The Managed Servers are configured to run the various Oracle Fusion Middleware products, such as Oracle Access Management and Oracle Identity Governance, depending on your choice of products in the enterprise deployment.
- A data tier, consisting of two or more physical hosts that are hosting an Oracle RAC Database.

Receiving Requests Through External Load Balancer

The following topics describe the external load balancer and its role in an enterprise deployment.

- Purpose of the External Load Balancer (LBR)
- Summary of the Typical Load Balancer Virtual Server Names
- HTTPS Versus HTTP Requests to the External Virtual Server Name

Purpose of the External Load Balancer (LBR)

There are two types of load balancers, Local Load Balancers and Global Load Balancers. Load balancers can either be hardware devices such as Big IP, Cisco, Brocade, and so on—or they can be software applications. Most load balancer appliances can be configured for both local and global load balancers.

Load balancers should always be deployed in pairs to ensure that no single load balancer is a single point of failure. Most load balancers do this in an active-passive way. You should consult your load balancer documentation on how best to achieve this.

> **Note**
>
> Oracle does not certify against specific load balancers. The configuration information of load balancers given in the Enterprise Deployment guide are for guidance only and you should consult with your load balancer vendor about the best practices that are associated with the configuration of the device that you are using.

A local load balancer is used to distribute traffic within a site. It can distribute both HTTP and TCP traffic and the requirements of your deployment dictates which options you should use. Local load balancers often provide acceleration for SSL encryption and decryption as well as the ability to terminate or off-load SSL requests. This SSL termination at the load balancer provides a performance gain to applications at the cost of communications being secured in

Copyright © 2018, 2025, Oracle and/or its affiliates.

the rest of the tiers only by subnet restrictions in corporate networks. Given the increased security requirements applying nowadays to production deployments, this version of the Enterprise Deployment guide is implementing end-to-end SSL communication all the way from the external clients to the middle tier. HTTPS protocol is used for the communication between clients and the front-end load balancer, between the frontend load balancer and Oracle HTTP Servers and between Oracle HTTP Server and the WebLogic Servers . This comes at the cost of performance overhead that is especially relevant in SSL handshakes. Depending on the type of connections used by client requests (long lived, short lived, keep alive settings) this may be a factor in the performance of an Enterprise Deployment. It is considered however that security standards have increased nowadays and this guide will focus on the most secure deployment approach.

> **Note**
>
> For the purpose of providing the most realistic working environment possible this guide uses self-issued certificates in the web and application tiers. Notice, however, that self-issued digital certificates are not recommended in production mode. The sample steps provided in this book should be executed or substituted with appropriate certificates signed by a formal Certificate Authority in your production environment.

Enterprise Deployment guide environments always utilize a local load balancer.

The following topics describe the types of requests that are handled by the external load balancer in an Enterprise Deployment:

- HTTP Requests From the Internet to the Web Server Instances in the Web Tier
- Specific Internal-Only Communications Between the Components of the Application Tier

HTTP Requests From the Internet to the Web Server Instances in the Web Tier

The external load balancer balances the load on the web tier by receiving requests to a single virtual host name and then routing each request to one of the web server instances, based on a load balancing algorithm. In this way, the load balancer ensures that no one web server is overloaded with requests.

For more information about the purpose of specific virtual host names on the external load balancer, see Summary of the Typical Load Balancer Virtual Server Names.

HTTPS/encrypted requests are routed from the load balancer to the web tier. This guide provides instructions for SSL configuration between the load balancer and the web tier, and between the web tier and the application tier.

The load balancer provides high availability by ensuring that if one web server goes down, requests are routed to the remaining web servers that are up and running.

Further, in a typical highly available configuration, the external load balancers are configured such that a hot standby device is ready to resume service in case a failure occurs in the main load balancing appliance. This is important because for many types of services and systems, the external load balancer becomes the unique point of access to make invocations and, as a result, becomes a single point of failure (SPOF) for the whole system if it is not protected.

Specific Internal-Only Communications Between the Components of the Application Tier

In addition, the external load balancer routes specific communications between the Oracle Fusion Middleware components and applications on the application tier. The internal-only requests are also routed through the load balancer by using a unique virtual host name.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Summary of the Typical Load Balancer Virtual Server Names

In order to balance the load on servers and to provide high availability, the hardware load balancer is configured to recognize a set of virtual server names. By using the naming convention in Typical Enterprise Deployment Topology Diagram, the following virtual server names are recognized by the hardware load balancer in this topology:

- `product.example.com`: This virtual server name is used for all incoming traffic.

  Users enter this URL to access the Oracle Fusion Middleware product that you have deployed and the custom applications that are available on this server. The load balancer then routes these requests (by using a load balancing algorithm) to one of the servers in the web tier. In this way, the single virtual server name can be used to route traffic to multiple servers for load balancing and high availability of the web servers instances.

- `productinternal.example.com`: This virtual server name is for internal communications only.

  The load balancer uses its **Network Address Translation (NAT)** capabilities to route any internal communication from the application tier components that are directed to this URL. This URL is not exposed to external customers or users on the Internet. Each product has specific uses for the internal URL, so in the deployment instructions, the virtual server name is prefixed with the product name.

- `admin.example.com`: This virtual server name is for administrators who need to access the Oracle Enterprise Manager Fusion Middleware Control and Oracle WebLogic Remote Console interfaces.

  This URL is known only to internal administrators. It also uses the NAT capabilities of the load balancer to route administrators to the active Administration Server in the domain.

For a complete set of virtual server names that you must define for your topology, see the chapter that describes the product-specific topology.

HTTPS Versus HTTP Requests to the External Virtual Server Name

This Enterprise Deployment Guide uses SSL for all the virtual servers, hence the frontend port 80 is no longer used. It is a best practice to assign the main external URL (for example, `https://myapplication.example.com`) to the SSL port number 443.

> **Note**
>
> If port 80 remains open in the load balancer, then it is recommended to redirect any requests to it (non-SSL protocol) to port 443 (SSL protocol). Refer to your load balancer's specific documentation to implement this redirection. See Configuring Virtual Hosts on the External Load Balancer.

Understanding the Web Tier

The web tier of the reference topology consists of web servers that receive requests from the load balancer. In a typical enterprise deployment, at least two Oracle HTTP Server instances are configured in the web tier. The following topics provide more detail.

- Benefits of Using a Web Tier to Route Requests
- Alternatives to Using a Web Tier

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Configuration of Oracle HTTP Server in the Web Tier
- About Mod_WL_OHS

Benefits of Using a Web Tier to Route Requests

A web tier with Oracle HTTP Server is not a requirement for many of the Oracle Fusion Middleware products. You can route traffic directly from the hardware load balancer to the WLS servers in the Application Tier. However, a web tier provides several advantages, which is why it is recommended as part of the reference topology.

- The web tier provides faster fail-over in the event of a WebLogic Server instance failure. The plug-in actively learns about the failed WebLogic Server instance by using the information supplied by its peers. It avoids the failed server until the peers notify the plug-in that it is available. Load balancers are typically more limited and their monitors cause higher overhead.
- The web tier provides DMZ public zone, which is a common requirement in security audits. If a load balancer routes directly to the WebLogic Server, requests move from the load balancer to the application tier in one single HTTP jump, which can cause security concerns.
- The web tier allows the WebLogic Server cluster membership to be reconfigured (new servers added, others removed) without having to change the web server configuration (as long as at least some of the servers in the configured list remain alive).
- Oracle HTTP Server delivers static content more efficiently and faster than WebLogic Server; it also provides the ability to create virtual hosts and proxies via the Oracle HTTP Server configuration files.
- The web tier provides HTTP redirection over and above what the WebLogic Server provides. You can use Oracle HTTP Server as a front end against many different WebLogic Server clusters, and in some cases, control the routing by using content-based routing.
- Oracle HTTP Server provides the ability to integrate single sign-on capabilities into your enterprise deployment. For example, you can later implement single sign-on for the enterprise deployment by using Oracle Access Manager, which is part of the Oracle Identity and Access Management family of products.
- A web tier with Oracle HTTP Server provides support for WebSocket connections deployed within the WebLogic Server.

For more information about Oracle HTTP Server, see Introduction to Oracle HTTP Server in *Administering Oracle HTTP Server*.

Alternatives to Using a Web Tier

Although a Web tier provides a variety of benefits in an enterprise topology, Oracle also supports routing requests directly from the hardware load balancer to the Managed Servers in the middle tier.

This approach provide the following advantages:

- Lower configuration and processing overhead than using a front-end Oracle HTTP Server Web tier front-end.
- Monitoring at the application level since the LBR can be configured to monitor specific URLs for each Managed Server (something that is not possible with Oracle HTTP Server).

Copyright © 2018, 2025, Oracle and/or its affiliates.

You can potentially use this load balancer feature to monitor SOA composite application URLs. Note that this enables routing to the Managed Servers only when all composites are deployed, and you must use the appropriate monitoring software.

Configuration of Oracle HTTP Server in the Web Tier

This enterprise deployment guide provides information about configuring the Oracle HTTP Server instances as separate standalone domains, one on each web tier host. In each OHS Host the corresponding Node Manager listens only on localhost and manages only its local HTTP Server. See Configuring Oracle HTTP Server for an Enterprise Deployment. Oracle recommends this approach instead of configuring Oracle HTTP Server instances as part of the application tier domain.

See About Oracle HTTP Server in *Installing and Configuring Oracle HTTP Server*.

About Mod_WL_OHS

As shown in the diagram, the Oracle HTTP Server instances use the WebLogic Proxy Plug-In (`mod_wl_ohs`) for proxying HTTPs requests from Oracle HTTP Server to the Oracle WebLogic Server Managed Servers in the Application tier.

See What are Oracle WebLogic Server Proxy Plug-Ins? in *Using Oracle WebLogic Server Proxy Plug-Ins*.

About the Application Tier

The application tier consists of two physical host computers, where Oracle WebLogic Server and the Oracle Fusion Middleware products are installed and configured. The application tier computers reside in the secured zone between firewall 1 and firewall 2.

The following topics provide more information:

- Configuration of the Administration Server and Managed Servers Domain Directories
- Best Practices and Variations on the Configuration of the Clusters and Hosts on the Application Tier
- About the Node Manager Configuration in a Typical Enterprise Deployment
- About Using Unicast for Communications within the Application Tier
- About OPSS and Requests to the Authentication and Authorization Stores

Configuration of the Administration Server and Managed Servers Domain Directories

Unlike the Managed Servers in the domain, the Administration Server uses an active-passive high availability configuration. This is because only one Administration Server can be running within an Oracle WebLogic Server domain.

In the topology diagrams, the Administration Server on HOST1 is in the active state and the Administration Server on HOST2 is in the passive (inactive) state.

To support the manual fail over of the Administration Server in the event of a system failure, the typical enterprise deployment topology includes:

- A Virtual IP Address (VIP) for the routing of Administration Server requests.
- The configuration of the Administration Server domain directory on a shared storage device.

Copyright © 2018, 2025, Oracle and/or its affiliates.

In the event of a system failure (for example a failure of HOST1), you can manually reassign the Administration Server VIP address to another host in the domain, mount the Administration Server domain directory on the new host, and then start the Administration Server on the new host.

However, unlike the Administration Server, there is no benefit to storing the Managed Servers on shared storage. In fact, there is a potential performance impact when Managed Server configuration data is not stored on the local disk of the host computer.

As a result, in the typical enterprise deployment, after you configure the Administration Server domain on shared storage, a copy of the domain configuration is placed on the local storage device of each host computer, and the Managed Servers are started from this copy of the domain configuration. You create this copy by using the Oracle WebLogic Server pack and unpack utilities.

The resulting configuration consists of separate domain directories on each host: one for the Administration Server (on shared storage) and one for the Managed Servers (on local storage). Depending upon the action required, you must perform configuration tasks from one domain directory or the other.

For more information about structure of the Administration Server domain directory and the Managed Server domain directory, as well as the variables used to reference these directories, see Understanding the Recommended Directory Structure for an Enterprise Deployment.

There is an additional benefit to the multiple domain directory model. It allows you to isolate the Administration Server from the Managed Servers. By default, the primary enterprise deployment topology assumes the Administration Server domain directory is on one of the application tier hosts, but if necessary, you could isolate the Administration Server further by running it from its own host, for example in cases where the Administration Server is consuming high CPU or RAM. Some administrators prefer to configure the Administration Server on a separate, dedicated host, and the multiple domain directory model makes that possible.

Best Practices and Variations on the Configuration of the Clusters and Hosts on the Application Tier

In a typical enterprise deployment, you configure the Managed Servers in a cluster on two or more hosts in the application tier. For specific Oracle Fusion Middleware products, the enterprise deployment reference topologies demonstrate best practices for the number of Managed Servers, the number of clusters, and the services that are targeted for each cluster.

These best practices consider typical performance, maintenance, and scale-out requirements for each product. The result is the grouping of Managed Servers into an appropriate set of clusters within the domain.

Variations of the enterprise deployment topology allow the targeting of specific products or components to additional clusters or hosts for improved performance and isolation.

For example, you can consider hosting the Administration Server on a separate and smaller host computer, which allows the FMW components and products to be isolated from the Administration Server.

These variations in the topology are supported, but the enterprise deployment reference topology uses the minimum hardware resources while keeping high availability, scalability, and security in mind. Perform the appropriate resource planning and sizing, based on the system requirements for each type of server and the load that the system must sustain. Based on these decisions, you must adapt the steps to install and configure these variations accordingly from the instructions presented in this guide.

Copyright © 2018, 2025, Oracle and/or its affiliates.

The Enterprise deployment guide is focused on a static cluster-based topology. Static clusters, also known as "configured clusters" are conventional clusters where you manually configure and add each server instance. Dynamic clusters (consisting of server instances that can be dynamically scaled up to meet the resource needs of your application) are out of the scope of this guide.

About the Node Manager Configuration in a Typical Enterprise Deployment

Oracle WebLogic can use either a per domain Node Manager or a per host Node Manager. The following sections of this topic provide more information on the impact of the Node Manager configuration on a typical enterprise deployment.

> **Note**
>
> For general information about these two types of Node Managers, see Overview in *Administering Node Manager for Oracle WebLogic Server*.

About Using a Per Domain Node Manager Configuration

In a per domain Node Manager configuration—as opposed to a per host Node Manager configuration—you actually start two Node Manager instances on the Administration Server host: one from the Administration Server domain directory and one from the Managed Servers domain directory. In addition, a separate Node Manager instance runs on each of the other hosts in the topology.

The Node Manager that controls the Administration Server uses the listen address of the virtual host name created for the Administration Server. The Node Manager that controls the Managed Servers uses the listen address of the physical host. When the Administration Server fails over to another host, an additional instance of Node Manager is started to control the Administration Server on the failover host.

The key advantages of the per domain configuration are an easier and simpler initial setup of the Node Manager and the ability to set Node Manager properties that are unique to the Administration Server. This last feature was important in previous releases because some features, such as Crash Recovery, applied only to the Administration Server and not to the Managed servers. In the current release, Oracle FMW products such as Oracle SOA Suite can be configured for Automated Service Migration, rather than Whole Server Migration. This means the Managed Servers, as well as the Administration Server, can take advantage of Crash Recovery, so there is no need to apply different properties to the Administration Server and Managed Server domain directories.

Another advantage is that the per domain Node Manager provides a default SSL configuration for Node Manager-to-Server communication, based on the Demo Identity store created for each domain.

About Using a Per Host Node Manager Configuration

In a per host Node Manager configuration, you start a single Node Manager instance to control the Administration Server and all Managed Servers on a host, even those that reside in different domains. This reduces the footprint and resource utilization on the Administration Server host, especially in those cases where multiple domains coexist on the same computer.

A per host Node Manager configuration allows all Node Managers to use a listen address of ANY, so they listen on all addresses available on the host. This means that when the Administration Server fails over to a new host, no additional configuration is necessary.

Copyright © 2018, 2025, Oracle and/or its affiliates.

The per host Node Manager configuration requires additional configuration steps. If you want SSL for Node Manager-to-Server communication, then you must configure an additional Identity and Trust store, and it also requires using Subject Alternate Names (SAN), because the Node Manager listens on multiple addresses.

For scalability and manageability reasons, this Enterprise Deployment Guide uses Per Host Node manager configuration. The sections in the different chapters will provide the required steps for using a single Node manager in each host of the topology.

About Using Unicast for Communications within the Application Tier

Oracle recommends the unicast communication protocol for communication between the Managed Servers and hosts within the Oracle WebLogic Server clusters in an enterprise deployment. Unlike multicast communication, unicast does not require cross-network configuration and it reduces potential network errors that can occur from multicast address conflicts as well.

When you consider using the multicast or unicast protocol for your own deployment, consider the type of network, the number of members in the cluster, and the reliability requirements for cluster membership. Also consider the following features of each protocol.

Features of unicast in an enterprise deployment:

- Uses a group leader that every server sends messages directly to. This leader is responsible for retransmitting the message to every other group member and other group leaders, if applicable.
- Works out of the box in most network topologies
- Requires no additional configuration, regardless of the network topology.
- Uses a single missed heartbeat to remove a server from the cluster membership list.

Features of multicast in an enterprise deployment:

- Multicast uses a more scalable peer-to-peer model, where a server sends each message directly to the network once and the network makes sure that each cluster member receives the message directly from the network.
- Works out of the box in most modern environments, where the cluster members are in a single subnet.
- Requires additional configuration in the routers and WebLogic Server (that is, Multicast TTL) if the cluster members span more than one subnet.
- Uses three consecutive missed heartbeats to remove a server from the cluster membership list.

Depending on the number of servers in your cluster and on whether the cluster membership is critical for the underlying application (for example, in session-replication intensive applications or clusters with intensive RMI invocations across the cluster), each model may act better.

Consider whether your topology is going to be part of an active-active disaster recovery system or if the cluster is going to traverse multiple subnets. In general, unicast acts better in those cases.

For more information about multicast and unicast communication types, see the following resources:

- Configuring Multicast Messaging for WebLogic Server Clusters in *High Availability Guide*

Copyright © 2018, 2025, Oracle and/or its affiliates.

- One-to-Many Communication Using Unicast in *Administering Clusters for Oracle WebLogic Server*

About OPSS and Requests to the Authentication and Authorization Stores

Many of the Oracle Fusion Middleware products and components require an Oracle Platform Security Services (OPSS) security store for authentication providers (an identity store), policies, credentials, keystores, and for audit data. As a result, communications must be enabled so the application tier can send requests to and from the security providers.

For authentication, this communication is to an LDAP directory, such as Oracle Unified Directory (OUD), which typically communicates over port 389 or 636. When you configure an Oracle Fusion Middleware domain, the domain is configured by default to use the WebLogic Server Authentication provider. However, for an enterprise deployment, you must use a dedicated, centralized LDAP-compliant authentication provider.

For authorization (and the policy store), the location of the security store varies, depending upon the tier:

- For the application tier, the authorization store is database-based, so frequent connections from the Oracle WebLogic Server Managed Servers to the database are required for the purpose of retrieving the required OPSS data.
- For the web tier, the authorization store is file-based, so connections to the database are not required.

For more information about OPSS security stores, see the following sections of *Securing Applications with Oracle Platform Security Services*:

- Authentication Basics
- The Security Model

About the Data Tier

In the data tier, an Oracle RAC database runs on the two hosts (DBHOST1 and DBHOST2). The database contains the schemas required by the Oracle FMW Products and the Oracle Platform Security Services (OPSS) policy store.

You can define multiple services for the different products and components in an enterprise deployment to isolate and prioritize throughput and performance accordingly. In this guide, one database service is used as an example. Furthermore, you can use other high availability database solutions to protect the database:

- Oracle Data Guard: See Introduction to Oracle Data Guard in *Oracle Data Guard Concepts and Administration*.
- Oracle RAC One Node: See Overview of Oracle RAC One Node in *Oracle Real Application Clusters Administration and Deployment Guide*.

These solutions above provide protection for the database beyond the information provided in this guide, which focuses on using an Oracle RAC Database, given the scalability and availability requirements that typically apply to an enterprise deployment.

For more information about using Oracle Databases in a high availability environment, see Database Considerations in *High Availability Guide*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

3

About the IAM Enterprise Deployment

Learn about deploying Oracle Identity and Access Management topologies on commodity hardware. These topologies represent specific reference implementations of the concepts described in About a Typical Enterprise Deployment.
This chapter includes the following topics:

- About the Primary and Build-Your-Own Enterprise Deployment Topologies
  There are two primary reference topologies for Oracle Identity and Access Management. While the components installed into each topology are the same, the exact Oracle Identity and Access Management topology you install and configure for your organization may vary.
- Diagram of Oracle Identity and Access Management on Distributed Hardware
  The sample topology diagram in this section shows an eight-node distributed topology.
- About the Primary Oracle Identity and Access Management Topology Diagrams
  The primary enterprise deployment topology is the main Oracle reference topology for Oracle Identity and Access Management. It provides a solution which is both highly available and scalable.
- About the Forgotten Password Functionality
  This section provides information about the forgotten password functionality.
- Integrating Oracle LDAP, Oracle Access Manager, and Oracle Identity Governance
  Integration of Oracle Identity Governance and Oracle Access Manager with LDAP directories is done by using LDAP Connector.
- Roadmap for Implementing the Primary IAM Suite Topologies
  This section provides a roadmap for implementing the primary IAM suite topologies on commodity hardware.
- Building Your Own Oracle Identity and Access Management Topology
  These step-by-step instructions help you configure the two primary enterprise topologies for Oracle Identity and Access Management. However, Oracle recognizes that the requirements of your organization may vary, depending on the specific set of Oracle Fusion Middleware products you purchase and the specific types of applications you deploy.
- About Using SSL Certificates in the Oracle Identity Management Enterprise Topology
  The Oracle Identity Management Enterprise Deployment Topology can use SSL all the way from the external clients to the backend WebLogic Servers.
- About Using JDBC Persistent Stores
  Oracle recommends that you use JDBC stores. This leverages the consistency, data protection, and high availability features of an Oracle database that makes stores available for all the servers in the cluster.
- About Using Service or Server Migration to Enable High Availability of the Enterprise Topology
  To ensure high availability of the Oracle Identity Governance products and components, this guide recommends that you enable Oracle WebLogic Server Whole Server Migration for the Oracle Identity Manager and Oracle SOA Suite clusters that you create as part of the reference topology.

Copyright © 2018, 2025, Oracle and/or its affiliates.

About the Primary and Build-Your-Own Enterprise Deployment Topologies

There are two primary reference topologies for Oracle Identity and Access Management. While the components installed into each topology are the same, the exact Oracle Identity and Access Management topology you install and configure for your organization may vary.

This guide provides step-by-step instructions for installing and configuring these topologies.

The components installed into each topology are the same. The difference being that one deployment is concentrated onto a small number of highly specified servers and the second is distributed amongst a larger number of smaller machines.

To simplify the installation and configuration process, this guide utilizes the IAM deployment wizard, which once you tell it how to layout your topology, will automatically configure it for you.

After you have created your deployment, this guide will show you how to extend it to include additional IAM products, which you may want to use. The procedures in this book do not cover every IAM product. The steps in this guide can easily be adapted to any other IAM product you may want to include.

Diagram of Oracle Identity and Access Management on Distributed Hardware

The sample topology diagram in this section shows an eight-node distributed topology.

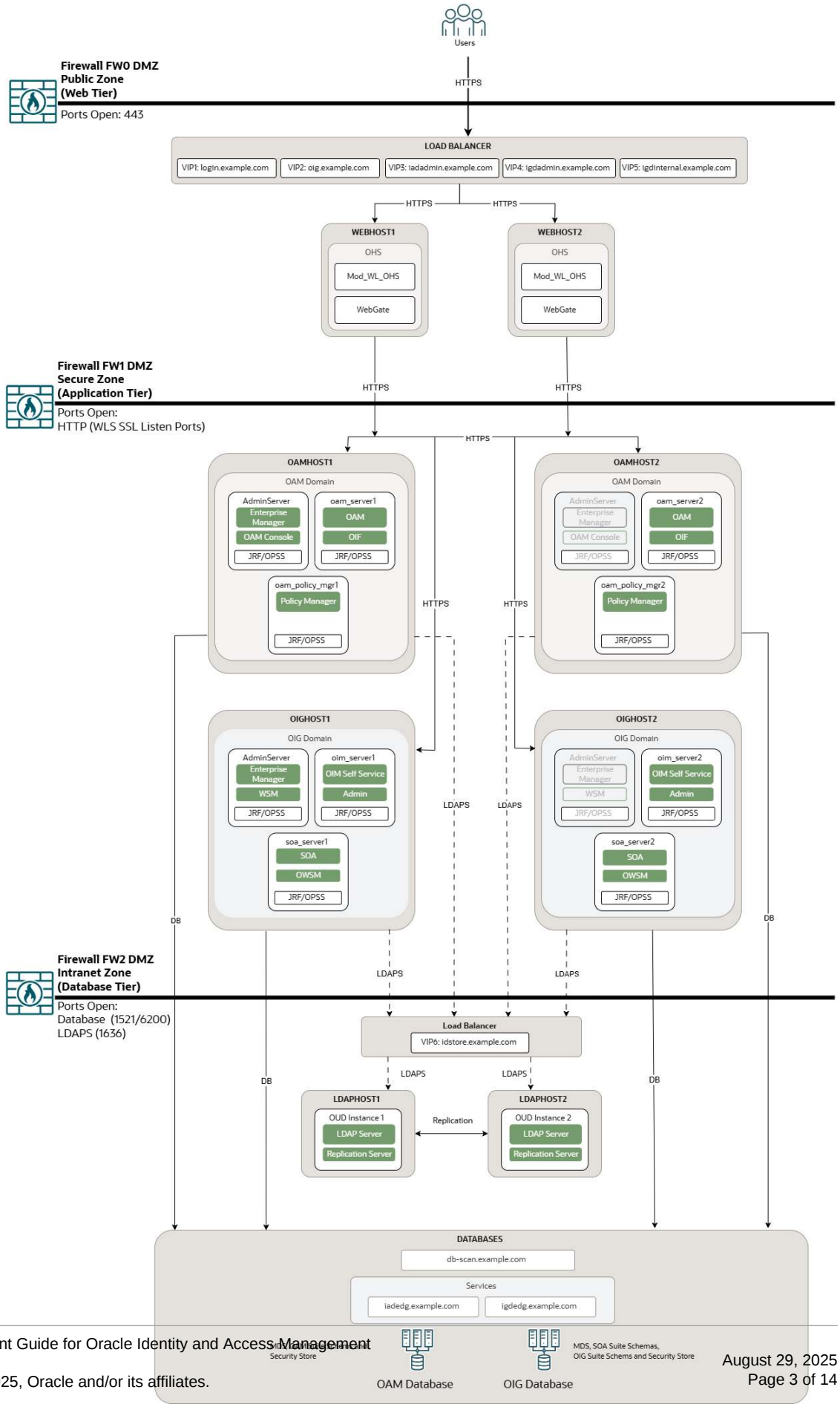
In this topology, the software is distributed across eight hosts: two hosts for the web tier and four hosts for the application tier, and two hosts for the directory services.

This topology is a typical deployment if you are using virtual machines (VMs) that are less powerful, but easy to create and manage. You deploy key components, such as Oracle Access Management, Oracle Identity Governance, and Directory Services on their own dedicated hosts.

The following illustrations shows the Oracle Identity and Access Management topologies for End to End SSL, and for SSL Termination. The diagrams are shown with complete separation of components. If you wish to use less hardware, then you can collocate the components as required. For information about the system requirements for each host, see Host Computer Hardware Requirements.

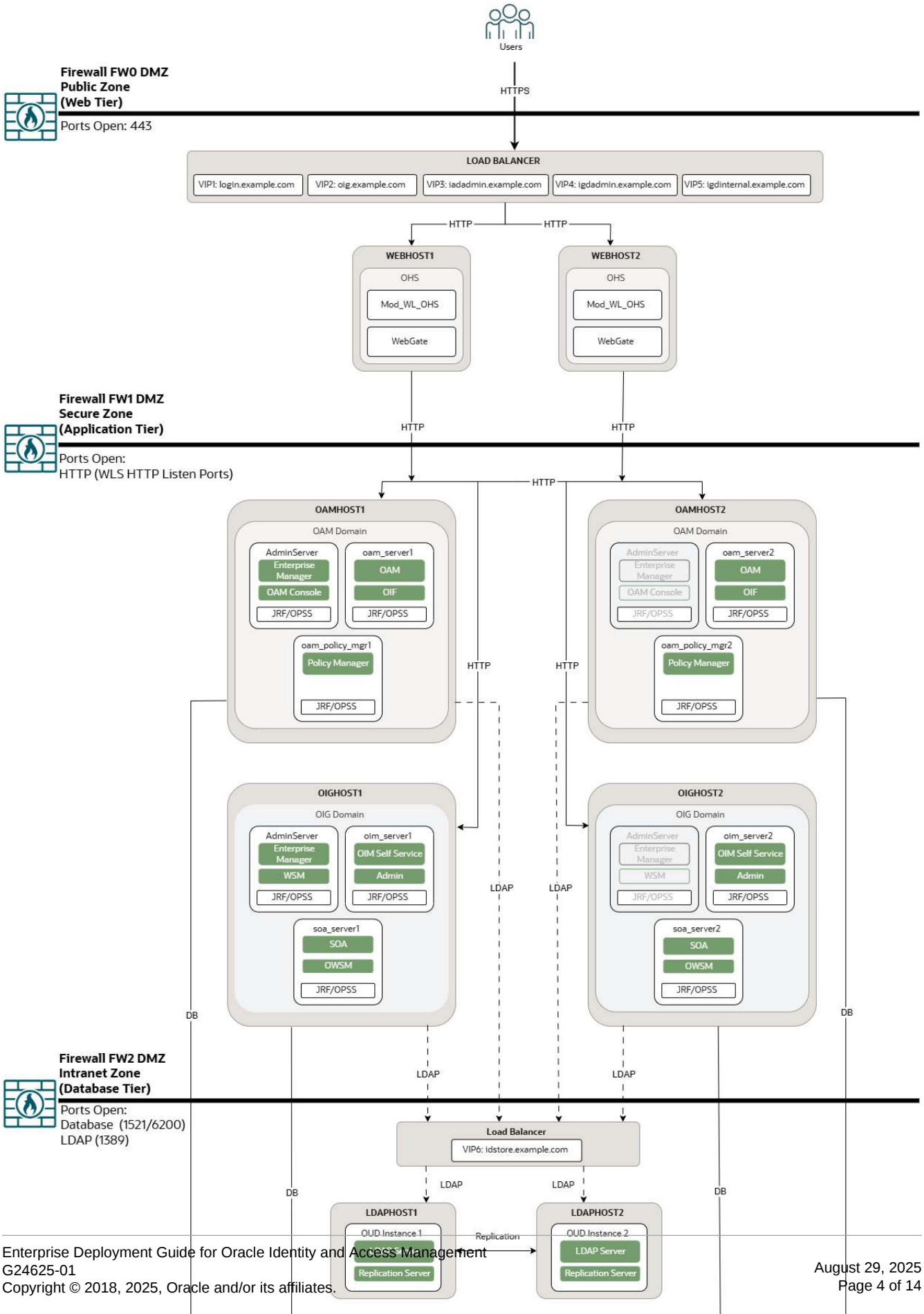
Copyright © 2018, 2025, Oracle and/or its affiliates.

Figure 3-1 Oracle Identity and Access Management End to End SSL Topology



Copyright © 2018, 2025, Oracle and/or its affiliates.

Figure 3-2 Oracle Identity and Access Management SSL Terminated Topology



Copyright © 2018, 2025, Oracle and/or its affiliates.

About the Primary Oracle Identity and Access Management Topology Diagrams

The primary enterprise deployment topology is the main Oracle reference topology for Oracle Identity and Access Management. It provides a solution which is both highly available and scalable.

- Product Separation
- Understanding the Directory Tier
- About Oracle Unified Directory Assured Replication
- Summary of Oracle Identity and Access Management Load Balancer Virtual Server Names
- About the igdinternal Virtual Server Name
- Summary of the Managed Servers and Clusters on the Application Tier Hosts

Product Separation

An IAM deployment is made up of a number of components. These components include:

- Web Servers
- WebLogic Application Servers
- LDAP
- Database

The Oracle Identity and Access Management components are split into two different domains: IAMAccessDomain and IAMGovernanceDomain. Products are distributed as follows:

- IAMAccessDomain contains Oracle Access Management (OAM).
- IAMGovernanceDomain contains Oracle Identity Governance.
- OHSDomain used to host the Oracle HTTP Servers.
- OUDSMDomain used when OUDSM is deployed.

This split is due to the different operational and availability requirements demanded of the individual components. Typically components in the IAMAccessDomain have a higher availability requirement than those in the IAMGovernanceDomain. By separating these components out, you can manage the availability requirements differently. You can patch governance components independently of access components, and you can shut down the Governance instance without impacting the access components. It is not supported to have Oracle Access Manager and Oracle Identity Governance in the same domain.

In addtion to the domains above, note that Oracle Unified Directory is deployed without a domain.

A further benefit of this separation is that you can build a topology in a modular fashion. You can start with a directory and extend it to Access components, then later extend it to Governance components, without needing to affect the deployed software or configuration of existing components, unless you are wiring them together.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Understanding the Directory Tier

The Directory tier consists of two or more physical host computers, where an LDAP compliant directory is installed. Typically, this is Oracle Unified Directory (OUD).

The Directory tier is often combined with the Data tier.

This release of the Enterprise Deployment Guide supports three different LDAP directories. You may be creating this directory for the first time, or you may be using an existing directory from within the organization. The Oracle Unified Directory (OUD) directory is supported.

The directory you choose will be organization dependent.

About Oracle Unified Directory Assured Replication

Oracle Unified Directory server instances natively use replication to keep their embedded databases in sync. By default, replication employs a loose consistency model in which the updates are replicated to replicas AFTER returning the operation result to the application. In this model it is therefore possible to write some data to a replica, and read outdated information from another replica for a short time after the write. Great efforts have been made in Oracle Unified Directory replication to ensure that the replication process is fast and can achieve replication in the order of one millisecond.

Oracle Unified Directory can be configured to use the Assured Replication model, which has been developed to guarantee that the data in the replicas is consistent. When using the Safe Read mode of Assured Replication, applications have the guarantee that the replication process is completed before returning the result of a write operation.

Using Assured Replication has a negative impact on the response time of write operations because it requires some communications with remote replicas before returning the operation result. The amount of the delay varies, depending on the network being used and the capacity of the servers hosting Oracle Unified Directory. Using Assured replication has little if any impact on read operations.

If you expect to regularly perform large writes to your directory, consider configuring your load balancer to distribute requests to your Oracle Unified Directory instances in an active/passive mode. This will remove the chance of you reading out of date data from a replica, but could result in overall performance degradation if your Oracle Unified Directory host is not capable of processing all of the requests.

For the purposes of this Guide, it is assumed that the ability to have multiple servers processing requests is more important than the extra overhead incurred with writing requests in Assured mode. To that end, this Guide shows the configuration of Oracle Unified Directory using Assured Replication. Both of the following Oracle Unified Directory configurations, however, are supported:

- Active/Active in an assured configuration
- Active/Passive in a non assured configuration

For more information, see the Assured Replication section of *Oracle Fusion Middleware Administrator's Guide for Oracle Unified Directory*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Summary of Oracle Identity and Access Management Load Balancer Virtual Server Names

In order to balance the load on servers and to provide high availability, a load balancer is required. This load balancer should exist and be in a redundant configuration to ensure maximum availability. The load balancer must be configured to recognize a set of virtual server names.

The load balancer in Oracle Identity and Access Management deployments must recognize the following virtual server names.

- `login.example.com` - This virtual server name is used for all incoming Access traffic. It acts as the access point for all HTTP traffic to the runtime Access Management components. The load balancer routes all requests to this virtual server name over SSL. As a result, clients access this service using the following secure address:

  `login.example.com:443`

- `oig.example.com` - This virtual server name is used for all incoming Governance traffic. It acts as the access point for all HTTP traffic to the runtime Governance components. The load balancer routes all requests to this virtual server name over SSL. As a result, clients access this service using the following secure address:

  `oig.example.com:443`

  In previous releases of the Enterprise Deployment Guide, `login.example.com` and `oig.example.com` were the same entry point. This release allows for them to be separated out. This will enable smarter routing from the load balancer, allow a more modular deployment and will facilitate future Multi-datacenter deployments. If desired these two entry points can still be combined to provide a single point of entry into the IAM deployment.

- `iadadmin.example.com` - This virtual server name is for administrators who need to access the Oracle Enterprise Manager Fusion Middleware Control, Oracle WebLogic Remote Console, and Oracle Access Manager Console.

  As a result, clients access this service by using the following secure address:

  `iadadmin.example.com:443`

  Create rules in the firewall to block outside traffic from accessing the `/console` and `/em` URLs using this virtual host. Only traffic inside the DMZ should be able to access these URLs on the `iadadmin.example.com` virtual host.

  > **Note**
  >
  > In certain circumstances you may wish to expose the administration URLs to certain users on the internet to provide remote access. Whilst this is highly discouraged, if you need to do this you should have a different listen port for each administration address, for example `iadadmin.example.com:444`. This must have very strict security policies in place to ensure that it is available only to the users required.

- `igdadmin.example.com` - This virtual server name is for administrators who need to access the Oracle Enterprise Manager Fusion Middleware Control, Oracle WebLogic Remote Console, and Oracle Identity Governance System Administration Console.

  As a result, clients access this service by using the following secure address:

Copyright © 2018, 2025, Oracle and/or its affiliates.

`igdadmin.example.com:443`

Create rules in the firewall to block outside traffic from accessing the `/console` and `/em` URLs using this virtual host. Only traffic inside the DMZ should be able to access these URLs on the `iadadmin.example.com` virtual host.

> **Note**
>
> In certain circumstances you may wish to expose the administration URLs to certain users on the internet to provide remote access. Whilst this is highly discouraged, if you need to do this you should have a different listen port for each administration address, for example `iadadmin.example.com:445`. This must have very strict security policies in place to ensure that it is available only to the users required.

- `igdinternal.example.com` - This virtual server name is for internal communications between the application tier components in the Governance Domain only and is not exposed to the Internet. This virtual server is used for both Oracle Identity Governance Suite and Oracle SOA Suite internal communications.

  For SSL terminated installations the traffic from clients to this virtual servers is not SSL-enabled.

  `igdinternal.example.com:7777`

  For End to end SSL installations the following:

  `igdinternal.example.com:443`

- `idstore.example.com` - This virtual server name is used for all incoming identity store traffic. It acts as the access point to the LDAP directory instances. This virtual server is not exposed to the internet.

  The traffic from clients to this virtual server may or may not be SSL-enabled, depending on the type of LDAP directory in use.

To provide an easier mapping between frontend hostnames and backend OHS listeners, this guide presents different listeners (443,444,445 and 446) in the frontend load balancer that map to the 4443, 4444, 4445 and 4446 listeners in OHS. This allows easier segregation of traffic rules and SLAs for each type of access (LBR listener on 443 for apps, 444 for internal access, and 445 for WLS domain administration). Alternatively, the same port (443) can be used in the frontend load balancer for all the different frontend hostnames. However, that OHS cannot host more than one SSL virtual host on the same IP address and port so it will still require separate listeners for each virtual host. For more information about the OHS configuration for the different products, see the following chapters.

This guide provides instructions that explain how to perform the following tasks:

- Configuring the load balancer to recognize and route requests to the virtual host names.
- Configuring the Oracle HTTP Server instances on the web tier to recognize and properly route requests to the virtual host names and the correct host computers.

About the igdinternal Virtual Server Name

The `igdinternal.example.com` virtual server name functions the same as the `oig.example.com`, except that it is invoked by intranet clients and callbacks only. This topic provides additional details.

Copyright © 2018, 2025, Oracle and/or its affiliates.

The igdinternal.example.com virtual server name is not used explicitly during the installation and configuration of the enterprise deployment, but custom systems often expose services that should be consumed by internal-only clients. In such cases, for efficiency and security reasons you must avoid using an external URL such as `oig.example.com`. Instead, you should use an address that cannot be invoked by Internet clients. SOA composite applications can use this internal URL in their end points either directly or through deployment plans.

When you use the `igdinternal.example.com` address, there are implications for the front end address specified for the system. Web services optimizations (for example, direct RMI invocation instead of invocations that involve a full loopback to the load balancer endpoint) are triggered when the front end address for the cluster matches the invocation endpoint. For this reason, depending on the number and relevance of the expected internal invocations, consider setting the front end URL for the cluster and the ServerURL and HTTPServerURL properties to either the external or internal.

You can set the front end URL for a cluster when you create the cluster in the Configuration Wizard. You can also modify it later, by using the WebLogic Remote Console. See Configure HTTP in *Oracle WebLogic Remote Console Online Help*.

Summary of the Managed Servers and Clusters on the Application Tier Hosts

The Application tier hosts the Administration Server and Managed Servers in the Oracle WebLogic Server domains.

Depending upon the components you select, the Oracle WebLogic Server domain for the Oracle Identity and Access Management consists of the clusters shown in Table 3-1. These clusters function as active-active high availability configurations.

Table 3-1 Domain Clusters and Managed Servers

Domain	Cluster	Managed Servers
IAMAccessDomain	Oracle Access Manager	oam_server1, oam_server2
	Oracle Policy Manager	oam_policy_mgr1, oam_policy_mgr2
IAMGovernanceDomain	Oracle Identity Governance	oim_server1, oim_server2
	Oracle SOA Suite	soa_server1, soa_server2

About the Forgotten Password Functionality

This section provides information about the forgotten password functionality.

There are two methods to manage forgotten passwords.

- Oracle Access Management (OAM) and Oracle Identity Governance (OIG) integration:

  In this scenario, when the users first sign in, they can set a number of challenge questions stored in OIG. If users forget their password, they can answer the challenge questions. If they answer their challenge questions successfully, they will be given the option to reset their password by using OIG.

- Oracle Access Management:

  In this scenario, OAM is wired to the Oracle User Messaging Service. Each user is associated with either a telephone number and or an email address. When users request

Copyright © 2018, 2025, Oracle and/or its affiliates.

for a forgotten password link, they are sent a one time PIN which they can enter into the application to reset the password.

This guide describes how to set up both the scenarios.

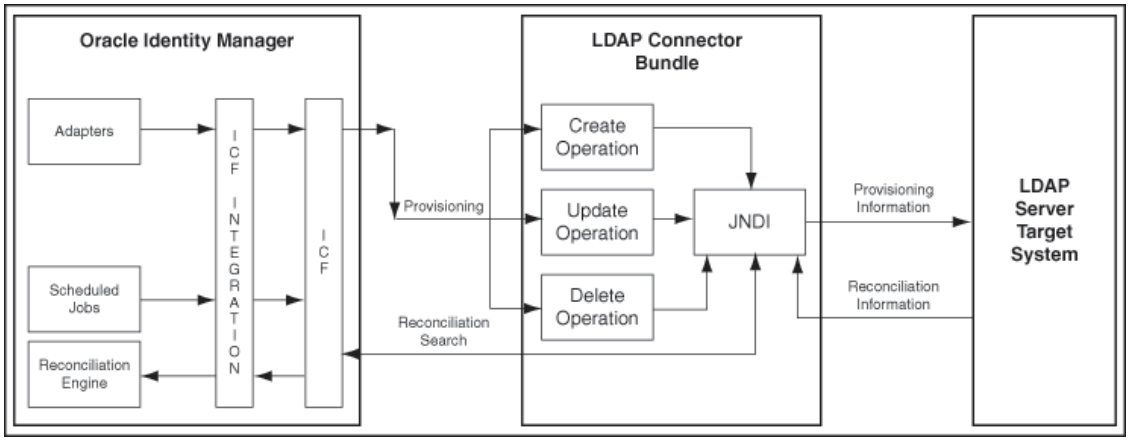
Integrating Oracle LDAP, Oracle Access Manager, and Oracle Identity Governance

Integration of Oracle Identity Governance and Oracle Access Manager with LDAP directories is done by using LDAP Connector.

To enable termination of user sessions upon disablement or termination of a user, download the 12.2.1.3 version of the LDAP Connector.

This section describes how to obtain, install, and configure the Oracle Connector for LDAP.

About the Oracle Connector



The LDAP Connector is implemented by using the Identity Connector Framework (ICF). The ICF is a component that provides basic reconciliation and provisioning operations that are common to all Oracle Identity Governance connectors. The ICF is shipped along with Oracle Identity Governance. Therefore, you need not configure or modify the ICF.

The LDAP Connector uses JNDI to access the target system. This connector can be configured to run in one of the following modes:

- **Identity Reconciliation**: Identity reconciliation is also known as authoritative or trusted source reconciliation. In this form of reconciliation, OIG Users are created or updated corresponding to the creation of and updates to users on the target system.

  > **Note**
  >
  > The identity reconciliation mode supports the reconciliation of user objects only.

- **Account Management**: Account management is also known as target resource management. This mode of the connector enables provisioning and target resource reconciliation.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Provisioning

Provisioning involves creating, updating, or deleting users, groups, roles, and organizational units (OUs) on the target system through Oracle Identity Governance.

When you allocate (or provision) a target system resource to an OIM user, the operation results in the creation of an account on the target system for that user. In the Oracle Identity Governance context, the term **provisioning** is also used to mean updates (for example enabling or disabling) made to the target system account through Oracle Identity Governance.

Users and organizations are organized in a hierarchical format on the target system. Before you can provision users to (that is, create users in) the required organizational units (OUs) on the target system, you must fetch into Oracle Identity Governance the list of OUs used on the target system. This is achieved by using the LDAP Connector OU lookup Reconciliation scheduled job for lookup synchronization.

Similarly, before you can provision users to the required groups or roles on the target system, you must fetch into Oracle Identity Governance the list of all groups and roles used on the target system. This is achieved by using the LDAP Connector Group Lookup Reconciliation and LDAP Connector Role Lookup Recon scheduled jobs for lookup synchronization.

Target resource reconciliation

To perform target resource reconciliation, the LDAP Connector User Search Reconciliation or LDAP Connector User Sync Reconciliation scheduled jobs is used. The connector applies filters to locate users to be reconciled from the target system and then fetches the attribute values of these users.

Depending on the data that you want to reconcile, you use different scheduled jobs. For example, you use the LDAP Connector User Search Reconciliation scheduled job to reconcile user data in the target resource mode.

You can deploy the Oracle LDAP Connector either locally in Oracle Identity Governance or remotely in the connector server. A **connector server** enables remote execution of an Identity Connector. This guide explains the steps to install and configure the connector locally in Oracle Identity Governance.

Roadmap for Implementing the Primary IAM Suite Topologies

This section provides a roadmap for implementing the primary IAM suite topologies on commodity hardware.

Table 3-2 below outlines a roadmap for implementing deploying IAM in an Enterprise Deployment.

Table 3-2 Roadmap for Implementing Primary IAM Suite Topologies on Commodity Hardware

Scenario	Tasks	For More Information, See
Creating an IAM Enterprise Deployment manually on commodity hardware	Understand a typical enterprise deployment and review the primary deployment topologies.	Enterprise Deployment Overview About a Typical Enterprise Deployment About the IAM Enterprise Deployment

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 3-2 (Cont.) Roadmap for Implementing Primary IAM Suite Topologies on Commodity Hardware

Scenario	Tasks	For More Information, See
	Review the hardware and software requirements and procure the resources for the enterprise deployment.	Procuring Resources for an Enterprise Deployment or Procuring Resources for an Oracle Cloud Infrastructure Deployment
	Download and configure the WebLogic Remote Console.	Download and Configure the WebLogic Remote Console
	Prepare the load balancers and firewalls.	Preparing the Load Balancer and Firewalls for an Enterprise Deployment
	Prepare the storage and understand the directory structure.	Preparing the File System for an Enterprise Deployment
	Configure the host computers.	Preparing the Host Computers for an Enterprise Deployment
	Prepare the Oracle Cloud Infrastructure for an enterprise deployment (if using).	Preparing the Oracle Cloud Infrastructure for an Enterprise Deployment
	Prepare the database.	Preparing the Database for an Enterprise Deployment
	Install and configure Oracle LDAP.	Configuring Oracle LDAP for an Enterprise Deployment
	Install and configure Oracle Unified Directory Services Manager.	Configuring Oracle Unified Directory Service Manager
	Install and configure Oracle Access Management	Configuring Oracle Access Management
	Install and configure Oracle Identity Governance.	Configuring Oracle Identity Governance
	Install and Configure Oracle HTTP Server.	Configuring Oracle HTTP Server for an Enterprise Deployment
	Perform common configuration and management tasks.	Common Configuration and Management Tasks for an Enterprise Deployment
	Configure server migration settings.	Using Service Migration in an Enterprise Deployment
	Learn about scaling procedures.	Scaling Procedures for an Enterprise Deployment
	Configure Single Sign-On (SSO).	Configuring Single Sign-On for an Enterprise Deployment

Building Your Own Oracle Identity and Access Management Topology

These step-by-step instructions help you configure the two primary enterprise topologies for Oracle Identity and Access Management. However, Oracle recognizes that the requirements of

Copyright © 2018, 2025, Oracle and/or its affiliates.

your organization may vary, depending on the specific set of Oracle Fusion Middleware products you purchase and the specific types of applications you deploy.

For information on the primary enterprise topologies for Oracle Identity and Access Management, see #unique_49/unique_49_Connect_42_CHDFBIDE.

In many cases, you can install and configure an alternative topology, one that includes additional components, or one that does not include all the Oracle Identity and Access Management products shown in the primary topology diagrams.

A few alternatives to the reference Oracle Identity and Access Management topologies you can implement by using the steps provided in this guide are:

- OAM Only with an Existing Directory
- OIG Only with an Existing Directory
- OAM/OIG Integrated in a Modular Deployment
- OAM/OIG Integrated with an Existing Directory

About Using SSL Certificates in the Oracle Identity Management Enterprise Topology

The Oracle Identity Management Enterprise Deployment Topology can use SSL all the way from the external clients to the backend WebLogic Servers.

The HTTPS protocol is used for the communication between the clients and the front-end load balancer, the front-end load balancer and Oracle HTTP Servers, and the Oracle HTTP Servers and the WebLogic Servers.

For these communications to take place, each access point (whether the frontend load balancer, Oracle HTTP Server or Oracle WebLogic Server) needs to certify its identity and provide a public key to encrypt traffic. This is done using the appropriate SSL certificate in each access point. These SSL certificates are typically provided by formal certificate authorities (CAs). It is expected that in a real enterprise deployment, certificates are issued by one of these commercial entities and are present in the pertaining end points. Since each Certificate Authority has its own mechanisms to issue a SSL certificate, it is out of the scope of this Enterprise Deployment Guide to use and describe such processes. However, to provide a working end to end configuration with a solid encryption in the different communications across tiers, this guide provides SSL configuration steps using a self issued Certificate Authority. This approach is considered secure and solid enough since the endpoints involved (OHS and WLS listeners) reside behind their corresponding firewalls and are not really exposed to the public. The frontend load balancer is considered a much more exposed endpoint and it is expected in all cases that users acquire the pertaining certificate for it from a formal Certificate Authority.

By default, the different chapters in this guide follow a flow based on the creation and usage of certificates using a self issued CA. Scripts are provided to simplify most of the complexity around creating these certificates and adding them to the corresponding stores and wallets.

About Using JDBC Persistent Stores

Oracle recommends that you use JDBC stores. This leverages the consistency, data protection, and high availability features of an Oracle database that makes stores available for all the servers in the cluster.

Refer to the technical details in the Using Persistent Stores for TLOGs and JMS in an Enterprise Deployment section in the *Common Configuration and Management Tasks for an*

Copyright © 2018, 2025, Oracle and/or its affiliates.

Enterprise Deployment chapter. These technical details provide some guidance on possible performance implications and tunning recommendations.

In the Configuration Wizard, select the following when creating and extending domains in the **High Availability Options** screen:

- Set **JTA Transaction Log Persistence** to **JDBC TLog Store**.
- Set **JMS Server Persistence** to **JMS JDBC Store**.

This ensures that the JDBC persistent stores are used for both JMS and TLOGS.

About Using Service or Server Migration to Enable High Availability of the Enterprise Topology

To ensure high availability of the Oracle Identity Governance products and components, this guide recommends that you enable Oracle WebLogic Server Whole Server Migration for the Oracle Identity Manager and Oracle SOA Suite clusters that you create as part of the reference topology.

Service Migration is configured using the Configuration Wizard HA Options screen. When you select the **Enable Automatic Service Migration** option in the Configuration Wizard HA Options screen, it configures migratable target definitions that are required for automatic service migration.

In the same screen, you use **JTA Transaction Log Persistence** and **JMS Server Persistence** options to configure them with JDBC stores automatically. Oracle recommends that you enable these options when you configure clusters in the SOA enterprise deployment.

For more information about service migration, see Using Service Migration in an Enterprise Deployment.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Part II
Preparing for an Enterprise Deployment

It is important to understand the tasks that need to be performed to prepare for an enterprise deployment.

This part of the enterprise deployment guide contains the following topics.

- Procuring Resources for an Oracle Cloud Infrastructure Deployment
  Before you deploy Oracle Identity and Access Management on Oracle Cloud Infrastructure (OCI), you need to ensure that you have sufficient OCI resources at your disposal.
- Procuring Resources for an Enterprise Deployment
  It is essential to procure the required hardware, software, and network settings before you configure the Oracle Identity and Access Management reference topology.
- Download and Configure the WebLogic Remote Console
  This chapter describes how to download and configure the WebLogic Remote Console.
- Preparing the Load Balancer and Firewalls for an Enterprise Deployment
  It is important to understand how to configure the external load balancer and ports that must be opened on the firewalls for an enterprise deployment.
- Preparing the File System for an Enterprise Deployment
  Preparing the file system for an enterprise deployment involves understanding the requirements for local and shared storage, as well as the terminology that is used to reference important directories and file locations during the installation and configuration of the enterprise topology.
- Preparing the Host Computers for an Enterprise Deployment
  It is important to perform a set of tasks on each computer or server before you configure the enterprise deployment topology. This involves verifying the minimum hardware and operating system requirements for each host, configuring operating system users and groups, enabling Unicode support, mounting the required shared storage systems to the host and enabling the required virtual IP addresses on each host.
- Preparing the Oracle Cloud Infrastructure for an Enterprise Deployment
  If you plan to deploy Identity and Access Management on Oracle Cloud Infrastructure (OCI), you have to configure OCI to facilitate the deployment. Create the required OCI components to perform the deployment.
- Preparing the Database for an Enterprise Deployment
  Preparing the database for an enterprise deployment involves ensuring that the database meets specific requirements, creating database services, using SecureFiles for large objects in the database, and creating database backup strategies.

Copyright © 2018, 2025, Oracle and/or its affiliates.

4

Procuring Resources for an Oracle Cloud Infrastructure Deployment

Before you deploy Oracle Identity and Access Management on Oracle Cloud Infrastructure (OCI), you need to ensure that you have sufficient OCI resources at your disposal.

For information about the resources, see Preparing the Oracle Cloud Infrastructure for an Enterprise Deployment.

This chapter includes the following topics:

- Procuring Resources for OCI
  It is important to understand the resource requirements for an Oracle Cloud Infrastructure deployment. These resources include load balancer, compute instances, network, gateways, and databases.
- Sizing
  The sizing guidelines provide the performance recommendations and sizing requirements for Oracle Identity and Access Management, Release 14.1.2.1.0.

Procuring Resources for OCI

It is important to understand the resource requirements for an Oracle Cloud Infrastructure deployment. These resources include load balancer, compute instances, network, gateways, and databases.

For an illustration of the OCI layout depicting the use of these resources, see #unique_85/unique_85_Connect_42_FIG_AHX_WTD_XPB.

- Load Balancer Requirements
  You will require two load balancers. One for internal traffic and the other for external traffic. The Shape of the load balancer should be sufficient for your expected traffic volume.
- Network
  You will require one Virtual Cloud Network (VCN). This network is sub-divided into several subnets to increase security. Each subnet has security lists and route tables.
- Compute Instances
  You will require a minimum of nine compute instances.
- Gateway
  You require one public gateway and one service gateway.
- Database
  The number of databases you require depends on the disaster recovery strategy you plan to use. If you have a traditional Active/Passive solution, then you can use a single container database (CDB) with two pluggable databases (PDB).

Load Balancer Requirements

You will require two load balancers. One for internal traffic and the other for external traffic. The Shape of the load balancer should be sufficient for your expected traffic volume.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Network

You will require one Virtual Cloud Network (VCN). This network is sub-divided into several subnets to increase security. Each subnet has security lists and route tables.

Oracle virtual cloud networks (VCNs) provide customizable and private cloud networks in Oracle Cloud Infrastructure (OCI). Just like a traditional data center network, the VCN provides you complete control over your cloud networking environment. You can assign private IP address spaces, create subnets and route tables, and configure stateful firewalls.

Compute Instances

You will require a minimum of nine compute instances.

- Bastion Node: The bastion node is used to set up the environment and to provide you access to the internal resources in the Kubernetes cluster; you cannot access them directly. The bastion node may be the smallest shape available as it does not perform any day-to-day work. The bastion node can also be used as the administrative node for Oracle Identity Role Intelligence.
- Web Tier Nodes: You will require a minimum of two web tier compute instances. These instances require sufficient resources to handle the expected traffic flow. You can size them the same way as their on-premise equivalents.
- Directory Nodes: You will require a minimum of two directory compute instances. These will reside in the Database Tier.
- Application Nodes: You will require a minimum of two compute instances for Oracle Access Manager and a minimum of two compute instances for Oracle Identity Governance. These will reside in the application tier.

Gateway

You require one public gateway and one service gateway.

Public Gateway: A public gateway is a virtual router you can add to your VCN to enable direct connectivity to the internet. The gateway supports connections initiated from within the VCN (Egress) and connections initiated from the internet (Ingress).

Service Gateway: A service gateway enables cloud resources without public IP addresses to privately access Oracle services.

Database

The number of databases you require depends on the disaster recovery strategy you plan to use. If you have a traditional Active/Passive solution, then you can use a single container database (CDB) with two pluggable databases (PDB).

If the disaster recovery strategy is to use Oracle Access Manager Active/Active and Oracle Identity Governance Active/Passive, then you will require two separate databases.

The databases you create should be highly available real application cluster databases. For more information about the database requirements, see Preparing the Database for an Enterprise Deployment.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Sizing

The sizing guidelines provide the performance recommendations and sizing requirements for Oracle Identity and Access Management, Release 14.1.2.1.0.

For sizing guidelines, see Deep Dive into Oracle Access Management 12.2.1.4.0 Performance.

Copyright © 2018, 2025, Oracle and/or its affiliates.

5

Procuring Resources for an Enterprise Deployment

It is essential to procure the required hardware, software, and network settings before you configure the Oracle Identity and Access Management reference topology.

This chapter provides information on how to reserve the required IP addresses and identify and obtain software downloads for an enterprise deployment.

- Hardware and Software Requirements for the Enterprise Deployment Topology
  It is important to understand the hardware load balancer requirements, host computer hardware requirements, and operating system requirements for the enterprise deployment topology.
- Reserving the Required IP Addresses for an Enterprise Deployment
  You have to obtain and reserve a set of IP addresses before you install and configure the enterprise topology. The set of IP addresses that need to be reserved are listed in this section.
- Identifying and Obtaining Software Distributions for an Enterprise Deployment
  Before you begin to install and configure the enterprise topology, you must obtain the software distributions that you need to implement the topology.
- Obtaining SSL Certificates
  The Enterprise Deployment Topology uses SSL and hence you must obtain SSL certificates. SSL can be terminated at the load balancer or can be enabled all the way from the external clients to the backend application servers.

Hardware and Software Requirements for the Enterprise Deployment Topology

It is important to understand the hardware load balancer requirements, host computer hardware requirements, and operating system requirements for the enterprise deployment topology.

This section includes the following sections.

- External Load Balancer Requirements
  The section lists the requirements of the external load balancer.
- Host Computer Hardware Requirements
  This section provides information to help you procure host computers that are configured to support the enterprise deployment topologies.
- Operating System Requirements for an Enterprise Deployment Topology
  This section provides details about the operating system requirements.
- About Private Networks
  A private network enables you to keep inter-application communications within the private network, providing communication that is both faster and more secure. By keeping inter-application traffic inside the private network, you do not expose traffic to the internet. To use a private network, you have to create a private VLAN.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- About Virtual Server Templates
  You can customize the values of the Virtual Server Templates depending on the results of your capacity planning.

External Load Balancer Requirements

The section lists the requirements of the external load balancer.

The enterprise topology uses an external load balancer. The features of the external load balancer are:

- Ability to load-balance traffic to a pool of real servers through a virtual host name: Clients access services by using the virtual host name (instead of using actual host names). The load balancer can then load balance requests to the servers in the pool.
- Port translation configuration should be possible so that incoming requests on the virtual host name and port are directed to a different port on the backend servers.
- Monitoring of ports on the servers in the pool to determine availability of a service.
- Ability to configure names and ports on your external load balancer. The virtual server names and ports must meet the following requirements:
  - The load balancer should allow configuration of multiple virtual servers. For each virtual server, the load balancer should allow configuration of traffic management on more than one port. For example, for Oracle HTTP Server in the web tier, the load balancer needs to be configured with a virtual server and ports for HTTP and HTTPS traffic.
  - The virtual server names must be associated with IP addresses and be part of your DNS. Clients must be able to access the external load balancer through the virtual server names.
- Ability to detect node failures and immediately stop routing traffic to the failed node.
- It is highly recommended that you configure the load balancer to be in fault-tolerant mode.
- It is highly recommended that you configure the load balancer virtual server to return immediately to the calling client when the backend services to which it forwards traffic are unavailable. This is preferred over the client disconnecting on its own after a timeout based on the TCP/IP settings on the client machine.
- Ability to maintain sticky connections to components. Examples of this include cookie-based persistence, IP-based persistence, and so on.
- The load balancer should be able to terminate SSL requests at the load balancer and forward traffic to the backend real servers by using the equivalent non-SSL protocol (for example, HTTPS to HTTP).
- In this Enterprise Deployment Guide, SSL listeners are used for the Oracle HTTP Servers and the Oracle WebLogic Servers. The load balancer should therefore be able to establish SSL communication with the backend servers in its pools.
- The ability to route TCP/IP requests. This is a requirement for LDAP
- SSL acceleration (this feature is recommended, but not required for the enterprise topology).

Copyright © 2018, 2025, Oracle and/or its affiliates.

Host Computer Hardware Requirements

This section provides information to help you procure host computers that are configured to support the enterprise deployment topologies.

It includes the following topics.

- General Considerations for Enterprise Deployment Host Computers
  This section specifies the general considerations that are required for the enterprise deployment host computers.
- Reviewing the Oracle Fusion Middleware System Requirements
  This section provides reference to the system requirements information to help you ensure that the environment meets the necessary minimum requirements.
- Typical Memory, File Descriptors, and Processes Required for an Enterprise Deployment
  This section specifies the typical memory, number of file descriptors, and operating system processes and tasks details that are required for an enterprise deployment.
- Typical Disk Space Requirements for an Enterprise Deployment
  This section specifies the disk space that is typically required for this enterprise deployment.

General Considerations for Enterprise Deployment Host Computers

This section specifies the general considerations that are required for the enterprise deployment host computers.

Before you start the process of configuring an Oracle Fusion Middleware enterprise deployment, you must perform the appropriate capacity planning to determine the number of nodes, CPUs, and memory requirements for each node depending on the specific system's load as well as the throughput and response requirements. These requirements vary for each application or custom Oracle Identity and Access Management system being used.

The information in this chapter provides general guidelines and information that helps you determine the host computer requirements. It does not replace the need to perform capacity planning for your specific production environment.

> **Note**
>
> As you obtain and reserve the host computers in this section, note the host names and system characteristics in the Enterprise Deployment workbook. You will use these addresses later when you enable the IP addresses on each host computer. See Using the Enterprise Deployment Workbook.

Reviewing the Oracle Fusion Middleware System Requirements

This section provides reference to the system requirements information to help you ensure that the environment meets the necessary minimum requirements.

Review the *Oracle Fusion Middleware System Requirements and Specifications* to ensure that your environment meets the minimum installation requirements for the products that you are installing.

Copyright © 2018, 2025, Oracle and/or its affiliates.

The Requirements and Specifications document contains information about general Oracle Fusion Middleware hardware and software requirements, minimum disk space and memory requirements, database schema requirements, and the required operating system libraries and packages.

It also provides some general guidelines for estimating the memory requirements for your Oracle Fusion Middleware deployment.

Typical Memory, File Descriptors, and Processes Required for an Enterprise Deployment

This section specifies the typical memory, number of file descriptors, and operating system processes and tasks details that are required for an enterprise deployment.

The following table summarizes the memory, file descriptors, and processes required for the Administration Server and each of the Managed Servers computers in a typical Oracle Identity and Access Management enterprise deployment. These values are provided as an example only, but they can be used to estimate the minimum amount of memory required for an initial enterprise deployment.

The example in this topic reflects the minimum requirements for configuring the Managed Servers and other services required on OAMHOST1, as depicted in the reference topologies.

When you procure systems, use the information in the **Approximate Top Memory** column as a guide when determining the minimum physical memory that each host computer should have available.

After you procure the host computer hardware and verify the operating system requirements, review the software configuration to be sure that the operating system settings are configured to accommodate the number of open files listed in the **File Descriptors** column and the number processes listed in the **Operating System Processes and Tasks** column.

See Setting the Open File Limit and Number of Processes Settings on UNIX Systems.

Table 5-1 Typical Memory, File Descriptors, and Processes Required for an Enterprise Deployment

Managed Server, Utility, or Service	Approximate Top Memory	Number of File Descriptors	Operating System Processes and Tasks
Access Administration Server	3.5 GB	2300	180
Governance Administration Server	4.5 GB	2100	100
soa_server	6.0 GB	3100	240
oim_server	8 GB	1400	190
oam_server	2.7 GB	2000	170
oam_policy_mgr	2.88 GB	1700	160
WLST (connection to the Node Manager)	1.5 GB	910	20
Configuration Wizard	1.5 GB	700	20
Node Manager	1.0 GB	300	20
TOTAL	31.58 GB*	14510	1100

Copyright © 2018, 2025, Oracle and/or its affiliates.

* Approximate total, with consideration for Operating System and other additional memory requirements.

Typical Disk Space Requirements for an Enterprise Deployment

This section specifies the disk space that is typically required for this enterprise deployment.

For the latest disk space requirements for the Oracle Fusion Middleware 14c (14.1.2.1.0) products, including the Oracle Identity and Access Management products, review the *Oracle Fusion Middleware System Requirements and Specifications*.

In addition, the following table summarizes the disk space that is typically required for an Oracle Identity and Access Management enterprise deployment.

Use the this information and the information in Preparing the File System for an Enterprise Deployment to determine the disk space requirements required for your deployment.

Server	Disk
Database	nXm n = number of disks, at least 4 (striped as one disk) m = size of the disk (minimum of 30 GB)
WEBHOST*n*	10 GB
OAMHOST*n*	10 GB*
OIGHOST*n*	10 GB*
LDAPHOST*n*	10 GB*

* For a shared storage Oracle home configuration, two installations suffice by making a total of 20 GB.

Operating System Requirements for an Enterprise Deployment Topology

This section provides details about the operating system requirements.

The Oracle Fusion Middleware software products and components that are described in this guide are certified on various operating systems and platforms, which are listed in *Oracle Fusion Middleware System Requirements and Specifications*.

> **Note**
>
> This guide focuses on the implementation of the enterprise deployment reference topology on Oracle Linux systems.
>
> The topology can be implemented on any certified, supported operating system, but the examples in this guide typically show the commands and configuration steps as they should be performed by using the bash shell on Oracle Linux.

About Private Networks

A private network enables you to keep inter-application communications within the private network, providing communication that is both faster and more secure. By keeping inter-

Copyright © 2018, 2025, Oracle and/or its affiliates.

application traffic inside the private network, you do not expose traffic to the internet. To use a private network, you have to create a private VLAN.

About Virtual Server Templates

You can customize the values of the Virtual Server Templates depending on the results of your capacity planning.

The following are the typical virtual server templates that you can customize.

Table 5-2 Virtual Server Templates

Type	Description	Memory (GB)	Number of Virtual CPUs
VERY_LARGE	Large Memory Intensive Applications	20	6
EXTRA_LARGE	CPU Intensive Applications	16	6
LARGE	Average Intensity Applications	8	2
SMALL	Low intensity applications	4	1

Reserving the Required IP Addresses for an Enterprise Deployment

You have to obtain and reserve a set of IP addresses before you install and configure the enterprise topology. The set of IP addresses that need to be reserved are listed in this section.

Before you begin installing and configuring the enterprise topology, you must obtain and reserve a set of IP addresses:

- Physical IP (IP) addresses for each of the host computers that you have procured for the topology
- A virtual IP (VIP) address for the Administration Servers and a virtual host name mapped to this VIP
- VIPs are not required for any of the Managed Servers in the FMW IAM Enterprise Deployment since all components support Automatic Service Migration.

  For Fusion Middleware 14*c* products that support Automatic Service Migration, VIPs for the Managed Servers are typically not necessary.

You can then work with your network administrator to be sure that these required VIPs are defined in your DNS server. Alternatively, for non-production environments, you can use the /etc/hosts file to define these virtual hosts.

For more information, see the following topics.

- What is a Virtual IP (VIP) Address?
  This section defines the virtual IP address and specifies its purpose.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Why Use Virtual Host Names and Virtual IP Addresses?
  For an enterprise deployment, in particular, it is important that a set of VIPs and the virtual host names to which they are mapped are reserved and enabled on the corporate network.
- Physical and Virtual IP Addresses Required by the Enterprise Topology
  This section describes the physical IP (IP) and virtual IP (VIP) addresses that are required for the Administration Server and each of the Managed Servers in a typical Oracle Identity and Access Management enterprise deployment topology.

What is a Virtual IP (VIP) Address?

This section defines the virtual IP address and specifies its purpose.

A virtual IP address is an unused IP Address that belongs to the same subnet as the host's primary IP address. It is assigned to a host manually. If a host computer fails, the virtual address can be assigned to a new host in the topology. For the purposes of this guide, *virtual* IP addresses are referenced, which can be reassigned from one host to another, and *physical* IP addresses are referenced, which are assigned permanently to hardware host computer.

Why Use Virtual Host Names and Virtual IP Addresses?

For an enterprise deployment, in particular, it is important that a set of VIPs and the virtual host names to which they are mapped are reserved and enabled on the corporate network.

Alternatively, host names can be resolved through the appropriate `/etc/hosts` file propagated through the different nodes.

The Oracle Identity Governance product supports Automatic Service Migration. As a result, it is no longer necessary to reserve VIPs for each of the Managed Servers in the domain. Instead, a VIP is required for the Administration Server only.

In the event of the failure of the host computer where the IP address is assigned, the IP address can be assigned to another host in the same subnet so that the new host can take responsibility for running the Admin Server. The reassignment of virtual IP address for the Administration Server must be performed manually.

> **Note**
>
> Regardless the use of virtual or physical IPs, Oracle also recommends that you use aliases to map to different IPs in different data centers in preparation for disaster recovery. It is recommended to use these aliases to configure the listen address for the components. This approach will be used in this guide.

Physical and Virtual IP Addresses Required by the Enterprise Topology

This section describes the physical IP (IP) and virtual IP (VIP) addresses that are required for the Administration Server and each of the Managed Servers in a typical Oracle Identity and Access Management enterprise deployment topology.

Before you begin to install and configure the enterprise deployment, reserve a set of host names and IP addresses that correspond to the VIPs in Table 5-3.

You can assign any unique host name to the VIPs, but in this guide, each VIP is referenced by using the suggested host names in the table.

Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> As you obtain and reserve the IP addresses and their corresponding virtual host names in this section, note the values of the IP addresses and host names in the Enterprise Deployment workbook. You will use these addresses later when you enable the IP addresses on each host computer. See Using the Enterprise Deployment Workbook .

Table 5-3 Summary of the Virtual IP Addresses Required for the Enterprise Deployment

Virtual IP	VIP Maps to...	Description
VIP1	IADADMINVHN	IADADMINVHN is the virtual host name used as the listen address for the Administration Server used by the IAMAccessDomain and fails over with manual failover of the Administration Server. It is enabled on the node where the Administration Server process is running.
VIP2	IGDADMINVHN	IGDADMINVHN is the virtual host name used as the listen address for the Administration Server used by the IAMGovernacneDomain and fails over with manual failover of the Administration Server. It is enabled on the node where the Administration Server process is running.

Identifying and Obtaining Software Distributions for an Enterprise Deployment

Before you begin to install and configure the enterprise topology, you must obtain the software distributions that you need to implement the topology.

The following table lists the distributions used in this guide.

For general information about how to obtain Oracle Fusion Middleware software, see Obtaining Product Distributions in *Planning an Installation of Oracle Fusion Middleware*.

For more specific information about locating and downloading specific Oracle Fusion Middleware products, see the *Oracle Fusion Middleware Download, Installation, and Configuration Readme Files* on OTN.

> **Note**
>
> The information in this guide is meant to complement the information contained in the Oracle Fusion Middleware certification matrixes. If there is a conflict of information between this guide and the certification matrixes, then the information in the certification matrixes must be considered the correct version, as they are frequently updated.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 5-4 Oracle Fusion Middleware Distributions

Distribution	Description	Installer File Name	Mandatory Patches
Oracle Identity Management quick installer 14c (14.1.2.1.0)	Download this distribution to install the Oracle Fusion Middleware Infrastructure, Oracle SOA Suite, and Oracle Identity and Access Management. This distribution also installs the Repository Creation Utility (RCU), which in previous Oracle Fusion Middleware releases was packaged in its own distribution.	`fmw_14.1.2.1.0_idmquickstart.jar`	October 2025 Stack Bundle Patch or later. For patch details, see Doc ID 2657920.2 on My Oracle Support.
Oracle Fusion Middleware 14c (14.1.2.1.0) Infrastructure	Download this distribution to install the Oracle Fusion Middleware Infrastructure, Oracle Identity and Access Management, Oracle SOA Suite. Each product can be downloaded separately but this quite will use the quick installer. This distribution also installs the Repository Creation Utility (RCU), which in previous Oracle Fusion Middleware releases was packaged in its own distribution.	`fmw_14.1.2.1.0_infrastructure_generic.jar`	October 2025 Stack Bundle Patch or later. For patch details, see Doc ID 2657920.2 on My Oracle Support.
Oracle HTTP Server 14c (14.1.2.0.0)	Download this distribution to install the Oracle HTTP Server software on the Web Tier.	`fmw_14.1.2.0.0_ohs_linux64.bin`	October 2025 Bundle Patch or later.
Oracle Unified Directory 14c (14.1.2.1.0)	Download this distribution to install the Oracle Unified Directory software.	`fmw_14.1.2.1.0_oud.jar`	October 2025 Stack Bundle Patch or later. For patch details, see Doc ID 2657920.2 on My Oracle Support. Patch 38047590 - LDAPConfigTool

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 5-4 (Cont.) Oracle Fusion Middleware Distributions

Distribution	Description	Installer File Name	Mandatory Patches
Oracle Internet Directory 14c (14.1.2.1.0)	Download this distribution to install the Oracle Internet Directory software.	`fmw_14.1.2.1.0_oid_linux64.bin`	October 2025 Stack Bundle Patch or later. For patch details, see Doc ID 2657920.2 on My Oracle Support. Patch 38047590 - LDAPConfigTool
Oracle Identity and Access Management 14c (14.1.2.1.0)	Download this distribution to install the Oracle Identity and Access Management software.	`fmw_14.1.2.1.0_idm.jar`	October 2025 Stack Bundle Patch or later. For patch details, see Doc ID 2657920.2 on My Oracle Support.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 5-4 (Cont.) Oracle Fusion Middleware Distributions

Distribution	Description	Installer File Name	Mandatory Patches
Oracle SOA Suite 14c (14.1.2.0.0)	Download this distribution to install the Oracle SOA Suite software.	`fmw_14.1.2.0.0_soa.jar`	October 2025 Stack Bundle Patch or later. For patch details, see Doc ID 2657920.2 on My Oracle Support.

Note

This is not required if you are using idmquickinsta

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 5-4 (Cont.) Oracle Fusion Middleware Distributions

Distribution	Description	Installer File Name	Mandatory Patches
		l l e r .	
Oracle Internet Directory Connector (12.2.1.3+)	Download this distribution to integrate with Oracle Internet Directory and Oracle Unified Directory.	`oid-12.2.1.3.0.zip`	

Obtaining SSL Certificates

The Enterprise Deployment Topology uses SSL and hence you must obtain SSL certificates. SSL can be terminated at the load balancer or can be enabled all the way from the external clients to the backend application servers.

The SSL protocol (HTTPS/LDAPS) is used for the communication between the clients and the front-end load balancer, the front-end load balancer and Oracle HTTP Servers (OHS), and the Oracle HTTP Servers and the WebLogic Servers and LDAP directories.

For these communications to take place, each access point (whether the frontend load balancer, Oracle HTTP Server or Oracle WebLogic Server) needs to certify its identity and provide a public key to encrypt traffic. This is done using the appropriate SSL certificate used at each access point. These SSL certificates are typically provided by commercial Certificate Authorities (CAs). It is expected that in a real enterprise deployment, the certificates used by each endpoint are issued by one of these commercial CAs. Since each Certificate Authority has its own mechanism to issue a SSL certificate, it is out of the scope of this Enterprise Deployment Guide to use and describe such processes. However, to provide SSL encryption between the different components across tiers, this guide provides SSL configuration steps using a self-issued Certificate Authority. This approach is considered secure and solid enough since the endpoints involved (OHS and WLS listeners) reside behind their corresponding firewalls and are not really exposed to the public. The front-end load balancer is considered a much more exposed endpoint and it is expected in all cases that users acquire the load balancer certificate from a commercial Certificate Authority.

For more information, see the following topics:

- Understanding SSL
- Types of Certificate
- Keystores and Truststores
- Using Certificates in WebLogic Servers
- Creating a Certificate Authority (CA)
- Creating Certificates
- Creating a PKCS12 Keystore
- Creating a PKCS12 Truststore

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Certificates Used in this Guide

Understanding SSL

There are two methods of implementing SSL communication: One-way authentication where a client establishes trust to a server, and two-way authentication where a client establishes trust to a server and the server in turn establishes trust to the client. This guide utilizes one-way SSL authentication.

Every communication in a system has a client and a server for the purposes of authentication. A given component may act as both a client and a server depending on the action it is performing, for example:

A Web-browser is always a client. A user types in a URL to access a web-resource. The Web-browser contacts a load balancer, which in this case is acting as a server. Once the load balancer has received the request it changes role and becomes a client whilst it transfers the request to an Oracle HTTP server, whose role in this transaction is a server. The Oracle HTTP server becomes a client whilst it interacts with a WebLogic Server which is the client, and so on.

In one-way SSL a client will contact a server and initiate a SSL handshake. After establishing a common SSL protocol and ciphersuite, the server will send the SSL certificate to the client. The client will then look at its truststore and see if it has a Certificate Authority (CA) which it can use to verify the SSL certificate that has been presented to it. The client verifies the certificate by checking the validity date, that the Common Name (CN) in the certificate matches the hostname of the server, and the digital signature is valid. If all the certificate checks pass, the transaction will be allowed to proceed. Some clients such as browsers, will report that a handshake has failed and provide you the option to manually trust the certificate. This commonly happens with self-issued Certificate Authorities which are not automatically trusted by browsers.

> **Note**
>
> The majority of commercially issued Certificate Authorities are loaded into a browsers truststore automatically.

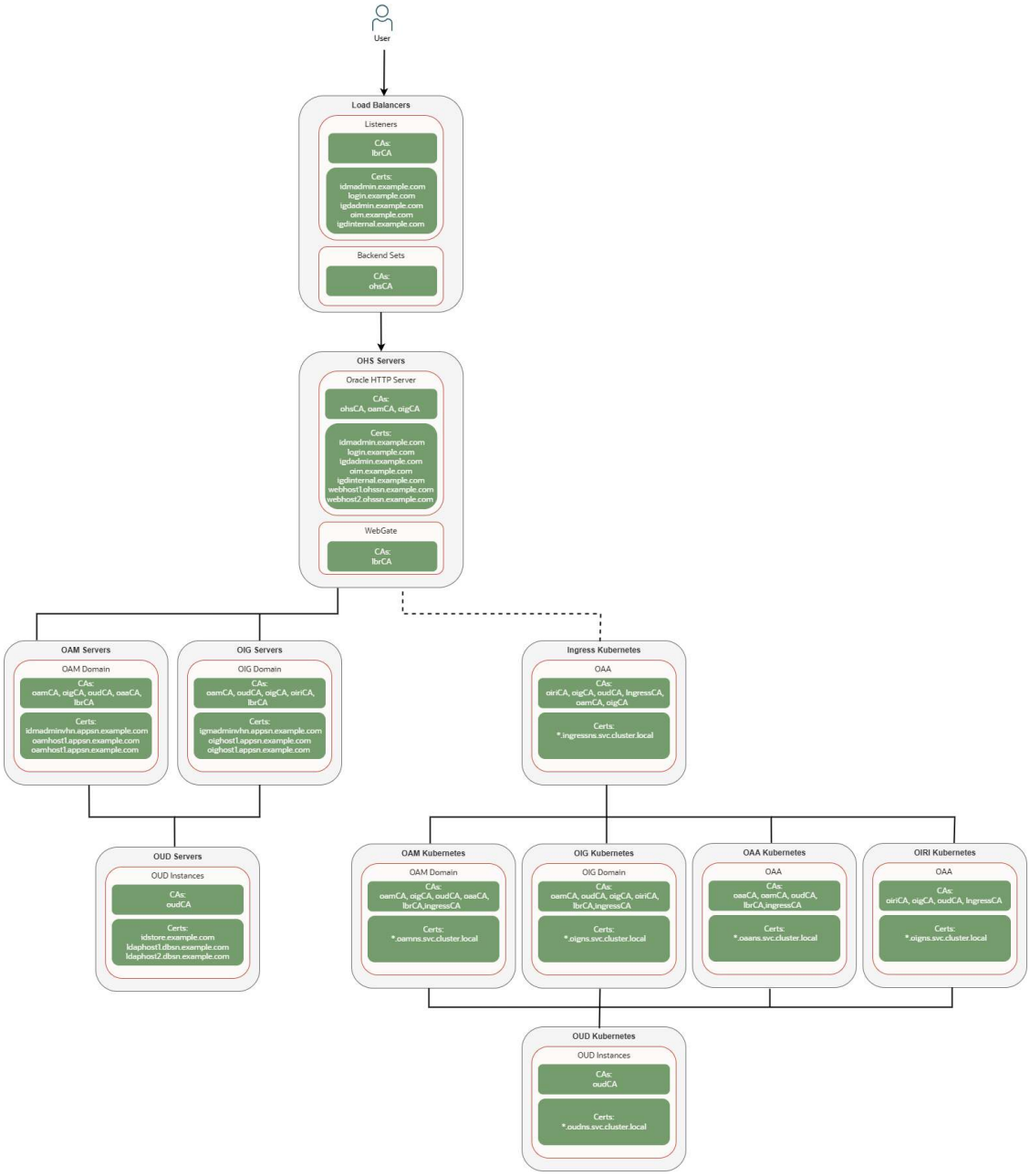
Every client has an SSL truststore. This truststore contains a list of Certificate Authorities that the client will trust. The more Certificate Authorities you have issuing certificates, the more entries you need in a truststore. If you are issuing your own certificates for a deployment, then this needs to be considered.

When using self-issued certificates in an identity management deployment, it is recommended that a single Certificate Authority is used to issue all certificates in the identity management deployment. This means only one Certificate Authority needs to be trusted by each component. If you are using self-issued certificates for your internal components and a commercial certificate for your load balancer, you will need to trust both Certificate Authorities.

The diagram below shows how you need to trust the Certificate Authorities of a typical identity management deployment:

Copyright © 2018, 2025, Oracle and/or its affiliates.

Figure 5-1 Certificates in a Typical Identity and Access Management Topology



Types of Certificate

There are 3 types of SSL certificates:

- Host based certificates – Host based certificates are certificates issued to a single host name. They are the most granular, but every new host needs a new certificate created. The more certificates you have the greater the administrative overhead.
- Wildcard certificates – Wildcard certificates are certificates which are issued for a specific domain or subnet. For example `*.example.com` can be used for `host1.example.com` and `host2.example.com`. It is important to note that wildcard certificates are not recursive so

Copyright © 2018, 2025, Oracle and/or its affiliates.

`*.example.com` could not be used for `host1.subnet1.example.com` and `host2.subnet2.example.com` - in this situation you would need two wildcard certificates `*.subnet1.example.com` and `*.subnet2.example.com`. Wildcard certificates are easy to manage but can be a bit vague. They are perfect for Kubernetes deployments where you can create a wildcard certificate per namespace.

- SAN based certificates – SAN (Subject Alternative Name) certificates are certificates which can have a list of specified hosts not necessarily in the same subnet. These certificates are a great compromise because they are easy to manage but not too encompassing.

> **Note**
>
> Some products, namely LDAP and OHS, require SAN certificates which have both the specific host name and virtual host name included, for example: `login.example.com` and `ohs1.ohssn.com`.

> **Note**
>
> In an enterprise deployment where you are using commercial certificates for the load balancer and internal certificates elsewhere, you may need to have two certificates for a specific host name, for example `login.example.com` being the commercial certificate used by the load balancer, and `login.example.com` being a self-issued certificate issued for the Oracle HTTP Server.

Keystores and Truststores

A keystore is a collection of certificates stored in a single location. A truststore is a collection of Certificate Authorities stored in a single location.

Stores can either be stored in a file, in a database, or a certificate management system. Using a certificate management system can make certificate reissuing easier.

Stores in files can use several formats, for example:

- JKS - Java Keystore.
- PKCS12 – Public-Key Cryptography Standard, a more modern standard used across many platforms.
- Wallets – A bespoke store used by products such as Oracle HTTP Server.

This guide uses PKCS12 stores for most of the products and a wallet for Oracle HTTP Server.

Using Certificates in WebLogic Servers

The following guidelines outline using custom or third party SSL certificates with the WebLogic Servers:

- The SSL certificate used by each WebLogic server (identity key, private key) must be issued to that server's listen address. For example, if the server oam_server1 listens on `oamhost1.example.com`, the CN of its SSL certificate must be that hostname or the wildcard name valid for that hostname.
- Oracle recommends using an identity keystore that is shared by all the servers in the same domain, where you import all the private keys used by the different WebLogic servers, each mapped to a different alias.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Oracle recommends using a trust keystore shared by all the servers in the domain. You must import the Certificate Authority's (CA) certificate, and intermediate CA's if needed, into this trust keystore.
- You must specify the identity keystore, alias of the identity key, and the trust keystore for each WebLogic server in the WebLogic domain's configuration. Use the WebLogic Remote Console to configure these SSL settings for each server.
- Start the WebLogic servers using the appropriate java options to point to the trusted keystore. This is required so the WebLogic servers can communicate with external SSL endpoints that use the Certificate Authorities included in such a trust store.

Creating a Certificate Authority (CA)

A Certificate Authority (CA) is used to issue certificates. The CA is also placed in the truststore of clients interacting with those certificates. To create a Certificate Authority, use the following commands:

```
openssl genrsa -out idmCA.key 4096
```

```
openssl req -x509 -new -key idmCA.key -out idmCA.crt -subj
"C=<CountryName>/ST=<StateOrProvinceName>/L=<Locality>/O=<Organization>/
CN=IDM Certificate Authority" -days 50000
```

For example:

```
openssl req -x509 -new -key idmCA.key -out idmCA.crt -subj "/C=US/ST=CA/
L=Redwood Shores/O=Oracle/CN=IDM Certificate Authority" -days 50000
```

Creating Certificates

The following topics explain how to create the different type of certificates:

- Creating a Host Based Certificate
- Creating a SAN Based Certificate
- Creating a Wildcard Based Certificate

Creating a Host Based Certificate

To create a host-based certificate perform the following steps:

1. Export the following environment variable substituting `myhost.example.com` with the required hostname:

   ```
   export host=myhost.example.com
   ```

2. Generate a private key:

   ```
   openssl genrsa -out $host.key 4096
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

3. Create a Certificate Signing Request (CSR):

```
openssl req -new -key $host.key -out $host.csr -subj "/C=<CountryName>/
ST=<StateOrProvinceName>/L=<Locality>/O=<Organization>/CN=$host"
```

For example:

```
openssl req -new -key $host.key -out $host.csr -subj "/C=US/ST=California/
L=Redwood Shores/O=Oracle/CN=$host"
```

> **Note**
>
> If you wish to enter the subject interactively you can do so without specifying the `-subj` clause.

4. Sign the CSR with the Certificate Authority (CA):

```
openssl x509 -req -in $host.csr -CA idmCA.crt -CAkey idmCA.key -
CAcreateserial -out $host.crt -days 30000
```

Creating a SAN Based Certificate

A SAN based certificate includes multiple hosts in the same certificate. You can create multiple SAN certificates for individual components, or you can create a single SAN certificate to encompass all of the hosts and virtual hosts in your deployment.

> **Note**
>
> Even if you are using host-based deployments you need to create the following SAN certificates:
>
> - An LDAP certificate that contains the host name and the virtual host name, for example `ldaphost1.example.com` and `idstore.example.com`.
> - An OHS certificate containing the host name and the virtual host name, for example `webhost1.example.com` and `login.example.com`.

To create a SAN based certificate perform the following steps:

1. Export the following environment variable:

```
export cert_name=idmCerts
```

2. Create a certificate configuration file called `$cert_name.cnf` with the following contents:

Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> Change the `[ req_distinguished_name]` parameters accordingly. Ensure that all the hostnames you wish to include are in the `alt_names` section.

```
[ req ]
default_bits           = 4096
distinguished_name     = req_distinguished_name
v3_reqensions         = v3_req

[ req_distinguished_name ]
countryName            = <CountryName>
stateOrProvinceName    = < StateOrProvinceName>
localityName           = <Locality>
organizationName       = <Organization>
commonName             = ldaphost1.dbsn.example.com

[ v3_req ]
keyUsage = critical, digitalSignature, nonRepudiation, keyEncipherment,
dataEncipherment, keyAgreement, keyCertSign, cRLSign
extendedKeyUsage = serverAuth, clientAuth
subjectAltName = @alt_names

[alt_names]

DNS.1   = ldaphost1.dbsn.example.com
DNS.2   = ldaphost2.db.example.com
DNS.3   = oamhost1.appsn.example.com
DNS.4   = oamhost2.appsn.example.com
DNS.5   = oighost1.appsn.example.com
DNS.6   = oighost1.appsn.example.com
DNS.7   = webhost1.ohssn.example.com
DNS.8   = webhost2.ohssn.example.com
DNS.9   = iadadminvhn.ohssn.example.com
DNS.10   = igdadminvhn.ohssn.example.com
DNS.11   = idstore.example.com
DNS.12   = iadadmin.example.com
DNS.13   = igdadmin.example.com
DNS.14   = login.example.com
DNS.15   = oig.example.com
DNS.16   = igdinternal.example.com
```

3. Generate a Private Key:

   ```
   openssl genrsa -out $host.key 4096
   ```

   This will create a file called out `$cert_name.key`.

4. Create a Certificate Signing Request (CSR) using the certificate configuration file:

   ```
   openssl req -new -key $cert_name.key -out $cert_name.csr -subj "/
   C=<CountryName>/ST=<StateOrProvinceName>/L=<Locality>/O=<Organization>/
   CN=$host" -config $cert_name.cnf -addext "subjectAltName = DNS:$cert_name"
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

For example:

```
openssl req -new -key $cert_name.key -out $cert_name.csr -subj "/C=US/
ST=California/L=Redwood Shores/O=Oracle/CN=$cert_name" -
config $cert_name.cnf -addext "subjectAltName = DNS:$cert_name"
```

This will create a file called `$cert_name.csr`.

5. Sign the CSR with the Certificate Authority (CA):

```
openssl x509 -req -in $cert_name.csr -CA idmCA.crt -CAkey idmCA.key -
CAcreateserial -out $cert_name.crt -days 30000 -setalias $cert_name -
extfile $cert_name.cnf -extensions v3_req
```

This will create a file called `$cert_name.crt`. This file will be in Privacy Enhanced Mail (PEM) format.

6. Verify that each host is included in the certificate using the command:

```
openssl x509 -text -noout -in $cert_name.crt -certopt
no_subject,no_header,no_version,no_serial,no_signame,no_validity,no_issuer,
no_pubkey,no_sigdump,no_aux
```

Creating a Wildcard Based Certificate

To create a wildcard based certificate perform the following steps:

1. Export the following environment variable:

```
export cert_name=examplecom
```

2. Create a certificate configuration file called `$cert_name.cnf` with the following contents:

> **Note**
>
> Change the `[ req_distinguished_name]` parameters accordingly.

```
[req]
default_md = sha256
prompt = no
req_extensions = req_ext
distinguished_name = req_distinguished_name

[req_distinguished_name]
commonName = *.example.com
countryName = <CountryName>
stateOrProvinceName = <StateOrProvince>
localityName = <Locality>
organizationName = <Organization>

[req_ext]

keyUsage=critical,digitalSignature,keyEncipherment
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
extendedKeyUsage=critical,serverAuth,clientAuth
subjectAltName = @alt_names
[alt_names]
DNS.1=example.com
DNS.2=*.example.com
```

3. Generate a Private Key:

   ```
   openssl genrsa -out $host.key 4096
   ```

   This will create a file called out `$cert_name.key`.

4. Create a Certificate Signing Request (CSR) from the private key:

   ```
   openssl req -new -nodes -$cert_name.key -config $cert_name.cnf -
   out $cert_name.csr
   ```

   This will create a file called `$cert_name.csr`.

5. Sign the CSR with the Certificate Authority (CA):

   ```
   openssl x509 -req -in $cert_name.csr -CA idmCA.crt -CAkey idmCA.key -
   CAcreateserial -out $cert_name.crt -days 30000 -setalias $cert_name -
   extfile $cert_name.cnf -extensions req_ext
   ```

   This will create a file called `$cert_name.crt`. This file will be in Privacy Enhanced Mail (PEM) format.

6. Verify that each host is included in the certificate using the command:

   ```
   openssl x509 -text -noout -in $cert_name.crt -certopt
   no_subject,no_header,no_version,no_serial,no_signame,no_validity,no_issuer,
   no_pubkey,no_sigdump,no_aux
   ```

Creating a PKCS12 Keystore

A keystore contains a private key, certificates and Certificate Authorities (CA's). You can create a keystore using the following commands:

1. Export the following environment variables:

   > **Note**
   >
   > Change `password` to a password of your choice.

   ```
   export cert_name=idmcerts.crt
   export keystorepwd=password
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> Keystore Passwords MUST be unique to keystores, DO NOT use a password that you are planning on using for user accounts.

2. Create the keystore:

```
openssl pkcs12 -export -out $cert_name.p12 -inkey $cert_name.key -
in $cert_name.crt -chain -CAfile idmCA.crt -passout pass:$keystorepwd -
name $cert_name
```

   This will create a keystore file called `$cert_name.p12`. This file will be in PKCS12 format.

3. Oracle recommends that WebLogic Domains use a single keystore with each of the host or SAN based certificates contained within. Each certificate is identified by an alias. For host based certificates this is the hostname. For SAN based certificates this is an arbitrary name such as `idmcert`.

   To create a keystore with all of the certifcates, you import each of the pkcs12 certifcates you created above one after the other, for example:

```
keytool -importkeystore -srckeystore oamhost1.example.com.p12 -
srcstoretype PKCS12 -destkeystore idmcerts.p12 -deststoretype PKCS12 -
storepass <password_of_keystore> -srcstorepass
<password_of_source_p12_file>
```

4. Repeat the above step for each certificate to be added.

Creating a PKCS12 Truststore

A truststore contains all the Certificate Authorities (CA's) that you wish to trust. In a deployment where each of the certificates are created using the same Certificate Authority (CA), this will contain:

- The idmCA.
- Any CA's and Intermediate CA's associated with your load balancer These may be self-signed CA's or commercial CA's.

The simplest way to create the truststore is using the keytool command.

1. For each certificate you wish to add to the truststore, issue the following commands:

```
keytool -import -keystore idmTrustStore.p12 -alias idmCA -cert -rfc -file
idmCA.crt -storepass <password> -storetype pkcs12 -noprompt
```

   Where `-alias imdCA` is the user friendly name for the certificate, and `-storepass <password>` is a password for the keystore.

> **Note**
>
> Keystore Passwords MUST be unique to keystores, DO NOT use a password that you are planning on using for user accounts.

2. Repeat the above command for each CA certificate you wish to add to the truststore.

Copyright © 2018, 2025, Oracle and/or its affiliates.

3. Verify the contents of the keystore by issuing the command:

```
openssl pkcs12 -info -in idmTrustStore.p12 -password pass:<password>
```

Certificates Used in this Guide

To keep things as simple as possible, this guide assumes that you have the following certificates and Certificate Authorities (CA's):

- One CA for the load balancer which is commercially issued.
- One CA for the Identity Management Suite
- Two SAN certificates for the load balancer
- One SAN certificate for the rntire Identity Management Suite, or certificates for each host in the Oracle Identity Management topology.

The following table outlines all the certificates used in an enterprise deployment:

Table 5-5 Certificates Used in an IDM Enterprise Deployment

Purpose	Alias	Certificate Authority	SAN Certificate (Host Names)
Public Load Balancer		lbrCA	`login.example.com` `oig.example.com`

Private Load Balancer		lbrCA	`iadadmin.example.com`
			`igdadmin.example.com`
			`login.example.com`
			`oig.example.com`
			`igdinternal.example.com`

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 5-5 (Cont.) Certificates Used in an IDM Enterprise Deployment

Purpose	Alias	Certificate Authority	SAN Certificate (Host Names)

Identity Management Suite	idmcert	idmCA	`iadadmin.example.com`
			`igdadmin.example.com`
			`login.example.com`
			`oig.example.com`
			`igdinternal.example.com`
			`idstore.example.com`
			`webhost1.ohssn.example.com`
			`webhost2.ohssn.example.com`
			`ldaphost1.dbsn.example.com`
			`ldaphost2.dbsn.example.com`
			`iadadminvh.appsn.example.com`
			`oamhost1.appsn.example.com`
			`oamhost2.appsn.example.com`
			`igdadminvh.appsn.example.com`
			`oighost1.appsn.example.com`
			`oighost2.appsn.example.com`

idmTrustStore		idmCA lbrCA	

> **Note**
>
> Alias is the alias assigned to the certificate when placed into a keystore.

An alternative approach would be to use per host based certificates in which case you would have:

Table 5-6 Per Host Certificates Used in an IDM Enterprise Deployment

Purpose	Alias	Certificate Authority	SAN Certificate (Host Names)
Public Load Balancer		lbrCA	`login.example.com` `oig.example.com`

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 5-6 (Cont.) Per Host Certificates Used in an IDM Enterprise Deployment

Purpose	Alias	Certificate Authority	SAN Certificate (Host Names)
Private Load Balancer		IbrCA	iadadmin.example.com igdadmin.example.com login.example.com oig.example.com igdinternal.example.com
ldaphost1	ldaphost1	idmCA	idstore.example.com ldaphost1.example.com
ldaphost2	ldaphost2	idmCA	idstore.example.com ldaphost2.example.com
oamhost1	oamhost1	idmCA	iadadminvhn.example.com oamhost1.example.com
oamhost2	oamhost2	idmCA	iadadminvhn.example.com oamhost2.example.com
oighost1	oighost1	idmCA	igdadminvhn.example.com oighost1.example.com
oighost2	oighost2	idmCA	igdadminvhn.example.com oighost2.example.com
webhost1		idmCA	iadadmin.example.com webhost1.example.com
webhost1		idmCA	igdadmin.example.com webhost1.example.com
webhost1		idmCA	login.example.com webhost1.example.com
webhost1		idmCA	oig.example.com webhost1.example.com
webhost1		idmCA	igdinternal.example.com webhost1.example.com
webhost2		idmCA	iadadmin.example.com webhost2.example.com
webhost2		idmCA	igdadmin.example.com webhost2.example.com
webhost2		idmCA	login.example.com webhost2.example.com
webhost2		idmCA	oig.example.com webhost2.example.com
webhost2		idmCA	igdinternal.example.com webhost2.example.com
idmTrustStore		idmCA IbrCA	

Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> Alias is the alias assigned to the certificate when placed into a keystore.

Certificates Used in a Kubernetes Deployment

To keep things as simple as possible, in a Kubernetes deployment the following Certificates and Certificate Authorities are recommended:

- One CA for the load balancer which is commercially issued.
- One CA for the Identity Management Suite.
- Wild card certificates for each namespace.

Purpose	Alias	Certificate Authority	SAN Certificate / Wild Card Names
Public Load Balancer		lbrCA	`login.example.com` `oig.example.com`

Private Load Balancer		lbrCA	`iadadmin.example.com`
			`igdadmin.example.com`
			`login.example.com`
			`oig.example.com`
			`igdinternal.example.com`

webhost1		idmCA	`iadadmin.example.com` `webhost1.example.com`
webhost1		idmCA	`igdadmin.example.com` `webhost1.example.com`
webhost1		idmCA	`login.example.com` `webhost1.example.com`
webhost1		idmCA	`oig.example.com` `webhost1.example.com`
webhost1		idmCA	`igdinternal.example.com` `webhost1.example.com`
webhost2		idmCA	`iadadmin.example.com` `webhost2.example.com`

Copyright © 2018, 2025, Oracle and/or its affiliates.

Purpose	Alias	Certificate Authority	SAN Certificate / Wild Card Names
webhost2		idmCA	igdadmin.example.com webhost2.example.com
webhost2		idmCA	login.example.com webhost2.example.com
webhost2		idmCA	oig.example.com webhost2.example.com
webhost2		idmCA	igdinternal.example.com webhost2.example.com
idmTrustStore		idmCA lbrCA	
Ingress Namespace		idmCA	WildCard: *.ingressns.svc.cluster.local
OUD Namespace		idmCA	WildCard: *.oudns.svc.cluster.local
OAM Namespace		idmCA	WildCard: *.oamns.svc.cluster.local
OIG Namespace		idmCA	WildCard: *.oigns.svc.cluster.local

Copyright © 2018, 2025, Oracle and/or its affiliates.

6

Download and Configure the WebLogic Remote Console

This chapter describes how to download and configure the WebLogic Remote Console.

> **Note**
>
> For the initial configuration steps required in this Enterprise Deployment Guide, you will need access to the Administration Server listen address and administration port. Later on you will configure access from a frontend load balancer.

Perform the following steps to download and configure the WebLogic Remote Console:

1. Uninstall any previous versions of the WebLogic Remote Console from your computer.
2. Download the WebLogic Remote Console for your Operating System from https://github.com/oracle/weblogic-remote-console/releases.
3. Run the installer.
4. Install the WebLogic Remote Console extension in the WebLogic Server domain. The WebLogic Remote Console extension provides additional functionality when using the WebLogic Remote Console to manage WebLogic domains.

   > **Note**
   >
   > This step is optional.

   a. Create a `management-services-ext` directory under the domain home.
   b. Download the latest WebLogic Remote Console extension, `console-rest-ext-<version>.war`, from https://github.com/oracle/weblogic-remote-console/releases and save it inside the `management-services-ext` directory you created in the previous step. If you have an earlier version of the extension already downloaded, delete it and replace it with the latest version.
   c. Reboot the Administration Server if it is already running.
5. Launch the WebLogic Remote Console application. For example:

   ```
   ./weblogic-remote-console
   ```

   After configuring OAM and OIG later in this guide, you must connect to the domain provider using the Administration Servers listen address.

Copyright © 2018, 2025, Oracle and/or its affiliates.

7

Preparing the Load Balancer and Firewalls for an Enterprise Deployment

It is important to understand how to configure the external load balancer and ports that must be opened on the firewalls for an enterprise deployment.

- Configuring Virtual Hosts on the External Load Balancer
  The external load balancer configuration facilitates to recognize and route requests to several virtual servers and associated ports for different types of network traffic and monitoring.
- Configuring the Firewalls and Ports for an Enterprise Deployment
  As an administrator, it is important that you become familiar with the port numbers that are used by various Oracle Fusion Middleware products and services. This ensures that the same port number is not used by two services on the same host, and that the proper ports are open on the firewalls in the enterprise topology.

Configuring Virtual Hosts on the External Load Balancer

The external load balancer configuration facilitates to recognize and route requests to several virtual servers and associated ports for different types of network traffic and monitoring.

The following topics explain how to configure the external load balancer, provide a summary of the virtual servers that are required, and provide additional instructions for these virtual servers:

- Overview of the External Load Balancer Configuration
- Typical Procedure for Configuring the External Load Balancer
- Load Balancer Health Monitoring
- Summary of the Virtual Servers Required for an Enterprise Deployment

Overview of the External Load Balancer Configuration

As shown in the topology diagrams, you must configure the external load balancer to recognize and route requests to several virtual servers and associated ports for different types of network traffic and monitoring.

In the context of a load-balancing device, a virtual server is a construct that allows multiple physical servers to appear as one for load-balancing purposes. It is typically represented by an IP address and a service, and it is used to distribute incoming client requests to the servers in the server pool.

The virtual servers should be configured to direct traffic to the appropriate host computers and ports for the various services that are available in the enterprise deployment.

In addition, you should configure the load balancer to monitor the host computers and ports for availability so that the traffic to a particular server is stopped as soon as possible when a service is down. This ensures that incoming traffic on a given virtual host is not directed to an unavailable service in the other tiers. At the same time, this monitoring should not overload the

Copyright © 2018, 2025, Oracle and/or its affiliates.

backend system with too frequent health requests. In the end, a trade off needs to be made between how fast the death detection occurs and how much overhead is introduced on the systems that are monitored.

Note that after you configure the load balancer, you can later configure the web server instances in the web tier to recognize a set of virtual hosts that use the same names as the virtual servers that you defined for the load balancer. For each request coming from the external load balancer, the web server can then route the request appropriately, based on the server name included in the header of the request. See Configuring Oracle HTTP Server for Administration and Oracle Web Services Manager.

Typical Procedure for Configuring the External Load Balancer

The following procedure outlines the typical steps for configuring a external load balancer for an enterprise deployment.

Note that the actual procedures for configuring a specific load balancer will differ, depending on the specific type of load balancer. There may also be some differences depending on the type of protocol that is being load balanced. For example, TCP virtual servers and HTTP virtual servers use different types of monitors for their pools. Refer to the vendor-supplied documentation for actual steps.

1. Create a pool of servers. This pool contains a list of servers and the ports that are included in the load-balancing definition.

   For load balancing between the web hosts, create a pool of servers that would direct requests to hosts WEBHOST1 and WEBHOST2 to each port used in the OHS. For example, a pool to WEBHOST1 and WEBHOST2 to port 4443 for access to applications like OAM and OIG, a pool to WEBHOST1 and WEBHOST2 to port 4446 for internal accesses, a pool to WEBHOST1 and WEBHOST2 to port 4444 for access to OAM admin consoles, and a pool to WEBHOST1 and WEBHOST2 to port 4445 for access to OIG admin consoles.

   If you are using SSL Termination, then create a pool of servers that would direct requests to hosts WEBHOST1 and WEBHOST2 on port 7777.

2. Create rules to determine whether a given host and service is available and assign it to the pool of servers that are described in Step 1.

3. Create the required virtual servers on the load balancer for the addresses and ports that receive requests for the applications.

   For a complete list of the virtual servers required for the enterprise deployment, see Summary of the Virtual Servers Required for an Enterprise Deployment.

   When you define each virtual server on the load balancer, consider the following:

   a. If your load balancer supports it, specify whether the virtual server is available internally, externally, or both. Ensure that internal addresses are only resolvable from inside the network.

   b. Configure SSL Termination, if applicable, for the virtual server.

   c. Configure SSL for the virtual server for end-to-end SSL.

   d. Configure SSL for the communication with the pool of servers for end-to-end SSL.

      Some load balancers may need to be provided with the backend's certificate (the SSL certificate used by the OHS listeners in the backend pool) to establish the appropriate SSL communication. In that case you may need to add the OHS's CA certificate to the load balancer as a trusted certificate. Since this guide uses example certificates based on the WebLogic per-domain CA, you can add this after the domain is created.

Copyright © 2018, 2025, Oracle and/or its affiliates.

e. Assign the pool of servers created in Step 1 to the virtual server.

Load Balancer Health Monitoring

The load balancer must be configured to check that the services in the Load Balancer Pool are available. Failure to do so will result in requests being sent to hosts where the service is not running.

The following table shows examples of how to determine whether a service is available:

Table 7-1 Examples Showing How to Determine Whether a Service is Available

Service	Monitor Type	Monitor Mechanism
OUD	ldap	ldapbind to cn=oudadmin
OHS	http	check for GET /healtcheck.html\r\n

Summary of the Virtual Servers Required for an Enterprise Deployment

This topic provides details of the virtual servers that are required for an enterprise deployment.

The following table provides a list of the virtual servers that you must define on the external load balancer for the Oracle Identity and Access Management enterprise topology:

Virtual Host	Server Pool (SSL Terminated)	External	Server Pool (end to end SSL)	Other Required Configuration/ Comments

`login.example.com:443`	`webhost1.example.com:7777`
	`webhost2.example.com:7777`
		Yes	`webhost1.example.com:4443`
			`webhost2.example.com:4443`
				Identity Management requires that the following be added to the HTTP header:
				`Header Name: IS_SSL`
				`Header Value: ssl`
				`Header Name: WL-Proxy-SSL`
				`Header Value: true`

`oig.example.com:443`	`webhost1.example.com:7777`
	`webhost2.example.com:7777`
		Yes	`webhost1.example.com:4444`
			`webhost2.example.com:4444`
				Identity Management requires that the following be added to the HTTP header:
				`Header Name: IS_SSL`
				`Header Value: ssl`
				`Header Name: WL-Proxy-SSL`
				`Header Value: true`

`iadadmin.example.com:80`	`webhost1.example.com:7777` `webhost2.example.com:7777`	No	`webhost1.example.com:4445` `webhost2.example.com:4445`	

Copyright © 2018, 2025, Oracle and/or its affiliates.

Virtual Host	Server Pool (SSL Terminated)	External	Server Pool (end to end SSL)	Other Required Configuration/ Comments
igdadmin.example.com:80	webhost1.example.com:7777 webhost2.example.com:7777	No	webhost1.example.com:4446 webhost2.example.com:4446	
igdinternal.example.com:7777	webhost1.example.com:7777 webhost2.example.com:7777	No	webhost1.example.com:4447 webhost2.example.com:4447	
idstore.example.com:1389	ldaphost1.example.com:1389 ldaphost2.example.com:1389	No		
idstore.example.com:1636	ldaphost1.example.com:1636 ldaphost2.example.com:1636	No		

Additional Instructions for iadadmin.example.com and igdadmin.example.com

This section provides additional instructions that are required for the virtual server `iadadmin.example.com` and `igdadmin.example.com`.

Perform the following steps when you configure this virtual server on the load balancer:

- Enable address and port translation.
- Enable reset of connections when services or hosts are down.

Additional Instructions for login.example.com and oig.example.com

Perform the following steps when you configure this virtual server on the external load balancer:

- Use port 80 and port 443. Any request that is directed to port 80 (non-SSL protocol) should be redirected to port 443 (SSL protocol).
- Enable address and port translation.
- Enable reset of connections when services and nodes are down.
- Create rules to filter out access to /management and /em on this virtual server.

  These context strings direct requests to the WebLogic Remote Console and to the Oracle Enterprise Manager Fusion Middleware Control and must be used only when you access the system from `iadadmin.example.com` and `igdadmin.example.com`.

> **Note**
>
> Oracle recommends that you configure LBR for cookie-based persistence because session persistence is required for some web applications of SOA, such as BPM Worklist (/integration/worklistapp), SOA Composer (/soa/composer), BPM Composer (/bpm/composer), BPM Workspace (/bpm/workspace), and so on.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Additional Instructions for igdinternal.example.com

Perform the following steps when you configure this virtual server on the external load balancer:

- Enable address and port translation.
- Enable reset of connections when services or nodes are down.

As with the `login.example.com`, create rules to filter out access to /console and /em on this virtual server.

Configuring the Firewalls and Ports for an Enterprise Deployment

As an administrator, it is important that you become familiar with the port numbers that are used by various Oracle Fusion Middleware products and services. This ensures that the same port number is not used by two services on the same host, and that the proper ports are open on the firewalls in the enterprise topology.

The following tables lists the ports that you must open on the firewalls in the topology:

Firewall notation:

- FW0 refers to the outermost firewall.
- FW1 refers to the firewall between the web tier and the application tier.
- FW2 refers to the firewall between the application tier and the data tier.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 7-2 Firewall Ports for SSL Terminated Enterprise Deployments

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Browser request	FW0	80	HTTP / Load Balancer **Note** You need this option only if redirection from port 8	Inbound	Timeout depends on the size and type of HTML content.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 7-2 (Cont.) Firewall Ports for SSL Terminated Enterprise Deployments

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
			0 to port 443 is used.		
Browser request	FW0	443	HTTPS / Load Balancer	Inbound	Timeout depends on the size and type of HTML content.
Browser request	FW1	443	HTTPS / Load Balancer	Outbound (for intranet clients)	Timeout depends on the size and type of HTML content.
Callbacks and Outbound invocations	FW1	443	HTTPS / Load Balancer	Outbound	Timeout depends on the size and type of HTML content.
Load balancer to Oracle HTTP Server	n/a	7777	HTTP	n/a	n/a
Session replication within a WebLogic Server cluster	n/a	n/a	n/a	n/a	By default, this communication uses the same port as the server's listen address.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 7-2 (Cont.) Firewall Ports for SSL Terminated Enterprise Deployments

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
WebLogic Remote Console and Enterprise Manager Console	FW1	7001	HTTP/t3	Both	You should tune this timeout based on the type of access to the Remote console (whether you plan to use the Oracle WebLogic Remote Console from the application tier clients or clients external to the application tier).
OAM Server Access	FW1	14100	HTTP	Inbound	Set the timeout to a short period (5-10 seconds).
OAM Policy Manager	FW1	14150	HTTP	Inbound	Set the timeout to a short period (5-10 seconds).
OIG Administration Console Access	FW1	7101	HTTP	Inbound	Set the timeout to a short period (5-10 seconds).
OIM Server Access	FW1	14000	HTTP	Inbound	Set the timeout to a short period (5-10 seconds).
SOA Server Access	FW1	7003	HTTP	Inbound	Set the timeout to a short period (5-10 seconds).
Database access	FW2	1521	SQL*Net	Both	Timeout depends on database content and on the type of process model used for SOA.
Oracle Notification Server (ONS)	FW2	6200	ONS	Both	Required for Gridlink. An ONS server runs on each database server.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 7-2 (Cont.) Firewall Ports for SSL Terminated Enterprise Deployments

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Coherence for deployment	n/a	9991 Coherence requires the following connectivity between members: • Port 9991 for both UDP and TCP for both multicast and unicast configurations. • TCP port 7. • Ephemereal ports 32768-60999 for both udp and tcp.	n/a	n/a	n/a
Oracle Unified Directory access	FW2	1389	LDAP	Inbound	You should tune the directory server's parameters based on load balancer, and not the other way around.

Table 7-3 Firewall Ports for End to End SSL Enterprise Deployments

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Browser request	FW0	443	HTTPS / Load Balancer	Inbound	Timeout depends on the size and type of HTML content.
Browser request	FW1	443	HTTPS / Load Balancer	Outbound (for intranet clients)	Timeout depends on the size and type of HTML content.
Callbacks and Outbound invocations	FW1	443	HTTPS / Load Balancer	Outbound	Timeout depends on the size and type of HTML content.
Load balancer to Oracle HTTP Server	n/a	4445-4449	HTTPS	n/a	n/a

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 7-3 (Cont.) Firewall Ports for End to End SSL Enterprise Deployments

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
OHS registration with Administration Server	FW1	7002	HTTPS / t3s	Inbound	Set the timeout to a short period (5-10 seconds).
OHS management by Administration Server	FW1	OHS Admin Port (7779)	TCP / HTTP	Outbound	Set the timeout to a short period (5-10 seconds).
Session replication within a WebLogic Server cluster	n/a	n/a	n/a	n/a	By default, this communication uses the same port as the server's listen address.
OAM Administration Console access	FW1	9001	HTTPS / Administration Server and Enterprise Manager https/t3s	Both	You should tune this timeout based on the type of access to the WebLogic Remote Console (whether you plan to use the WebLogic Remote Console from the application tier clients or clients external to the application tier).
OAM Server Access	FW1	14101	HTTPS	Inbound	Set the timeout to a short period (5-10 seconds).
OAM Policy Manager	FW1	14151	HTTPS	Inbound	Set the timeout to a short period (5-10 seconds).
OIG Administration Console Access	FW1	9102	HTTPS	Inbound	Set the timeout to a short period (5-10 seconds).
OIM Server Access	FW1	14001	HTTPS	Inbound	Set the timeout to a short period (5-10 seconds).
SOA Server Access	FW1	7004	HTTPS	Inbound	Set the timeout to a short period (5-10 seconds).
Database access	FW2	1521	SQL*Net	Both	Timeout depends on database content and on the type of process model used for SOA.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 7-3 (Cont.) Firewall Ports for End to End SSL Enterprise Deployments

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Oracle Notification Server (ONS)	FW2	6200	ONS	Both	Required for Gridlink. An ONS server runs on each database server.
Coherence for deployment	n/a	9991 Coherence requires the following connectivity between members: • Port 9991 for both UDP and TCP for both multicast and unicast configurations. • TCP port 7. • Ephemereal ports 32768-60999 for both udp and tcp.	n/a	n/a	n/a
Oracle Unified Directory access	FW2	1636	LDAPS	Inbound	You should tune the directory server's parameters based on load balancer, and not the other way around.

Copyright © 2018, 2025, Oracle and/or its affiliates.

8

Preparing the File System for an Enterprise Deployment

Preparing the file system for an enterprise deployment involves understanding the requirements for local and shared storage, as well as the terminology that is used to reference important directories and file locations during the installation and configuration of the enterprise topology.

This chapter describes how to prepare the file system for an Oracle Fusion Middleware enterprise deployment.

- Overview of Preparing the File System for an Enterprise Deployment
  It is important to set up your storage in a way that makes the enterprise deployment easy to understand, configure, and manage.
- Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment
  Oracle recommends that you implement certain guidelines regarding shared storage when you install and configure an enterprise deployment.
- About the Recommended Directory Structure for an Enterprise Deployment
  The diagrams in this section show the recommended directory structure for a typical Oracle Fusion Middleware enterprise deployment.
- File System and Directory Variables Used in This Guide
  Understanding the file system directories and the directory variables used to reference these directories is essential for installing and configuring the enterprise deployment topology.
- About Creating and Mounting the Directories for an Enterprise Deployment
  Oracle recommends that you implement certain best practices when you create or mount the top-level directories in an enterprise deployment.
- Summary of the Shared Storage Volumes in an Enterprise Deployment
  It is important to understand the shared volumes and their purpose in a typical Oracle Fusion Middleware enterprise deployment.

Overview of Preparing the File System for an Enterprise Deployment

It is important to set up your storage in a way that makes the enterprise deployment easy to understand, configure, and manage.

This chapter provides an overview of the process of preparing the file system for an enterprise deployment. Oracle recommends setting up your storage according to information in this chapter. The terminology defined in this chapter is used in the diagrams and procedures throughout the guide.

Use this chapter as a reference to understand the directory variables that are used in the installation and configuration procedures.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Other directory layouts are possible and supported, but the model adopted in this guide was designed for maximum availability, providing both the best isolation of components and symmetry in the configuration and facilitating backup and disaster recovery. The rest of the document uses this directory structure and directory terminology.

Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment

Oracle recommends that you implement certain guidelines regarding shared storage when you install and configure an enterprise deployment.

Before you implement the detailed recommendations in this chapter, be sure to review the recommendations and general information about using shared storage in the *High Availability Guide*.

The recommendations in this chapter are based on the concepts and guidelines described in the *High Availability Guide*.

Table 8-1 lists the key sections that you should review and how those concepts apply to an enterprise deployment.

Table 8-1 Shared Storage Resources in the High Availability Guide

Section in *High Availability Guide*	Importance to an Enterprise Deployment
Shared Storage Prerequisites	Describes guidelines for disk format and the requirements for hardware devices that are optimized for shared storage.
Using Shared Storage for Binary (Oracle Home) Directories	Describes your options for storing the Oracle home on a shared storage device that is available to multiple hosts. For an enterprise deployment, Oracle recommends that you use redundant Oracle homes on separate storage volumes. If a separate volume is not available, a separate partition on the shared disk should be used to provide redundant Oracle homes to application tier hosts.
Using Shared Storage for Domain Configuration Files	Describes the concept of creating separate domain homes for the Administration Server and the Managed Servers in the domain. For an enterprise deployment, the Administration Server domain home location is referenced by the ASERVER_HOME variable.
Shared Storage Requirements for JMS Stores and JTA Logs	Provides instructions for setting the location of the transaction logs and JMS stores for an enterprise deployment.
Introduction to Zero Downtime Patching	Describes the Zero Downtime feature and the procedure to configure and monitor workflows.

Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> Zero Downtime Patching (ZDT Patching) provides an automated mechanism to orchestrate the rollout of patches while avoiding downtime or loss of sessions. ZDT reduces risks and downtime of mission-critical applications that require availability and predictability while applying patches.
>
> By using the workflows that you define, you can patch or update any number of nodes in a domain with little or no manual intervention. Changes are rolled out to one node at a time. This preemptively allows for session data to be migrated to compatible servers in the cluster and allows service migration of singleton services, such as JTA and JMS.
>
> When you patch the Oracle home, the current Oracle home must be installed locally on each node that is included in the workflow. Although it is not required, Oracle also recommends that the Oracle home be in the same location on each node.

About the Recommended Directory Structure for an Enterprise Deployment

The diagrams in this section show the recommended directory structure for a typical Oracle Fusion Middleware enterprise deployment.

The directories shown in the diagrams contain binary files that are installed on the disk by the Oracle Fusion Middleware installers, domain-specific files generated through the domain configuration process, as well as domain configuration files that are propagated to the various host computers through the Oracle WebLogic Server `pack` and `unpack` commands.

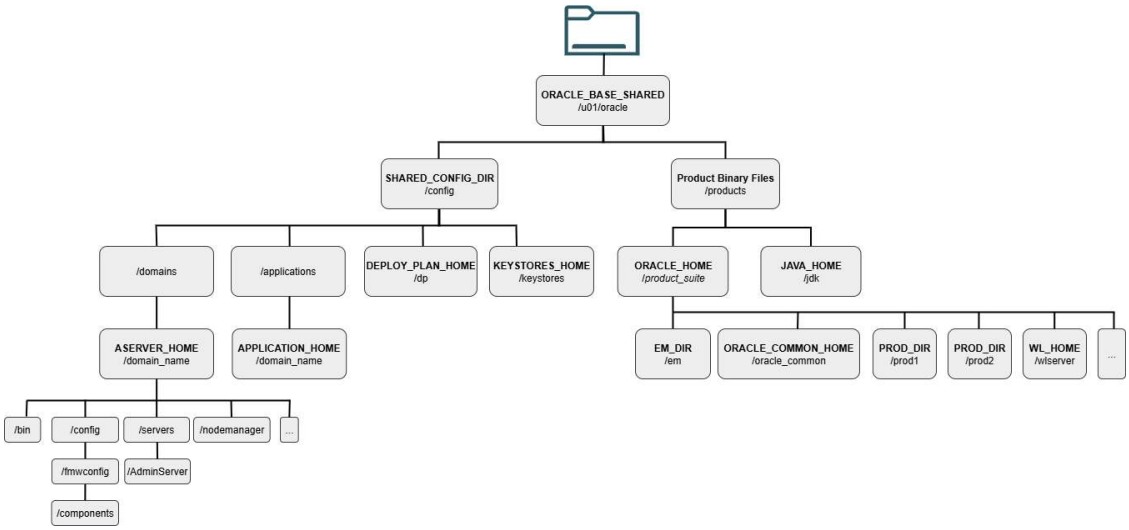
The diagrams are used to indicate:

- Figure 8-1 shows the resulting directory structure on the shared storage device after you have installed and configured a typical Oracle Fusion Middleware enterprise deployment. The shared storage directories are accessible by the application tier host computers.
- Figure 8-2 shows the resulting directory structure on the local storage device for a typical application tier host after you have installed and configured an Oracle Fusion Middleware enterprise deployment. The Managed Servers in particular are stored on the local storage device for the application tier host computers.
- Figure 8-4 shows the resulting directory structure on the local storage device for a typical web tier host after you have installed and configured an Oracle Fusion Middleware enterprise deployment. Note that the software binaries (in the Oracle home) are installed on the local storage device for each web tier host.

Where applicable, the diagrams also include the standard variables used to reference the directory locations in the installation and configuration procedures in this guide.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Figure 8-1 Recommended Shared Storage Directory Structure for an Enterprise Deployment

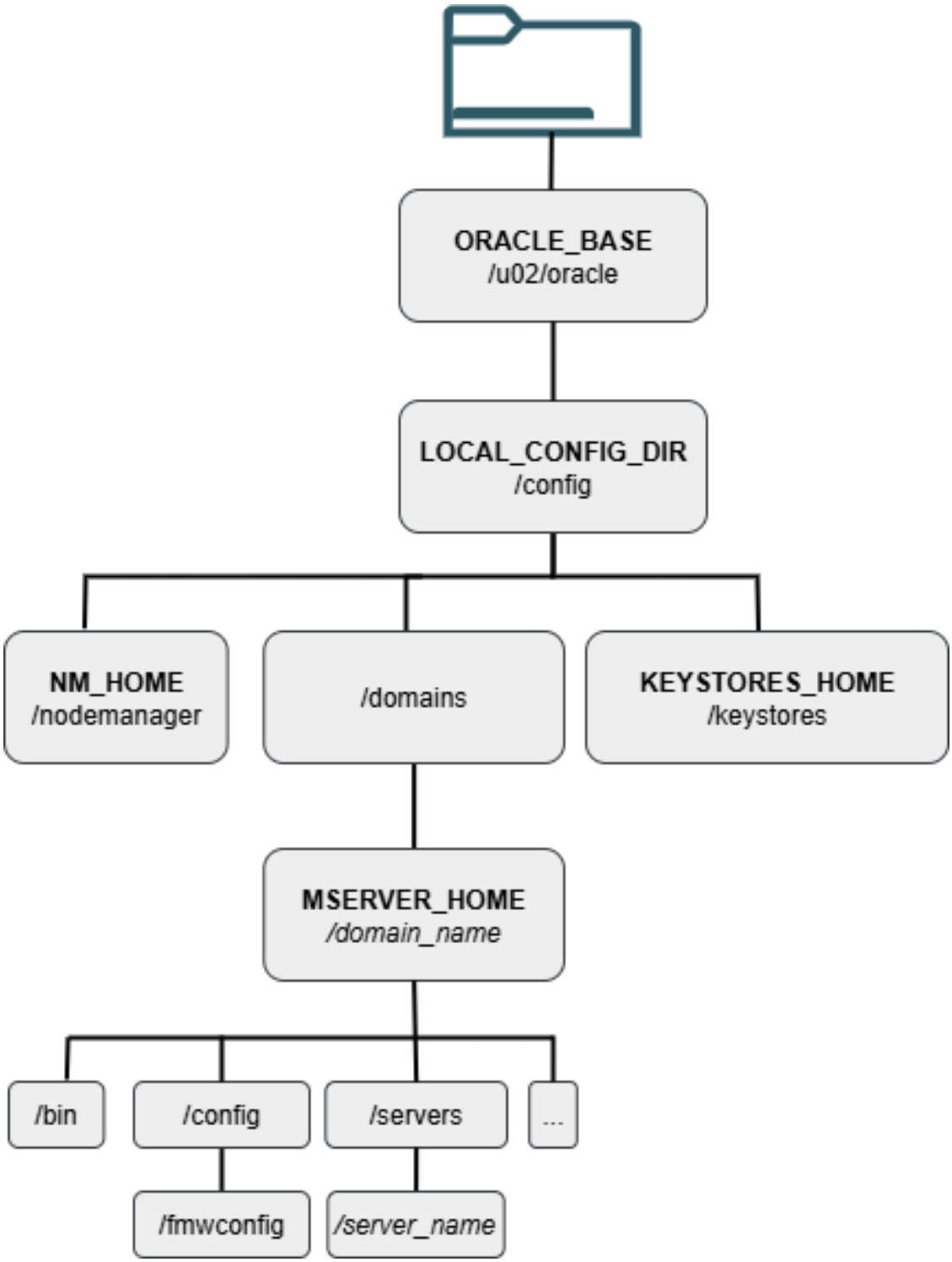


> **Note**
>
> - In a deployment which uses more than one domain, it is recommended that different copies of the binaries are used for each domain. For Oracle Identity Management, it is recommended that the following ORACLE_HOMEs be created:
>   - `DIR_ORACLE_HOME` for Directory components (`/oud`)
>   - `IAD_ORACLE_HOME` for Access components (`/oam`)
>   - `IGD_ORACLE_HOME` for Identity Governance components (`/oig`)
> - Oracle Identity Management uses multiple Domains. The diagram above depicts ASERVER_HOME as a generic location. This guide will often prefix this value with a domain abbreviation for example:
>   - `IAD_ASERVER_HOME` for Oracle Access Manager
>   - `IGD_ASERVER_HOME` for Oracle identity Governance

*See About the Node Manager Configuration in a Typical Enterprise Deployment.

Copyright © 2018, 2025, Oracle and/or its affiliates.

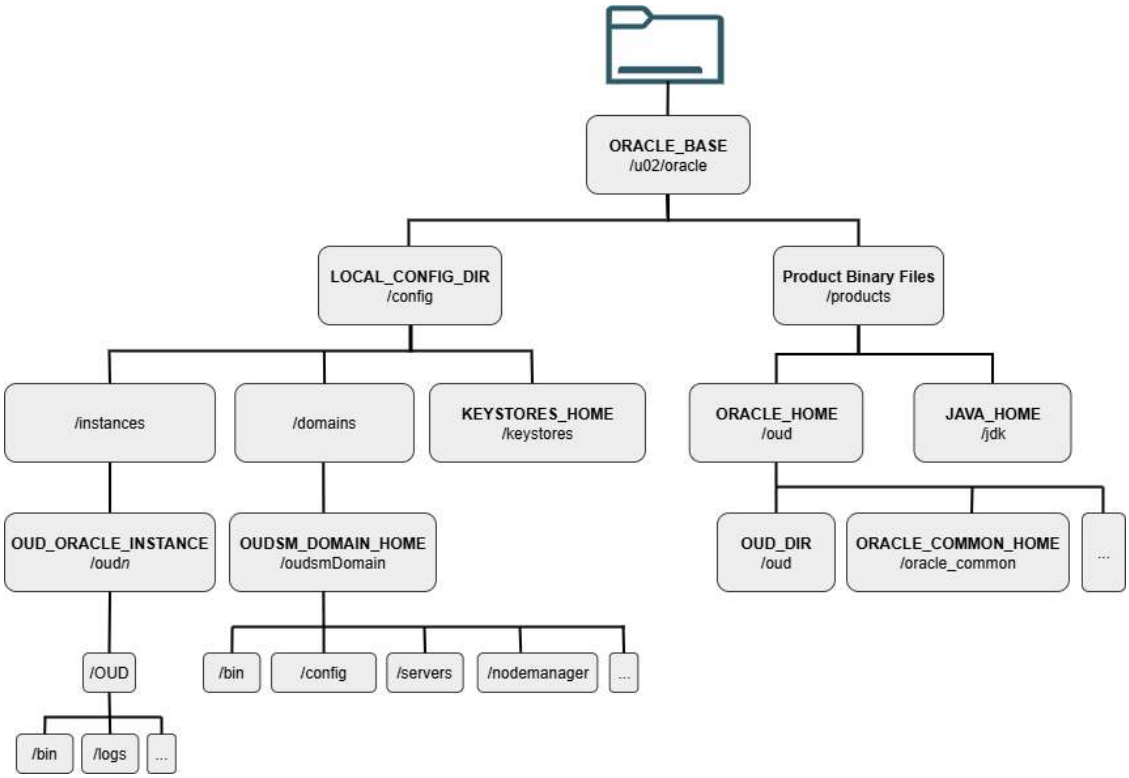
Figure 8-2 Recommended Local Storage Directory Structure for an Application Tier Host Computer in an Enterprise Deployment



See About the Node Manager Configuration in a Typical Enterprise Deployment.

Copyright © 2018, 2025, Oracle and/or its affiliates.

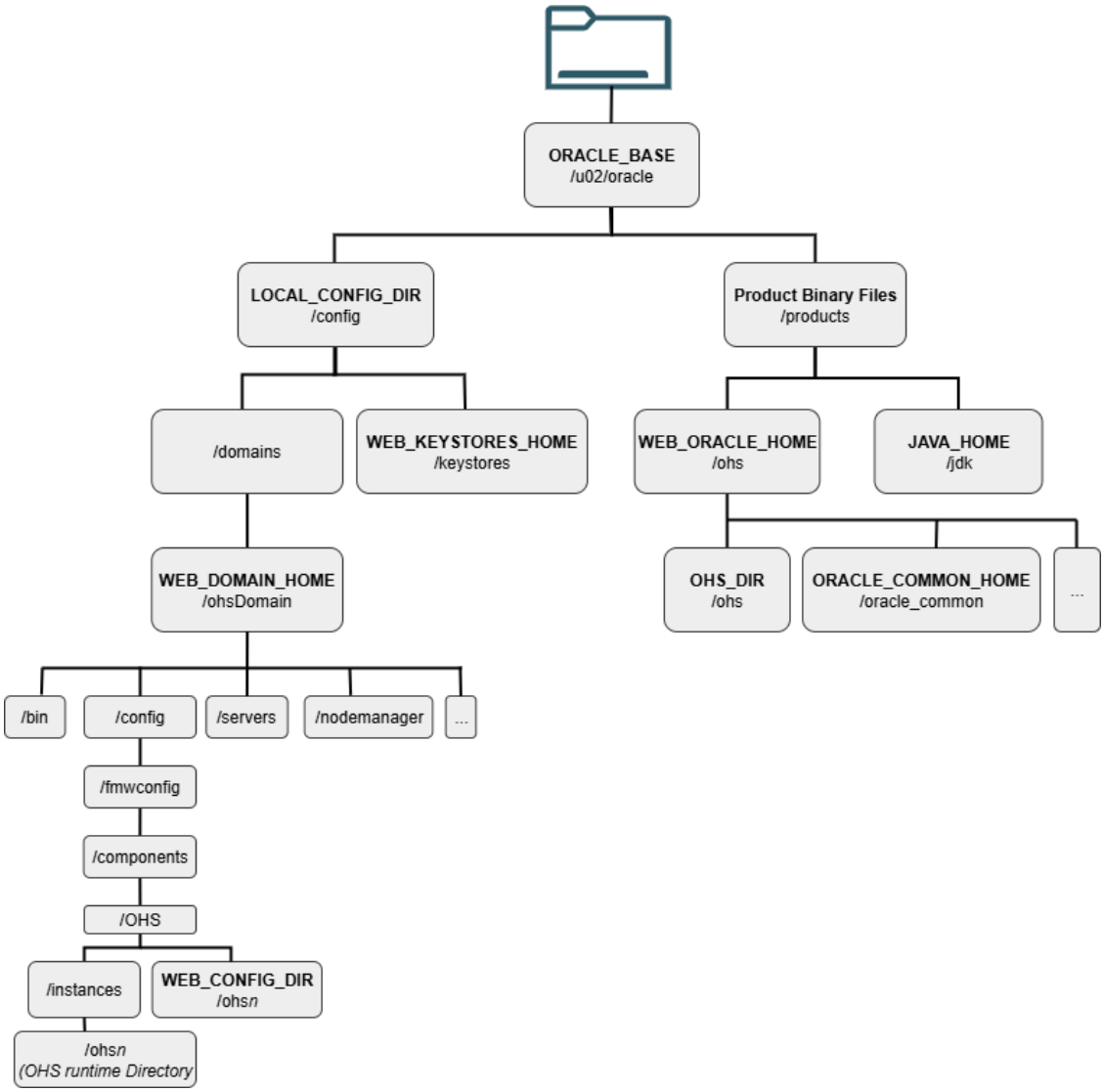
Figure 8-3 Recommended Local Storage Directory Structure for a Directory Tier Host Computer in an Enterprise Deployment



> **Note**
>
> `oudDomain` exists only if OUDSM is used.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Figure 8-4 Recommended Local Storage Directory Structure for a Web Tier Host Computer in an Enterprise Deployment



File System and Directory Variables Used in This Guide

Understanding the file system directories and the directory variables used to reference these directories is essential for installing and configuring the enterprise deployment topology.

Table 8-2 lists the file system directories and the directory variables that are used to reference the directories on the application tier. Table 8-3 lists the file system directories and variables that are used to reference the directories on the web tier.

For additional information about mounting these directories when you use shared storage, see About Creating and Mounting the Directories for an Enterprise Deployment.

Throughout this guide, the instructions for installing and configuring the topology refer to the directory locations that use the variables shown here.

You can also define operating system variables for each of the directories listed in this section. If you define system variables for the particular UNIX shell that you are using, you can then

Copyright © 2018, 2025, Oracle and/or its affiliates.

use the variables as they are used in this document, without having to map the variables to the actual values for your environment.

> **Note**
>
> As you configure your storage devices to accommodate the recommended directory structure, note the actual directory paths in the Enterprise Deployment workbook. You will use these addresses later when you enable the IP addresses on each host computer.
>
> See Using the Enterprise Deployment Workbook.

Table 8-2 Sample Values for Key Directory Variables on the Application Tier

Directory Variable	Description	Relative Path	Sample Value on the Application Tier
ORACLE_BASE	The base directory, under which Oracle products are installed.	N/A	`/u01/oracle` - OAM, OIG `/u02/oracle` - OUD, OUDSM, OID

ORACLE_HOME	The read-only location for the product binaries. For the application tier host computers, it is stored on shared disk.
	The Oracle home is created when you install the Oracle Fusion Middleware Infrastructure software.
	You can then install additional Oracle Fusion Middleware products into the same Oracle home.
		ORACLE_BASE`/products/<product>`	`/u01/oracle/products/oam`
			`/u01/oracle/products/oig`
			`/u02/oracle/products/oud`
			`/u02/oracle/products/oudsm`
			`/u02/oracle/products/oid`

ORACLE_COMMON_HOME	The directory within the Oracle Fusion Middleware Oracle home where common utilities, libraries, and other common Oracle Fusion Middleware products are stored.	*ORACLE_HOME*`/oracle_common`	`/u01/oracle/products/oam/oracle_common`
			`/u01/oracle/products/oig/oracle_common`
			`/u02/oracle/products/oud/oracle_common`
			`/u02/oracle/products/oudsm/oracle_common`
			`/u02/oracle/products/oid/oracle_common`

WL_HOME	The directory within the Oracle home where the Oracle WebLogic Server software binaries are stored.	*ORACLE_HOME*`/wlserver`	`/u01/oracle/products/oam/wlserver` `/u01/oracle/products/oig/wlserver`

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 8-2 (Cont.) Sample Values for Key Directory Variables on the Application Tier

Directory Variable	Description	Relative Path	Sample Value on the Application Tier
PROD_DIR	Individual product directories for each Oracle Fusion Middleware product that you install.	*ORACLE_HOME*/*prod_dir*	/u01/oracle/products/oam /u01/oracle/products/oig /u02/oracle/products/oud /u02/oracle/products/oudsm /u02/oracle/products/oid
EM_DIR	The product directory used to store the Oracle Enterprise Manager Fusion Middleware Control software binaries.	*ORACLE_HOME*/em	/u01/oracle/products/oam/em /u01/oracle/products/oig/em
JAVA_HOME	The location where you install the supported Java Development Kit (JDK).	*ORACLE_BASE*/products/jdk	/u01/oracle/products/jdk - OAM, OIG /u02/oracle/products/jdk - OUD, OUDSM, OID
SHARED_CONFIG_DIR	The shared parent directory for shared environment configuration files, including domain configuration, keystores, runtime artifacts, and application deployments	*ORACLE_BASE*/config	/u01/oracle/config
LOCAL_CONFIG_DIR	The local or nfs-mounted configuration directory unique to a given host containing the machine-specific domain directory for the managed servers (MSERVER_HOME). Directory variable: LOCAL_CONFIG_DIR	/u02/oracle/config - OAM, OIG	/u02/oracle/config
ASERVER_HOME	The Administration Server domain home, which is installed on a shared disk.	*SHARED_CONFIG_DIR*/domains/*domain_name*	/u01/oracle/config/domains/oam /u01/oracle/config/domains/oig
IAD_ASERVER_HOME	The Administration Server domain home for Oracle Access Management, which is installed on a shared disk.	*SHARED_CONFIG_DIR*/domains/*domain_name*	/u01/oracle/config/domains/oam
IGD_ASERVER_HOME	The Administration Server domain home for Oracle Identity Governance, which is installed on a shared disk.	*SHARED_CONFIG_DIR*/domains/*domain_name*	/u01/oracle/config/domains/oig

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 8-2 (Cont.) Sample Values for Key Directory Variables on the Application Tier

Directory Variable	Description	Relative Path	Sample Value on the Application Tier

MSERVER_HOME	The Managed Server domain home, which is created by using the `unpack` command on the local disk of each application tier host.	*LOCAL_CONFIG_DIR*`/domains/`*domain_name*	`/u02/oracle/config/domains/oam`
			`/u02/oracle/config/domains/oig`
			In this example, replace *domain_name* with the name of the WebLogic Server domain.

IAD_MSERVER_HOME	The Managed Server domain home, for Oracle Access Management which is created by using the `unpack` command on the local disk of each application tier host.	*LOCAL_CONFIG_DIR*`/domains/`*domain_name*	`/u02/oracle/config/domains/oam`
IGD_MSERVER_HOME	The Managed Server domain home, for Oracle Identity Governance which is created by using the `unpack` command on the local disk of each application tier host.	*LOCAL_CONFIG_DIR*`/domains/`*domain_name*	`/u02/oracle/config/domains/oig`

APPLICATION_HOME	The Application home directory, which is installed on shared disk, so the directory is accessible by all the application tier host computers.	*SHARED_CONFIG_DIR*`/applications/`*domain_name*	`/u01/oracle/config/applications/oam`
			`/u01/oracle/config/applications/oig`
			In this example, replace *domain_name* with the name of the WebLogic Server domain.

NM_HOME	The directory used by the Per Machine Node Manager start script and configuration files. **Note:** This directory is necessary only if you are using a Per Machine Node Manager configuration. See About the Node Manager Configuration in a Typical Enterprise Deployment.	*LOCAL_CONFIG_DIR*`/node_manager`	`/u02/oracle/config/node_manager`
DEPLOY_PLAN_HOME	The deployment plan directory, which is used as the default location for application deployment plans. **Note:** This directory is required only when you are deploying custom applications to the application tier.	*SHARED_CONFIG_DIR*`/dp`	`/u01/oracle/config/dp`
KEYSTORE_HOME	The shared location for custom certificates and keystores.	*SHARED_CONFIG_DIR*`/keystores`	`/u01/oracle/config/keystores`

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 8-3 Sample Values for Key Directory Variables on the Web Tier

Directory Variable	Description	Sample Value on the Web Tier
WEB_ORACLE_HOME	The read-only location for the Oracle HTTP Server product binaries. For the web tier host computers, this directory is stored on the local disk. The Oracle home is created when you install the Oracle HTTP Server software .	`/u02/oracle/products/ohs`
ORACLE_COMMON_HOME	The directory within the Oracle HTTP Server Oracle home where common utilities, libraries, and other common Oracle Fusion Middleware products are stored.	`/u02/oracle/products/ohs/oracle_common`
WL_HOME	The directory within the Oracle home where the Oracle WebLogic Server software binaries are stored.	`/u02/oracle/products/ohs/wlserver`
PROD_DIR	Individual product directories for each Oracle Fusion Middleware product that you install.	`/u02/oracle/products/ohs`
JAVA_HOME	The location where you install the supported Java Development Kit (JDK).	`/u02/oracle/products/jdk`
WEB_DOMAIN_HOME	The Domain home for the standalone Oracle HTTP Server domain, which is created when you install Oracle HTTP Server on the local disk of each web tier host.	`/u02/oracle/config/domains/ohsDomain`
WEB_CONFIG_DIR	This is the location where you edit the Oracle HTTP Server configuration files (for example, `httpd.conf` and `moduleconf/*.conf`) on each web host. Note that this directory is also referred to as the OHS Staging Directory. Changes made here are later propagated to the OHS Runtime Directory. See Staging and Run-time Configuration Directories in the *Administering Oracle HTTP Server*.	`/u02/oracle/config/domains/ohsDomain/config/fmwconfig/components/OHS/`*`instance/instance_name`*
WEB_KEYSTORE_HOME	If you use Oracle HTTP Server as your web server, this is the location for custom certificates and keystores.	`/u02/oracle/config/keystores`

About Creating and Mounting the Directories for an Enterprise Deployment

Oracle recommends that you implement certain best practices when you create or mount the top-level directories in an enterprise deployment.

- For the application tier, install the Oracle home, which contains the software binaries, on a second shared storage volume or second partition that is mounted to OAMHOST2. Be sure the directory path to the binaries on OAMHOST2 is identical to the directory path on OAMHOST1.

  For example:

  ```
  /u01/oracle/products/oam/
  ```

  See Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment.
- This enterprise deployment guide assumes that the Oracle Web tier software is installed on a local disk.

Copyright © 2018, 2025, Oracle and/or its affiliates.

The Web tier installation is typically performed on local storage to the WEBHOST nodes. When you use shared storage, you can install the Oracle Web tier binaries (and create the Oracle HTTP Server instances) on a shared disk. However, if you do so, then the shared disk *must* be separate from the shared disk used for the application tier, and you must consider the appropriate security restrictions for access to the storage device across tiers.

As with the application tier servers (OAMHOST1 and OAMHOST2), use the same directory path on both computers.

For example:

```
/u02/oracle/products/ohs
```

Summary of the Shared Storage Volumes in an Enterprise Deployment

It is important to understand the shared volumes and their purpose in a typical Oracle Fusion Middleware enterprise deployment.

The following table summarizes the shared volumes and their purpose in a typical Oracle Fusion Middleware enterprise deployment.

See, Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment.

Table 8-4 Shared Storage Volumes in an Enterprise Deployment

Shared Volume Name	Exports	Mounted to Host	Mount Directories	Description and Purpose
oamBinaries1	`/exports/oamBinaries1`	OAMHOST1 OAMHOST3	`/u01/oracle/products/`	Shared storage for the OAM product binaries to be used. This is where the Oracle home directory and product directories are installed.
oamBinaries2	`/exports/oamBinaries2`	OAMHOST2 OAMHOST4	`/u01/oracle/products/`	Shared storage for the OAM product binaries to be used. This is where the Oracle home directory and product directories are installed.
oigBinaries1	`/exports/oigBinaries1`	OIGHOST1 OIGHOST3	`/u01/oracle/products/`	Shared storage for the OIG product binaries to be used. This is where the Oracle home directory and product directories are installed.
oigBinaries2	`/exports/oigBinaries2`	OIGHOST2 OIGHOST4	`/u01/oracle/products/`	Shared storage for the OIG product binaries to be used. This is where the Oracle home directory and product directories are installed.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 8-4 (Cont.) Shared Storage Volumes in an Enterprise Deployment

Shared Volume Name	Exports	Mounted to Host	Mount Directories	Description and Purpose
oamSharedConfig	`/exports/oamSharedConfig`	OAMHOST1 OAMHOST2 (failover) OAMHOST3 (failover) OAMHOST4 (failover)	`/u01/oracle/config`	Administration Server domain configuration, mounted to all OAM hosts; used initially by OAMHOST1 but can be failed over to any host.
oigSharedConfig	`/exports/oigSharedConfig`	OIGHOST1 OIGHOST2 (failover) OIGHOST3 (failover) OIGHOST4 (failover)	`/u01/oracle/config`	Administration Server domain configuration, mounted to all OIG hosts; used initially by OIGHOST1, but can be failed over to any host.
installers	`/exports/installers`	ALL HOSTS	`/installers`	Convenient location used to stage Oracle Software Installers during installation. Can be unmounted following installation. This is optional.

Note

Directory Binaries can either be created locally or shared. If shared, create a separate volume for the Directory binaries.

If Shared storage is being used for local storage, to make the backups easier or to have the files stored on more fault tolerant hardware, then you must create an NFS volume for each of the hosts in the topology. It should then be mounted to the host in the location *ORACLE_BASE*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

9

Preparing the Host Computers for an Enterprise Deployment

It is important to perform a set of tasks on each computer or server before you configure the enterprise deployment topology. This involves verifying the minimum hardware and operating system requirements for each host, configuring operating system users and groups, enabling Unicode support, mounting the required shared storage systems to the host and enabling the required virtual IP addresses on each host.

This chapter describes the tasks that you must perform from each computer or server that is hosting the enterprise deployment.

- Verifying the Minimum Hardware Requirements for Each Host
  After you procure the required hardware for the enterprise deployment, it is important to ensure that each host computer meets the minimum system requirements.
- Verifying Linux Operating System Requirements
  You can review the typical Linux operating system settings for an enterprise deployment in this section.
- Enabling Unicode Support
  It is recommended to enable Unicode support in your operating system so as to allow processing of characters in Unicode.
- Setting the DNS Settings
  You should configure the host to access your corporate DNS hosts.
- Configuring Users and Groups
  You should create the groups and users either locally or in your NIS or LDAP server. This user is the Oracle software owner.
- Configuring a Host to Use an NTP (time) Server
  All servers in the deployment must have the same time. The best way to achieve this is to use an NTP server.
- Configuring a Host to Use an NIS/YP Host
  If you are using NFS version 4, configure a directory service or an NIS (Network Information Server).
- Creating Local Directories
- Creating Mount Points for Shared Storage
- Mounting the Required Shared File Systems on Each Host
  It is important to understand how to mount the shared storage to all the servers that require access.
- Enabling the Required Virtual IP Addresses on Each Host
  You must enable the required virtual IP addresses on each host in order to prepare the host for the enterprise deployment.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Verifying the Minimum Hardware Requirements for Each Host

After you procure the required hardware for the enterprise deployment, it is important to ensure that each host computer meets the minimum system requirements.

After you have procured the required hardware for the enterprise deployment, log in to each host computer and verify the system requirements listed in Hardware and Software Requirements for the Enterprise Deployment Topology.

Ensure that you have sufficient local disk storage and shared storage configured as described in Preparing the File System for an Enterprise Deployment.

Allow sufficient swap and temporary space; specifically:

- **Swap Space**–The system must have at least 500 MB.
- **Temporary Space**–There must be a minimum of 500 MB of free space in the `/tmp` directory.

Verifying Linux Operating System Requirements

You can review the typical Linux operating system settings for an enterprise deployment in this section.

To ensure the host computers meet the minimum operating system requirements, ensure that you have installed a certified operating system and that you have applied all the necessary patches for the operating system.

In addition, review the following sections for typical Linux operating system settings for an enterprise deployment.

- Setting Linux Kernel Parameters
- Setting the Open File Limit and Number of Processes Settings on UNIX Systems
- Verifying IP Addresses and Host Names in DNS or Hosts File

Setting Linux Kernel Parameters

The kernel-parameter and shell-limit values shown in Table 9-1 are recommended values only. Oracle recommends that you tune these values to optimize the performance of the system. See your operating system documentation for more information about tuning kernel parameters.

Kernel parameters must be set to a minimum of those in Table 9-1 on all nodes in the topology.

The values in the following table are the current Linux recommendations. For the latest recommendations for Linux and other operating systems, see *Oracle Fusion Middleware System Requirements and Specifications*.

If you deploy a database onto the host, you might need to modify additional kernel parameters. See Configuring Kernel Parameters in *Oracle Grid Infrastructure Installation Guide for Linux*.

Table 9-1 UNIX Kernel Parameters

Parameter	Value
kernel.sem	256 32000 100 142

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 9-1 (Cont.) UNIX Kernel Parameters

Parameter	Value
kernel.shmmax	4294967295

To set these parameters:

1. Sign in as `root` and add or amend the entries in the `/etc/sysctl.conf` file.
2. Save the file.
3. Activate the changes by entering the following command:

   ```
   /sbin/sysctl -p
   ```

Setting the Open File Limit and Number of Processes Settings on UNIX Systems

On UNIX operating systems, the `Open File Limit` is an important system setting, which can affect the overall performance of the software running on the host computer.

For guidance on setting the `Open File Limit` for an Oracle Fusion Middleware enterprise deployment, see Host Computer Hardware Requirements.

> **Note**
>
> The following examples are for Linux operating systems. Consult your operating system documentation to determine the commands to be used on your system.

For more information, see the following sections.

- Viewing the Number of Currently Open Files
- Setting the Operating System Open File and Processes Limits

Viewing the Number of Currently Open Files

You can see how many files are open with the following command:

```
/usr/sbin/lsof | wc -l
```

To check your open file limits, use the following commands.

C shell:

```
limit descriptors
```

Bash:

```
ulimit -n
```

Setting the Operating System Open File and Processes Limits

To change the Open File Limit values on Oracle Enterprise Linux 6 or greater:

Copyright © 2018, 2025, Oracle and/or its affiliates.

1. Sign in as `root` user and edit the following file:

   `/etc/security/limits.d/*-nproc.conf`

   For example:

   `/etc/security/limits.d/20-nproc.com`

   > **Note**
   >
   > The number can vary from host to host.

2. Add the following lines to the file. (The values shown here are for example only):

   ```
   * soft  nofile  4096
   * hard  nofile  65536
   * soft  nproc   2047
   * hard  nproc   16384
   ```

   The `nofiles` values represent the open file limit; the `nproc` values represent the number of processes limit.

3. Save the changes, and close the file.

4. Re-login into the host computer.

Verifying IP Addresses and Host Names in DNS or Hosts File

Before you begin the installation of the Oracle software, ensure that the IP address, fully qualified host name, and the short name of the host are all registered with your DNS server. Alternatively, you can use the local `hosts` file and add an entry similar to the following:

`IP_Address Fully_Qualified_Name Short_Name`

For example:

```
10.229.188.205  host1.example.com  host1
```

Oracle also recommends that you use aliases to map to different IPs in different data centers in preparation for disaster recovery. You can also use these aliases to configure the listen address for some of the components.

In this guide, the abstract hostnames that are provided on the **Hardware - Host Computers** tab of the workbook (OIGHOST*n* and ADMINVHN) are used for these aliases, so the `/etc/hosts` can be similar to this example:

```
10.229.188.203 iadadminvhn.example.com iadadminvhn
10.229.188.204 igdadminvhn.example.com igdadminvhn
10.229.188.205 oighost1.example.com oighost1
10.229.188.206 oighost2.example.com oighost2
10.229.188.207 oighost3.example.com oighost3
10.229.188.208 oighost4.example.com oighost4
10.229.188.209 webhost1.example.com webhost1
10.229.188.210 webhost2.example.com webhost2
10.229.188.211 oamhost1.example.com oamhost1
10.229.188.212 oamhost2.example.com oamhost2
10.229.188.213 oamhost3.example.com oamhost3
10.229.188.214 oamhost4.example.com oamhost4
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

Enabling Unicode Support

It is recommended to enable Unicode support in your operating system so as to allow processing of characters in Unicode.

Your operating system configuration can influence the behavior of characters supported by Oracle Fusion Middleware products.

On UNIX operating systems, Oracle highly recommends that you enable Unicode support by setting the `LANG` and `LC_ALL` environment variables to a locale with the UTF-8 character set. This enables the operating system to process any character in Unicode. Oracle Identity and Access Management technologies, for example, are based on Unicode.

If the operating system is configured to use a non-UTF-8 encoding, Oracle Identity and Access Management components may function in an unexpected way. For example, a non-ASCII file name might make the file inaccessible and cause an error. Oracle does not support problems caused by operating system constraints.

Setting the DNS Settings

You should configure the host to access your corporate DNS hosts.

To do this, update DNS settings by updating the file `/etc/resolv.conf`.

Configuring Users and Groups

You should create the groups and users either locally or in your NIS or LDAP server. This user is the Oracle software owner.

The instructions below are for creating the user locally. Refer to the NIS documentation for information about creating these groups and user in your NIS server.

Groups

You must create the following groups on each node.

- `oinstall`
- `dba`

To create the groups, use the following command as root:

```
groupadd groupname
```

For example

```
groupadd -g 500 oinstall
groupadd -g 501 dba
```

User

You must create the following user on each node.

- `oracle` - The owner of the Oracle software. You may use a different name. The primary group for this account must be `oinstall`. The account must also be in the `dba` group.

Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> - The group `oinstall` must have write privileges to all the file systems on shared and local storage that are used by the Oracle software.
> - Each group must have the same Group ID on every node.
> - Each user must have the same User ID on every node.
> - The user and group should exists at the NIS server due to the NFSv4 mount requirement.

To create a local user, use the following command as `root`:

```
useradd -g primary group -G optional groups -u userid username
```

For example:

```
useradd -g oinstall -G dba -u 500 oracle
```

> **Note**
>
> To create this user in NIS, refer to the NIS documentation.

Configuring a Host to Use an NTP (time) Server

All servers in the deployment must have the same time. The best way to achieve this is to use an NTP server.

To configure a host to use an NTP server:

1. Determine the name of the NTP server(s) you wish to use. For security reasons, ensure that these are inside your organization.
2. Log in to the host as the root user.
3. Edit the file `/etc/ntp.conf` to include a list of the time servers. After editing, the file appears as follows:

   ```
   server ntphost1.example.com
   server ntphost2.example.com
   ```

4. Run the following command to synchronize the system clock to the NTP server:

   ```
   /usr/sbin/ntpdate ntpserver1.example.com
   /usr/sbin/ntpdate ntpserver2.example.com
   ```

5. Start the NTP client using the following command:

   ```
   service ntpd start
   ```

6. Validate that the time is set correctly using the date command.
7. To make sure that the server always uses the NTP server to synchronize the time. Set the client to start on reboot by using the following command:

   ```
   chkconfig ntpd on
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

Configuring a Host to Use an NIS/YP Host

If you are using NFS version 4, configure a directory service or an NIS (Network Information Server).

Once you have configured your NIS host, configure each compute node to use it. Before beginning, determine the host names of the NIS servers you are going to use.

1. Login to the host as root.
2. Edit the `/etc/idmapd.conf` configuration file:

   ```
   vi /etc/idmapd.conf
   ```

   Set the domain value, as in the following example:

   ```
   Domain = example.com
   ```

3. Restart the `rpcidmapd` service:

   ```
   service rpcidmapd restart
   ```

4. Update the `/etc/yp.conf` configuration file, and set the correct domain value, as in the following example:

   ```
   vi /etc/yp.conf
   ```

   Add the following line:

   ```
   domain example.com server NIS_Server_hostname_or_IP
   ```

   Where `example.com` is the example domain and *NIS_Server_hostname_or_IP* is the host name or IP address of the NIS host. You must replace these sample values with values appropriate for your environment.

5. Set NIS domain name on the command line:

   ```
   domainname NIS_DOMAIN_NAME
   ```

   For example:

   ```
   domainname nisdomain.example.com
   ```

6. Edit the `/etc/nsswitch.conf` configuration file:

   ```
   vi /etc/nsswitch.conf
   ```

   Add `nis` to each of the following entries:

   > **Note**
   >
   > The first value may be `compat` or `files` depending on your OS and enterprise requirements.

   ```
   passwd:     files nis
   shadow:     files nis
   group:      files nis
   automount:  files nis nisplus
   aliases:    files nis nisplus
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

7. Restart the `rpcidmapd` service:

   ```
   service rpcidmapd restart
   ```

8. Restart the `ypbind` service by running the following command:

   ```
   service ypbind restart
   ```

9. Check the `yp` service by running this command:

   ```
   ypwhich
   ```

10. Verify if you can access Oracle user accounts:

    ```
    ypcat passwd
    ```

11. Add `ypbind` to your boot sequence, so that it starts automatically after rebooting.

    ```
    chkconfig ypbind on
    ```

Creating Local Directories

If your deployment requires the use of local directories, for example local binaries, local instance/domain files or certificates, then create local directories.

If you are using `/u02` as described in this Enterprise Deployment Guide, you will need to create these local directories as `root` and then change the ownership to your `oracle` account.

```
sudo mkdir -p /u02/oracle/config /u02/oracle/products
```

```
sudo chown -R oracle:oinstall /u02
```

For a list of locations to create, see Preparing the File System for an Enterprise Deployment.

Creating Mount Points for Shared Storage

Before you can mount shared storage to a host you must pre-create the location to which it will be attached. For example:

```
sudo mkdir -p /u01/oracle/products
```

```
sudo chown -R oracle:oinstall /u01/oracle
```

For a list of locations to create, see Preparing the File System for an Enterprise Deployment.

Mounting the Required Shared File Systems on Each Host

It is important to understand how to mount the shared storage to all the servers that require access.

The shared storage configured, as described in Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment, must be available on the hosts that use it.

Copyright © 2018, 2025, Oracle and/or its affiliates.

In an enterprise deployment, it is assumed that you have a hardware storage filer, which is available and connected to each of the host computers that you have procured for the deployment

Each host must have appropriate privileges set within the Network Attached Storage (NAS) or Storage Area Network (SAN) so that it can write to the shared storage.

Follow the best practices of your organization for mounting shared storage. This section provides an example of how to do this on Linux by using NFS storage.

You must create and mount shared storage locations so that OAMHOST1 and OAMHOST2 can see the same location if it is a binary installation in two separate volumes.

You use the following command to mount shared storage from a NAS storage device to a Linux host. If you are using a different type of storage device or operating system, refer to your manufacturer documentation for information about how to do this.

> **Note**
>
> The user account used to create a shared storage file system owns and has read, write, and execute privileges for those files. Other users in the operating system group can read and process the files, but they do not have write privileges.
>
> See Selecting an Installation User in the *Oracle Fusion Middleware Installation Planning Guide*.

In the following example, `nasfiler` represents the shared storage filer. Also note that these are examples only. Typically, the mounting of these shared storage locations should be done by using the `/etc/fstabs` file on UNIX systems, so that the mounting of these devices survives a reboot. Refer to your operating system documentation for more information.

To mount the shared storage on Linux:

1. Create the mount directories on OAMHOST1, as described in Summary of the Shared Storage Volumes in an Enterprise Deployment, and then mount the shared storage. For example:

   ```
   mount-t nfs nasfiler:/exports/oamBinaries1 /u01/oracle/products/
   ```

2. Repeat the procedure on OAMHOST2 using `oamBinaries2`.
3. Repeat the procedure on OIGHOST1 and OIGHOST2 using the shares `oigBinaries1` and `oigBinaries2`.

- Validating the Shared Storage Configuration

Validating the Shared Storage Configuration

Ensure that you can read and write files to the newly mounted directories by creating a test file in the shared storage location that you just configured.

For example:

```
$ cd <newly mounted directory>
$ touch testfile
```

Verify that the owner and permissions are correct:

```
$ ls -l testfile
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

Then remove the file:

```
$ rm testfile
```

> **Note**
>
> The shared storage can be a NAS or SAN device. The following example illustrates creating storage for a NAS device from OAMHOST1. The options may differ depending on the specific storage device.
>
> ```
> mount -t nfs -o rw,bg,hard,nointr,tcp,vers=3,timeo=300,rsize=32768,wsize=32768
> nasfiler:/exports/oamBinaries1 /u01/oracle/products
> ```
>
> Contact your storage vendor and machine administrator to learn about the appropriate options for your environment.

Enabling the Required Virtual IP Addresses on Each Host

You must enable the required virtual IP addresses on each host in order to prepare the host for the enterprise deployment.

To prepare each host for the enterprise deployment, you must enable the virtual IP (VIP) addresses that are described in Reserving the Required IP Addresses for an Enterprise Deployment.

It is assumed that you have already reserved the VIP addresses and host names and that they have been enabled by your network administrator. You can then enable the VIPs on the appropriate host.

> **Note**
>
> The virtual IP addresses used for the enterprise topology are not persisted because they are managed by Whole Server Migration (for selected Managed Servers and clusters) or by manual failover (for the Administration Server).

Starting with Oracle Enterprise Linux 6, the "`ifconfig`" command is deprecated and is replaced with the "`ip`" command.

To enable the VIP addresses on each host, run the following commands as `root`:

1. Determine the CIDR notation of the netmask. Each Netmask has a CIDR notation. For example, 255.255.240.0 has a CIDR of 20.

   If the netmask you are adding is the same as the interface, the fastest way to determine this is to examine the existing IP address that are assigned to the network card. You can do this by using the following command:

   ```
   ip addr show
   ```

   Sample output:

   ```
   1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
   default qlen 1000
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9000 qdisc mq state UP
group default qlen 1000
    link/ether 02:00:17:01:0c:69 brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    inet 10.0.10.96/24 brd 10.0.10.255 scope global dynamic noprefixroute
ens3
       valid_lft 59480sec preferred_lft 59480sec
    inet6 fe80::17ff:fe01:c69/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

In this example your IP Address `10.0.10.96` is assigned to ethernet device `ens3`. If you are unsure of the CIDR value, contact your network administrator.

2. Configure the additional IP address on the appropriate network interface card with an appropriately suffixed label using the following command:

```
ip addr add VIP/CIDR dev nic# label nic#:n
```

> **Note**
>
> For each VIP/VHN that you need to add, increment the :n suffix starting with :1

For Example: For VIP IP of 10.0.10.97 CIDR 20 and the ens3 NIC:

```
ip addr add 10.0.10.97/24 dev ens3 label ens3:1
```

3. For each of the virtual IP addresses that you define, update the ARP caches by using the following command:

```
arping -b -A -c 3 -I ens3 10.0.10.97
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

10

Preparing the Oracle Cloud Infrastructure for an Enterprise Deployment

If you plan to deploy Identity and Access Management on Oracle Cloud Infrastructure (OCI), you have to configure OCI to facilitate the deployment. Create the required OCI components to perform the deployment.

> **Note**
>
> The instructions provided in this guide are correct at the time of publishing. Due to the evolving nature of the OCI interface, you may find minor changes in the options. See the Oracle Cloud Infrastructure documentation to obtain the latest steps.

This chapter includes the following topics:

- About the OCI Deployment
  It is important to understand the OCI components that you require to deploy Oracle Identity and Access Management on OCI.
- Creating an SSH Key Pair
  You must create an SSH key pair in order to connect to the bastion node, compute instances, and database hosts
- Creating an OCI Compartment
  Create a container in your OCI tenancy to hold the deployment.
- Creating OCI Networking
  Create the network components in your OCI tenancy for the deployment.
- Creating Compute Instances
  Create the compute instances in your OCI tenancy for the deployment.
- Creating File Systems and Mount Targets
  You need to create NFS file systems for Persistent Volumes and Oracle HTTP Server installations.
- Creating Load Balancers
  You need to create two OCI load balancers. One of these load balances is used to direct public traffic and the other for internal call backs. The load balancer used for internal traffic is not available outside the OCI container.
- Creating a Database
  You must create a database to store all the schema information for Oracle Identity Management.
- Creating a DNS Server
  It is important that all host names are resolvable, including the load balancer virtual hosts. You can make them resolvable by adding entries to the local hosts files. However, in OCI, using a private DNS server is the simpler method.
- Validating Your Environment
  Perform checks to ensure that your environment is ready for a deployment.

Copyright © 2018, 2025, Oracle and/or its affiliates.

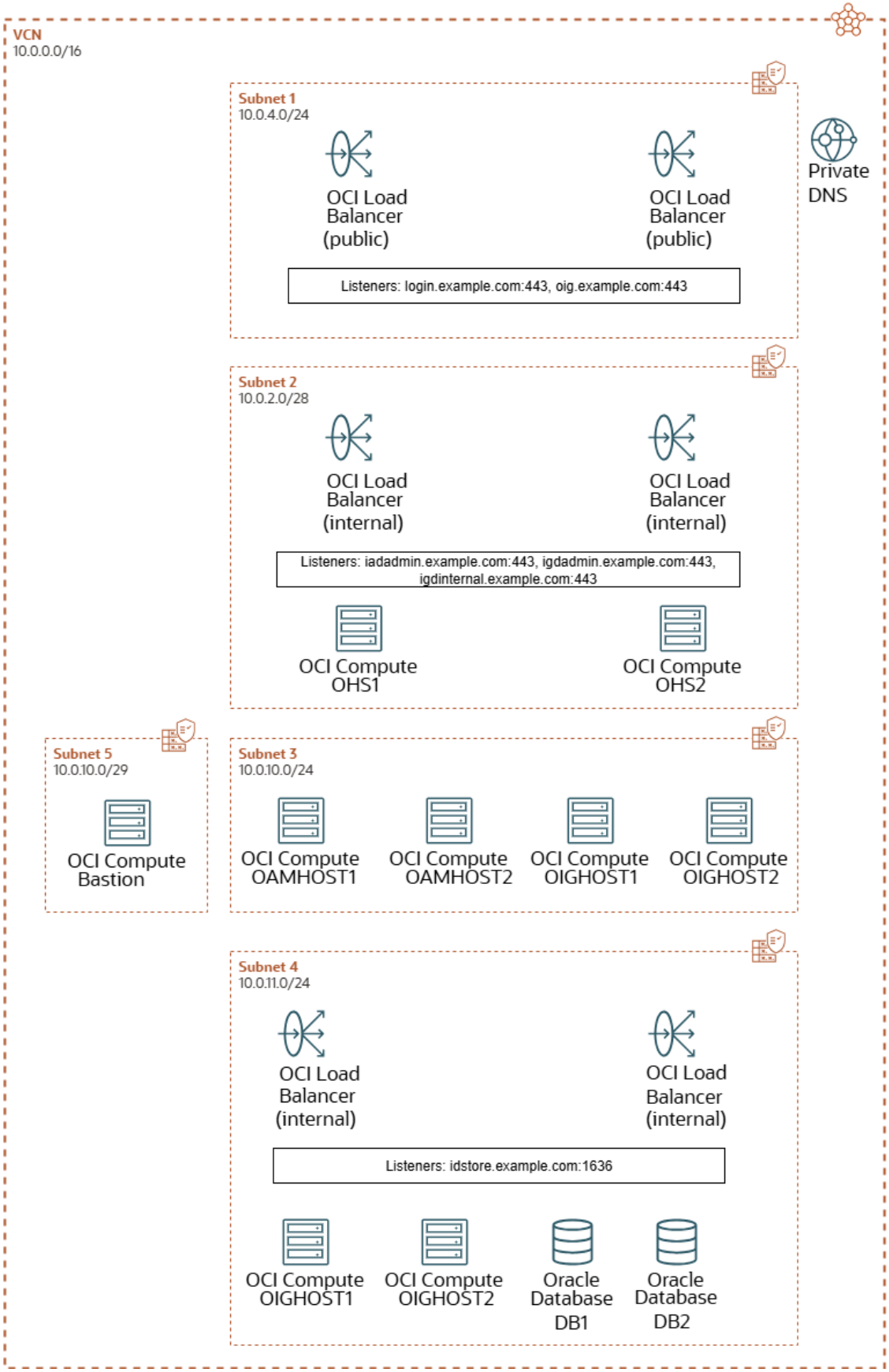
About the OCI Deployment

It is important to understand the OCI components that you require to deploy Oracle Identity and Access Management on OCI.

This illustrations below shows all the required OCI components. It shows the different network requirements and how the OCI components fit into those networks. Each subnet is protected by security lists.

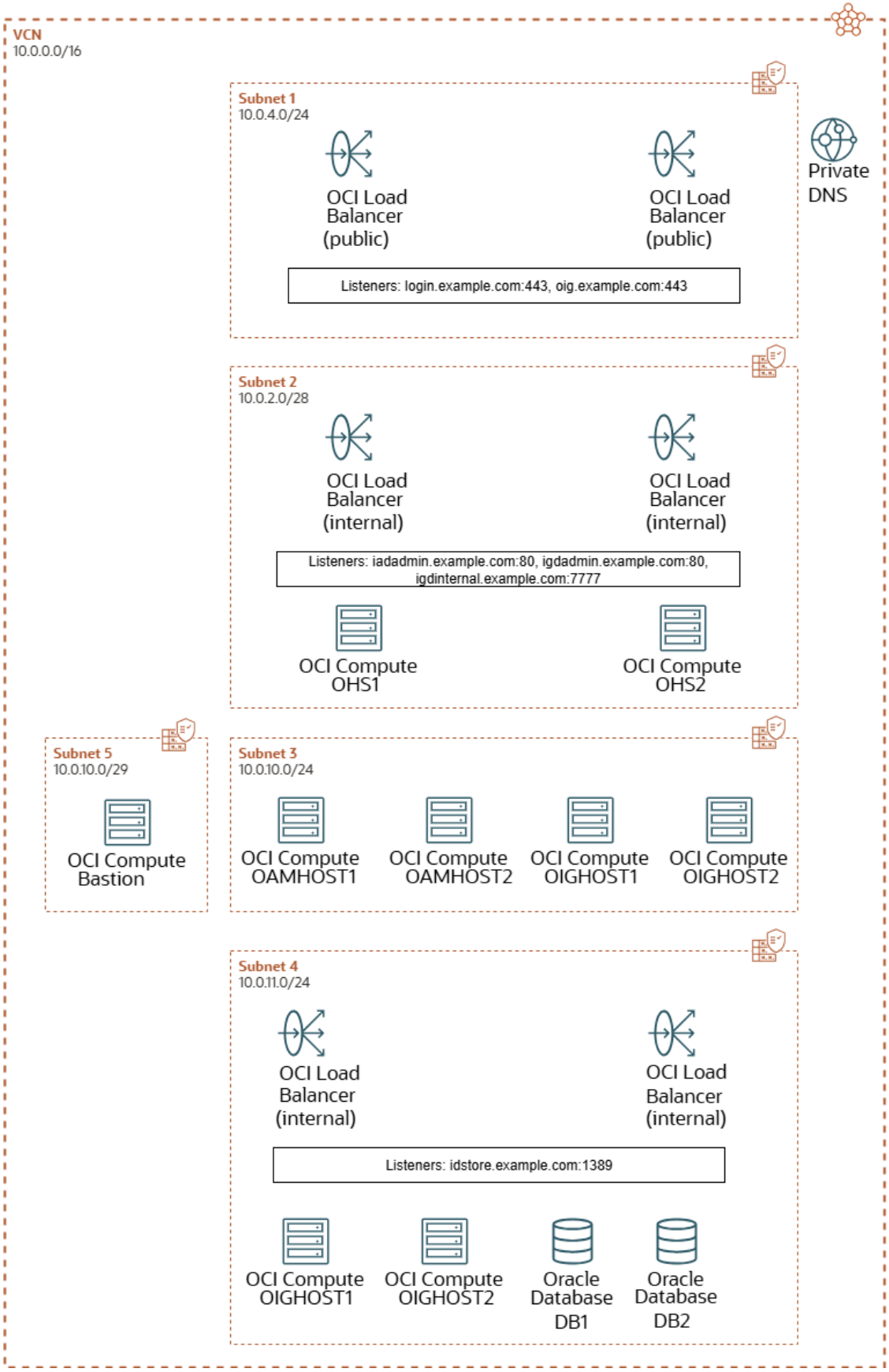
Copyright © 2018, 2025, Oracle and/or its affiliates.

Figure 10-1 An Illustration of the OCI Layout in an End to End SSL Topology



Copyright © 2018, 2025, Oracle and/or its affiliates.

Figure 10-2 An Illustration of the OCI Layout in a SSL Terminated Topology



Copyright © 2018, 2025, Oracle and/or its affiliates.

When deploying Oracle Identity and Access Management in OCI, you have to set up the OCI environment with the following characteristics:

- **VCN**: There will be one public Virtual Cloud Network which provides external access to the environment. For security reasons, the VCN is broken down into a several subnets.
- **Subnets**: The VCN is divided into several subnets to ensure that the network traffic is routed only to the areas requiring it. For instance, traffic to the database subnet will not be available directly from the internet. Traffic is available only to the Application tier, which interacts with the database subnet.
- **Security Lists**: Security lists provide an additional layer of security that allows traffic only into and out of a subnet, based on the ports and protocols permitted.
- **Bastion Node**: The Bastion node is a compute instance inside the VCN that you can log in to. The Bastion node can communicate with all the components inside the deployment. The Bastion node is used for setting up the environment and for ongoing management. Therefore, you must lock down access to the Bastion node to ensure that it is accessed only by clients on your corporate network who are registered with it using an SSL key pair.
- **Load balancer**: The three LBaaS services are created within the OCI framework. The public-facing load balancer is used to access the Oracle Identity and Access Management deployment from the internet. The private load balancers are for internal traffic and routing it is not available outside of the VCN. The public load balancer is the only internet-facing part of your deployment (except for the Bastion node).
- **Compute Instances**: You require a minimum of two compute instances to host your Oracle HTTP servers. These are placed into a demilitarized zone (DMZ) below the load balancers. The load balancers send requests to the OHS servers which pass traffic onto the applications residing in the application compute instances.You require a minimum of two compute instances for your applications. For example, two compute Instances for OAM, two compute instances for OIG and two compute instances for LDAP.
- **Database**: The database(s) are present in a dedicated subnet.
- **DNS**: The DNS server is optional. It is used internally to provide name resolution. You can achieve name resolution by maintaining entries in the individual host files.

The following sections describe the procedure to set up the components depicted in this illustration:

Creating an SSH Key Pair

You must create an SSH key pair in order to connect to the bastion node, compute instances, and database hosts

You can configure OCI by using the Oracle Cloud Console and a bastion node. The SSL certificates provide a secure access to the bastion node, compute instances, and database hosts. You have to create an SSL certificate on the host you use to configure OCI. This host could be a laptop or a desktop.

After you create the certificate on the device, share it with the OCI resources to enable access to the resources and to manage them. If you use more than one device, you have to register the SSL keys for all those devices.

If you do not have an SSL certificate for the device you are using, create the cetificate using the following command:

```
ssh-keygen -t rsa -N "" -b 2048 -f id_rsa
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

This command creates two files `id_rsa` and `id_rsa.pub` in the `.ssh` directory under the home directory. These are the certificate files you will use to access the OCI resources.

Creating an OCI Compartment

Create a container in your OCI tenancy to hold the deployment.

To create a compartment:

1. Log in to the Oracle Cloud Infrastructure Console, select **Identity and Security**, and then select **Compartments** under **Identity**.
2. Click **Create Compartment**.
3. Specify a **Name** and **Description**.
4. Click **Create Compartment**.

You will create all the OCI objects inside this compartment.

Creating OCI Networking

Create the network components in your OCI tenancy for the deployment.

- Creating an Oracle Virtual Cloud Network
- Creating Gateways
- Creating Security Lists
- Creating Route Tables
- Creating Subnets

Creating an Oracle Virtual Cloud Network

You need to create a Virtual Cloud Network (VCN). A VCN is a customizable, software-defined network that you create in an OCI region. It's analogous to a traditional data center network, offering you control over your cloud network environment. You can define your own private IP address space, create subnets, configure routing, and set up security rules.

To create an Oracle Virtual Cloud Network:

1. Log in to the Oracle Cloud Infrastructure Console.
2. Select **Networking** and then **Virtual Cloud Networks**.
3. Click **Create VCN**.
4. Enter the following information in the wizard:
   - **Name**: Select a name for the network. For example `idm`.
   - **Compartment**: Select the name of the compartment you created earlier. See Creating an OCI Compartment.
   - **IPv4 CIDR Blocks**: Enter the internal CIDR block you want to use for your network. For example: `10.0.0.0/16`.
   - **Use DNS Hostnames in this VCN**: Select this option.
5. Click **Create VCN**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

6. Review the summary information of the details specified and click **Create**.
7. When complete, click **View Virtual Cloud Network**.

These steps will create a public and private subnet.

Creating Gateways

You need to create a gateway to allow data to flow from one network to another. Gateways serve as an entry and exit point for a network as all data going outside of a network must pass through it. As the name suggests it acts as a gate between two networks.

There are four types of Gateways in Oracle Cloud Infrastructure (OCI):

- **Internet Gateway**: Used to access the internet from a VCN (say, network) in OCI. It supports connections initiated from within the VCN (egress) and connections initiated from the internet (ingress).
- **NAT Gateway**: Used to provide resources without public IP addresses, access to the Internet without exposing these resources to the incoming internet connections.
- **Service Gateway**: Gives resources in your VCN and on-premises network, private access to multiple Oracle services within OCI without the traffic going over the internet. Any traffic from your VCN that is destined for one of the supported public services uses the instance's private IP address for routing, travels over the Oracle Cloud Infrastructure network fabric, and never traverses the internet.
- **Dynamic Routing Gateway (DRG)**: Used to connect to your existing on-premises network to your VCN. A DRG provides a single point of entry for remote network paths coming into the VCN. It provides a path for VCNs to communicate across regions or outside the region to On-premise. Each VCN can have a single DRG.

To create the required gateways:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Select **Networking** > **Virtual Cloud Networks** and click the name of your Virtual Cloud Network.
3. Select **Gateways** from the list of resources.
4. To create an Internet Gateway:
   a. Click **Create Internet Gateway**.
   b. Enter a name, for example `Internet Gateway`.
   c. Click **Create Internet Gateway**.
5. To create a Service Gateway:
   a. Click **Create Service Gateway**.
   b. Enter a name, for example `Service Gateway`.
   c. From the drop down list, select **ALL Services in Oracle Services Network**.
   d. Click **Create Service Gateway**.

Creating Security Lists

You need to create security lists which enable different subnets to communicate with each other.

Copyright © 2018, 2025, Oracle and/or its affiliates.

This section describes the minimum steps you need to perform to enable this access. You should harden your security lists to ensure that only certain machines/networks have access to this node. This part is outside the scope of this guide.

- Required Security Lists
- Creating the Required Security Lists

Required Security Lists

You need to create security lists which enable the OCI components to communicate with each other across different subnets and where necessary the internet.

The following tables show the security lists that need to be created.

bastion-private-seclist

Table 10-1 Description of Ingress and Egress Rules for Security List *bastion-private-seclist*

Rule Type	Type	Source CIDR	Destination CIDR	Protocol	Destination Port Range
Ingress	CIDR	10.0.1.0/29		TCP SSH Connections	22
Ingress	CIDR	10.0.1.0/29		ICMP	
Egress	CIDR		0.0.0.0/0	All Protocols	

> **Note**
>
> 10.0.1.0 is the subnet you will use for the bastion node. You can change this value if required.

bastion-public-seclist

Table 10-2 Description for Ingress and Egress Rules for Security List *bastion-public-seclist*

Rule Type	Type	Source CIDR	Destination CIDR	Protocol	Destination Port Range	Type
Ingress	CIDR	0.0.0.0/0		TCP	22	
Ingress	CIDR	10.0.1.0/29		ICMP		3
Egress	CIDR		0.0.0.0/0	All Protocols		

> **Note**
>
> 10.0.1.0 is the subnet you will use for the bastion node. You can change this value if required. Unless otherwise stated, leave the values blank.

Copyright © 2018, 2025, Oracle and/or its affiliates.

setup-seclist

During the set up of Oracle Identity and Access Management, the bastion node requires access to some of the services that get created as part of the build process. The access is not required after the build process is complete. For manageability reasons, a separate security list is created for this purpose. This way, after the setup is complete, you just have to remove the security list from the subnet. If further setups are required, you can add as needed.

The security list should be added to the following subnets:

- Private subnet for Node Manager
- db-subnet

Table 10-3 Description for Ingress Rules for Security List *setup-seclist*

Rule Type	Type	Source CIDR	Destination CIDR	Protocol	Source Port Range	Destination Port Range - SSL Terminated	Destination Port Range - End to End SSL	Comment
Ingress	CIDR	10.0.1.0/29		TCP		7001	9002 and 7002	OAM Administration Server
Ingress	CIDR	10.0.10.0/24		TCP		111	111	
Ingress	CIDR	10.0.10.0/24		TCP		2048-2050	2048-2050	
Ingress	CIDR	10.0.10.0/24		UDP		111	111	
Ingress	CIDR	10.0.10.0/24		UDP		2048	2048	
Ingress	CIDR	10.0.1.0/29		TCP		1389	1636	LDAP Connections
Ingress	CIDR	10.0.2.0/24		TCP		111	111	
Ingress	CIDR	10.0.2.0/24		TCP		2048-2050	2048-2050	
Ingress	CIDR	10.0.2.0/24		UDP		111	111	
Ingress	CIDR	10.0.2.0/24		UDP		2048	2048	
Egress	CIDR	10.0.2.0/24		TCP		111	111	
Egress	CIDR	10.0.2.0/24		TCP		2048-2050	2048-2050	
Egress	CIDR	10.0.2.0/24		UDP		111	111	
Egress	CIDR	10.0.2.0/24		UDP		2048-2050	2048-2050	

Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> The destination ports listed above are dependent on the values you provide to your installation. Sample values will be used for consistency within this guide.

app-seclist

Table 10-4 Description for Ingress Rules for Security List *app-seclist*

Rule Type	Type	Source CIDR	Destination CIDR	Protocol	Destination Port Range - SSL Terminated	Destination Port Range - End to End SSL	Type	Code	Comment
Ingress	CIDR	10.0.10.0/24		All protocols					
Ingress	CIDR	10.0.0.0/28		ICMP			3	4	
Ingress	CIDR	0.0.0.0/0		TCP	22	22			SSH Connections
Ingress	CIDR	10.0.2.0/28		TCP		9002			OAM Admin Server Secure Port
Ingress	CIDR	10.0.2.0/28		TCP	7001	7002			OAM Admin Server Port
Ingress	CIDR	10.0.2.0/28		TCP	14100	14101			OAM Server Port
Ingress	CIDR	10.0.2.0/28		TCP	14150	14151			OAM Policy Manager Port
Ingress	CIDR	10.0.2.0/28		TCP		9102			OIG Admin Server Secure Port
Ingress	CIDR	10.0.2.0/28		TCP	7101	9102, 7102			OIG Admin Server Port
Ingress	CIDR	10.0.2.0/28		TCP	14001	14002			OIG OIM Port
Ingress	CIDR	10.0.2.0/28		TCP	7003	7004			OIG SOA Port

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 10-4 (Cont.) Description for Ingress Rules for Security List *app-seclist*

Rule Type	Type	Source CIDR	Destination CIDR	Protocol	Destination Port Range - SSL Terminated	Destination Port Range - End to End SSL	Type	Code	Comment
Ingress	CIDR	10.0.10.0/24		TCP	5556	5556			Node Manager
Ingress	CIDR	10.0.10.0/24		TCP	All	All			All communication within the subnet
Egress	CIDR		10.0.10.0/24	All protocols					
Egress	CIDR		10.0.0.0/28	TCP		6433			
Egress	CIDR		10.0.0.0/28	TCP		12250			
Egress	CIDR		10.0.0.0/28	ICMP			3	4	
Egress	CIDR		10.0.11.0/24	TCP			1521	1521	Database Connections
Egress	Service		All Services in Oracle Service Network	TCP		443 Loadbalancer			

public-lbr-seclist

This security list determines who can access the load balancer and where the load balancer is allowed to send requests.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 10-5 Description for Ingress Rules for Security List *public-lbr-seclist*

Rule Type	Type	Source CIDR	Destination CIDR	Protocol	Description
Ingress	CIDR	0.0.0.0/0		TCP	HTTP requests from anywhere
Ingress	CIDR	0.0.0.0/0		TCP	HTTPS requests from anywhere
Egress	CIDR		10.0.2.0/28	TCP	Oracle HTTP Server. Only for SSL terminated environments.
Ingress	CIDR	10.0.10.0/24	-	TCP	Application Tier LDAP calls
Ingress	CIDR	10.0.11.0/24	-	TCP	Database Tier LDAP calls

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 10-5 (Cont.) Description for Ingress Rules for Security List *public-lbr-seclist*

Rule Type	Type	Source CIDR	Destination CIDR	Protocol	DDCommentee ss tt i i m aa tt i i oo m n PP oo rr tt RR aa m gg ee -- SE Sn Ld Tt eo rE m m i d nS aS tL e d
Egress	CIDR	-	10.0.2.0/28	TCP	84Oracle HTTP 04Server calls , 4 43 4- 34 , 4 74 79 7 7

storage-seclist

Create a security list for the OHS mount target.

Table 10-6 Description for Ingress Rules for Security List *storage-seclist*

Rule Type	Type	Source CIDR	Destination CIDR	Protocol	Source Port Range	Destination Port Range
Ingress	CIDR	10.0.10.0/24		TCP		111

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 10-6 (Cont.) Description for Ingress Rules for Security List *storage-seclist*

Rule Type	Type	Source CIDR	Destination CIDR	Protocol	Source Port Range	Destination Port Range
Ingress	CIDR	10.0.10.0/24		TCP		2048-2050
Ingress	CIDR	10.0.10.0/24		UDP		111
Ingress	CIDR	10.0.10.0/24		UDP		2048
Ingress	CIDR	10.0.1.0/29		TCP		111
Ingress	CIDR	10.0.1.0/29		TCP		2048-2050
Ingress	CIDR	10.0.1.0/29		UDP		111
Ingress	CIDR	10.0.1.0/29		UDP		2048
Ingress	CIDR	10.0.2.0/28		TCP		111
Ingress	CIDR	10.0.2.0/28		TCP		2048-2050
Ingress	CIDR	10.0.2.0/28		UDP		111
Ingress	CIDR	10.0.2.0/28		UDP		2048
Egress	CIDR		10.0.10.0/24	TCP	111	
Egress	CIDR		10.0.10.0/24	TCP	2048-2050	
Egress	CIDR		10.0.10.0/24	UDP	111	
Egress	CIDR		10.0.1.0/29	TCP	111	
Egress	CIDR		10.0.1.0/29	TCP	2048-2050	
Egress	CIDR		10.0.1.0/29	UDP	111	
Egress	CIDR		10.0.2.0/28	TCP	111	
Egress	CIDR		10.0.2.0/28	TCP	2048-2050	
Egress	CIDR		10.0.2.0/28	UDP	111	

db-seclist

Create a security list for the Database.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 10-7 Description for Ingress Rules for Security List *db-seclist*

Rule Type	Type	Source CIDR	Destination CIDR	Protocol	Destination Port Range - SSL Terminated	Description
Ingress	CIDR	0.0.0.0/0		TCP	22	SSH Calls
Ingress	CIDR	10.0.11.0/24		TCP	1521	
Ingress	CIDR	10.0.11.0/24		TCP	6200	
Ingress	CIDR	10.0.10.0/24		TCP	1521	
Ingress	CIDR	10.0.10.0/24		TCP	6200	
Ingress	CIDR	10.0.10.0/24		TCP	1389	1636
Ingress	CIDR	10.0.11.0/24		TCP	1389	1636

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 10-7 (Cont.) Description for Ingress Rules for Security List *db-seclist*

Rule Type	Type	Source CIDR	Destination CIDR	Protocol	Destination Port Range - SSL Terminated	Description
Ingress	CIDR	10.0.11.0/24		TCP	4444	4O4U4D4A dministrationPort

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 10-7 (Cont.) Description for Ingress Rules for Security List *db-seclist*

Rule Type	Type	Source CIDR	Destination CIDR	Protocol	Destination Port Range - SSL Terminated	Description
Ingress	CIDR	10.0.11.0/24		TCP	8989	8O 9U 8D 9R Replication Port

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 10-7 (Cont.) Description for Ingress Rules for Security List *db-seclist*

Rule Type	Type	Source CIDR	Destination CIDR	Protocol	Destination Port Range - SSL Terminated	Description
Ingress	CIDR	10.0.1.0/29		TCP	1521 :	Used for orsetup only
Egress	CIDR		0.0.0.0/0	All Protocols		

web-seclist

To be assigned to web-subnet (just adding this from review comment so I can add to relevant section later)

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 10-8 Description for Ingress Rules for Security List *web-seclist*

Rule Type	Type	Source CIDR	Destination CIDR	Protocol	Destination Port Range
Ingress	CIDR	0.0.0.0/0		TCP	22
Ingress	CIDR	10.0.2.0/28		TCP	111
Ingress	CIDR	10.0.2.0/28		TCP	2048-2050
Ingress	CIDR	10.0.2.0/28		UDP	111
Ingress	CIDR	10.0.2.0/28		UDP	2048
Ingress	CIDR	10.0.2.0/28		TCP	4445-4449
Ingress	CIDR	10.0.4.0/24		TCP	4445-4449
Egress	CIDR	10.0.2.0/28		TCP	111
Egress	CIDR	10.0.2.0/28		TCP	2048-2050
Egress	CIDR	10.0.2.0/28		UDP	111
Egress	CIDR	10.0.2.0/28		UDP	2048

Creating the Required Security Lists

To create the security lists in Required Security Lists:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Select **Networking** > **Virtual Cloud Networks** and click the name of your Virtual Cloud Network.
3. Select the **Security** tab.
4. Click **Create Security List**.
5. Enter the following information:
   - **Name**: Name of the security list to be created. See Required Security Lists.
   - **Compartment**: Select the name of the compartment you created earlier. See Creating an OCI Compartment.
   - Click **Add Another Ingress Rule** to add an Ingress rule as described in Required Security Lists.
   - Click **Add Another Egress Rule** to add an Egress rule as described in Required Security Lists.
6. Click **Create Security List** once all the rules are created.

Creating Route Tables

You need to create route tables which enable subnets to communicate with other subnets in the internet. In addition, you should enable access to the bastion node from the internet.

- Required Route Tables
- Creating the Required Route Tables

Copyright © 2018, 2025, Oracle and/or its affiliates.

Required Route Tables

The following table shows the route tables that need to be created.

Table 10-9 Description of Route Tables

Name	Target Type	Destination CIDR Block / Destination Service	Target Internet Gateway
bastion-route-table	Internet Gateway	0.0.0.0/0	Internet Gateway
lbr-route-table	Internet Gateway	0.0.0.0/0	Internet Gateway
web-route-table	Service Gateway	All Services in Oracle Services Network	Service Gateway
db-route-table	Service Gateway	All Services in Oracle Services Network	Service Gateway
app-route-table	Service Gateway	All Services in Oracle Services Network	Service Gateway

Creating the Required Route Tables

To create the route tables in Required Route Tables:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Select **Networking** > **Virtual Cloud Networks** and click the name of your Virtual Cloud Network.
3. Select the **Routing** tab.
4. Click **Create Route Table**.
5. Enter the following information:
   - **Name**: Name of the route table to be created. See Required Route Tables.
   - **Compartment**: Select the name of the compartment you created earlier. See Creating an OCI Compartment.
6. Click **Add Another Route Rule**.
7. In the **Rule** drop-down menu, enter the following information as described in Required Route Tables:
   - **Target Type**
   - **Destination CIDR Block** or **Destination Service**
   - **Target Internet Gateway**
8. Click **Create**.

Creating Subnets

After you create the security lists and route tables, you must create subnets and assign the route tables and security lists to them.

- Required Subnets
- Creating the Required Subnets

Copyright © 2018, 2025, Oracle and/or its affiliates.

Required Subnets

The following table shows the subnets that need to be created.

Table 10-10 Description of Subnets

Name	Subnet Type	IPv4 CIDR Block	Route Table	Subnet Access	Security Lists
bastion-subnet	Regional	10.0.1.0/29	bastion-route-table	Public	bastion-public-seclist setup-seclist
app-subnet	Regional	10.0.10.0/24	app-route-table	Private	bastion-private-seclist app-seclist storage-seclist web-seclist setup-seclist
web-subnet	Regional	10.0.2.0/28	web-route-table	Private	web-seclist storage-seclist setup-seclist
db-subnet	Regional	10.0.11.0/24	db-route-table	Private	db-seclist setup-seclist storage-seclist
lbr-subnet	Regional	10.0.4.0/24	lbr-route-table	Public	public-lbr-seclist

> **Note**
>
> setup-seclist is only required during initial setup.

Creating the Required Subnets

To create the subnets in Required Subnets:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Select **Networking** > **Virtual Cloud Networks** and click the name of your Virtual Cloud Network.
3. Select the **Subnets** tab.
4. Click **Create Subnet**.
5. Enter the following information as per Required Subnets:
   - **Name**: Name of the subnet to be created..
   - **Compartment**: Select the name of the compartment you created earlier. See Creating an OCI Compartment.
   - **Subnet Type**
   - **IPv4 CIDR Block**
   - **Route Table**

Copyright © 2018, 2025, Oracle and/or its affiliates.

- **Subnet Access**
- **DNS Resolution**: Select **Use DNS Hostnames in the subnet**.
- **DHCP Options**: Select **Default DHCP Options for the VCN**.
- **Security Lists**: Select the secuirty list, and click **Add Another Security List** if needed.

6. Click **Create Subnet**.

Creating Compute Instances

Create the compute instances in your OCI tenancy for the deployment.

- Required Compute Instances
- Creating the Required Compute Instances
- Assigning a Secondary IP Address to a Compute Instance
- Connecting to a Compute Instance
- Configuring the Compute Instances (General)
- Configuring the Bastion Node
- Configuring the OHS Nodes

Required Compute Instances

The following table shows the compute instances that need to be created.

Table 10-11 Compute Instances to Create

Name	Min CPUs	Min Memory (GB)	Subnet	Public IP Address
bastion	1	8	bastion-subnet	Yes
webhost1	1	16	web-subnet	No
webhost1	1	16	web-subnet	No
oamhost1	4	32	app-subnet	No
oamhost2	4	32	app-subnet	No
oighost1	4	32	app-subnet	No
oighost2	4	32	app-subnet	No
ldaphost1	2	16	db-subnet	No
ldaphost2	2	16	db-subnet	No

Creating the Required Compute Instances

To create the compute instances in Required Compute Instances :

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Select **Compute** > **Instances**.
3. Click **Create Instance**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

4. In **Basic Information** enter the following information using the details in Required Compute Instances :
   - **Name**: Name of the compute instance to be created. .
   - **Compartment**: Select the name of the compartment you created earlier. See Creating an OCI Compartment.
   - **Placement**: Select an **Availability Domain**.
   - **Image**: Select the operating system image you want to use. For example: **Oracle Linux 9**
   - **Shape**: Click **Change Shape** Select an architecture and shape you want to use. For example: **VM.Standard.E4.Flex**.
     On the selected image, expand the selected **Shape name** to customize the number of CPUs and memory to assign to the Compute Instance.
   - Click **Next** and **Next** again.
5. In **Networking** enter the following information:
   - **Network** : Select your VCN.
   - **Subnet**: Select which Subnet to use. If you need a public IP address select**Automatically assign public IPv4 Address**.
   - **Add SSH Keys**: Select**Paste public key**. Copy the contents of the `id_rsa.pub` file that you created earlier. See Creating an SSH Key Pair.
6. Click **Next** and review the summary.
7. Click **Create**.

The summary screen displays the IP addresses assigned to the compute instance. Make a note of this address. You will need it for connecting to the node.

Assigning a Secondary IP Address to a Compute Instance

In an Enterprise deployment you need to assign an additional IP address to the hosts OAMHOST1 and OIGHOST1.

This IP address is used for Admin Server Failover. The Admin Server listens on this secondary IP address. If the host to which the IP address becomes unavailable, the IP address can be assigned to a different host and the admin server is started there.

Before you can add a secondary IP address, you need to look for an IP address in the same subnet as the main IP address that is not being used. For example, if the IP address of your OAMHOST1 is 10.0.10.96 then you may see that IP address 10.0.10.97 is not being used, and this can therefore be used as a Virtual IP address.

To add the Virtual IP address to the network card, perform the following steps:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Select **Compute** and click **Instances**.
3. Click on the Compute instance you want to add the IP address to, for example OAMHOST1.
4. Select the **Networking** tab.
5. Click on the VNIC listed in the **Attached VNICs**. This will most likely have the same name as the compute instance.
6. In the **VNIC Information** select the **IP administration** tab.

Copyright © 2018, 2025, Oracle and/or its affiliates.

7. In **IPv4 Addresses**, click **Assign Secondary IP Address**
8. Enter the IP Address you wish to add and a hostname.
9. Click **Assign**.
10. Repeat the above steps for OIGHOST1.

Connecting to a Compute Instance

You can now connect to the compute instances using the following SSH command:

```
ssh -i id_rsa opc@IPAddress
```

Alternatively, if you are using SSH agent forwarding, which enables you to use your local **SSH** keys instead of leaving the keys (without passphrases) on the server, then you can use the following command:

```
ssh -A opc@IPAddress
```

To install the Oracle binaries, you will need access to a graphical display. To get this access, you should use X11 Forwarding. Ensure that on your desktop/laptop, you install an 'X' Window server. For example: `XQuartz for MacOS`. You will then need to connect as the oracle user, for example:

```
ssh -AX oracle@IPAddress
```

Configuring the Compute Instances (General)

After creating the compute instances, you need to configure them.

You must perform the following steps on all the compute instances unless otherwise stated:

- Installing X11 and Additional Packages
- Enabling X11 Forwarding
- Editing Compute Instance Fault Domain
- Creating a Software Owner Account
- Creating Local Directories
- Creating Mount Points for Shared Storage
- Preparing the Compute Instance
- Using the Firewall

Installing X11 and Additional Packages

In order to install the Identity Management products on the compute notes, you must install some Linux packages on all the compute nodes (LDAPHOSTn, OAMHOSTn, OIGHOSTn, WEBHOSTn).

Copyright © 2018, 2025, Oracle and/or its affiliates.

Run the following command to install the packages on each of the compute nodes:

```
sudo yum install -y libXrender libXtst xauth xterm nc gtk3* at-spi2-cor*
libXScrnSaver* libnotify* openldap* java
```

For the webhost (WEBHOST1 and WEBHOST2), you must also install additional packages. Run the following command to install the packages on the webhosts:

```
sudo yum install -y xorg-x11* libaio-devel* gcc-c++-* ksh* libnsl* xorg-x11-
utils* libstdc++-devel sysstat*
```

Enabling X11 Forwarding

Configure SSHD to not use localhost for X11:

1. Open `/etc/ssh/sshd_config` in your preferred editor.

   ```
   sudo vi /etc/ssh/sshd_config
   ```

2. Search for the line that has "X11UseLocalhost yes" (it is commented out).
3. Remove the comment from the beginning of the line.
4. Change the `yes` to `no`.
5. Save the file.
6. Restart SSHD by using the following command:

   ```
   sudo systemctl restart sshd
   ```

Editing Compute Instance Fault Domain

Each WEBHOST, OAMHOST, OIGHOST and LDAPHOST instance should reside in a different fault domain. Perform the following steps:

1. Click the compute instance.
2. Click **Actions** > **More Actions** and select **edit**.
3. Ensure each compute instance of a specific type is in a different fault domain, if not click **Edit Fault domain** and choose a different fault domain.
4. Click **Save changes**.

   Repeat the steps for each compute instance.

Creating a Software Owner Account

It is not good practice to install the Oracle software using the OPC user. It is better to create a custom user to own the software. You can create a custom user by running the following commands:

```
sudo adduser -u 1001 oracle
sudo groupadd -g 1002 oinstall
sudo usermod -a -G oinstall oracle
sudo usermod -g oinstall oracle
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

Creating Local Directories

If your deployment requires the use of local directories, for example local binaries, local instance/domain files or certificates, then create local directories.

If you are using `/u02` as described in this Enterprise Deployment Guide, you will need to create these local directories as `root` and then change the ownership to your `oracle` account.

```
sudo mkdir -p /u02/oracle/config /u02/oracle/products
```

```
sudo chown -R oracle:oinstall /u02
```

For a list of locations to create, see Preparing the File System for an Enterprise Deployment.

Creating Mount Points for Shared Storage

Before you can mount shared storage to a host you must pre-create the location to which it will be attached. For example:

```
sudo mkdir -p /u01/oracle/products
```

```
sudo chown -R oracle:oinstall /u01/oracle
```

For a list of locations to create, see Preparing the File System for an Enterprise Deployment.

Preparing the Compute Instance

For each compute instance you must also perform the tasks outlined in Preparing the Host Computers for an Enterprise Deployment.

Using the Firewall

The compute instance is created using an Oracle Linux image. The image comes with a built-in firewall, which is enabled by default. Even though you have security rules defined in your network, the Linux server rejects these requests because of the built-in Linux firewall.

You can decide to use this extra firewall or rely on your OCI security rules.

- Opening the Ports in the Firewall
- Disabling the Firewall

Opening the Ports in the Firewall

If you decide to use the firewall, you need to add firewall rules that enable every port coming in to the server to be allowed.

1. For every port that needs to be accessed, execute the following command:

   ```
   sudo firewall-cmd --permanent --add-port=YOUR PORT/tcp
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

For example:

```
sudo firewall-cmd --permanent --add-port=7777/tcp
```

2. Restart the firewall service after you configure all the ports. Use the following command to restart:

```
sudo systemctl restart firewalld
```

3. Validate the firewall configuration by executing the following command:

```
sudo firewall-cmd --list-ports
```

Disabling the Firewall

To disable the firewall, run the following commands:

```
sudo systemctl stop firewalld


sudo systemctl disable firewalld
```

Configuring the Bastion Node

Perform the following steps to configure the bastion node:

- Installing Git
- Setting Up the Hosts File

Installing Git

Git contains sample code to deploy Oracle Fusion Middleware. Install GIT using the following command on the bastion only:

```
sudo yum install git -y
```

Setting Up the Hosts File

When setting up, you should make `curl` commands to the load balancer. Because the bastion node uses the private DNS, the IP addresses returned for the load balancer end points is through the internal network that the bastion host does not have access to. To get around this issue, create an entry in the bastion hosts file for each entry point that points to the public IP address of the load balancer.

> **Note**
>
> You cannot perform this step until you have created the load balancers. See Creating Load Balancers.

Copyright © 2018, 2025, Oracle and/or its affiliates.

For example, if the public IP address of the load balancer is 129.1.1.3, add the following entry to the bastion hosts file:

```
129.1.1.3 login.example.com oig.example.com iadadmin.example.com
igdadmin.example.com
```

Configuring the OHS Nodes

Perform the following steps to configure the OHS nodes:

- Preparing the Compute Instance for Use by Oracle HTTP Server
- Preparing the Hosts File

Preparing the Compute Instance for Use by Oracle HTTP Server

You may also need to install additional Linux packages required to install the Oracle HTTP server, as well as setting the kernel parameters. See Preparing the Host Computers for an Enterprise Deployment.

Preparing the Hosts File

The nature of the networks in an OCI environment means that the Oracle HTTP Server instances will not have access to the public load balancer. This can cause issues when the Oracle HTTP Server tries to access some virtual hosts.

In later sections, you will create a public load balancer for connections from the outside world to your system. See Creating a Public Load Balancer.

You will also create a private load balancer to allow you to route requests from the private subnets to this load balancer. See Creating a Private Load Balancer.

To ensure that the requests from the Oracle HTTP Server are directed to the private load balancer rather than the public, you should create an entry in the `/etc/hosts` file on the web hosts, which looks as follows:

```
PRIVATE_LOAD_BALANCER_IP login.example.com
```

For example:

```
10.0.2.7 login.example.com
```

Creating File Systems and Mount Targets

You need to create NFS file systems for Persistent Volumes and Oracle HTTP Server installations.

The filesystems that you have to create are described in Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment.

- Overview of Preparing the File System for an Enterprise Deployment
- Summary of File Systems
- Creating a File System

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Setting the Mount Target Storage Reporting
- Mounting File Systems on Hosts

Overview of Preparing the File System for an Enterprise Deployment

It is important to set up your storage in a way that makes the enterprise deployment easy to understand, configure, and manage.

This chapter provides an overview of the process of preparing the file system for an enterprise deployment. Oracle recommends setting up your storage according to information in this chapter. The terminology defined in this chapter is used in the diagrams and procedures throughout the guide.

Summary of File Systems

The following table lists a summary of the hosts and file systems to be mounted.

You have to mount the file systems to the bastion node only during the initial set up.

Table 10-12 Summary of Hosts and the File Systems to be Mounted

Mount Host	File Systems	Exports	Mount Directories	Comments
All nodes	installers	`/exports/installers`	`/installers`	Used as a staging directory to temporarily store installers and patches.
oamhost1	oamBinaries1	`/exports/oamBinaries1`	`/u01/oracle/products`	
oamhost1	oamConfig	`/exports/oamSharedConfig`	`/u01/oracle/config`	
oamhost2	oamBinaries2	`/exports/oamBinaries2`	`/u01/oracle/products`	
oighost1	oigBinaries1	`/exports/oigBinaries1`	`/u01/oracle/products`	
oighost1	oigConfig	`/exports/oigSharedConfig`	`/u01/oracle/config`	
oighost2	oigBinaries2	`/exports/oigBinaries2`	`/u01/oracle/products`	

Creating a File System

To create a file system:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Select **Storage** and click **File Systems**.
3. Click **Create File System**.
4. Select **Filesystem for NFS**.
5. Click **Edit Details** in the **File System Information** section.
6. Enter the following details:

Copyright © 2018, 2025, Oracle and/or its affiliates.

- **Name**: Provide a name for the file system. For example: `oamBinaries1`.
- **Compartment**: Select the compartment you created earlier. See Creating an OCI Compartment.

7. Click **Edit Details** in the **Export Information** section.
8. Enter the following:
   - **Export Path**: This is the path you want to export. For example: `/exports/oamBinaries1`.
9. Click **Edit Details** in the **Mount Target Information** section.
10. Enter the following details:
    - **Mount Target Name**: Specify a name for the mount target. For example: `APPVolumes` (Application Servers).
    - **Virtual Cloud Network**: Select the **VCN**.
    - **Subnet**: For the persistent volumes, select the `app-subnet`.
11. Click **Create**.

> **Note**
>
> Create a new mount target only for the first persistent volume (PV). Subsequent PVs should use the same mount target.

Setting the Mount Target Storage Reporting

When you install Oracle products, the installer checks the available disk storage. This check fails when you use an OCI file system. The system displays a message saying that there is insufficient disk space. To overcome this error, you can configure OCI to report a specified amount of free space.

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Select a **Mount Target**, for example `AppVolumes`
3. Select **Storage** and click **Mount Targets**.
4. Click the **NFS** tab.
5. Click **Edit** next to the **Reported Size (GB)** (it looks like a pencil).
6. Set an arbitrary size value. For example: 20.

   This value ensures that the file system, when mounted on the OHS nodes, reports 20GB of free space. This enables the OHS installer to proceed.
7. Click **Save**.
8. Repeat the above steps for each mount target.

Mounting File Systems on Hosts

Each mount target has a different IP address. To determine how to mount a given file system:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Select **Storage** and click **File Systems**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

3. Select a file system.
4. On the File System screen, select an export from the list of exports.
5. Click **Mount Commands** at the top of the screen, to view examples of the mount command.
6. For OAM hosts, place the entries in `/etc/fstab` with the following mount options:

   Sample OAM `/etc/fstab` entry:

   ```
   <IP>:/exports/oamBinaries1 /u01/oracle/products nfs
   auto,rw,bg,hard,nointr,tcp,vers=3,timeo=300,rsize=32768,wsize=32768
   <IP>:/exports/oamSharedConfig /u01/oracle/config nfs
   auto,rw,bg,hard,nointr,tcp,vers=3,timeo=300,rsize=32768,wsize=32768
   ```

   Before you can use the file system ensure that you can write to it. Mount the file system to the bastion node and write to it. If you are unable to write, use the `chmod` command to enable writing to the file system.

   For example:

   ```
   sudo mkdir -p /u01/oracle/products /u01/oracle/config


   sudo mount -a


   sudo chmod -R 777 /u01/oracle
   ```

7. Run the following command to pick up the changes:

   ```
   sudo systemctl daemon-reload
   ```

Creating Load Balancers

You need to create two OCI load balancers. One of these load balances is used to direct public traffic and the other for internal call backs. The load balancer used for internal traffic is not available outside the OCI container.

For more information about load balancers, see Getting Started with Load Balancing.

- Creating a Public Load Balancer
- Creating a Private Load Balancer
- Creating an LDAP Load Balancer

Creating a Public Load Balancer

This load balancer directs traffic from the internet to the Oracle HTTP Servers, which in turn pass on the traffic to the Application tier.

The public load balancer will send traffic to and from the user via SSL. After the traffic moves inside the OCI Virtual Network, it is sent unencrypted in an SSL terminated topology, and encrypted in an end to end SSL topology. You will need to provide your own SSL certificate or create a self-signed certificate for testing purposes.

Copyright © 2018, 2025, Oracle and/or its affiliates.

To create a public load balancer, perform the following steps:

- Creating a Self-Signed Certificate
- Creating a Load Balancer
- Uploading Load Balancer Certificates
- Creating Host Names
- Creating Backend Sets
- Creating Listeners
- Updating the Default Listener

Creating a Self-Signed Certificate

You can create a self-signed certificate on any host which has access to the `openssl` packages. The following example is from a Linux box (in this case the bastion server was used).

For more information, see Doc ID 2617046.1.

If you prefer, you can also use a certificate provided by a recognized certificate authority.

To create a self-signed certificate:

1. Create the CA (certificate authority) private key by using the following command:

```
openssl genrsa -out ca.key 2048


Generating RSA private key, 2048 bit long modulus
.....................+++
.....+++
e is 65537 (0x10001)
```

2. Create the Certificate Signing Request (CSR).

```
openssl req -new -key ca.key -out ca.csr


You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:CR
State or Province Name (full name) []:SJO
Locality Name (eg, city) [Default City]:
Organization Name (eg, company) [Default Company Ltd]:mycompany
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:*.example.com
Email Address []:
Please enter the following 'extra' attributes
to be sent with your certificate request
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
A challenge password []:
An optional company name []:
```

3. Create the CA SIGN certificate that will be used to sign the new certificates.

```
openssl x509 -req -in ca.csr -signkey ca.key -out ca.crt


Signature ok
subject=/C=CR/ST=SJO/L=Default City/O=mycompany/CN=*.example.com
Getting Private key
```

4. Create the private key for the load balancer.

```
openssl genrsa -out loadbalancer.key 2048


Generating RSA private key, 2048 bit long modulus
.....................+++
.....+++
e is 65537 (0x10001)
```

5. Create the CSR for the load balancer.

```
openssl req -new -key loadbalancer.key -out loadbalancer.csr


You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:CR
State or Province Name (full name) []:SJO
Locality Name (eg, city) [Default City]:
Organization Name (eg, company) [Default Company Ltd]:mycompany
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:*.example.com
Email Address []:
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

6. Sign the certificate with the CA certificate.

```
openssl x509 -req -in loadbalancer.csr -CA ca.crt -CAkey ca.key -
CAcreateserial -out loadbalancer.crt -days 50000


Signature ok
subject=/C=CR/ST=SJO/L=Default City/O=mycompany/CN=*.example.com
Getting CA Private Key
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

7. Check that the certificate is signed by the CA.

```
openssl x509 -in loadbalancer.crt -text


Certificate:
Data:
Version: 1 (0x0)
Serial Number:
df:e7:c9:6a:56:e5:e4:c9
Signature Algorithm: sha256WithRSAEncryption
Issuer: Issuer: C=CR, ST=SJO, L=Default City, O=mycompany,
CN=*.example.com <==== here signed by my ca..
Validity
Not Before: Dec 3 16:34:58 2019 GMT
Not After : Oct 25 16:34:58 2156 GMT
Subject: =/C=CR/ST=SJO/L=Default City/O=mycompany/CN=*.example.com
Subject Public Key Info:
Public Key Algorithm: rsaEncryption
Public-Key: (2048 bit)
Modulus:
00:da:62:ce:69:77:ff:45:b0:84:9f:af:53:44:97:
13:28:91:44:cd:0b:1d:e5:a1:f6:a3:ef:f8:98:19:
8d:c2:56:a0:e1:80:1c:e0:0e:ae:34:9a:a8:ae:52:
d4:71:a4:da:10:8b:fd:df:73:0d:8e:98:ef:d4:7b:
36:f1:1c:5a:d7:24:88:63:f5:b2:6b:7a:62:50:3a:
e7:3a:3d:9a:b7:41:db:8e:f5:e8:91:46:48:cf:0c:
54:da:7b:da:20:76:b6:eb:4b:cb:fa:36:09:f7:94:
ea:c9:53:3f:b2:bc:66:4c:6d:7f:3f:09:cc:cd:c2:
10:1f:39:0f:6c:1d:49:7c:db:99:d9:d9:7d:48:dd:
09:52:50:9d:f5:44:fd:2e:48:f2:78:22:20:3c:07:
b6:a1:4d:f8:17:82:67:a1:45:52:0a:21:78:ed:1b:
ca:45:79:16:21:c9:e3:2f:a4:93:d4:bf:67:68:7a:
b6:d9:8f:e1:53:35:31:a6:17:38:f2:a6:79:b5:12:
6b:36:f2:2d:69:56:c2:d9:c0:89:d9:31:6b:06:0c:
1e:ba:a6:30:88:32:7b:92:e4:af:11:ab:37:1a:cb:
cf:4b:4c:7d:ff:a7:4d:f8:be:cd:98:17:63:83:06:
cf:e7:ae:4a:d5:6e:6b:e4:0d:f3:6f:70:52:2b:8b:
12:83
Exponent: 65537 (0x10001)
Signature Algorithm: sha256WithRSAEncryption
d8:36:2e:2e:42:72:76:15:ec:a8:3a:e9:dd:2d:2e:28:42:97:
48:4e:6f:33:ec:df:3e:a3:11:19:8b:62:d5:89:07:af:b5:ff:
b6:de:d7:5c:8b:7a:46:37:46:da:b7:44:7f:b6:cc:c8:a9:1e:
f9:ca:0f:76:2b:29:d2:4c:6a:af:18:9b:1a:62:42:87:e6:21:
b7:09:15:8d:b3:1d:05:4a:4d:1b:d1:07:00:cd:69:40:92:ed:
f9:3d:24:c9:b7:b9:00:7e:c3:f9:73:42:7f:13:34:a8:d1:e4:
32:91:08:51:07:a5:d0:ab:42:fb:83:c4:a7:b5:94:0f:2a:56:
8b:95:34:1b:63:5b:39:59:88:9b:9f:34:91:98:dc:8c:0a:0e:
01:f9:b2:6e:fd:2e:95:28:4c:76:dd:fe:a0:3f:f1:16:3b:88:
cd:e5:0a:f3:dd:52:0d:39:2a:60:2c:f0:5d:79:3b:7e:99:43:
3b:47:33:85:f9:7c:f1:e8:cb:3d:cd:ab:4c:1f:a2:72:99:70:
f4:8d:92:4a:24:9e:37:96:ad:24:d5:13:33:05:32:ae:d5:58:
ed:3e:32:6f:a7:1e:a8:61:a5:fb:73:ea:54:46:b7:07:77:07:
9a:9d:af:eb:66:5c:55:f1:50:23:fb:da:d9:b7:4b:0b:6d:bb:
c7:39:18:ae
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
-----BEGIN CERTIFICATE-----
MIIDUDCCAXXXXXXXXXXXXXXXXXXXXXXXXXXXXXAQsFADBaMQswCQYDVQQGEwJD
UjEMMAoGA1UECAwDU0pPMRUwEwYDVQQHDAxEZWZhdWx0IENpdHkxDzANBgNVBAoM
Bm9yYWNsZTEVMBMGA1UEAwwMKi5vcmFjbGUuY29tMCAXDTE5MTIwMzE2MzQ1OFoY
DzIxNTYxMDI1MTYzNDU4WjB4MQswCQYDVQQGEwJDUjEMMAoGA1UECAwDU0pPMRUw
EwYDVQQHDAxEZWZhdWx0IENpdHkxHDAaBgNVBAoME0RlZmF1bHQgQ29tcGFueSBM
dGQxDzANBgNVBAsMBm9yYWNsZTEVMBMGA1UEAwwMKi5vcmFjbGUuY29tMIIBIjAN
BgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA2mLOaXf/RbCEn69TRJcTKJFEzQsd
5aH2o+/4mBmNwlag4YAc4A6uNJqorlLUcaTaEIv933MNjpjv1Hs28Rxa1ySIY/Wy
a3piUDrnOj2at0HbjvXokUZIzwxU2nvaIHa260vL+jYJ95TqyVM/srxmTG1/PwnM
zcIQHzkPbB1JfNuZ2dl9SN0JUlCd9UT9LkjyeCIgPAe2oU34F4JnoUVSCiF47RvK
RXkWIcnjL6ST1L9naHq22Y/hUzUxphc48qZ5tRJrNvItaVbC2cCJ2TFrBgweuqYw
iDJ7kuSvEas3GsvPS0x9/6dN+L7NmBdjgwbP565K1W5r5A3zb3BSK4sSgwIDAQAB
MA0GCSqGSIb3DQEBCwUAA4IBAQDYNi4uQnJ2FeyoOundLS4oQpdITm8z7N8+oxEZ
i2LViQevtf+23tdci3pGN0bat0R/tszIqR75yg92KynSTGqvGJsaYkKH5iG3CRWN
sx0FSk0b0QcAzWlAku35PSTJt7kAfsP5c0J/EzSo0eQykQhRB6XQq0L7g8SntZQP
KlaLlTQbY1s5WYibnzSRmNyMCg4B+bJu/S6VKEx23f6gP/EWO4jN5Qrz3VINOSpg
LPBdeTt+mUM7RzOF+Xzx6Ms9zatMH6JymXD0jZJKJJ43lq0k1RMzBTKu1VjtPjJv
px6oYaX7c+pURrcHdweana/rZlxV8VAj+9rZt0sLbbvHORiu
-----END CERTIFICATE-----
```

This procedure creates the following files to be used later. See Uploading Load Balancer Certificates.

- `ca.crt`
- `loadbalancer.crt`
- `loadbalancer.key`

Creating a Load Balancer

To create a load balancer:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking** and click **Load Balancers**.
3. Select **Load Balancer** and click **Create Load Balancer**.
4. Enter the following information:
   - **Name**: Enter a name for the load balancer. For example: `public-loadbalancer`.
   - **Visibility Type**: Select **Public**.
   - Select **Assign a Public IP Address** and **Ephemeral IP Address** unless you want to use a specific IP address, in which case select **Reserved IP Address**.
   - **Shapes**: Select **Flexible Shape**.
   - **Bandwidth**: Select the anticipated bandwidth.
   - **Virtual Cloud Network**: Select the Virtual Cloud Network.
   - **Subnet**: Select the load balancer subnet you created earlier. See Creating Subnets for the Load Balancer.
5. Click **Next**.
6. On the Choose Back Ends screen, select your preferred **Load Balancing Policy**.
7. Click **Add Back Ends**.
   a. Select **Web Server Instances**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

b. Click **Add Selected Back Ends**.

c. Change the port to the listen port for Oracle HTTP Server. For example, `7777` (SSL Terminated) and `4447` (End to End SSL).

8. In the Specify Health Check Policy screen, change the port to the port used in the backend set. For example, `7777` or `4447`.
9. In the URL path enter `/health-check.html`.
10. Enter a name for the back-end set. For example: `ohs_servers`.

> **Note**
>
> In an SSL Terminated deployment, all listeners use the same backend set. In End to End SSL deployments, there is be a backend set for each host name. For consistency, call the backend set `ohs_hostname`. For example, `ohs_login`.

11. Select **Enable load balancer cookie persistence** in the Session Persistence section.
12. Click **Next**.
13. On the Configure Listener screen, enter the following information:

> **Note**
>
> You will need one listener for each entry point. However, you can add only one listener at this point.

- **Name**: Select a name for the listener. For example: `login_listener`.
- **Traffic Type**: Select `HTTPS`.
- **Port**: Select the load balancer port. `login.example.com` uses port `443`.
- **Certificate Resource**: Load Balancer Managed Certificate.
- **SSL Certificate**: Paste the contents of the certificate (.crt) file of the certificate you have generated/purchased for the host `login.example.com`.
- **CA Certificate**: Paste the contents of the CA Certificate that was used to generate the SSL certificate above.
- **Private Key**: Paste the contents of the key file which was used in the generation of the SSL certificate.

14. Click **Next**.
15. Ensure that the Compartment is set to your compartment.
16. In the Manage Logging screen, ensure that **Create a New Log Group** is selected.
17. Change **Name**. For example: `Public_Lbr`.
18. Change **Log Name**. For example: `Public_lbr_error`.
19. Click **Next**
20. Click **Submit**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Uploading Load Balancer Certificates

As the load balancer routes SSL requests, you need to upload the certificates for the load balancer. If you have created a self-signed certificate, add the details of that certificate. If you have your own certificates, upload those.

To upload the certificates:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking** and click **Load Balancers**.
3. Click the load balancer. For example: **public_loadbalancer**.
4. Select **Certificates** from the Certificates and Ciphers tab.
5. Select **Load Balancer Managed Certificate**.
6. Click **Add Certificate**.
7. Enter the following information. You can either upload the files directly or paste the contents of the files.
   - **Certificate Name**: Enter a name for the certificate. For example: `login_cert`.
   - **SSL Certificate**: Include the contents of the `login.crt` file.
   - **CA Certificate**: Select the **Specify CA Certificate** check box to include the contents of the `ca.crt` file.
   - **Private Key**: Select the **Specify Private Key** check box to include the contents of the `login.key` file.

   See Creating a Self-Signed Certificate.
8. Click **Add Certificate**.
9. Repeat the above step for each certificate you wish to upload.

Creating Host Names

Host names are used to filter the different entry points into the load balancer. You need to create a host name for each load balancer virtual host described in Summary of the Virtual Servers Required for an Enterprise Deployment.

You have to create the following host names:

- `login.example.com`
- `oig.example.com`
- `iadadmin.example.com`
- `igdadmin.example.com`
- `igdinternal.example.com`

To create the load balancer host names:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking** and click **Load Balancers**.
3. Click the load balancer. For example: **public_loadbalancer**.
4. Select the **Hostnames** tab.

Copyright © 2018, 2025, Oracle and/or its affiliates.

5. Click **Create Hostname**.
6. Enter the following information:
   - **Name**: Enter a name for the host name. For example: `login_hostname`.
   - **Hostname**: Enter the fully qualified host name. For example: `login.example.com`.
7. Click **Create**.
8. Repeat for each host name to be created.

> **Note**
>
> If you want to limit the admin access to users inside the network, you should create the hosts `iadadmin.example.com` and `igdadmin.example.com` in the private load balancer.

Creating Backend Sets

In End to End SSL deployments, you need to create additional backend sets that points to each OHS virtual host as outlined below. If using SSL Terminated, you can ignore this section.

Table 10-13 End to End SSL Deployments

Name	SSL	Healthcheck	Backends	Backend Port
ohs_login	Yes	https:4447	webhost1, webhost2	4447
ohs_oig	Yes	https:4448	webhost1, webhost2	4448

Perform the following steps to create a backend set:

1. Log into the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking** and click **Load Balancers**.

   For example, **public_loadbalancer**.
3. Click the **Backends** tab.
4. Click **Create Backend Set**.
5. Enter information for each field as described below:

   Name - For example, `ohs_iadadmin`.

   Use SSL - Select to use SSL.

   Certificate Resource - Select **Load Balancer Managed Certificates**.

   Certificate name - Enter the certificate name corresponding to the Oracle HTTP Servers CA. For example, `ohs_ca`. As all OHS certificates will have been created using the same CA, the CA created as part of the config wizard can be reused, This will have a system generated name similar to `cert_bs_<DATESTAMP>`.

   Verify Peer Certificate - Select **Enabled**.

   Verify Depth - Select **1**.

   Session Persistence - Select **Load balancer cookie persistence**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Health Check - Enter **https** and the OHS listen port. For example, `4445`.

URL Path - `/heathcheck.html`.

6. Click **Create backend set**.
7. After the backend set is created, click its name. For example, iadadmin.
8. Select the **Backends** tab.
9. Click **Add Backends**.
10. Select **Compute Instances** and click **Add Instances**.
11. Select the compute instances where your OHS servers reside. For example, **webhost1** and **webhost2**.
12. Click **Add Instances**.
13. In the port field, select the OHS port you want to use. For example, `4445`.
14. Click **Add**

Creating Listeners

You need to create a listener for each host name you have created earlier. See Creating Host Names. The `login` listener has been created at the time of creating the load balancer. See Creating a Load Balancer.

Table 10-14 SSL Terminated Deployments

Name	Protocol	Port	SSL	Backend Set	Host Name
login (default)	https	443	Yes	ohs_servers	login_hostname
oig	https	443	Yes	ohs_servers	oig_hostname

Table 10-15 End to End SSL Deployments

Name	Protocol	Port	SSL	Backend Set	Host Name
login_listener (default)	https	443	Yes	ohs_login	login_hostname
oig_listener	https	443	Yes	ohs_oig	oig_hostname

> **Note**
>
> In both of the above deployments the administration URLs are not included in the public loadbalancer. This ensures that it is physically impossible to access administration functions from the internet. They are available from inside the network or via a browser running on your bastion host.
> Should you have a requirement to expose these end points to the internet you will need to add in extra listeners for the administration functions. Access to these URLs should however be tightly controlled via appropriate security lists.

To create the load balancer listeners:

1. Log into the Oracle Cloud Infrastructure Console for your tenancy.

Copyright © 2018, 2025, Oracle and/or its affiliates.

2. Navigate to **Networking** and click **Load Balancers**.
3. Click the load balancer. For example: **public_loadbalancer**.
4. Select the **Listeners** tab.
5. Click **Create Listener**.
6. Enter the following information:
   - **Name**: Enter a name for the listener. For example: `oig_listener`.
   - **Protocol**: Select **https**.
   - **Port**: Specify `443`.
   - **Certificate Resource**: Select **Load balancer managed certificate**.
   - **Certificate Name**: Ensure that the certificate you created for the load balancer is displayed. If not displayed, select the certificate.

     > **Note**
     >
     > This option will be available only if you use the `HTTPS` protocol.

   - **Verify Peer Certificate** - Select **Enabled**.
   - **Verify Depth** - Select **1**.
   - **Hostname**: Select the hostname associated with your listener. For example: `oig`.
   - **Backend Set**: Select the backend set. For example: **ohs_oig**.
7. Click **Create Listener**.
8. Repeat the steps to create the remaining listeners.

> **Note**
>
> If you want to limit the admin access to users inside the network, create the listeners `iadadmin_listener` and `igdadmin_listener` in the private load balancer.

Updating the Default Listener

When you created the load balancer, a default listener also gets created. You have to assign the newly created host name to this listener.

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking** and click **Load Balancers**.
3. Click the load balancer, for example: **public-loadbalancer**.
4. Select the **Listeners** tab.
5. To edit the listener, click the three dots next to the name, for example **login**, and then click **Edit**.
6. Set the host name to the host name you created earlier, for example: **login**. See Creating Host Names.
7. Click **Save changes**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Creating a Private Load Balancer

The private load balancer, which is used to route internal call backs, resides in the same subnet as the Oracle web servers. This load balancer services requests generated from inside the application.

> **Note**
>
> Web servers issue `curl` commands to `login.example.com`. Therefore, you also need to define this on the private load balancer because the web servers do not have direct access to the public load balancer. You can use the same certificates that you used when you created the public load balancer.

To create a private load balancer, perform the following steps:

- Creating a Load Balancer
- Creating Host Names
- Updating the Default Listener
- Uploading Load Balancer Certificates
- Creating Backend Sets
- Creating Listeners

Creating a Load Balancer

To create a load balancer:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking** and click **Load Balancers**.
3. Click **Create Load Balancer**.
4. Select **Load Balancer** and click **Create Load Balancer**.
5. Enter the following information:
   - **Name**: Enter a name for the load balancer. For example: `internal-loadbalancer`.
   - **Visibility Type**: Select **Private**.
   - **Shapes**: Select **Flexible Shape**.
   - **Bandwidth**: Select the anticipated bandwidth.
   - **Virtual Cloud Network**: Select the Virtual Cloud Network.
   - **Subnet**: Select the same subnet as the web servers. For example: `web-subnet`. See Creating Subnets for the Load Balancer.
6. Click **Next**.
7. On the Choose Back Ends screen, select your preferred **Load Balancing Policy**.
8. Click **Add Instances**.
   a. Select **Web Server Instances**.
   b. Click **Add Instances**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

c. Change the port to the listen port for Oracle HTTP Server. For example: `7777` (SSL Terminated) and `4447` (End to End SSL).

9. In the Specify Health Check Policy section, change the port to the port used in the backend set. For example, `7777` and `4447`.
10. In the Back End Set section, enter a name for the back-end set. For example: `ohs_servers`.

> **Note**
>
> In an SSL Terminated deployment, all listeners use the same backend set. In End to End SSL deployments, there is a backend set for each host name. For consistency, call the backend set old_hostname. For example, ohs_login.

11. In the Session Persistence section select **Enable Loadbalancer cookie persistence**.
12. Click **Next**.
13. On the Configure Listener screen, enter the following information:
    - **Name**: Select a name for the listener. For example: `login`.
    - **Traffic Type**: Select the traffic type the listener uses. `login_listener` uses **HTTPS**.
    - **Port**: Select the load balancer port. `login.example.com` uses port `443`.
    - **Certificate Resource**: Load Balancer Managed Certificate.
    - **SSL Certificate**: Paste the contents of the certificate (.crt) file of the certificate you have generated/purchased for the host `login.example.com`.
    - **CA Certificate**: Paste the contents of the CA Certificate that was used to generate the SSL certificate above.
    - **Private Key**: Paste the contents of the key file which was used in the generation of the SSL certificate.
14. Click **Next**.
15. In the Manage Logging screen, ensure that **Create a New Log Group** is selected.
16. Change **Name**. For example: `lbr_internal`.
17. Change **Log Name**. For example: `lbr_internal_error`.
18. Click **Next**
19. Click **Submit**.
20. Review the Summary and Click **Submit**.

Creating Host Names

Host names are used to filter the different entry points into the load balancer. You need to create a host name for each load balancer virtual host described in Summary of the Virtual Servers Required for an Enterprise Deployment.

You have to create the following host names:

- `igdinternal.example.com`
- `login.example.com`
- `iadadmin.example.com`

Copyright © 2018, 2025, Oracle and/or its affiliates.

- `igdadmin.example.com`
- `oig.example.com`

> **Note**
>
> `login.example.com` is defined here for internal traffic routing. The EDG uses network segregation. If you do not define it here, calls to `login.example.com` will attempt to communicate using the public network and fail.

To create the load balancer host name:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking** and click **Load Balancers**.
3. Click the load balancer. For example: **internal_loadbalancer**.
4. Select the **Hostnames** tab from the resource list.
5. Click **Create Hostname**.
6. Enter the following information:
   - **Name**: Enter a name for the host name. For example: `igdinternal_hostname`.
   - **Hostname**: Enter the fully qualified host name. For example: `igdinternal.example.com`.
7. Click **Create**.
8. Repeat Steps 5 through 7 to create each of the required host names.

Updating the Default Listener

When you created the load balancer, a default listener also gets created. You have to assign the newly created host name to this listener.

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking** and click **Load Balancers**.
3. Click the load balancer. For example: **internal_loadbalancer**.
4. Click on the **Listeners** tab.
5. To edit the listener, click the three dots next to the name, and then click **Edit**.
6. Set the host name to the host name you created earlier. For example: `login`. See Creating Host Names.
7. Click **Save changes**.

Uploading Load Balancer Certificates

As the load balancer routes SSL requests, you need to upload the certificates for the load balancer. If you have created a self-signed certificate, add the details of that certificate. If you have your own certificates, upload those.

To upload the certificates:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking** and click **Load Balancers**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

3. Click the load balancer. For example: **internal_loadbalancer**.
4. Select the **Certificates and ciphers** tab.
5. Click **Add Certificate**.
6. Enter the following information. You can either upload the files directly or paste the contents of the files.
   - **Name**: Enter a name for the certificate. For example: `login_cert`.
   - **SSL Certificate**: Include the contents of the `login.crt` file.
   - **CA Certificate**: Select the **Specify CA Certificate** check box to include the contents of the `ca.crt` file.
   - **Private Key**: Select the **Specify Private Key** check box to include the contents of the `login.key` file.

   See Creating a Self-Signed Certificate.
7. Click **Add Certificate**.
8. Repeat the above for each certificate you need to add.

Creating Backend Sets

In end to end SSL deployments, you need a backend set that points to each OHS virtual host. You have already created a backend set for login when you created the listener, you now need to create the following backend sets.

Table 10-16 End to End SSL Deployments

Name	SSL	Healthcheck	Backends	Backend Port
ohs_iadadmin	Yes	https:4445	webhost1, webhost2	4445
ohs_igdadmin	Yes	https:4446	webhost1, webhost2	4446
ohs_login (default)	Yes	https:4447	webhost1, webhost2	4447
ohs_oig	Yes	https:4448	webhost1, webhost2	4448
ohs_igdinternal	Yes	https:4449	webhost1, webhost2	4449

Perform the following steps to create a backend set:

1. Log into the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking** and click **Load Balancers**.

   For example, **internal_loadbalancer**.
3. Select the **Backend Sets** tab.
4. Click **Create Backend Set**.
5. Enter information for each field as described below:

   Name - For example, `ohs_iadadmin`.

   Use SSL - Select to use SSL.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Certificate Resource - Select **Load Balancer Managed Certificates**.

Certificate Name - Enter the certificate name corresponding to the Oracle HTTP Servers CA. For example, `ohs_ca`. As all OHS certificates will have been created using the same CA, the CA created as part of the config wizard can be reused, This will have a system generated name similar to `cert_bs_<DATESTAMP>`

Verify Peer Certificate - Select **Enabled**.

Verify Depth - Select **1**.

Session Persistence - Select **Load balancer cookie persistence**.

Health Check - Enter **https** and the OHS listen port. For example `4445`.

URL Path - `/heathcheck.html`.

6. Click **Create Backend Set**.
7. After the backend set is created, click its name. For example, **iadadmin**.
8. Select the **Backends** tab.
9. Click **Add backends**.
10. Select **Compute Instances** and click **Add Instances**.
11. Select the compute instances where your OHS servers reside. For example, **webhost1** and **webhost2**.
12. Click **Add Instances**.
13. In the port field, select the OHS port you want to use. For example, `4445`.

Creating Listeners

You need to create a listener for each host name you have created earlier. See Creating Host Names.

Table 10-17 SSL Terminated Deployments

Name	Protocol	Port	SSL	Backend Set	Host Name
igdinternal_listener	http	7777	No	ohs_servers	`igdinternal_hostname`
login_listener (default)	https	443	Yes	ohs_servers	`login_hostname`
iadadmin_listener	http	80	No	ohs_servers	`iadadmin_hostname`
igdadmin_listener	http	80	No	ohs_servers	`igdadmin_hostname`

Table 10-18 End to End SSL Deployments

Name	Protocol	Port	SSL	Backend Set	Host Name
igdinternal_listener	https	443	Yes	ohs_igdinternal	igdinternal_hostname
login_listener (default)	https	443	Yes	ohs_login	login_hostname

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 10-18 (Cont.) End to End SSL Deployments

Name	Protocol	Port	SSL	Backend Set	Host Name
oig_listener	https	443	Yes	ohs_oig	oig_hostname
iadadmin_listener	https	443	Yes	ohs_iadadmin	iadadmin_hostname
igdadmin_listener	https	443	Yes	ohs_igdadmin	igdadmin_hostname

To create the load balancer listeners:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking** and click **Load Balancers**.
3. Click the load balancer. For example: **internal_loadbalancer**.
4. Select the **Listeners** tab.
5. Click **Create Listener**.
6. Enter the following information as per the table above:
   - **Name**: Enter the listener name. For example `iadadmin_listener`.
   - **Protocol**: Select **HTTPS**.
   - **Port**: Specify 443.
   - **Certificate Resource**: Select **Load balancer managed certificate**.
   - **Certificate Name**: Ensure that the certificate you created for the load balancer is displayed. If not displayed, select the certificate, for example `iadadmin_cert`
   - **Hostname**: Select **login**.
   - **Backend Set**: Select the back end set. For example:
     - SSL Terminated: `ohs_oig`
     - End to End SSL: `ohs_login`
7. Click **Create Listener**.

Creating an LDAP Load Balancer

The LDAP load balancer is used to route internal LDAP requests to the LDAP Servers. It resides in the same subnet as the LDAP Servers. This load balancer services requests generated from inside the application.

To create an LDAP load balancer, perform the following steps:

- Creating a Load Balancer
- Creating Host Names
- Updating the Default Listener

Creating a Load Balancer

To create a load balancer:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.

Copyright © 2018, 2025, Oracle and/or its affiliates.

2. Navigate to **Networking** and click **Load Balancers**.
3. Click **Create Load Balancer**.
4. Select **Load Balancer** and click **Create Load Balancer**.
5. Enter the following information:
   - **Name**: Enter a name for the load balancer. For example: `ldap-loadbalancer`.
   - **Visibility Type**: Select **Private**.
   - **Shapes**: Select **Flexible Shape**.
   - **Bandwidth**: Select the anticipated bandwidth.
   - **Virtual Cloud Network**: Select the Virtual Cloud Network.
   - **Subnet**: Select the database subnet. For example: `db-subnet`. See Creating Subnets for the Load Balancer.
6. Click **Next**.
7. On the Choose Back Ends screen, select your preferred **Load Balancing Policy**.
8. Click **Add Instances**.
   a. Select **LDAP Host Backends**.
   b. Click **Add Instances**.
   c. Change the port to the listen port for OUD. For example: `1389` (SSL Terminated) and `1636` (End to End SSL).
9. In the Specify Health Check Policy section, select **Protocol** `TCP`. Change the port to the port used in the backend set, for example, `1389` and `1636`. Set **Use SSL**: `false`
10. In the Back End Set section, enter a name for the back-end set. For example: `ldap_servers`.
11. In the Session Persistence section select **Enable Loadbalancer cookie persistence**.
12. Click **Next**.
13. On the Configure Listener screen, enter the following information:
    - **Name**: Select a name for the listener. For example: `idstore_listener`.
    - **Traffic Type**: Select the traffic type the listener uses. `idstore_listener` uses **TCP**.
    - **Port**: Select the load balancer port `1389` or `1636`.
14. Click **Next**.
15. In the Manage Logging screen, ensure that **Create a New Log Group** is selected.
16. Change **Name**. For example: `lbr_ldap`.
17. Change **Log Name**. For example: `lbr_ldap_error`.
18. Click **Next**
19. Click **Submit**.
20. Review the Summary and Click **Submit**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Creating Host Names

Host names are used to filter the different entry points into the load balancer. You need to create a host name for each load balancer virtual host described in Summary of the Virtual Servers Required for an Enterprise Deployment.

You have to create the following host names:

- `idstore.example.com`

To create the load balancer host name:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking** and click **Load Balancers**.
3. Click the load balancer. For example: **ldap_loadbalancer**.
4. Select the **Hostnames** tab from the resource list.
5. Click **Create Hostname**.
6. Enter the following information:
   - **Name**: Enter a name for the host name. For example: `iidstore_hostname`.
   - **Hostname**: Enter the fully qualified host name. For example: `idstore.example.com`.
7. Click **Create**.

Updating the Default Listener

When you created the load balancer, a default listener also gets created. You have to assign the newly created host name to this listener.

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking** and click **Load Balancers**.
3. Click the load balancer. For example: **ldap_loadbalancer**.
4. Click on the **Listeners** tab.
5. To edit the listener, click the three dots next to the name, and then click **Edit**.
6. Set the host name to the host name you created earlier. For example: `idstore`. See Creating Host Names.
7. Click **Save changes**.

Creating a Database

You must create a database to store all the schema information for Oracle Identity Management.

There are several different databases that you can create in OCI. For this example, a bare metal RAC database will be created. You may need to create one or more databases.

See Preparing the Database for an Enterprise Deployment for details on the databases and services you should create. This section shows an example of creating one of these databases in OCI.

- Creating the Database

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Creating a Secondary Pluggable Database
- Connecting to the Database Node
- Configuring the Database

Creating the Database

After establishing the network, you can now create the database.

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Oracle Database** and click **Oracle Base Database Service**.
3. Click on **Create DB System**.
4. Enter the following information:
   - **Compartment**: Select the name of the compartment you created earlier. See Creating an OCI Compartment.
   - **DB System Name**: Enter a name for the database infrastructure. For example: `Identity_Management_Databases`.
   - **Availability Domain**: Select an availability domain.
   - **Shape**: This value depends on your sizing requirements.
   - **Storage**: Select **Change Storage** then select **Oracle Grid Infrastructure**. Choose your storage performance, for example **Balanced**.
   - Click **Save**.
   - In the **DB System Configuration** section choose the `Total Node Count`. This should be at least 2 for High Availability.
     Select the **Database Software Edition**. you wish to use.
   - **Configure Storage**: Select the sizing requirements for your storage.
   - In the **SSH Keys** box, select **Paste SSH Keys**.
   - Copy the contents of the `id_rsa.pub` file that you created earlier. See Creating an SSH Key Pair.
   - **License Type**: Select the type of database license you have.
   - **Virtual Cloud Network**: Select your VCN, for example `idm-vcn`.
   - **Client Subnet**: Select the DB subnet. For example: `db-subnet`.
   - **Host Name Prefix**: Select a host name prefix. For example: `db`.
   - **Database Name**: Enter a name for you database, for example `idmdb`
   - **Database Unique Name Suffix:** Set it to a value unique to your system, which is especially important if you are going to create a disaster recovery site. The best practice is to set the suffix to the abbreviated region. For example: `lon` for London.
   - **Database Image:** Select the database release you want to use. For example: `23ai`.
   - **PDB Name** : Enter a name for the Oracle Access Manager PDB you want to create. For example: `oampdb`.
   - **Sys Password** : Select a password you want to assign to the database SYS account.
   - **Database Backups**: Select **Enable automatic backups**.
   - **Backup Retention Period:** Specify the period for which you want to keep the database backups.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- **Backup Scheduling:** Specify the preferred time to initiate the backup.

5. Click **Create**.
6. After the database is created, note the following values for use at a later point:
   - **SCAN DNS Name**: This is the host name you use to connect to the database.
   - **db node 1** and **db node 2**: To obtain the names/IP addresses of these nodes, click **Nodes** from the resources list.

> **Note**
>
> After the database is created, OCI adds a suffix to the database name. Ensure that you use the complete name including this suffix when configuring the database as described in Preparing the Database for an Enterprise Deployment.

Creating a Secondary Pluggable Database

When you create the database, it creates a single pluggable database (PDB). A single PDB may be sufficient for your needs.

However, if you require more PDBs so that OAM and OIG use different PDBs in the same database, you have to create additional PDBs. See Creating a PDB Using an Existing PDB as a Template.

You can do this either at the database level or, if you are using Oracle OCI, through the OCI console. For adding a PDB at the database level, see Creating a PDB Using an Existing PDB as a Template.

Alternatively, you can create extra pluggable databases by using the OCI Console.

1. Log in to the Oracle Cloud Infrastructure for your tenancy.
2. Navigate to **Oracle Database** and click **Oracle Base Database Service**.
3. Click the DB system hosting the database.
4. Click on the **Databases** tab
5. Click the **Container Database Name** from the list of displayed databases.
6. Click the **Pluggable Databases** tab.
7. Click **Create**.
8. Add a name for the new pluggable database, for example `oigpdb`, and enter the TDE wallet password for the container database. This password may be the same as the database SYS password if you opt not to set an explicit value.
9. Click **Create**.

Connecting to the Database Node

You can now connect to the database node using the following SSH command:

```
ssh -A opc@databaseNodeIPAdddress
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

Connect to DB node 1 from the bastion node using the command:

```
ssh -A opc@dbnode1
```

After you connect to DB node 1 as `opc`, connect to the oracle user using the following command:

```
sudo su - oracle
```

Configuring the Database

After you create the skeletal database, you should configure the database as described in Preparing the Database for an Enterprise Deployment.

Creating a DNS Server

It is important that all host names are resolvable, including the load balancer virtual hosts. You can make them resolvable by adding entries to the local hosts files. However, in OCI, using a private DNS server is the simpler method.

By default, the compute hosts are configured to use a private DNS server. You have to add the entries only for the local hosts.

- Creating a DNS Zone
- Creating DNS Records

Creating a DNS Zone

To create a DNS zone:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking**, select **DNS Management**, and then click **Zones**.
3. Select the **Private Zones** tab.
4. Click **Create Zone**.
5. Enter the following information:
   - **Name**: Enter a name for the zone. For example: `example.com`.
   - Select **Existing DNS Private View**.
   - **DNS Private View**: Select **Virtual Cloud Network**.
6. Click **Create**.

Creating DNS Records

After you create the zone, you can create records in the zone for each host. There are two types of DNS records that have to be created:

- **A Record**: This is an IP address association with a host name.
- **CNAME**: This is an alias for the A Record.

If you have multiple hosts using the same IP address, Oracle recommends you to create one 'A Record' and multiple 'CNAME' records.

Copyright © 2018, 2025, Oracle and/or its affiliates.

To create a record:

1. Log in to the Oracle Cloud Infrastructure Console for your tenancy.
2. Navigate to **Networking**, select **DNS Management**, and then click **Zones**.
3. Select the **Private Zones** tab.
4. Click on the newly created zone `example.com`.
5. Click **Manage Records**.
6. Click **Add Record**.
7. Select the **Record Type**: **A** or **CNAME**.
8. Specify the name of the host in the domain. For example: `loadbalancer.example.com`.
9. Specify the **Address** which is the IP Address of the host. For example: The IP address of the public load balancer.

   OR

   Specify the **Target** which is the name of the **A** record with which you want to associate the alias.
10. Set the **TTL** value to `86400`. If the **TTL** field is disabled, select the lock icon at the end of the row to specify a value.
11. Click **Add Record**.

> **Note**
>
> To continue adding another record, select the **ADD ANOTHER RECORD** check box. After you click **Submit**, the Add Record screen remains open to add another record.

You have to create the following entries:

Table 10-19 DNS Record Type and the Associated Host Name

Name	Type	Target	Address
`loadbalancer`	A		IP address of the Internal load balancer.
`iadadmin`	CNAME	`loadbalancer.example.com`	
`igdadmin`	CNAME	`loadbalancer.example.com`	
`login`	CNAME	`loadbalancer.example.com`	
`oig`	CNAME	`loadbalancer.example.com`	
`idstore`	A		IP Address of the internal LDAP load balancer
`igdinternal`	CNAME		IP address of the Internal load balancer.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 10-19 (Cont.) DNS Record Type and the Associated Host Name

Name	Type	Target	Address
webhost1	A		IP address of *WEBHOST1*.
webhost2	A		IP address of *WEBHOST2*.
ldaphost1	A		IP Address of *LDAPHOST1*
ldaphost2	A		IP Address of *LDAPHOST2*
oamhost1	A		IP Address of *OAMHOST1*
oamhost2	A		IP Address of *OAMHOST2*
iadadminvh	A		Floating IP Address of the OAM Admin Server
oighost1	A		IP Address of *OIGHOST1*
oighost2	A		IP Address of *OIGHOST2*
igdadminvh	A		Floating IP Address of the OIG Admin Server

> **Note**
>
> The hostnames *ldaphostn*, *oamhostn*, *oighostn*, *iadadminvhn* and *igdadminvhn* must be the same on both a primary and standby sites for disaster recovery, regardless of the underlying physical hostname.

12. After entering all your entries, click **Publish Changes**.
13. Click **Confirm Publish Changes** to make the records active.

Validating Your Environment

Perform checks to ensure that your environment is ready for a deployment.

For the bastion node

- Check the network connectivity:

```
ping webhost1.example.com
```

```
ping webhost2.example.com
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

- Resolve the public address of the load balancer:

  ```
  ping login.example.com
  ```

  ```
  ping oig.example.com
  ```

- Verify DNS is working by issuing the command:

  ```
  nslookup idstore.example.com
  ```

  Verify that the IP address returned is that of your LDAP load balancer.

From the Web Tiers

- Verify that the web tiers can communicate with the Application compute instances, which includes the port that you communicate on.

  Use the following command to check network connectivity:

  ```
  nc -zv OAMHOSTn <PORT>
  ```

  or:

  ```
  nc -zv OIGHOSTn <PORT>
  ```

  For example, if your OAM Administration Server for OAMHOST1 is using port 9002 then the command is as follows:

  ```
  nc -zv oamhost1.example.com 9002
  ```

  This will check that WEBHOSTn can communicate with the OAMHOST1 on port 9002 which is the secure mode port of the OAM Administration Server.

  The output will be similar to the following:

  ```
  Ncat: Version 7.92 ( https://nmap.org/ncat )
  Ncat: Connection refused.
  ```

  The Connection refused message is expected because you have not yet configured the OAM Admin Server. If the command hangs then this means that there is no pathway between the WEBHOST and the OAMHOST using port 9002. This indicates an issue with your security rules.

- Resolve the public address of the load balancer

  ```
  ping login.example.com
  ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

11

Preparing the Database for an Enterprise Deployment

Preparing the database for an enterprise deployment involves ensuring that the database meets specific requirements, creating database services, using SecureFiles for large objects in the database, and creating database backup strategies.

This chapter provides information about the database requirements, creating database services, and about the database backup strategies.

- About Preparing the Database for an Enterprise Deployment
  You have to configure a supported database as part of an Oracle Fusion Middleware enterprise deployment. Most Oracle Fusion Middleware products require a specific set of schemas that must be installed in a supported database. The schemas are installed by using the Oracle Fusion Middleware Repository Creation Utility (RCU).
- About Database Requirements
  Before you configure the enterprise deployment topology, you have to verify that the database meets all necessary requirements.
- Applying the Database Patches
  Ensure that you apply all the necessary database patches. The patches are required only if you are using Oracle Identity Governance.
- Creating Database Services
  When multiple Oracle Fusion Middleware products are sharing the same database, each product should be configured to connect to a separate, dedicated database service. This service should be different from the default database service.
- Creating a PDB Using an Existing PDB as a Template
  You must create a secondary pluggable database for OIG.
- Using SecureFiles for Large Objects (LOBs) in an Oracle Database
  SecureFiles is a LOB storage architecture. It is recommended to use SecureFiles for the Oracle Fusion Middleware schemas, in particular for the Oracle SOA Suite schemas.
- About Database Backup Strategies
  Performing a database backup at key points in the installation and configuration of an enterprise deployment enables you to recover quickly from any issue that might occur in the later configuration steps.

About Preparing the Database for an Enterprise Deployment

You have to configure a supported database as part of an Oracle Fusion Middleware enterprise deployment. Most Oracle Fusion Middleware products require a specific set of schemas that must be installed in a supported database. The schemas are installed by using the Oracle Fusion Middleware Repository Creation Utility (RCU).

In an enterprise deployment, Oracle recommends a highly available Real Application Clusters (Oracle RAC) database for the Oracle Fusion Middleware product schemas.

Copyright © 2018, 2025, Oracle and/or its affiliates.

About Database Requirements

Before you configure the enterprise deployment topology, you have to verify that the database meets all necessary requirements.

- Supported Database Versions
- Additional Database Software Requirements
- Databases Required
- Minimum Initialization Parameters

Supported Database Versions

Use the following information to verify what databases are supported by each Oracle Fusion Middleware release and which version of the Oracle database you are currently running:

- For a list of all certified databases, refer to *Oracle Fusion Middleware Supported System Configurations*.
- To check the release of your database, query the `PRODUCT_COMPONENT_VERSION` view:

```
SQL> SELECT VERSION FROM SYS.PRODUCT_COMPONENT_VERSION WHERE PRODUCT LIKE 'Oracle%';
```

Oracle Fusion Middleware requires that the database supports the AL32UTF8 character set. Check the database documentation for information on choosing a character set for the database.

For enterprise deployments, Oracle recommends that you use GridLink data sources to connect to Oracle RAC databases.

> **Note**
>
> For more information about using GridLink data sources and SCAN, see Using Active GridLink Data Sources in *Administering JDBC Data Sources for Oracle WebLogic Server*.
>
> Use of Active GridLink has specific licensing requirements, including a valid WebLogic Suite license. See Oracle WebLogic Server data sheet.

Additional Database Software Requirements

In the enterprise topology, there are two database host computers in the data tier that host the two instances of the RAC database. These hosts are referred to as DBHOST1 and DBHOST2.

Before you install or configure the enterprise topology, you must ensure that the following software is installed and available on DBHOST1 and DBHOST2:

- **Oracle Clusterware**

  See Installing Oracle Grid Infrastructure for a New Cluster in *Oracle Grid Infrastructure Installation Guide for Linux*.

- **Oracle Real Application Clusters**

  See Installing Oracle RAC and Oracle RAC One Node in *Oracle Real Application Clusters Installation Guide for Linux and UNIX*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- **Time synchronization between Oracle RAC database instances**

  The clocks of the database instances must be in sync if they are used by servers in a Fusion Middleware cluster configured with server migration.

- **Automatic Storage Management** (optional)

  See Introducing Oracle Automatic Storage Management in *Oracle Automatic Storage Management Administrator's Guide*.

General Database Characteristics

- Character Set – The character set must be unicode compliant. For example: AL32UTF8.
- Database Options – The following database options must be installed into the database:
  - Oracle JVM
  - Oracle Text
- Database Views – The following database view must be created on the database:
  - XAVIEWS
- Database Packages – The following database package must exist in the database:
  - DBMS_SHARED_POOL
- Transparent Data Encryption - This is required by the Oracle Privileged Account Manager.

Databases Required

For Oracle Identity and Access Management, a number of separate databases are recommended. Table 11-1 provides a summary of these databases. Which database or databases you use depends on the topology that you are implementing.

For this release of Oracle Identity and Access Management, you must use a separate RCU schema prefix for each domain. This allows different products to use a the same or different databases if required.

If you are planning on creating a Multi-Datacenter, you should use separate databases for Access and Governance. This allows different replication mechanisms to be used for each.

Table 11-1 Mapping between Databases and Schemas

Database Names	Database Hosts	Scan Address	Service Name	RCU Prefix	Schemas in Database
IADDB	DBHOST1 DBHOST2	*DBSCAN*	`iadedg.example.com`	EDGIAD	OAM, IAU, MDS, OPSS
IGDDB	DBHOST1 DBHOST2	*DBSCAN*	`igdedg.example.com`	EDGIGD	OIM, SOAINFRA, MDS, OPSS, ORASDPM, BI, ODS

Note

- Databases can be combined if you are not planning on creating a multi-datacenter deployment.
- Databases can be pluggable.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Minimum Initialization Parameters

The databases must have the following minimum initialization parameters defined:

Table 11-2 Minimum Initialization Parameters for Oracle Databases

Parameter	Small Value	Medium Value	Large Value
aq_tm_processes	1	1	1
dml_locks	200	200	200
job_queue_processes	12	12	12
open_cursors	1600	1600	1600
session_max_open_files	50	50	50
sessions	4000	4000	4000
processes	5000	5000	5000
sga_target	28G	58G	118G
pga_aggregate_target	7G	14G	29G
sga_max_size	8G	8G	8G
session_cached_cursors	800	800	800
db_keep_cache_size	800M	800M	800M
cursor_sharing	FORCE	FORCE	FORCE
query_rewrite_integrity	TRUSTED	TRUSTED	TRUSTED
query_rewrite_enabled	TRUE	TRUE	TRUE
max_dispatchers	0	0	0
max_shared_servers	0	0	0
_active_session_legacy_behavior	TRUE	TRUE	TRUE
disk_asynch_io	FALSE	FALSE	FALSE
_b_tree_bitmap_plans	FALSE	FALSE	FALSE
parallel_max_servers	1	1	1
shared_servers	0	0	0
db_securefile	ALWAYS	ALWAYS	ALWAYS
plsql_code_type	NATIVE	NATIVE	NATIVE
_active_session_legacy_behavior	TRUE	TRUE	TRUE

Oracle recommends you to set these parameters in the database configuration assistant when creating the database. If not, you can adjust them after creating the database, by using the `alter system` database command. For example:

```
sqlplus / as sysdba

alter system set aq_tm_processes=1 scope=spfile;
```

After making changes in the `spfile`, restart the database. For example

```
srvctl stop database -d iaddb
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
srvctl start database -d iaddb
```

> **Note**
>
> For guidelines on setting up optimum parameters for the Database, see Tuning Database Parameters in *Tuning Performance*.

Applying the Database Patches

Ensure that you apply all the necessary database patches. The patches are required only if you are using Oracle Identity Governance.

Include the database patches from Oracle Text Mandatory Patches.

Also, ensure that you include the latest patch set bundles for the release of database that you are using.

Creating Database Services

When multiple Oracle Fusion Middleware products are sharing the same database, each product should be configured to connect to a separate, dedicated database service. This service should be different from the default database service.

A different service name enables you to create role-based database services for Disaster Recovery and Multi-Datacenter topologies.

> **Note**
>
> The instructions in this section are for the Oracle Database 19*c* release. If you are using another supported database, refer to the appropriate documentation library for more up-to-date and release-specific information.

With Data Guard you can automatically control the startup of database services on primary and standby database by assigning a database role to each service. This service is in addition to the default service created when the database was commissioned. A role based database service will automatically start upon database startup if the management policy of the service is AUTOMATIC and if one of the roles assigned to that service matches the current role of the database; for example, if the database is running as a primary.

Creating a database service in this way means that, the service is started whenever the database with the role `primary` is started. The service will move between sites as the underlying databases roles are moved through switchover or failover.

If you are planning to use a standard disaster recovery solution as described in Disaster Recovery Guide, then each database service should be defined as a Role Based Database service.

If you are planning on using a multi-datacenter deployment, then the database service created for the Oracle Identity Governance (IGDDB) database should be a role based service.

For more information about connecting to Oracle databases using services, see Overview of Using Dynamic Database Services to Connect to Oracle Databases in *Real Application Clusters Administration and Deployment Guide*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

In addition, the database service should be different from the default database service. For complete instructions on creating and managing database services for an Oracle Database 19*c* database, see Overview of Automatic Workload Management with Dynamic Database Services in *Real Application Clusters Administration and Deployment Guide*.

Runtime connection load balancing requires configuring Oracle RAC Load Balancing Advisory with service-level goals for each service for which load balancing is enabled.

You can configure the Oracle RAC Load Balancing Advisory for `SERVICE_TIME` or `THROUGHPUT`. Set the connection load-balancing goal to **SHORT**.

You create and modify Oracle Database services by using the `srvctl` utility.

To create and modify a database service:

1. Add the service to the database and assign it to the instances by using `srvctl`:

   ```
   srvctl add service -db iaddb -service iamedg.example.com -preferred iaddb1,iaddb2
   ```

   If you use PDB, change the command to the following:

   ```
   srvctl add service -db iamdb1 -pdb iadpdb -service iadedg.example.com -
   preferred iamdb1, iamdb2
   ```

   > **Note**
   >
   > For the Service Name of the Oracle RAC database, use lowercase letters, followed by the domain name. For example: `iamedg.example.com`.

2. Start the service:

   ```
   srvctl start service –db iaddb –service iamedg.example.com
   ```

   > **Note**
   >
   > For complete instructions on creating and managing database services with SRVCTL, see Creating Services with SRVCTL in the *Real Application Clusters Administration and Deployment Guide*.

3. Modify the service so that it uses the Load Balancing Advisory and the appropriate service-level goals for runtime connection load balancing.

   Use the following resources in the Oracle Database 19*c* *Real Application Clusters Administration and Deployment Guide* to set the SERVICE_TIME and THROUGHPUT service-level goals:

   - Overview of the Load Balancing Advisory
   - Configuring Your Environment to Use the Load Balancing Advisory

   For example:

   Check the default configuration of the service by using this command:

   ```
   srvctl config service -db iaddb -service iamedg.example.com
   ```

   Several parameters are shown. Check the following parameters:

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Connection Load Balancing Goal: Long
- Runtime Load Balancing Goal: NONE

You can modify these parameters by using the following command:

```
srvctl modify service -db iaddb -service iamedg.example.com -rlbgoal
SERVICE_TIME -clbgoal SHORT
```

4. Restart the service:

```
srvctl stop service -db iamdb -service iamedg.example.com


srvctl start service -db iamdb -service iamedg.example.com
```

5. Verify the change in the configuration:

```
srvctl config service -db iamdb -service iamedg.example.com


Runtime Load Balancing Goal: SERVICE_TIME
  Service name: iamedg.example.com
  Service is enabled
  Server pool: iamdb_iamedg.example.com
  ...
  Connection Load Balancing Goal: SHORT
  Runtime Load Balancing Goal: SERVICE_TIME
  ...
```

Creating a PDB Using an Existing PDB as a Template

You must create a secondary pluggable database for OIG.

When you create a database, an empty PDB is created at the same time. This example used the empty PDB to create a secondary PDB with the same configuration.

This example shows the commands used to create a secondary PDB by using the empty PDB. The new database is called OIG and the existing one is called OAM:

```
sqlplus / as sysdba
```

> **Note**
>
> You may need to set *ORACLE_HOME* and *ORACLE_SID* to the database you just created to ensure that you are connecting to the correct database, especially if you have more than one database in the compartment. Creating a Database.

```
create pluggable database IGDPDB from IADPDB keystore identified by
syspassword;
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> 'syspassword' is the database sys password.

```
Pluggable database created.

show pdbs;

CON_ID     CON_NAME      OPEN MODE      RESTRICTED
-------    ----------    ----------     -----------
 2          PDB$SEED      READ ONLY      NO
 3          IADPDB        READ WRITE     NO
 5          IGDPDB        MOUNTED

alter pluggable database IGDPDB open read write;

Pluggable database altered.

show pdbs;

CON_ID     CON_NAME        OPEN MODE      RESTRICTED
-------    ----------      ----------     ----------
  2        PDB$SEED        READ ONLY      NO
  3        IADPDB          READ WRITE     NO
  5        IGDPDB          READ WRITE     NO
```

Using SecureFiles for Large Objects (LOBs) in an Oracle Database

SecureFiles is a LOB storage architecture. It is recommended to use SecureFiles for the Oracle Fusion Middleware schemas, in particular for the Oracle SOA Suite schemas.

See Using Oracle LOB Storage in the *SecureFiles and Large Objects Developer's Guide*.

In Oracle 19*c* Databases, the default setting for using SecureFiles is `PREFERRED` . This means that the database attempts to create a SecureFiles LOB unless a BasicFiles LOB is explicitly specified for the LOB or the parent LOB (if the LOB is in a partition or sub-partition). The Oracle Fusion Middleware schemas do not explicitly specify BasicFiles, which means that Oracle Fusion Middleware LOBs defaults to SecureFiles when installed in an Oracle 19*c* database.

Note that the SecureFiles segments require tablespaces managed with automatic segment space management (ASSM). This means that LOB creation on SecureFiles will fail if ASSM is not enabled. However, the Oracle Fusion Middleware tablespaces are created by default with

Copyright © 2018, 2025, Oracle and/or its affiliates.

ASSM enabled. As a result, with the default configuration, nothing needs to be changed to enable SecureFiles for the Oracle Fusion Middleware schemas.

About Database Backup Strategies

Performing a database backup at key points in the installation and configuration of an enterprise deployment enables you to recover quickly from any issue that might occur in the later configuration steps.

At key points in the installation and configuration of an enterprise deployment, this guide recommends that you back up your current environment. For example, after you install the product software and create the schemas for a particular Oracle Fusion Middleware product, you should perform a database backup. Performing a backup allows you to perform a quick recovery from any issue that might occur in the later configuration steps.

You can choose to use your own backup strategy for the database, or you can simply make a backup by using operating system tools or RMAN for this purpose.

Oracle recommends that you use Oracle Recovery Manager for the database, particularly if the database was created using Oracle Automatic Storage Management. If possible, you can also perform a cold backup by using operating system tools such as tar.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Part III
Configuring the Enterprise Deployment

You have to perform a series of steps to configure your enterprise deployment. For example, creating the initial infrastructure domain, which you can use as the starting point for the deployment. This section provides instructions to complete each of these steps, in detail.

Part III contains the following chapters:

- Configuring Oracle LDAP for an Enterprise Deployment
  Install and configure Oracle Unified Directory (OUD) for use with Oracle Identity and Access Management.
- Configuring Oracle Unified Directory Service Manager
  Install and configure Oracle Unified Directory Service Manager (OUDSM) in order to manage Oracle Unified Directory (OUD).
- Configuring Oracle Access Management
- Configuring Oracle Identity Governance
- Configuring Oracle HTTP Server for an Enterprise Deployment
  Install and configure Oracle HTTP Server (OHS) to direct HTTP(S) requests from the hardware load balancer to specific Managed Servers in the application tier.

Copyright © 2018, 2025, Oracle and/or its affiliates.

12

Configuring Oracle LDAP for an Enterprise Deployment

Install and configure Oracle Unified Directory (OUD) for use with Oracle Identity and Access Management.

This chapter includes the following topics:

- Configuring Oracle Unified Directory
  Install and configure Oracle Unified Directory (OUD). In an enterprise deployment, each OUD instance is configured on a separate host. OUD is not installed into a domain.
- Preparing an Existing LDAP Directory
  Before you can use an LDAP directory with Oracle Identity and Access Management, it must be extended with object classes required by Oracle Access Manager.

Configuring Oracle Unified Directory

Install and configure Oracle Unified Directory (OUD). In an enterprise deployment, each OUD instance is configured on a separate host. OUD is not installed into a domain.

- Variables Used When Configuring Oracle Unified Directory
- Setting Environment Variables
- Installing a Supported JDK
- Installing Oracle Unified Directory
- Configuring the Oracle Unified Directory Instances

Variables Used When Configuring Oracle Unified Directory

The procedures for installing and configuring Oracle Unified Directory (OUD) reference a series of variables that you can replace with the actual values used in your environment.

The following table outlines the OUD variables used:

Table 12-1 Variables Used in This Chapter

Variable	Sample Value	Description
DIR_ORACLE_HOME	`/u02/oracle/products/oud`	Oracle Home for the Oracle Unified Directory.
OUD_ORACLE_INSTANCE	`/u02/oracle/config/instances/oud1`	The path to the OUD instance home.
OUD_REPLICATION_PORT	`8989`	The replication port you wish to use.
LDAP_PORT	`1389`	The LDAP port you wish to use.
LDAP_SSL_PORT	`1636`	The LDAP port you wish to use for SSL.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 12-1 (Cont.) Variables Used in This Chapter

Variable	Sample Value	Description
LDAP_ADMIN_PORT	4444	The LDAP port you wish to use for the administration port.
JAVA_HOME	`/u02/oracle/products/jdk`	The JDK home directory.
INSTANCE_NAME	`../../../../u02/oracle/config/instances/oud1`	The path to the instance home. **Note** The tool creates the instance home relative to the DIR_ORACLE_HOME, so you must include previous directories to get the instance created in LOCAL_CONFIG_DIR/instances.
LOCAL_CONFIG_DIR	`/u02/oracle/config`	The local or nfs-mounted configuration directory unique to a given host containing the machine-specific domain directory (`MSERVER_HOME`).

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 12-1 (Cont.) Variables Used in This Chapter

Variable	Sample Value	Description
OHS_DOMAIN_HOME	`/u02/oracle/config/domains/domain_name`	The Domain home for the standalone Oracle HTTP Server domain, which is created when you install Oracle HTTP Server on the local disk of each web tier host.
IDSTORE_HOST	`idstore.example.com`	The host of your Identity Store directory. When preparing the Identity Store these should point to one of the LDAP instances. When configuring components such as OAM or OIG they should point to the load balancer entry point.
IDSTORE_PORT	`1636`	The port of your Identity Store directory. When preparing the Identity Store these should point to one of the LDAP instances. When configuring components such as OAM or OIG they should point to the load balancer entry point.
IDSTORE_DIRECTORYTYPE	`OUD`	The type of directory you are using. Valid value is OUD.
IDSTORE_BINDDN	`cn=oudadmin`	An administrative user in the Identity Store Directory.
IDSTORE_SEARCHBASE	`dc=example,dc=com`	The location in the directory where Users and Groups are stored.
IDSTORE_LOGINATTRIBUTE	`uid`	The LDAP attribute, which contains the users Login name.
IDSTORE_USERSEARCHBASE	`cn=Users,dc=example,dc=com`	The location in the directory where Users are Stored.
IDSTORE_GROUPSEARCHBASE	`cn=Groups,dc=example,dc=com`	The location in the directory where Groups are Stored.
IDSTORE_SYSTEMIDBASE	`cn=SystemIDs,dc=example,dc=com`	The location of a container in the directory where system users can be placed when you do not want them in the main user container.
IDSTORE_USERNAMEATTRIBUTE	`cn`	The name of the LDAP attribute which stores a users name.
IDSTORE_ADMIN_PORT	`4444`	The administration port of your Oracle Unified Directory instance. If you are not using Oracle Unified Directory, you can leave out this parameter.
IDSTORE_KEYSTORE_FILE	`/u02/oracle/config/instances/oud1/config/admin-keystore`	The location of the LDAP Truststore for LDAP connections.
IDSTORE_KEYSTORE_PASSWORD	`password`	The password of the IDSTORE_KEYSTORE_FILE.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 12-1 (Cont.) Variables Used in This Chapter

Variable	Sample Value	Description
IDSTORE_NEW_SETUP	`true`	This parameter is used when preparing a directory for the first time.
IDSTORE_OAMADMINUSER	`oamadmin`	The name of the user you want to administer OAM.
IDSTORE_OAMSOFTWAREUSER	`oamLDAP`	The name of the user that gets created in LDAP that is used when OAM is running to connect to the LDAP server.
OAM_IDSTORE_ROLE_SECURITY_ADMIN	`OAMAdministrators`	The name of the group, which is used to allow access to the OAM console. Only users assigned to this group will be able to access the OAM Console.
OAM_SERVER_LOGIN_ATTRIBUTE	`uid`	The name of the LDAP attribute where userids are stored. This should be the same as the IDSTORE_LOGIN_ATTRIBUTE.
IDSTORE_WLSADMINUSER	`weblogic_iam`	The username to be created for logging in to the web logic domain once it is enabled by Single Sign-On.
IDSTORE_WLSADMINGROUP	`WLSAdministrators`	The name of the group to which users who are allowed to log in to the WebLogic system components, such as the WLS Console and EM, belong.
ORACLE_HOME		Oracle Home value to set when instructed below.

Setting Environment Variables

To help navigate this guide, to be able to copy sample commands without modification you can set the following environment variables, replacing the values with values appropriate to your environment.

```
export DIR_ORACLE_HOME=/u02/oracle/products/oud
export ORACLE_HOME=/u02/oracle/products/oud
export ORACLE_COMMON_HOME=$ORACLE_HOME/oracle_common
export JAVA_HOME=/u02/oracle/products/jdk
export PATH=$JAVA_HOME/bin:$PATH
export LDAPCONFIG=$ORACLE_HOME/idmtools
export OUD_ORACLE_INSTANCE=/u02/oracle/config/instances/oud1
export LDAP_BASE_DN=dc=example,dc=com
export LDAPHOST1=ldaphost1.example.com
export LDAPHOST2=ldaphost2.example.com
export LOCAL_CONFIG_DIR=/u02/oracle/config
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

Installing a Supported JDK

Oracle Fusion Middleware requires that a certified Java Development Kit (JDK) is installed on your system.

- Locating and Downloading the JDK Software
- Installing the JDK Software

Locating and Downloading the JDK Software

To find a certified JDK, see the certification document for your release on the Oracle Fusion Middleware Supported System Configurations page.

After you identify the Oracle JDK for the current Oracle Fusion Middleware release, you can download an Oracle JDK from the following location on Oracle Technology Network:

```
https://www.oracle.com/java/technologies/downloads/
```

Be sure to navigate to the download for the Java SE JDK.

Installing the JDK Software

Oracle Fusion Middleware requires you to install a certified Java Development Kit (JDK) on your system.

For more information about the recommended location for the JDK software, see Understanding the Recommended Directory Structure for an Enterprise Deployment.

To install JDK 21.0:

1. Change directory to the location where you downloaded the JDK archive file.

   ```
   cd download_dir
   ```

2. Unpack the archive into the JDK home directory, and then run the following commands:

   ```
   tar -xzvf jdk-21.0.4+8_linux-x64_bin.tar.gz
   ```

   Note that the JDK version listed here was accurate at the time this document was published. For the latest supported JDK, see the *Oracle Fusion Middleware System Requirements and Specifications* for the current Oracle Fusion Middleware release.

3. Move the JDK directory to the recommended location in the directory structure.

   For example:

   ```
   ln -s jdk-21.0.4 /u02/oracle/products/jdk
   ```

   See File System and Directory Variables Used in This Guide.

4. Run the following command to verify that the appropriate `java` executable is in the path and your environment variables are set correctly:

   ```
   java -version
   ```

   The Java version in the output should be displayed as “21.0.4”.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Installing Oracle Unified Directory

You can install Oracle Unified Directory by using an interactive graphical wizard provided by the Oracle Universal Installer.

- Starting the Oracle Unified Directory Installer
- Navigating the Oracle Unified Directory Installation Screens
- Installing the Stack Bundle Patch
- Installing the Software on Other Host Computers
- Verifying the Installation
  After you complete the installation, you nust verify it.

Starting the Oracle Unified Directory Installer

To start the installation program:

1. Log in to LDAPHOST1.
2. Go to the directory in which you downloaded the installer.
3. Run the following Java command to launch the installation wizard:

   ```
   java -d64 -jar fmw_14.1.2.1.0_oud.jar
   ```

   Replace the JDK location in the above command with the actual JDK location on your system. For information about downloading the software and locating the actual installer file name for your product, see Identifying and Obtaining Software Distributions for an Enterprise Deployment.

Navigating the Oracle Unified Directory Installation Screens

The following table describes how to use the installer screens to install Oracle Unified Directory.

If you need additional help with any of the installation screens, click the screen name.

Screen	Description
Welcome	This screen introduces you to the product installer. Click **Next**.
Auto Updates	Select whether or not you want to receive automatic updates for this product.
Installation Location	For the purposes of this enterprise deployment, enter the value of the *DIR_ORACLE_HOME* variable listed in Table 8-2. Note that run-time processes cannot write to this directory.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Screen	Description
Installation Type	Use this screen to select the type of installation and as a consequence, the products and feature sets you want to install. If you plan to manage OUD through WebLogic server or OUDSM, select **Collocated Oracle Unified Directory Server (Managed through WebLogic server)**. **Note** If you select Collocated mode, you must also install Oracle Fusion Middleware Infrastructure. See Installing the Oracle Fusion Middleware Infrastructure. If you plan to manage OUD independently of WebLogic server, select **Standalone Oracle Unified Directory Server (Managed independently of WebLogic server)**. Click **Next**.
Prerequisite Checks	The installer analyzes the host computer to ensure that the prerequisites are fulfilled. The results of the prerequisite checks are displayed on this screen. If a prerequisite check fails, an error or warning message is displayed. • Fix the error and click **Rerun**. • To ignore the error or warning and continue with the installation, click **Skip**. • To stop the prerequisite checking process, click **Stop**. Click **Next** to continue.
Installation Summary	This screen displays the Oracle home directory that you specified earlier. It also indicates the amount of disk space that will be used for the installation and the free space available. Review information on this screen. To save the settings specified so far in the installation wizard in a text file (called a *response* file), click **Save**. If necessary, you can use the response file to perform the same installation from the command line. Click **Install** to begin the installation. For more information about silent or command line installation, see "Using the Oracle Universal Installer in Silent Mode" in Installing Software with the Oracle Universal Installer.
Installation Progress	This screen shows the progress and status of the installation process. If you want to cancel the installation, click **Cancel**. The files that were copied to your system before you canceled the installation will remain on the system; you should remove them manually. Click **Next** to continue.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Screen	Description
Installation Complete	Click **Finish**.

Installing the Stack Bundle Patch

After installing the software binaries you must download and apply the July 2025 Stack Bundle Patch or later. For more details, see Identifying and Obtaining Software Distributions for an Enterprise Deployment.

To apply the patch run the following commands:

1. After downloading the patch, unzip it to your preferred location. For example:

   ```
   unzip p38184742_141210_Linux-x86-64.zip
   ```

   This location will be known as $PATCH_DIR.
2. Navigate to the $PATCH_DIR:

   ```
   cd $PATCH_DIR/tools/spbat/generic/SPBAT/
   ```
3. Apply the patch using the command:

   ```
   ./spbat.sh -type oud -phase downtime -mw_home $DIR_ORACLE_HOME -
   spb_download_dir $PATCH_DIR
   ```

Installing the Software on Other Host Computers

If you have configured a separate shared storage volume or partition for LDAPHOST2 , then you must also install the software on LDAPHOST2. For more information, see Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment.

Note that the location where you install the Oracle home (which contains the software binaries) varies, depending upon the host. To identify the proper location for your Oracle home directories, refer to the guidelines in File System and Directory Variables Used in This Guide.

You must install the Stack Bundle Patch and any other mandatory patches outlined in Identifying and Obtaining Software Distributions for an Enterprise Deployment.

Verifying the Installation

After you complete the installation, you nust verify it.

Perform the following tasks:

- Reviewing the Installation Log Files
- Checking the Directory Structure
- Viewing the Contents of Your Oracle Home

Copyright © 2018, 2025, Oracle and/or its affiliates.

Reviewing the Installation Log Files

Review the contents of the installation log files to make sure that no problems were encountered. For a description of the log files and where to find them, see Understanding Installation Log Files in *Installing Software with the Oracle Universal Installer*.

Checking the Directory Structure

After you install the Oracle Unified Directory and create the Oracle home, you should see the directory and sub-directories listed in this topic. The contents of your installation vary based on the options you selected during the installation.

To check the directory structure:

1. Navigate to the $DIR_ORACLE_HOME:

   ```
   cd $DIR_ORACLE_HOME
   ```

2. Enter the following command:

   ```
   ls --format=single-column
   ```

   If you installed using the colocated method, the directory structure on your system must match the structure shown in the following example:

   ```
   addons
   bat
   bin
   common
   config
   lib
   libForUpgrade
   oud-proxy-setup
   oud-proxy-setup.bat
   oud-replication-gateway-setup
   oud-replication-gateway-setup.bat
   oud-setup
   oud-setup.bat
   plugins
   snmp
   winlib
   ```

   If you installed using the standalone method, then the directory structure should match the structure shown below:

   ```
   cfgtoollogs
   inventory
   OPatch
   oracle_common
   oraInst.loc
   oud
   oui
   wlserver
   ```

   See What are the Key Oracle Fusion Middleware Directories? in *Understanding Oracle Fusion Middleware*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Viewing the Contents of Your Oracle Home

You can also view the contents of your Oracle home by using the `viewInventory` script. See Viewing the contents of an Oracle home in *Installing Software with the Oracle Universal Installer*.

Configuring the Oracle Unified Directory Instances

Follow these steps to configure Oracle Unified Directory (OUD) components in the directory tier on LDAPHOST1 and LDAPHOST2. During the configuration you will also configure Oracle Unified Directory replication servers.

The following are the two option available when you install Oracle Unified Directory:

- Standalone mode: Choose this option if you wish to manage OUD via command line tools.
- Co-located mode: Choose this option to associate Oracle Unified directory with a domain. If you choose to associate it with a domain, you have the option to manage OUD using Oracle Unified Directory Service Manager. If you wish to use OUDSM in the same domain as OUD, then you must select co-located mode.

This section contains the following topics:

- Configuring Oracle Unified Directory on LDAPHOST1
- Configuring Oracle Unified Directory Instance on LDAPHOST2

Configuring Oracle Unified Directory on LDAPHOST1

Ensure that ports 1389, 1636, 4444, and 8989 are not in use by any service on the computer by issuing these commands for the operating system you are using. If a port is not in use, no output is returned from the command.

```
netstat -an | grep "1389"
```

If the ports are in use (that is, if the command returns output identifying either port), you must free the port.

Remove the entries for ports 1389, 1636, 4444, and 8989 in the `/etc/services` file and restart the services or restart the computer.

Change directory to the following:

```
cd $DIR_ORACLE_HOME/oud
```

- Configuring OUD Using the GUI
- Configuring OUD Using the CLI
- Validating Oracle Unified Directory on LDAPHOST1

Configuring OUD Using the GUI

Set the environment variable INSTANCE_NAME as follows:

```
export INSTANCE_NAME=../../../../u02/oracle/config/instances/oud1
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> The tool creates the instance home relative to the `$DIR_ORACLE_HOME`, so you must include previous directories to get the instance created in `$LOCAL_CONFIG_DIR/instances`.

Start the Oracle Unified Directory configuration assistant by executing the command:

```
./oud-setup
```

The following table describes how to use the configuration assistant screens to configure Oracle Unified Directory.

Screen	Description
Welcome	This screen introduces you to the product configuration assistant. Click **Next**.
Server Administration Settings	Enter the following details of the server: • **Instance Path**: Enter the location of the OUD configuration files (`$OUD_INSTANCE_HOME`). • **Host Name**: Enter the name of the host where Oracle Unified Directory is running. For example: `ldaphost1.example.com`. • **Administration Port(s)**: The value of this field determines how you are going to administer OUD. The following are the optional values: – **Enable Administration only by LDAP**: Enter the LDAP port that will be used for administration traffic. The default LDAP administration port is 4444. – **Enable Administration by LDAP and HTTP**: Enter the LDAP and HTTP ports that will be used for administration traffic. The default administration ports are 4444 for LDAP and 8444 for HTTP. – **Enable Administration by HTTP**: Enter the HTTP port that will be used for administration traffic. The default HTTP administration port is 8444. • **LDAP Port**: Enter the port that you wish to use for administering OUD via LDAP. • **HTTP Port**: Enter the port that you wish to use for administering OUD via HTTP: 8444 (OUD_ADMIN_PORT). • **Root User DN**: Enter an administrative user. For example, `cn=oudadmin`. • **Password**: Enter the password you wish to assign to the `ouadmin` user. • **Password (Confirm)**: Repeat the password. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Screen	Description
Ports	Enter the following details: **LDAP** • **Enable**: Select if you wish to enable non SSL communications with OUD. • **on Port**: Select the Port you wish to use (`LDAP_PORT`). • **Enable Start TLS for LDAP**: Select Enable **StartTLS for LDAP** to specify that the LDAP connection handler should allow clients to use the StartTLS extended operation to initiate secure communication over an otherwise insecure connection. **LDAPS** • **Enable**: Select if you wish to enable SSL communications with OUD. • **on Port**: Select the Port you wish to use (`LDAP_SSL_PORT`). If you select this option, you must provide the SSL certificate information below. **Certificate** You can use the existing certificate or generate a self signed certificate. Self signed certificates are not recommended for production deployments. • **Generate self signed certificate**: Select this if you wish OUD to generate its own certificate. • **Use an existing certificate**: Select this if you are using an existing certificate. Select the type of certificate, the location of the Keystore, and the Keystore pin.
Topology Options	Enter the following details: • **This will server will be part of a replication topology**: Select this. • **Replication Port**: Enter the replication port. For example: `8989 (OUD_REPLICATION_PORT)`. • **Configure As Secure**: Select this if you wish the replication traffic to be encrypted. • There is already a server in the topology. Leave it unselected. Click **Next**.
Directory Data	Enter the following details: • **Directory Base DN**: `dc=example,dc=com` • **Directory Data**: Only create base entry. Click **Next**.
Oracle Components Integration	If you are planning to use the directory for integrating with other directories, Select **Enable for DIP**. If you are planning on using the directory for E Business Suite or for Oracle database name resolution, select **Enable for EBS (E-Business Suite), Database Net Services and DIP**. Click **Next**. If you are planning to use the directory for Enterprise User Security, select **Enable fo EUS (Enterprise User Security), EBS, Database Net Services and DIP**.
Server Tuning	You have the option of allocating specific resources to the OUD instance. You can choose to accept the default of change the resource allocations based on your deployment. If this server is used only for OUD as in a distributed deployment then be sure to check the box Dedicated Machine for OUD. Click **Next**.
Review	Verify that the information displayed is correct. If you wish the OUD server to be started after configuration, ensure that you select the option **Start the server**.
Finished	Click **Close**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Configuring OUD Using the CLI

You can also configure Oracle Unified Directory using the CLI and run the following command:

```
export INSTANCE=/u02/oracle/config/instances


$ORACLE_HOME/oud/oud-setup \
        --cli \
        --no-prompt \
        --noPropertiesFile \
        -I /u02/oracle/config/instances/oud1 \
        -h $LDAPHOST1 \
        -D cn=oudadmin \
        -j /home/oracle/oud.pwd \
        --usePkcs12keyStore /u02/oracle/config/keystores/idmcerts.p12 \
        --keyStorePasswordFile /u02/oracle/config/keystores/oud.pin \
        --keyPasswordFile /u02/oracle/config/keystores/oud.pin \
        --certNickname idmcerts \
        --ldapPort disabled \
        --adminConnectorPort 4444 \
        --ldapsPort 1636 \
        --baseDN $LDAP_BASE_DN \
        --addBaseEntry \
        --serverTuning systemMemory:75% \
        --offlineToolsTuning jvm-default \
        --doNotStart
/u02/oracle/config/instances/oud1/bin/start-ds
dsconfig -D "cn=oudadmin" -j /u02/oracle/config/keystores/oud.pin -X -n set-
trust-manager-provider-prop --provider-name=pkcs12 --set trust-store-
file:/u02/oracle/config/keystores/idmTrustStore.p12 --set trust-store-
pin:Manager1 --set enabled:true
dsconfig -D "cn=oudadmin" -j /u02/oracle/config/keystores/oud.pin -X -n set-
trust-manager-provider-prop --provider-name=jks --set enabled:false
XX=$(/u02/oracle/config/instances/oud1/bin/dsconfig -h ldaphost1 -p 4444 -D
cn=oudadmin -j ~/oud.pwd -X -n get-key-manager-provider-prop --provider-name
Administration --property key-store-pin --showKeystorePassword | grep "store-
pin" | cut -f2 -d:)
```

Validating Oracle Unified Directory on LDAPHOST1

After configuration, you can validate that Oracle Unified Directory is working by performing a simple search using the following command:

```
cd $OUD_ORACLE_INSTANCE/bin


./ldapsearch -h LDAPHOST1.example.com -p 1389 -D cn=oudadmin -b "" -s base
"(objectclass=*)" supportedControl
```

If Oracle Unified Directory is working correctly, you will see a list of `supportedControl` entries returned.

Copyright © 2018, 2025, Oracle and/or its affiliates.

If you have enabled SSL on the directory, you can test it using the command:

```
cd $OUD_ORACLE_INSTANCE/bin


./ldapsearch -h LDAPHOST1.example.com -p 1636 --useSSL -D cn=oudadmin -b "" -
s base "(objectclass=*)" supportedControl
```

Configuring Oracle Unified Directory Instance on LDAPHOST2

Ensure that ports 1389, 1636, 4444, and 8989 are not in use by any service on the computer by issuing these commands for the operating system you are using. If a port is not in use, no output is returned from the command.

```
netstat -an | grep "1389"
```

If the ports are in use (that is, if the command returns output identifying either port), you must free the port.

Remove the entries for ports `1389`, `1636`, `4444`, and `8989` in the `/etc/services` file and restart the services or restart the computer.

Navigate to the `$DIR_ORACLE_HOME/oud` directory:

```
cd $DIR_ORACLE_HOME/oud
```

Set the environment variable `INSTANCE_NAME` to `../../admin/oud2`.

For example:

```
export INSTANCE_NAME=../../../../u02/oracle/config/instances/oud2
```

Note the tool creates the instance home relative to the `$DIR_ORACLE_HOME`, so you must include previous directories to get the instance created in `$LOCAL_CONFIG_DIR/instances`.

Start the Oracle Unified Directory configuration assistant by executing the command:

```
./oud-setup
```

The following table describes how to use the configuration assistant screens to configure Oracle Unified Directory.

Screen	Description
Welcome	This screen introduces you to the product configuration assistant. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Screen	Description
Server Administration Settings	Enter the following details of the server: • **Instance Path**: Enter the location of the OUD configuration files (`$OUD_INSTANCE_HOME`). • **Host Name**: Enter the name of the host where Oracle Unified Directory is running. For example: `ldaphost2.example.com`. • **Administration Port(s)**: The value of this field determines how you are going to administer OUD. The following are the optional values: – **Enable Administration only by LDAP**: Enter the LDAP port that will be used for administration traffic. – **Enable Administration by LDAP and HTTP**: Enter the LDAP and HTTP ports that will be used for administration traffic. – **Enable Administration by HTTP**: Enter the HTTP port that will be used for administration traffic. • **LDAP Port**: Enter the port that you wish to use for administering OUD via LDAP. • **HTTP Port**: Enter the port that you wish to use for administering OUD via HTTP. For example, `8444` (`LDAP_ADMIN_PORT`). • **Root User DN**: Enter an administrative user. For example, `cn=oudadmin`. • **Password**: Enter the password you wish to assign to the ouadmin user. • **Password (Confirm)**: Repeat the password. Click **Next**.
Ports	Enter the following details: **LDAP** • **Enable**: Select if you wish to enable non SSL communications with OUD. • **on Port**: Select the Port you wish to use (`LDAP_PORT`). • **Enable Start TLS for LDAP**: Select **Enable StartTLS for LDAP** to specify that the LDAP connection handler should allow clients to use the StartTLS extended operation to initiate secure communication over an otherwise insecure connection. **LDAPS** • **Enable**: Select if you wish to enable SSL communications with OUD. • **on Port**: Select the Port you wish to use (`LDAP_SSL_PORT`). If you select this option, you must provide the SSL certificate information below. **Certificate** You can use the existing certificate or generate a self signed certificate. Self signed certificates are not recommended for production deployments. • **Generate self signed certificate**: Select this if you wish OUD to generate its own certificate. • **Use an existing certificate**: Select this if you are using an existing certificate. Select the type of certificate, the location of the Keystore, and the Keystore pin.
Topology Options	Enter the following details: • Select **This will server will be part of a replication topology**. • **Replication Port**: Enter the replication port. For example: `8989 (OUD_REPLICATION_PORT)`. • **Configure As Secure**: Select this if you wish the replication traffic to be encrypted. • There is already a server in the topology selected. Enter the following: – **Host Name**: Name of the existing Oracle Unified Directory server host. For example, `ldaphost1.example.com`. – **Administrator Connector Port**: `4444` (`LDAP_ADMIN_PORT`). – **Admin User**: Name of the Oracle Unified Directory admin user on `ldaphost1.example.com`. For example, `cn=oudadmin`. – **Admin Password**: Administrator password. Click **Next**. If you see a **Certificate Not Trusted** dialogue, it is because you are using self signed certificates. Click **Accept Permanently**. For more information, see Setting Up Replication During Installation.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Screen	Description
Create Global Administrator	Enter the following details: • **Global Administrator ID**: Enter the name of an account you want to use for managing Oracle Unified Directory replication. For example: `oudmanager`. • **Global Administrator Password / Confirmation**: Enter a password for this account. Click **Next**.
Data Replication	Select **dc=example,dc=com**. Click **Next**.
Oracle Components Integration	If you selected any products to integrate with, when you configured LDAPHOST1, then select the same option here. Click **Next**.
Server Tuning	You have the option of allocating specific resources to the OUD instance. You can choose to accept the default of change the resource allocations based on your deployment. If this server is used only for OUD as in a distributed deployment then be sure to check the box Dedicated Machine for OUD. Click **Next**.
Review	Verify that the information displayed is correct. If you wish the OUD server to be started after configuration, ensure that you select the option **Start the server**.
Finished	Click **Close**.

- Validating Oracle Unified Directory on LDAPHOST2

Validating Oracle Unified Directory on LDAPHOST2

After configuration you can validate that Oracle Unified Directory is working by performing a simple search. To do this issue the following command:

```
export $OUD_ORACLE_INSTANCE=/u02/oracle/config/instances/oud2


cd $OUD_ORACLE_INSTANCE/bin


./ldapsearch -h ldaphost2.example.com -p 1389 -D cn=oudadmin -b "" -s base
"(objectclass=*)" supportedControl
```

If Oracle Unified Directory is working correctly, you see a list `supportedControl` entries returned.

If you have enabled SSL on the directory, you can test it using the command:

```
export $OUD_ORACLE_INSTANCE=/u02/oracle/config/instances/oud2


cd $OUD_ORACLE_INSTANCE/bin


./ldapsearch -h ldaphost2.example.com -p 1636 --useSSL -D cn=oudadmin -b "" -
s base "(objectclass=*)" supportedControl
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

To check that Oracle Unified Directory replication is enabled, issue the command:

```
cd $OUD_ORACLE_INSTANCE/bin
```

```
./status
```

You are prompted for the Administrator bind DN (`cn=oudadmin`) and its password.

You then see output similar to the following example. Replication is set to enable.

```
--- Server Status ---
Server Run Status: Started
Open Connections: 2
          --- Server Details ---
Host Name: ldaphost1
Administrative Users: cn=oudadmin
Installation Path: /u02/oracle/products/oud/oud
Instance Path: /u02/oracle/config/instances/oud1
Version: Oracle Unified Directory 14.1.2.1.241204
Java Version: 21.0.4
Administration Connector: Port 4444 (LDAPS)
          --- Connection Handlers ---
Address:Port : Protocol : State
-------------:----------------------:---------
-- : LDIF : Disabled
8989 : Replication (secure) : Enabled
0.0.0.0:161 : SNMP : Disabled
0.0.0.0:389 : LDAP : Disabled
0.0.0.0:1636 : LDAPS : Enabled
0.0.0.0:1689 : JMX : Disabled
          --- Data Sources ---
Base DN: cn=virtual acis
Backend ID: virtualAcis
Entries: 0
Replication: Disabled
Base DN: dc=edg,dc=com
Backend ID: userRoot
Entries: 37
Replication: Enabled
Missing Changes: 0
Age Of Oldest Missing Change: <not available>
Subject:
```

Preparing an Existing LDAP Directory

Before you can use an LDAP directory with Oracle Identity and Access Management, it must be extended with object classes required by Oracle Access Manager.

In addition, certain users and groups need to be seeded into the directory. These users and groups will be used by the various Oracle Identity and Access Management products as described later.

The preparation of LDAP is performed using a tool called `ldapConfigTool` which is included in Patch 38047590. See, Identifying and Obtaining Software Distributions for an Enterprise Deployment.

This section includes the following topics:

- About the Enterprise Deployment Users and Groups

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Creating a Configuration File
- Preparing OUD as the Identity Store
- Creating Access Control Lists in Non-Oracle Directories

About the Enterprise Deployment Users and Groups

The following topics provide important information on the purpose and characteristics of the enterprise deployment administration users and groups.

- About Using Unique Administration Users for Each Domain

About Using Unique Administration Users for Each Domain

When you use a central LDAP user store, you can provision users and groups for use with multiple Oracle WebLogic Server domains. As a result, there is a possibility that one WebLogic administration user can have access to all the domains within an enterprise.

It is a best practice to create and assign a unique distinguished name (DN) within the directory tree for the users and groups that you provision for the administration of your Oracle Fusion Middleware domains.

For example, create two users called `oamLDAP` and `oimLDAP` which is used to connect the WebLogic domain to LDAP. This allows the domain to see the users and groups which exist in the directory. You can create a different user for each domain or use a single user for multiple domains. Under no circumstances should the default LDAP administration user be used for this purpose. You must create these users in the `systemids` container. This container is used for system users that are not normally visible to users. Placing the user into the `systemids` container ensures that customers who have Oracle Identity Governance do not reconcile this user.

Using a different user for Oracle Access Management (OAM) and Oracle Identity Governance (OIG) LDAP connections ensures that the user that OAM uses to connect to LDAP has a restricted privilege set.

Create a user called `weblogic_iam` and an administration group called `WLSAdministrators`. Users in the `WLSAdministrators` group will be allowed to access Oracle Fusion Middleware Control and the WebLogic Remote Console.

Create a user called `oamadmin` and an administration group called `OAMAdministrators`. Users in the `OAMAdministrators` group are allowed to access the Oracle Access Policy Manager and Oracle Access Manager Console.

Creating a Configuration File

Create a property file `iam.props`, to use when preparing the Identity Store and as a basis for later integration and configuration processes. The file will have the structure described in this section. When creating the file do not include any blank lines.

The property files in this section are complete examples. Some of the parameters specified in the file will not be used until later configuration steps in the guide. It is only necessary to include the properties for the products you are going to use.

This section includes the following topics:

- Oracle Unified Directory Example
- Explanation of Property Values

Copyright © 2018, 2025, Oracle and/or its affiliates.

Oracle Unified Directory Example

The following is and example configuration file for Oracle Unified Directory. For an explanation of the parameters, see Explanation of Property Values:

```
# Common
IDSTORE_HOST: idstore.example.com
IDSTORE_PORT: 1636
IDSTORE_ADMIN_PORT: 4444
IDSTORE_KEYSTORE_FILE: /u02/oracle/config/keystores/idmcerts.p12
IDSTORE_ADMIN_KEYSTORE_FILE: /u02/oracle/config/instances/oud1/config/admin-
keystore
IDSTORE_BINDDN: cn=oudadmin
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=example,dc=com
IDSTORE_SEARCHBASE: dc=example,dc=com
IDSTORE_USERNAMEATTRIBUTE: cn
IDSTORE_LOGINATTRIBUTE: uid
OAM_SERVER_LOGIN_ATTRIBUTE: uid
IDSTORE_USERSEARCHBASE: cn=Users,dc=example,dc=com
IDSTORE_NEW_SETUP: true
IDSTORE_DIRECTORYTYPE: OUD
# OAM
IDSTORE_OAMADMINUSER: oamadmin
IDSTORE_OAMSOFTWAREUSER: oamLDAP
OAM_IDSTORE_ROLE_SECURITY_ADMIN: OAMAdministrators
# OAM and OIG
IDSTORE_SYSTEMIDBASE: cn=SystemIDs,dc=example,dc=com
# OIG
IDSTORE_OIMADMINGROUP: OIMAdministrators
IDSTORE_OIMADMINUSER: oimLDAP
# WebLogic
IDSTORE_WLSADMINUSER : weblogic_iam
IDSTORE_WLSADMINGROUP : WLSAdministrators

LOG_LEVEL: ALL
```

Note

You can also specify the passwords directly in the file, if required. If you do not specify the passwords, you will be prompted for them at runtime. The parameters are:

- IDSTORE_KEYSTORE_PASSWORD
- IDSTORE_ADMIN_KEYSTORE_PASSWORD
- IDSTORE_PASSWD
- IDSTORE_OAMADMINUSER_PWD
- IDSTORE_OAMSOFTWAREUSER_PWD
- IDSTORE_OAMADMINUSER_PWD
- WLSPASSWD
- OAM_IDM_DOMAIN_WEBGATE_PASSWD

Copyright © 2018, 2025, Oracle and/or its affiliates.

Explanation of Property Values

The following table explains the configuration file property values required in this section.

LDAP Properties

Table 12-2 LDAP Variables Used in This Chapter

Variable	Sample Value	Description
IDSTORE_HOST	`idstore.example.com`	The host and port of your Identity Store directory. When preparing the Identity Store these should point to one of the LDAP instances. When configuring components such as OAM or OIG they should point to the load balancer entry point.
IDSTORE_PORT	`1636`	The port of your Identity Store directory. When preparing the Identity Store these should point to one of the LDAP instances. When configuring components such as OAM or OIG they should point to the load balancer entry point.
IDSTORE_DIRECTORYTYPE	`OUD`	The type of directory you are using. Valid value is OUD.
IDSTORE_BINDDN	`cn=oudadmin`	An administrative user in the Identity Store Directory.
IDSTORE_PASSWD	`password`	The the password of IDSTORE_BINDDN. You will be prompted for the value if not supplied.
IDSTORE_SEARCHBASE	`dc=example,dc=com`	The location in the directory where Users and Groups are stored.
IDSTORE_LOGINATTRIBUTE	`uid`	The LDAP attribute, which contains the users Login name.
IDSTORE_USERSEARCHBASE	`cn=Users,dc=example,dc=com`	The location in the directory where Users are Stored.
IDSTORE_GROUPSEARCHBASE	`cn=Groups,dc=example,dc=com`	The location in the directory where Groups are Stored.
IDSTORE_SYSTEMIDBASE	`cn=SystemIDs,dc=example,dc=com`	The location of a container in the directory where system users can be placed when you do not want them in the main user container.
IDSTORE_USERNAMEATTRIBUTE	`cn`	The name of the LDAP attribute which stores a users name.
IDSTORE_KEYSTORE_FILE	`/u02/oracle/config/keystores/idmcerts.p12`	The location of the LDAP Truststore for LDAP connections.
IDSTORE_KEYSTORE_PASSWORD	`password`	The password of the IDSTORE_KEYSTORE_FILE.

Copyright © 2018, 2025, Oracle and/or its affiliates.

OUD Properties

Table 12-3 OUD Variables Used in This Chapter

Variable	Sample Value	Description
IDSTORE_ADMIN_PORT	`4444`	The administration port of your Oracle Unified Directory instance. If you are not using Oracle Unified Directory, you can leave out this parameter.
IDSTORE_ADMIN_KEYSTORE_FILE	`/u02/oracle/config/instances/oud1/config/admin-keystore`	The location of the Oracle Unified Directory Keystore file. It is used to enable communication with Oracle Unified Directory using the Oracle Unified Directory administration port. It is called admin-keystore and is located in *OUD_INSTANCE_HOME/OUD/config*. If you are not using Oracle Unified Directory, you can leave out this parameter.
IDSTORE_ADMIN_KEYSTORE_PASSWORD	`password`	The encrypted password of the Oracle Unified Directory keystore. To obtain the password of the Administration Trust Store issue the command: `$OUD_INSTANCE_HOME/oud1/bin/dsconfig -h ldaphost1.example.com -p 4444 -D cn=oudadmin -j ~/oud.pwd -X -n get-key-manager-provider-prop --provider-name Administration --property key-store-pin --showKeystorePassword` Where `oud.pwd` is a file which contains your administrator password.
IDSTORE_NEW_SETUP	`true`	This parameter is used when preparing a directory for the first time.

Copyright © 2018, 2025, Oracle and/or its affiliates.

OAM Properties

Table 12-4 OAM Variables Used in This Chapter

Variable	Sample Value	Description
IDSTORE_OAMSOFTWAREUSER	`oamLDAP`	A user that gets created in LDAP that is used when Access Manager is running to connect to the LDAP server.
IDSTORE_PWD_OAMSOFTWAREUSER	`password`	The password you want to assign to the IDSTORE_OAMSOFTWAREUSER. If not supplied, you will be prompted for it.
OAM_IDSTORE_ROLE_SECURITY_ADMIN	`OAMAdministrators`	The name of the group, which is used to allow access to the OAM console. Only users assigned to this group will be able to access the OAM Console.
OAM_SERVER_LOGIN_ATTRIBUTE	`uid`	The name of the LDAP attribute where userids are stored, this should be the same as the IDSTORE_LOGIN_ATTRIBUTE.
OAM_IDM_DOMAIN_WEBGATE_PASSWD	`password`	The password you want to assign to the webgate.

OIG Properties

Table 12-5 OIG Variables Used in This Chapter

Variable	Sample Value	Description
IDSTORE_OIMADMINGROUP	`OIMAdministrators`	The name of the group you want to create to hold your Oracle Identity Governance administrative users.
IDSTORE_OIMADMINUSER	`oimLDAP`	The user that Oracle Identity Governance uses to connect to the Identity store.
IDSTORE_PWD_OIMADMINUSER	`password`	The password of IDSTORE_OIMADMINUSER. If there is no value, you will be prompted for it .

OIG Properties

Table 12-6 OIG Variables Used in This Chapter

Variable	Sample Value	Description
IDSTORE_WLSADMINUSER	`weblogic_iam`	The username to be created for logging in to the web logic domain once it is enabled by Single Sign-On.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 12-6 (Cont.) OIG Variables Used in This Chapter

Variable	Sample Value	Description
WLSPASSWD	`password`	The password you want to assign to the IDSTORE_WLSADMINUSER. If not supplied, you will be prompted for this value.
IDSTORE_WLSADMINGROUP	`WLSAdministrators`	The name of the group to which users who are allowed to log in to the WebLogic system components, such as the WLS Remote Console and EM, belong.

Preparing OUD as the Identity Store

Before an Oracle LDAP directory can be used with Oracle Identity and Access Management, the directory needs to be pre-configured.

This process involves the creation of additional object classes within the directory and the seeding of users that the Oracle Identity and Access Management suite will use to connect to the directory. There are two phases to the configuration process:

- Pre-configure: This adds the required object classes.
- Seeding of Users.

To do this, perform the following tasks on LDAPHOST1 if you are extending Oracle Unified Directory:

- Setting Environment Variables
- Directory Pre-Configuration
- Seeding Users and Groups
- Granting OUD changelog Access

Setting Environment Variables

To help navigate this section, to be able to copy sample commands without modification you can set the following environment variables, replacing the values with values appropriate to your environment.

```
export DIR_HOME=/u02/oracle/products/oud
export ORACLE_HOME=/u02/oracle/products/oud
export JAVA_HOME=/u02/oracle/products/jdk
export PATH=$JAVA_HOME/bin:$PATH
export LDAPCONFIG=$ORACLE_HOME/idmtools
```

Directory Pre-Configuration

Before an Oracle LDAP directory can be used with Oracle Identity and Access Management, the directory needs to be pre-configured.

Copyright © 2018, 2025, Oracle and/or its affiliates.

This process involves the creation of additional object classes within the directory and the seeding of users that the Oracle Identity and Access Management suite will use to connect to the directory. There are two phases to the configuration process:

- Pre-configure: This adds the required object classes.
- Seeding of Users.

To perform the pre-configuration task, perform the following on LDAPHOST1 if you are extending Oracle Unified Directory:

- Configure the Identity Store using the following `ldapConfigTool` command.

  ```
  cd $LDAPCONFIG/bin
  ```

  ```
  ./ldapConfigTool.sh -preConfigIDStore input_file=configfile
  ```

  For example:

  ```
  ./ldapConfigTool.sh -preConfigIDStore input_file=iam.props
  ```

  When the command runs, you are prompted to enter the password of the account you are connecting to the Identity Store with. This command might take some time to complete.

  Check the log file for any errors or warnings, and correct them. The file with the name `automation.log` is created in the directory from where you run the tool.

Seeding Users and Groups

You must seed the Identity Store with users and groups that are required by the Identity Management components.

To seed the Identity Store, perform the following tasks on LDAPHOST1:

- Seed the users and groups in the Identity Store using the following `ldapConfigTool` command.

  ```
  cd $LDAP_CONFIG_HOME/bin
  ```

  ```
  ./ldapConfigTool.sh -prepareIDStore mode=MODE input_file=configfile
  pwd_file=passwordfile
  ```

  The value selected for `MODE` determines the type of users to be created. Possible values for `MODE` are: `OAM`, `OIM`, and `WLS`.

  - In all topologies, when you enable single sign-on for your administrative consoles, you must ensure that there is a user in your Identity Store that has the permissions to log in to your WebLogic Remote Console and Oracle Enterprise Manager Fusion Middleware Control. Type:

    ```
    ./ldapConfigTool.sh -prepareIDStore mode=WLS input_file=iam.props
    ```

  - If your topology includes Oracle Access Management, you must seed the Identity Store with users that are required by Oracle Access Management. Type:

    ```
    ./ldapConfigTool.sh -prepareIDStore mode=OAM input_file=iam.props
    ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

- If your topology includes Oracle Identity Governance, you must seed the Identity Store with the `xelsysadm` user and assign it to an Oracle Identity Governance administrative group. You must also create a user outside of the standard `cn=Users` location to be able to perform reconciliation. This user is also the user that should be used as the bind DN when connecting to directories with Oracle Virtual Directory. Type

  ```
  ./ldapConfigTool.sh -prepareIDStore mode=OIM input_file=iam.props
  ```

> **Note**
>
> This command also creates a container in your Identity Store for reservations.

> **Note**
>
> When entering a password for `xelsysadm` ensure that it is the same as the OIG policy, in it must be at least 8 characters long, contain an uppercase character, and a number.

When the command runs, you are prompted to enter the password of the account you are connecting to the Identity Store with.

After running each command, check the log file for any errors or warnings and correct them. The file with the name `automation.log` is created in the directory from where you run the tool.

Granting OUD changelog Access

If you are using Oracle Unified Directory and wish to integrate with Oracle Identity Governance, you must grant access to the changelog by performing the following steps on each OUD instance.

1. Create a property file call `oudacl.props` with the following contents:

   ```
   IDSTORE_SEARCHBASE: dc=example,dc=com
   IDSTORE_HOST: ldaphost1.example.com
   IDSTORE_ADMIN_PORT: 4444
   IDSTORE_ADMIN_KEYSTORE_FILE: /u02/oracle/config/instances/oud1/config/
   admin-keystore
   IDSTORE_ADMIN_KEYSTORE_PASSWORD: password
   IDSTORE_BINDDN: cn=oudadmin
   IDSTORE_PASSWD: password
   ```

   For more information on these properties, see Explanation of Property Values.

2. Run the `ldapConfigTool` to create the ACLs in the directory schema:

   ```
   cd $LDAP_CONFIG_HOME/bin
   ```

   ```
   ./ldapConfigTool.sh -setupOUDacl input_file=oudacl.props log_level=FINEST
   ```

3. Repeat the above steps for each OUD instance.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Creating Access Control Lists in Non-Oracle Directories

In the preceding sections, you seeded the Identity Store with users and artifacts for the Oracle components. If your Identity Store is not Oracle Unified Directory, Oracle Directory Server Enterprise Edition, you must set up the access control lists (ACLs) to provide appropriate privileges to the entities you created, this is true even if using Oracle Virtual Directory in front of them. This section lists the artifacts created and the privileges required for the artifacts.

- Systemids. The System ID container is created for storing all the system identifiers. If there is another container in which the users are to be created, that is specified as part of the admin.
- Access Manager Admin User. This user is added to the OAM Administrator group, which provides permission for the administration of the Oracle Access Management Console. No LDAP schema level privileges are required, since this is just an application user.
- Access Manager Software User. This user is added to the groups where the user gets read privileges to the container. This is also provided with schema admin privileges.
- Oracle Identity Governance user oigLDAP under System ID container. Password policies are set accordingly in the container. The passwords for the users in the System ID container must be set up so that they do not expire.
- Oracle Identity Governance administration group. The Oracle Identity Governance user is added as its member. The Oracle Identity Governance admin group is given complete read/write privileges to all the user and group entities in the directory.
- WebLogic Administrator. This is the administrator of the IDM domain for Oracle Virtual Directory
- WebLogic Administrator Group. The WebLogic administrator is added as a member. This is the administrator group of the IDM domain for Oracle Virtual Directory.
- Reserve container. Permissions are provided to the Oracle Identity Governance admin group to perform read/write operations.

Copyright © 2018, 2025, Oracle and/or its affiliates.

13

Configuring Oracle Unified Directory Service Manager

Install and configure Oracle Unified Directory Service Manager (OUDSM) in order to manage Oracle Unified Directory (OUD).

Oracle Unified Directory Service Manager (OUDSM) is a Graphical User Interface (GUI) tool used to manage Oracle Unified Directory. It is not mandatory to be installed in the production environments; however, OUDSM makes managing Oracle Unified Directory easier.

It is recommended that, if you are installing OUDSM, create it in its own light-weight domain. As OUDSM is a management tool that does not need web access, it need not be accessed via the Oracle HTTP Server.

This chapter includes the following topics:

- Variables Used When Configuring Oracle Unified Directory
  The procedures for installing and configuring Oracle Unified Directory reference use a series of variables that you can replace with the actual values used in your environment.
- Setting Environment Variables
  Set environment variables used in this chapter.
- Installing the Oracle Fusion Middleware Infrastructure
  Use the following sections to install the Oracle Fusion Middleware Infrastructure software in preparation for configuring a new domain for Oracle Unified Directory Services Manager.
- Creating a Domain for Oracle Unified Directory Service Manager
  You can create a domain for Oracle Unified Directory Service Manager (OUDSM) without depending on the Oracle Database or the Repository Creation Utility (RCU) using the WebLogic Scripting Tool (WLST) command.
- Starting the Oracle Unified Directory Service Manager Domain
  After configuring the Oracle Unified Directory Service Manager domain, start the Administration Server to manage the domain.
- Configuring Oracle HTTP Server for Oracle Unified Directory Services Manager
  If you want to access the Oracle Unified Directory Services Manager (OUDSM) console through Oracle Web Servers, then you must add the necessary entry to one of your administrative virtual hosts.

Variables Used When Configuring Oracle Unified Directory

The procedures for installing and configuring Oracle Unified Directory reference use a series of variables that you can replace with the actual values used in your environment.

The following directory location variables are used in these procedures:

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 13-1 Variables Used in This Chapter

Variable	Sample Value	Description
ORACLE_HOME	`/u02/oracle/products/oudsm`	Oracle Home for the Oracle Unified Directory Services Manager.
JAVA_HOME	`/u02/oracle/products/jdk`	The JDK home directory.

Setting Environment Variables

Set environment variables used in this chapter.

To help navigate this guide and so you are able to copy sample commands without modification, you can set the following environment variables replacing the values with values appropriate to your environment.

```
export ORACLE_HOME=/u02/oracle/products/oud
export JAVA_HOME=/u02/oracle/products/jdk
export PATH=$JAVA_HOME/bin:$PATH
export LOCAL_CONFIG_DIR=/u02/oracle/config
```

Installing the Oracle Fusion Middleware Infrastructure

Use the following sections to install the Oracle Fusion Middleware Infrastructure software in preparation for configuring a new domain for Oracle Unified Directory Services Manager.

- Installing a Supported JDK
- Starting the Oracle Identity Management Quick Installer
- Navigating the Installation Screens
- Installing the Stack Bundle Patch
- Checking the Directory Structure
- Installing Oracle Fusion Middleware Infrastructure on the Other Host Computers

Installing a Supported JDK

Oracle Fusion Middleware requires that a certified Java Development Kit (JDK) is installed on your system.

- Locating and Downloading the JDK Software
- Installing the JDK Software

Locating and Downloading the JDK Software

To find a certified JDK, see the certification document for your release on the Oracle Fusion Middleware Supported System Configurations page.

After you identify the Oracle JDK for the current Oracle Fusion Middleware release, you can download an Oracle JDK from the following location on Oracle Technology Network:

Copyright © 2018, 2025, Oracle and/or its affiliates.

```
https://www.oracle.com/java/technologies/downloads/
```

Be sure to navigate to the download for the Java SE JDK.

Installing the JDK Software

Oracle Fusion Middleware requires you to install a certified Java Development Kit (JDK) on your system.

For more information about the recommended location for the JDK software, see Understanding the Recommended Directory Structure for an Enterprise Deployment.

To install JDK 21.0:

1. Change directory to the location where you downloaded the JDK archive file.

   ```
   cd download_dir
   ```

2. Unpack the archive into the JDK home directory, and then run the following commands:

   ```
   tar -xzvf jdk-21.0.4+8_linux-x64_bin.tar.gz
   ```

   Note that the JDK version listed here was accurate at the time this document was published. For the latest supported JDK, see the *Oracle Fusion Middleware System Requirements and Specifications* for the current Oracle Fusion Middleware release.

3. Move the JDK directory to the recommended location in the directory structure.

   For example:

   ```
   ln -s jdk-21.0.4 /u02/oracle/products/jdk
   ```

   See File System and Directory Variables Used in This Guide.

4. Run the following command to verify that the appropriate `java` executable is in the path and your environment variables are set correctly:

   ```
   java -version
   ```

   The Java version in the output should be displayed as “21.0.4”.

Starting the Oracle Identity Management Quick Installer

To start the installation program, perform the following steps.

1. Log into LDAPHOST1.
2. Go to the directory where you downloaded the installation program.
3. Launch the installation program by invoking the `java` executable from the JDK directory on your system, as shown in the example below.

   ```
   $JAVA_HOME/bin/java -jar fmw_14.1.2.0.0_infrastructure_generic.jar
   ```

   In this example:

   - If you download the distribution from the Oracle Technology Network (OTN), then the JAR file is typically packaged inside a downloadable ZIP file.

     To install the software required for the initial Infrastructure domain, the distribution you want to install is:

     fmw_14.1.2.0.0_infrastructure_generic.jar.

Copyright © 2018, 2025, Oracle and/or its affiliates.

For more information about the actual file names of each distribution, see Identifying and Obtaining Software Downloads for an Enterprise Deployment.

When the installation program appears, you are ready to begin the installation. See Navigating the Installation Screens for a description of each installation program screen.

Navigating the Installation Screens

The installation program displays a series of screens, in the order listed in the following table.

If you need additional help with any of the installation screens, click the screen name or click the **Help** button on the screen.

Table 13-2 Navigating the Infrastructure Installation Screens

Screen	Description
Installation Inventory Setup	On UNIX operating systems, this screen appears if you are installing any Oracle product on this host for the first time. Specify the location where you want to create your central inventory. Ensure that the operating system group name selected on this screen has write permissions to the central inventory location. See Understanding the Oracle Central Inventory in *Installing Software with the Oracle Universal Installer*. **Note** Oracle recommends that you configure the central inventory directory on the products shared volume. Example: `/u02/oracle/products/oraInventory` You may also need to execute the `createCentralinventory.sh` script as root from the `oraInventory` folder after the installer completes.
Welcome	This screen introduces you to the product installer.
Auto Updates	Use this screen to search My Oracle Support automatically for available patches or automatically search a local directory for patches that you have already downloaded for your organization.
Installation Location	Use this screen to specify the location of your Oracle home directory. For the purposes of an enterprise deployment, enter the value of the $ORACLE_HOME variable for the product listed in Table 8-2. For example, `/u02/oracle/products/oudsm`
Prerequisite Checks	This screen verifies that your system meets the minimum requirements. If there are any warning or error messages, refer to the Oracle Fusion Middleware System Requirements and Specifications document on the Oracle Technology Network (OTN).
Installation Summary	Use this screen to verify the installation options that you have selected. If you want to save these options to a response file, click **Save Response File** and provide the location and name of the response file. Response files can be used later in a silent installation situation. For more information about silent or command-line installation, see Using the Oracle Universal Installer in Silent Mode in *Installing Software with the Oracle Universal Installer*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 13-2 (Cont.) Navigating the Infrastructure Installation Screens

Screen	Description
Installation Progress	This screen allows you to see the progress of the installation.
Installation Complete	This screen appears when the installation is complete. Review the information on this screen, then click **Finish** to dismiss the installer.

Installing the Stack Bundle Patch

After installing the software binaries, you must apply the latest Stack Bundle Patch.

For an enterprise deployment you must download and apply the July 2025 Stack Bundle Patch or later. For more details, see Identifying and Obtaining Software Distributions for an Enterprise Deployment.

To apply the patch run the following commands:

1. After downloading the patch, unzip it to your preferred location. For example:

   ```
   unzip p38184742_141210_Linux-x86-64.zip
   ```

   This location will be known as $PATCH_DIR.
2. Navigate to the $PATCH_DIR:

   ```
   cd $PATCH_DIR/tools/spbat/generic/SPBAT/
   ```

3. Apply the patch using the command:

   ```
   ./spbat.sh -type oud -phase downtime -mw_home $ORACLE_HOME -
   spb_download_dir $PATCH_DIR
   ```

Checking the Directory Structure

After you install the Oracle Identity and Access Management and create the Oracle home, you should see the directory and sub-directories listed in this topic. The contents of your installation vary based on the options that you selected during the installation.

To check the directory structure:

1. Navigate to the $ORACLE_HOME:

   ```
   cd $ORACLE_HOME
   ```

2. Enter the following command:

   ```
   ls --format=single-column
   ```

   The directory structure on your system must match the structure shown in the following example:

   ```
   bin
   cfgtoollogs
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
coherence
domain-registry.xml
em
envPropertiesCache
idm
install
inventory
jdeveloper
jlib
lib
mft
OPatch
opmn
oracle_common
oraInst.loc
osb
oui
root.sh
soa
wlserver
```

See What are the Key Oracle Fusion Middleware Directories? in *Understanding Oracle Fusion Middleware*.

Installing Oracle Fusion Middleware Infrastructure on the Other Host Computers

If you have configured a separate shared storage volume or partition for secondary hosts, then you must install the Infrastructure on one of those hosts.

See Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment.

To install the software on the other host computers in the topology, log in to each host, and use the instructions in Starting the Oracle Identity Management Quick Installer and Navigating the Installation Screens to create the Oracle home on the appropriate storage device.

You must install the Stack Bundle Patch and any other mandatory patches outlined in Identifying and Obtaining Software Distributions for an Enterprise Deployment.

Creating a Domain for Oracle Unified Directory Service Manager

You can create a domain for Oracle Unified Directory Service Manager (OUDSM) without depending on the Oracle Database or the Repository Creation Utility (RCU) using the WebLogic Scripting Tool (WLST) command.

> **Note**
>
> This is the Oracle preferred approach to set up a domain for OUDSM. It is recommended not to extend this domain with any other products or components. In this approach, you do not have to run `config.sh`.

To set up the OUDSM domain using the WLST, do the following:

1. Launch the WLST by running the following command:

Copyright © 2018, 2025, Oracle and/or its affiliates.

On UNIX:

```
$ORACLE_HOME/oracle_common/common/bin/wlst.sh
```

2. Run the following command to create a compact domain for Oracle Unified Directory Services Manager:

```
createOUDSMDomain(domainLocation=path_to_domain_home,weblogicPort=wls_port,
weblogicSSLPort=ssl_port,weblogicUserName=wls_user,weblogicUserPassword=wls
_password)
```

In the above command, specify the values for the following parameters:

- `domainLocation`: This is the absolute path to the domain home. For example, `$LOCAL_CONFIG_DIR/domains/OUDSMDomain`.
- `weblogicPort`: This is the WebLogic port. This value must be unique to the server.
- `weblogicSSLPort`: This is the WebLogic SSL port. This parameter is optional and is not enabled if not passed.
- `weblogicUserName`: This is the WebLogic user name. This parameter is optional. If not specified, the default value `weblogic` is passed.
- `weblogicUserPassword`: This is the WebLogic Administration Server user password.

For example:

```
createOUDSMDomain(domainLocation="/u02/oracle/config/domains/
OUDSMDomain",weblogicPort=7001,weblogicSSLPort=7002,weblogicUserPassword='<pas
sword>')
```

3. Enter exit() to exit out of wlst.

Starting the Oracle Unified Directory Service Manager Domain

After configuring the Oracle Unified Directory Service Manager domain, start the Administration Server to manage the domain.

To do this, complete the following steps:

1. Start the Administration Server using the following command:

```
$LOCAL_CONFIG_DIR/domains/OUDSMDomain/bin/startWebLogic.sh
```

2. Verify that the Administration Server is up and running by accessing the Oracle Unified Directory Services Manager at the following URL:

```
http://ldaphost1.example.com:7001/oudsm
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

Configuring Oracle HTTP Server for Oracle Unified Directory Services Manager

If you want to access the Oracle Unified Directory Services Manager (OUDSM) console through Oracle Web Servers, then you must add the necessary entry to one of your administrative virtual hosts.

Once you have configured your Oracle HTTP server as described in Configuring Oracle HTTP Server for an Enterprise Deployment, then you can configure the Oracle HTTP Server to route requests to the Oracle Unified Directory Services Manager. To do this:

1. Add the following entries to the `oudsm.conf` file located at `$OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/moduleconf/`:

```
Listen 4450
<VirtualHost webhost1.example.com:4450>
    ServerName https://oudsm.example.com:443
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
    UseCanonicalName On

    <Location /oudsm>
      WebLogicHost ldaphost1.example.com
      WebLogicPort 7001
    </Location>
</VirtualHost>
```

> **Note**
>
> There are separate directories for configuration and runtime instance files. The runtime files under the `.../OHS/instances/ohsn/*` folder should not be edited directly. Edit only the `.../OHS/ohsn/*` configuration files.

2. Copy the `oudsm.conf` file to the following configuration directory of the second Oracle HTTP Server instance (ohs2):

   `$OHS_DOMAIN_HOME/config/fmwconfig/components/ohs2/moduleconf/`

3. Restart the Oracle HTTP server instances on WEBHOST1 and WEBHOST2.

Copyright © 2018, 2025, Oracle and/or its affiliates.

14

Configuring Oracle Access Management

Install and configure Oracle Access Management (OAM).

The following topics describe how to install and configure an initial domain, which can be used as the starting point for an enterprise deployment. Later chapters in this guide describe how to extend this initial domain with the various products and components that comprise the enterprise topology you are deploying.

A complete Oracle Identity and Access Management uses a split domain deployment, where there is a single domain for Oracle Access Management and a different one for Oracle Identity Governance. You must create a separate infrastructures for Access and Governance.

- About the Initial Infrastructure Domain
  Before you create the initial Infrastructure domain, ensure that you review the key concepts.
- Variables Used When Creating Infrastructure for Oracle Access Management
  As you perform the tasks in this chapter, you will be referencing the variables listed in this section.
- Setting Environment Variables
  Set environment variables used in this chapter.
- URLs Used in This Chapter
  This section describes the URLs used in this chapter.
- Installing the Oracle Fusion Middleware
  Install the Oracle Fusion Middleware software in preparation for configuring a new domain for Oracle Access Management.
- Configuring LDAP
  If you haven't already done so, you need to configure your LDAP directory.
- Creating the Database Schemas for Access Manager
  Oracle Fusion Middleware components require the existence of schemas in a database before you configure a Fusion Middleware Infrastructure domain for Oracle Access Management.
- Verifying Schema Access
- Configuring the Oracle Access Management Domain
  The following topics provide instructions for creating an Oracle Access Management domain using the Fusion Middleware Configuration wizard.
- Enabling SSL
  If you are configuring End to End SSL, you must perform additional steps.
- Configuring a Per Host Node Manager for an Enterprise Deployment
  For specific enterprise deployments, Oracle recommends that you configure a per-host Node Manager, as opposed to the default per-domain Node Manager.
- Configuring the Domain Directories and Starting the Servers
  After you have configured the domain and configured the Node Manager, you can start the Administration Server by using the Node Manager. In an enterprise deployment, the Node Manager is used to start and stop the Administration Server and all the Managed Servers in the domain.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Removing OAM Server from WebLogic Server defaultCoherenceCluster
  You must exclude all Oracle Access Management (OAM) clusters (including policy manager and OAM runtime server) from the default WebLogic Server coherence cluster using the WebLogic Remote Console.
- Tuning the WebLogic Server
  Tune the WebLogic Server for optimum performance by adding the Minimum Thread Constraint and removing the Max Thread and Capacity constraints.
- Adding a Load Balancer Certificate to the Oracle Keystore Service
  The Oracle Access Manager forgot password functionality requires that the SSL certificate used by the load balancer be added to the Oracle Keystore Service Trusted Certificates.
- Tuning the oamDS Data Source
  For optimium performance, increase the number of connections allowed by the OAM data source.
- Configuring the WebLogic Proxy Plug-In
  Before you can validate that requests are routed correctly through the Oracle HTTP Server instances, you must set the `WebLogic Plug-In Enabled` parameter.
- Configuring and Integrating with LDAP
  Configure OAM to use the LDAP directory.
- Updating WebGate Agents
  Update the WebGate SSO Agents to use the new security model.
- Updating Host Identifiers
  When you access your domain you enter using different load balancer entry points. Each of these entry points (virtual hosts) need to be added to the Policy list.
- Updating Idle Timeout Value
  The default timeout value set in Oracle Access Manager is often too long and can cause issues such as not logging a session out after that session has timed out. Therefore, it is recommended that this value is reduced to 15 minutes.
- Validating the Authentication Providers
  Set the order of identity assertion and authentication providers in the WebLogic Remote Console.
- Configuring Oracle ADF and OPSS Security with Oracle Access Manager
  Some Oracle Fusion Middleware management consoles use Oracle Application Development Framework (Oracle ADF) security, which can integrate with Oracle Access Manager Single Sign-on (SSO). These applications can take advantage of Oracle Platform Security Services (OPSS) SSO for user authentication, but you must first configure the domain-level `jps-config.xml` file to enable these capabilities.
- Propagating the Domain to OAMHOST2
  After you start and validate the Administration Server and Managed Servers on OAMHOST1, you can then perform the following tasks on OAMHOST2.
- Starting the Managed Servers in the Domain
  Start the Managed Servers.
- Validating Access Manager
  You can validate Oracle Access Manager by using the `oamtest` tool.
- Enabling Forgotten Password
  You can set up the One Time Pin forgotten password functionality which is provided with Oracle Access Manager

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Replacing Connect Strings with the Appropriate TNS Alias
  Oracle recommends using TNS Alias in the connection strings used by FMW components instead of repeating long JDBC strings across multiple connections pools.
- Backing Up the Configuration
  It is an Oracle best practices recommendation to create a backup after you successfully extended a domain or at another logical point. Create a backup after you verify that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

About the Initial Infrastructure Domain

Before you create the initial Infrastructure domain, ensure that you review the key concepts.

- About the Infrastructure Distribution
- Characteristics of the Domain

About the Infrastructure Distribution

You create the initial Infrastructure domain for an enterprise deployment by using the Oracle Fusion Middleware Infrastructure distribution. This distribution contains both the Oracle WebLogic Server software and the Oracle JRF software.

The Oracle JRF software consists of Oracle Web Services Manager, Oracle Application Development Framework (Oracle ADF), Oracle Enterprise Manager Fusion Middleware Control, the Repository Creation Utility (RCU), and other libraries and technologies that are required to support the Oracle Fusion Middleware products.

> **Note**
>
> The Access infrastructure does not use the Web Services Manager.

See Understanding Oracle Fusion Middleware Infrastructure in *Understanding Oracle Fusion Middleware*.

Characteristics of the Domain

The following table lists some of the key characteristics of the domain that you are about to create. Reviewing these characteristics helps you to understand the purpose and context of the procedures that are used to configure the domain.

Many of these characteristics are described in more detail in Understanding a Typical Enterprise Deployment.

Characteristic of the Domain	More Information
Uses a separate virtual IP (VIP) address for the Administration Server.	Configuration of the Administration Server and Managed Servers Domain Directories
Uses separate domain directories for the Administration Server and the Managed Servers in the domain.	Configuration of the Administration Server and Managed Servers Domain Directories
Uses a per domain Node Manager configuration.	About the Node Manager Configuration in a Typical Enterprise Deployment

Copyright © 2018, 2025, Oracle and/or its affiliates.

Characteristic of the Domain	More Information
Requires a separately installed LDAP-based authentication provider.	Understanding OPSS and Requests to the Authentication and Authorization Stores

Variables Used When Creating Infrastructure for Oracle Access Management

As you perform the tasks in this chapter, you will be referencing the variables listed in this section.

The following table explains the configuration file property values required in this section.

Table 14-1 LDAP Variables Used in This Chapter

Variable	Sample Value	Description
IDSTORE_HOST	`idstore.example.com`	The host and port of your Identity Store directory. When preparing the Identity Store these should point to one of the LDAP instances. When configuring components such as OAM or OIG they should point to the load balancer entry point.
IDSTORE_PORT	`1636`	The port of your Identity Store directory. When preparing the Identity Store these should point to one of the LDAP instances. When configuring components such as OAM or OIG they should point to the load balancer entry point.
IDSTORE_DIRECTORYTYPE	`OUD`	The type of directory you are using. Valid value is OUD.
IDSTORE_BINDDN	`cn=oudadmin`	An administrative user in the Identity Store Directory,
IDSTORE_PASSWD	`password`	The the password of IDSTORE_BINDDN. You will be prompted for the value if not supplied.
IDSTORE_SEARCHBASE	`dc=example,dc=com`	The location in the directory where Users and Groups are stored.
IDSTORE_LOGINATTRIBUTE	`uid`	The LDAP attribute, which contains the users Login name.
OAM_SERVER_LOGIN_ATTRIBUTE	`uid`	The LDAP attribute, which contains the users Login name.
OAM_IDSTORE_NAME	`OAMIDSTORE`	Name of the IDStore to create.
IDSTORE_USERSEARCHBASE	`cn=Users,dc=example,dc=com`	The location in the directory where Users are Stored.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-1 (Cont.) LDAP Variables Used in This Chapter

Variable	Sample Value	Description
IDSTORE_GROUPSEARCHBASE	`cn=Groups,dc=example,dc=com`	The location in the directory where Groups are Stored.
IDSTORE_NEW_SETUP	`true`	This parameter is used when preparing a directory for the first time.
IDSTORE_SYSTEMIDBASE	`cn=SystemIDs,dc=example,dc=com`	The location of a container in the directory where system users can be placed when you do not want them in the main user container.
IDSTORE_USERNAMEATTRIBUTE	`cn`	The name of the LDAP attribute which stores a users name.
IDSTORE_KEYSTORE_FILE	`/u01/oracle/config/keystores/idmTrustStore.p12`	The location on the *SHARED_CONFIG_DIR* of the LDAP Truststore for LDAP connections.
LOCAL_KEYSTORE_FILE	`/u02/oracle/config/keystores/idmTrustStore.p12`	The location on the *LOCAL_CONFIG_DIR* of the LDAP Truststore for LDAP connections.
IDSTORE_KEYSTORE_PASSWORD	`password`	The password of the IDSTORE_KEYSTORE_FILE.

Table 14-2 OAM Variables Used in This Chapter

Variable	Sample Value	Description
IAD_ORACLE_HOME	`/u01/oracle/products/oam`	The read-only location for the OAM product binaries stored on shared disk.
IAD_ASERVER_HOME	`/u01/oracle/config/domains/oam`	The Administration Server domain home, which is installed on a shared disk.
IAD_MSERVER_HOME	`/u02/oracle/config/domains/oam`	The Managed Server domain home, which is created by using the unpack command on the local disk of each application tier host.
APPLICATION_HOME	`/u01/oracle/config/applications/oam`	The Application home directory, which is installed on shared disk, so the directory is accessible by all the application tier host computers.
MS_APPLICATION_HOME	`/u02/oracle/config/applications/oam`	The Application home directory, which is installed on local disk, so the directory is accessible by all the application tier host computers.
JAVA_HOME	`/u01/oracle/products/jdk`	The location where you install the supported Java Development Kit (JDK).

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-2 (Cont.) OAM Variables Used in This Chapter

Variable	Sample Value	Description
ADMINVHN	`iadadminvhn.example.com`	The virtual host name used as the listen address for the Administration Server used by the IAMAccessDomain and fails over with manual failover of the Administration Server. It is enabled on the node where the Administration Server process is running.
OAMHOST1	`oamhost1.example.com`	The hostname of OAMHOST1.
OAMHOST2	`oamhost2.example.com`	The hostname of OAMHOST2.
DBHOST1	`dbhost1.example.com`	The hostname of DBHOST1.
DBHOST2	`dbhost2.example.com`	The hostname of DBHOST2.
SCAN_ADDRESS	`db-scan.example.com`	Address for the Oracle RAC Database.
PRIMARY_OAM_SERVERS	`oamhost1.example.com:5575`	A comma-separated list of all of the OAM managed servers that are in the deployment. The format of this is Server Running the OAM Managed Server: OAM Proxy port. Note the proxy port used is not the OAM managed server listen port. The OAM Proxy port can be found in the worksheet (OAM_PROXY_PORT).
WEBGATE_TYPE	`ohsWebgate14c`	The type of webgate profile to create. This should always be `ohsWebgate14c`.
ACCESS_GATE_ID	`Webgate_IDM`	The name of the Webgate Agent to create.
OAM_OIM_WEBGATE_PASSWD	`password`	The password you wish to assign to the webgate agent you will be creating.
COOKIE_DOMAIN	`.example.com`	The domain you wish to associate the OAM cookie with this is normally the same as the IDSTORE_SEARCH_BASE in domain format. The search base can be found in the worksheet (*REALM_DN*).
COOKIE_EXPIRY_INTERVAL	`120`	The amount of time before a cookie is expired.
OAM_WG_DENY_ON_NOT_PROTECTED	`true`	This should always be set to `true`. It ensures that any attempt to access a resource not explicitly stated in the OAM Resource list will be rejected.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-2 (Cont.) OAM Variables Used in This Chapter

Variable	Sample Value	Description
OAM_IDM_DOMAIN_OHS_HOST	`login.example.com`	The name of the Oracle HTTP Server (OHS) server which fronts the IAMAccessDomain. In the case of an enterprise deployment this will be the load balancer name.
OAM_IDM_DOMAIN_OHS_PORT	443	The port on which the OHS server fronting the IAMAccessDomain listens. In the case of an Enterprise Deployment, this will be the load balancer port. This is the IAD_HTTPS_PORT in the worksheet.
OAM_IDM_DOMAIN_OHS_PROTOCOL	`https`	This determines which process is being used when accessing the OHS server fronting the IAMAccessDomain. In the case of an Enterprise Deployment this will be the load balancer protocol. In the Enterprise Deployment Blueprint SSL is terminated at the load balancer. But the URL will always have the HTTPS prefix, so this value should be set to `https`.
OAM_SERVER_LBR_HOST	`login.example.com`	The name of the virtual host configured on the load balancer for logging in. This is usually the same as OAM_IDM_DOMAIN_OHS_HOST.
OAM_SERVER_LBR_PORT	443	The port of the virtual host configured on the load balancer for logging in. This is usually the same as OAM_IDM_DOMAIN_OHS_PORT.
OAM_SERVER_LBR_PROTOCOL	`https`	The protocol of the virtual host configured on the load balancer for logging in. This is usually the same as OAM_IDM_DOMAIN_OHS_PROTOCOL.
OAM_OAM_SERVER_TRANSFER_MODE	`Open`	This is the type of OAM security transport to be used. This should be set to `Open`. You can specify `Cert` if extra security is required. If you wish to use cert, refer to the Oracle Access Manager documentation for how to configure this.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-2 (Cont.) OAM Variables Used in This Chapter

Variable	Sample Value	Description
OAM_TRANSFER_MODE	`open`	The type of OAM security transport to be used. This should be the same as OAM_OAM_SERVER_TRANSPORT_MODE.
OAM_OAM_SSLENABLED	`true`	Set to `true` if using End to End SSL, and `false` is using SSL Terminated.
OAM_SSO_ONLY_FLAG	`false`	This is used to determine whether authentication mode is going to be used. For Enterprise Deployments this should be set to false.
OAM_IMPERSONATION_FLAG	`false`	Determines whether OAM be configured for impersonation. Impersonation is typically used in help desk type applications where a support user "impersonates" and actual user for the purposes of providing support.
OAM_IDM_DOMAIN_LOGOUT_URLS	`/console/jsp/common/logout.jsp,/em/targetauth/emaslogout.jsp`	A list of URLs that various products can invoke for the purposes of logging out.
OAM_OIM_INTEGRATION_REQ	`true`	If you are intending Oracle Identity Governance to handle forgotten password functionality then this parameter should be set to true. If you are using the new OAM forgotten password functionality then this value should be set to `false`.
OAM_OIM_OHS_URL	`https://oig.example.com:443/`	If you are planning on using OIM for Forgotten Password functionality then you need to specify the external entry point for OIG. This is the OIG URL to which OAM directs the requests. This URL is made up of the following values from the worksheet: `https://oig.example.com:IAG_HTTPS_PORT/`.
IDSTORE_OAMADMINUSER	`oamadmin`	The user of the Admin user account you are connecting to the Identity Store with.
IDSTORE_PWD_OAMADMINUSER	`password`	The password of the Admin user account IDSTORE_OAMADMINUSER you are connecting to the Identity Store with.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-2 (Cont.) OAM Variables Used in This Chapter

Variable	Sample Value	Description
IDSTORE_OAMSOFTWAREUSER	`oamLDAP`	The user of the OAM account you are connecting to the Identity Store with.
IDSTORE_PWD_OAMSOFTWAREUSER	`password`	The password of the account IDSTORE_OAMSOFTWAREUSER you are connecting to the Identity Store with.
OAM_IDSTORE_ROLE_SECURITY_ADMIN	`OAMAdministrators`	The name of the OAM Administrators security group.
OAM_WLS_ADMIN_PASSWD	`password`	The password of the WLS Admin account (`weblogic_iam`) .
IDSTORE_WLSADMINUSER	`weblogic_iam`	The user of the WLS account you are connecting to the Identity Store with.
IDSTORE_WLSADMINGROUP	`WLSAdministrators`	The name of the group you want to create to hold your WebLogic Server administrative users.
OAM_PROXY_PORT	`5575`	The OAM proxy port of the OAM Managed Server. The OAM Proxy port can be found in the worksheet (OAM_PROXY_PORT).
IAD_HTTP_PORT	`80`	The port number of the IAD Admin hostname (`iadadmin.example.com`) This is `80` for SSL Terminated and 443 for End to End SSL.

WLS Properties

Table 14-3 WLS Variables Used in This Chapter

Variable	Sample Value	Description
WLSHOST	`iadadminvhn.example.com`	Is the Admin Server listen address. For OAM configuration, this will be `iadadminvhn.example.com`.
WLSPORT	`9002`	The Administration port of the WebLogic Server when secure mode is enabled. This should always be 9002 for enterprise deployments. This is the IAD_WLS_PORT in the worksheet.
WLSADMIN	`weblogic`	The user used to connect to the Admin Server.
WLSPASSWD	`password`	The password for the weblogic admin user.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-3 (Cont.) WLS Variables Used in This Chapter

Variable	Sample Value	Description
WLS_IS_SSLENABLED	`true`	This flag is used to specify if the AdminServer is running in secure mode (SSL) or non-secure mode.
WLS_SSL_HOST_VERIFICATION	`true`	Whether to perform hostname verification with SSL certificates.
WLS_TRUSTSTORE	`/u01/oracle/config/keystores/idmTrustStore.p12`	The location of the WLS Truststore on shared storage.
LOCAL_WLS_TRUSTSTORE	`/u02/oracle/config/keystores/idmTrustStore.p12`	The location of the WLS Truststore on local storage.
WLS_TRUSTSTORE_PASSWORD	`password`	The password of the WLS_TRUSTSTORE.

OIG Properties

Table 14-4 OIG Variables Used in This Chapter

Variable	Sample Value	Description
IDSTORE_OIMADMINGROUP	`OIMAdministrators`	The name of the group you want to create to hold your Oracle Identity Governance administrative users.
IDSTORE_OIMADMINUSER	`oimLDAP`	The user that Oracle Identity Governance uses to connect to the Identity store.
IDSTORE_PWD_OIMADMINUSER	`password`	The password of IDSTORE_OIMADMINUSER. If there is no value, you will be prompted for it.

Setting Environment Variables

Set environment variables used in this chapter.

To help navigate this guide and so you are able to copy sample commands without modification, you can set the following environment variables replacing the values with values appropriate to your environment.

```
export IAD_ORACLE_HOME=/u01/oracle/products/oam
export ORACLE_HOME=$IAD_ORACLE_HOME
export ORACLE_COMMON_HOME=$IAD_ORACLE_HOME/oracle_common
export JAVA_HOME=/u01/oracle/products/jdk
export PATH=$JAVA_HOME/bin:$PATH
export IAD_ASERVER_HOME=/u01/oracle/config/domains/oam
export IAD_MSERVER_HOME=/u02/oracle/config/domains/oam
export NM_HOME=/u02/oracle/config/nodemanager
export APPLICATION_HOME=/u01/oracle/config/applications/oam
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
export MS_APPLICATION_HOME=/u02/oracle/config/applications/oam
export DB_HOST=db-scan.example.com
export DB_PORT=1521
export DB_SERVICE=oamsvc.example.com
export KEYSTORE_HOME=/u01/oracle/config/keystores
export LOCAL_KEYSTORE_HOME=/u02/oracle/config/keystores
export SHARED_CONFIG_DIR=/u01/oracle/config
export LOCAL_CONFIG_DIR=/u02/oracle/config
```

URLs Used in This Chapter

This section describes the URLs used in this chapter.

Table 14-5 SSL Terminated

Function	Component URL	Load Balancer URL
Remote Console connection	`http://iadadminvhn.example.com:7001/`	`http://iadadmin.example.com`
Enterprise Manager	`http://iadadminvhn.example.com:7001/em`	`http://iadadmin.example.com/em`
OAM Console	`http://iadadminvhn.example.com:7001/oamconsole`	`http://iadadmin.example.com/oamconsole`

Table 14-6 End to End SSL

Function	Component URL	Load Balancer URL
Remote Console connection	`https://iadadminvhn.example.com:9002/`	`https://iadadmin.example.com`
Enterprise Manager	`https://iadadminvhn.example.com:9002/em`	`https://iadadmin.example.com/em`
OAM Console	`https://iadadminvhn.example.com:9002/oamconsole`	`https://iadadmin.example.com/oamconsole`

Installing the Oracle Fusion Middleware

Install the Oracle Fusion Middleware software in preparation for configuring a new domain for Oracle Access Management.

- Installing a Supported JDK
- Installing the Oracle Fusion Middleware Infrastructure
  Use the following sections to install the Oracle Fusion Middleware Infrastructure software in preparation for configuring a new domain for Oracle Access Management.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Installing Oracle Access Management for an Enterprise Deployment
  The procedure for installing Oracle Access Management in an enterprise deployment domain is explained in this section.

Installing a Supported JDK

Oracle Fusion Middleware requires that a certified Java Development Kit (JDK) is installed on your system.

- Locating and Downloading the JDK Software
- Installing the JDK Software
  Oracle Fusion Middleware requires you to install a certified Java Development Kit (JDK) on your system.

Locating and Downloading the JDK Software

To find a certified JDK, see the certification document for your release on the Oracle Fusion Middleware Supported System Configurations page.

After you identify the Oracle JDK for the current Oracle Fusion Middleware release, you can download an Oracle JDK from the following location on Oracle Technology Network:

```
https://www.oracle.com/java/technologies/downloads/
```

Be sure to navigate to the download for the Java SE JDK.

Installing the JDK Software

Oracle Fusion Middleware requires you to install a certified Java Development Kit (JDK) on your system.

For more information about the recommended location for the JDK software, see Understanding the Recommended Directory Structure for an Enterprise Deployment.

To install JDK 21.0:

1. Change directory to the location where you downloaded the JDK archive file.

   ```
   cd download_dir
   ```

2. Unpack the archive into the JDK home directory, and then run the following commands:

   ```
   tar -xzvf jdk-21.0.4+8_linux-x64_bin.tar.gz
   ```

   Note that the JDK version listed here was accurate at the time this document was published. For the latest supported JDK, see the *Oracle Fusion Middleware System Requirements and Specifications* for the current Oracle Fusion Middleware release.

3. Move the JDK directory to the recommended location in the directory structure.

   For example:

   ```
   ln -s jdk-21.0.4 /u01/oracle/products/jdk
   ```

   See File System and Directory Variables Used in This Guide.

4. Run the following command to verify that the appropriate `java` executable is in the path and your environment variables are set correctly:

   ```
   java -version
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

The Java version in the output should be displayed as “21.0.4”.

Installing the Oracle Fusion Middleware Infrastructure

Use the following sections to install the Oracle Fusion Middleware Infrastructure software in preparation for configuring a new domain for Oracle Access Management.

- Starting the Oracle Identity Management Quick Installer
- Navigating the Installation Screens
- Installing Oracle Fusion Middleware Infrastructure on the Other Host Computers
- Checking the Directory Structure

Starting the Oracle Identity Management Quick Installer

To start the installation program, perform the following steps.

1. Log into OAMHOST1.
2. Go to the directory where you downloaded the installation program.
3. Launch the installation program by invoking the `java` executable from the JDK directory on your system, as shown in the example below.

   ```
   $JAVA_HOME/bin/java -jar fmw_14.1.2.0.0_infrastructure_generic.jar
   ```

   In this example:

   - If you download the distribution from the Oracle Technology Network (OTN), then the JAR file is typically packaged inside a downloadable ZIP file.

     To install the software required for the initial Infrastructure domain, the distribution you want to install is:

     fmw_14.1.2.0.0_infrastructure_generic.jar.

     For more information about the actual file names of each distribution, see Identifying and Obtaining Software Downloads for an Enterprise Deployment.

When the installation program appears, you are ready to begin the installation. See Navigating the Installation Screens for a description of each installation program screen.

Navigating the Installation Screens

The installation program displays a series of screens, in the order listed in the following table.

If you need additional help with any of the installation screens, click the screen name or click the **Help** button on the screen.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-7 Navigating the Infrastructure Installation Screens

Screen	Description
Installation Inventory Setup	On UNIX operating systems, this screen appears if you are installing any Oracle product on this host for the first time. Specify the location where you want to create your central inventory. Ensure that the operating system group name selected on this screen has write permissions to the central inventory location. See Understanding the Oracle Central Inventory in *Installing Software with the Oracle Universal Installer*. **Note** Oracle recommends that you configure the central inventory directory on the products shared volume. Example: `/u01/oracle/products/oraInventory` You may also need to execute the `createCentralinventory.sh` script as root from the `oraInventory` folder after the installer completes.
Welcome	This screen introduces you to the product installer.
Auto Updates	Use this screen to search My Oracle Support automatically for available patches or automatically search a local directory for patches that you have already downloaded for your organization.
Installation Location	Use this screen to specify the location of your Oracle home directory. For the purposes of an enterprise deployment, enter the value of the $ORACLE_HOME variable for the product listed in Table 8-2. For example, `/u01/oracle/products/oam`
Prerequisite Checks	This screen verifies that your system meets the minimum requirements. If there are any warning or error messages, refer to the Oracle Fusion Middleware System Requirements and Specifications document on the Oracle Technology Network (OTN).
Installation Summary	Use this screen to verify the installation options that you have selected. If you want to save these options to a response file, click **Save Response File** and provide the location and name of the response file. Response files can be used later in a silent installation situation. For more information about silent or command-line installation, see Using the Oracle Universal Installer in Silent Mode in *Installing Software with the Oracle Universal Installer*.
Installation Progress	This screen allows you to see the progress of the installation.
Installation Complete	This screen appears when the installation is complete. Review the information on this screen, then click **Finish** to dismiss the installer.

Installing Oracle Fusion Middleware Infrastructure on the Other Host Computers

If you have configured a separate shared storage volume or partition for secondary hosts, then you must install the Infrastructure on one of those hosts.

See Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment.

Copyright © 2018, 2025, Oracle and/or its affiliates.

To install the software on the other host computers in the topology, log in to each host, and use the instructions in Starting the Oracle Identity Management Quick Installer and Navigating the Installation Screens to create the Oracle home on the appropriate storage device.

You must install the Stack Bundle Patch and any other mandatory patches outlined in Identifying and Obtaining Software Distributions for an Enterprise Deployment.

Checking the Directory Structure

After you install the Oracle Identity and Access Management and create the Oracle home, you should see the directory and sub-directories listed in this topic. The contents of your installation vary based on the options that you selected during the installation.

To check the directory structure:

1. Navigate to the $ORACLE_HOME:

   ```
   cd $ORACLE_HOME
   ```

2. Enter the following command:

   ```
   ls --format=single-column
   ```

   The directory structure on your system must match the structure shown in the following example:

   ```
   bin
   cfgtoollogs
   coherence
   domain-registry.xml
   em
   envPropertiesCache
   idm
   install
   inventory
   jdeveloper
   jlib
   lib
   mft
   OPatch
   opmn
   oracle_common
   oraInst.loc
   osb
   oui
   root.sh
   soa
   wlserver
   ```

   See What are the Key Oracle Fusion Middleware Directories? in *Understanding Oracle Fusion Middleware*.

Installing Oracle Access Management for an Enterprise Deployment

The procedure for installing Oracle Access Management in an enterprise deployment domain is explained in this section.

This section contains the following procedures.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Starting the Oracle Identity and Access Management Installation Program
- Navigating the Installation Screens
- Installing the Stack Bundle Patch
- Installing Oracle Access Management on the Other Host Computers
- Verifying the Installation

Starting the Oracle Identity and Access Management Installation Program

To start the installation program:

1. Log in to OAMHOST1.
2. Go to the directory where you downloaded the installation program.
3. Launch the installation program by invoking the `java` executable from the JDK directory on your system, as shown in the example below.

   ```
   $JAVA_HOME/bin/java -jar fmw_14.1.2.1.0_idm_generic.jar
   ```

When the installation program appears, you are ready to begin the installation.

Navigating the Installation Screens

The installation program displays a series of screens, in the order listed in the following table.

If you need additional help with any of the installation screens, click the screen name.

Screen	Description
Installation Inventory Screen	If you did not create a central inventory when you installed the Oracle Fusion Middleware Infrastructure software, then this dialog box appears. Edit the **Inventory Directory** field so it points to the location of your local inventory, and then click **OK**.
Welcome	This screen introduces you to the product installer.
Auto Updates	Use this screen to automatically search My Oracle Support for available patches or automatically search a local directory for patches that you've already downloaded for your organization.
Installation Location	Use this screen to specify the location of your Oracle home directory. For Oracle Identity and Access Management, this should be set to *IAD_ORACLE_HOME*. For example, `/u01/oracle/products/oam`. For more information about Oracle Fusion Middleware directory structure, see "Selecting Directories for Installation and Configuration" in *Planning an Installation of Oracle Fusion Middleware*.
Installation Type	Use this screen to choose the type of installation you wish to deploy. • Select Collocated Oracle Identity and Access Manager (Managed through WebLogic Server):
Prerequisite Checks	This screen verifies that your system meets the minimum necessary requirements. If there are any warning or error messages, you can refer to one of the documents in the Roadmap for Verifying Your System Environment section in *Planning Your Oracle Fusion Middleware Infrastructure Installation.*
Installation Summary	Use this screen to verify the installation options you selected. Click **Install** to begin the installation.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Screen	Description
Installation Progress	This screen allows you to see the progress of the installation. Click **Next** when the progress bar reaches 100% complete.
Installation Complete	Review the information on this screen, then click **Finish** to dismiss the installer.

Installing the Stack Bundle Patch

After installing the software binaries, you must apply the latest Stack Bundle Patch.

For an enterprise deployment you must download and apply the July 2025 Stack Bundle Patch or later. For more details, see Identifying and Obtaining Software Distributions for an Enterprise Deployment.

To apply the patch run the following commands:

1. After downloading the patch, unzip it to your preferred location. For example:

   ```
   unzip p38184742_141210_Linux-x86-64.zip
   ```

   This location will be known as $PATCH_DIR.
2. Navigate to the $PATCH_DIR:

   ```
   cd $PATCH_DIR/tools/spbat/generic/SPBAT/
   ```

3. Apply the patch using the command:

   ```
   ./spbat.sh -type oam -phase downtime -mw_home $IAD_ORACLE_HOME -
   spb_download_dir $PATCH_DIR
   ```

Installing Oracle Access Management on the Other Host Computers

If you have followed the Enterprise Deployment Guide shared storage recommendations, there is a separate shared storage volume for product installations on OAMHOST2, and you must also install the software on OAMHOST2. See Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment.

You must install the Stack Bundle Patch and any other mandatory patches outlined in Identifying and Obtaining Software Distributions for an Enterprise Deployment.

Verifying the Installation

After you complete the installation, you can verify it by successfully completing the following tasks.

- Reviewing the Installation Log Files
- Checking the Directory Structure
- Viewing the Contents of Your Oracle Home

Copyright © 2018, 2025, Oracle and/or its affiliates.

Reviewing the Installation Log Files

Review the contents of the installation log files to make sure that no problems were encountered. For a description of the log files and where to find them, see Understanding Installation Log Files in *Installing Software with the Oracle Universal Installer*.

Checking the Directory Structure

The contents of your installation vary based on the options you selected during the installation.

The addition of Oracle Identity and Access Management will add the following directory and sub-directories:

```
$IAD_ORACLE_HOME/
OPatch
cfgtoollogs
coherence
em
idm
inventory
oraInst.loc
oracle_common
oui
wlserver

idm/
clone
common
connectors
designconsole
idmdiag
idmtools
jlib
libovd
mbeans
modules
oam
oic
opam-connectors
plugins
remote_manager
schema
server
upgrade
```

For more information about the directory structure you should see after installation, see "What are the Key Oracle Fusion Middleware Directories?" in *Understanding Oracle Fusion Middleware*.

Viewing the Contents of Your Oracle Home

You can also view the contents of your Oracle home by using the `viewInventory` script. See Viewing the contents of an Oracle home in *Installing Software with the Oracle Universal Installer*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Configuring LDAP

If you haven't already done so, you need to configure your LDAP directory.

To do this follow the steps in Preparing an Existing LDAP Directory.

Creating the Database Schemas for Access Manager

Oracle Fusion Middleware components require the existence of schemas in a database before you configure a Fusion Middleware Infrastructure domain for Oracle Access Management.

Install the schemas listed in this topic in a certified database for use with this release of Oracle Fusion Middleware.

The tool creates the following schemas:

- Metadata Services (MDS)
- Audit Services (IAU)
- Audit Services Append (IAU_APPEND)
- Audit Services Viewer (IAU_VIEWER)
- Oracle Platform Security Services (OPSS)
- User Messaging Service (UMS)
- WebLogic Services (WLS)
- Common Infrastructure Services (STB)
- Oracle Access Manager (OAM)

For more information about RCU and how the schemas are created and stored in the database, see Preparing for Schema Creation in *Creating Schemas with the Repository Creation Utility*.

- Installing and Configuring a Certified Database
- Creating the Database Schemas Using GUI
- Creating the Database Schemas Using CLI

Installing and Configuring a Certified Database

Make sure that you have installed and configured a certified database, and that the database is up and running.

See the Preparing the Database for an Enterprise Deployment.

Creating the Database Schemas Using GUI

- Starting the Repository Creation Utility (RCU)
- Navigating the RCU Screens to Create the Schemas

Starting the Repository Creation Utility (RCU)

To start the Repository Creation Utility (RCU):

Copyright © 2018, 2025, Oracle and/or its affiliates.

1. Navigate to the following directory on OAMHOST1:

   For Example

   ```
   $IAD_ORACLE_HOME/oracle_common/bin
   ```

2. Start RCU:

   ```
   ./rcu
   ```

> **Note**
>
> If your database has Transparent Data Encryption (TDE) enabled, and you want to encrypt your tablespaces created by the RCU, provide the `-encryptTablespace true` option when you start the RCU.
>
> This will default the appropriate RCU GUI Encrypt Tablespace checkbox selection on the Map Tablespaces screen without further effort during the RCU execution. See Encrypting Tablespaces in *Creating Schemas with the Repository Creation Utility*.

Navigating the RCU Screens to Create the Schemas

The RCU installation program displays a series of screens, in the order listed in the following table.

Screen	Description
Introducing RCU	Review the Welcome screen and verify the version number for RCU. Click **Next** to begin.
Selecting a Method of Schema Creation	If you have the necessary permission and privileges to perform DBA activities on your database, select **System Load and Product Load** on the Create Repository screen. The procedure in this document assumes that you have the necessary privileges. If you do not have the necessary permission or privileges to perform DBA activities in the database, you must select **Prepare Scripts for System Load** on this screen. This option will generate a SQL script, which can be provided to your database administrator. See Understanding System Load and Product Load in *Creating Schemas with the Repository Creation Utility*. Click **Next**. **Tip** For more information about the options on this screen, see Create repository in *Creating Schemas with the Repository Creation Utility*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Screen	Description
Providing Database Connection Details	Provide the database connection details for RCU to connect to your database. 1. As **Database Type**, select **Oracle Database** enabled for edition based redefinition. 2. In the **Host Name** field, enter the SCAN address of the Oracle RAC Database. 3. Enter the **Port** number of the RAC database scan listener, for example 1521. 4. Enter the RAC **Service Name** of the database. 5. Enter the **User Name** of a user that has permissions to create schemas and schema objects, for example SYS. 6. Enter the **Password** of the user name that you provided in step 5 7. If you have selected the SYS user, ensure that you set the role to SYSDBA. 8. Click **Next** to proceed, then click **OK** on the dialog window confirming that connection to the database was successful. **Tip** For more information about the options on this screen, see Database Connection Details in *Creating Schemas with the Repository Creation Utility*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Screen	Description
Specifying a Custom Prefix and Selecting Schemas	1. Specify the custom prefix you want to use to identify the Oracle Fusion Middleware schemas. The custom prefix is used to logically group these schemas together for use in this domain. For Oracle Access Management, use the prefix `IAD`. **Tip** Make a note of the custom prefix you choose to enter here; you will need this later, during the domain creation process. For more information about custom prefixes, see Understanding Custom Prefixes in *Creating Schemas with the Repository Creation Utility*. 2. Select the following schemas from the list of components: • **AS Common Schemas** When you select **AS Common Schemas**, all of the schemas in this section are automatically selected. If the schemas in this section are not automatically selected, then select the required schemas. – Metadata Services (MDS) – Audit Services (IAU) – Audit Services Append (IAU_APPEND) – Audit Services Viewer (IAU_VIEWER) – Oracle Platform Security Services (OPSS) – User Messaging Service (UMS) – WebLogic Services (WLS) – Common Infrastructure Services (STB) • Expand the group **IDM Schemas**, and then select the **Oracle Access Manager** schema. There are two mandatory schemas that are selected by default. You cannot deselect them: **Common Infrastructure Services** (the STB schema) and **WebLogic Services** (the WLS schema). The **Common Infrastructure Services** schema enables you to retrieve information from RCU during domain configuration. See Understanding the Service Table Schema in *Creating Schemas with the Repository Creation Utility*. **Tip** For more information about how to organize your schemas in a multi-domain environment, see Planning Your Schema Creation in *Creating Schemas with the Repository Creation Utility*. Click **Next** to proceed, then click **OK** on the dialog window confirming that prerequisite checking for schema creation was successful.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Screen	Description
Specifying Schema Passwords	Specify how you want to set the schema passwords on your database, then specify and confirm your passwords. Ensure that the complexity of the passwords meet the database security requirements before you continue. RCU will proceed at this point even if you do not meet the password polices. Hence, perform this check outside RCU itself.. Click **Next**. **Tip** You must make a note of the passwords you set on this screen; you will need them later on during the domain creation process.
Verifying the Tablespaces for the Required Schemas	You can accept the default settings on the remaining screens, or you can customize how RCU creates and uses the required tablespaces for the Oracle Fusion Middleware schemas. **Note** You can configure a Fusion Middleware component to use JDBC stores for JMS servers and Transaction Logs, by using the Configuration Wizard. These JDBC stores are placed in the WebLogic Services component tablespace. If your environment expects to have a high level of transactions and/or JMS activity, you can increase the default size of the <*PREFIX*>_WLS tablespace to better suit the environment load. Click **Next** to continue, and then click **OK** on the dialog window to confirm the tablespace creation. For more information about RCU and its features and concepts, see About the Repository Creation Utility in *Creating Schemas with the Repository Creation Utility*.
Creating Schemas	Review the summary of the schemas to be loaded and click **Create** to complete schema creation. **Note** If failures occurred, review the listed log files to identify the root cause, resolve the defects, and then use RCU to drop and re-create the schemas before you continue.
Reviewing Completion Summary and Completing RCU Execution	When you reach the Completion Summary screen, verify that all schema creations have been completed successfully, and then click **Close** to dismiss RCU.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Creating the Database Schemas Using CLI

Run the following commands to create the database schemas using the Repository Creation Assistant CLI in silent mode:

1. Create a password file `pwd.txt` that contains the the password for the database sysdba account and password assigned for the database schemas. This file should contain two lines as shown in the following example:

   ```
   sysdba_password
   schema_password
   ```

2. Run the following command to execute the RCU in silent mode:

   ```
   $ORACLE_COMMON_HOME/oracle_common/bin/rcu -silent -createRepository -
   databaseType ORACLE -connectString $DB_HOST:$DB_PORT/$DB_SERVICE -dbUser
   sys -dbRole sysdba -selectDependentsForComponents true -
   useSamePasswordForAllSchemaUsers true -schemaPrefix $RCU_PREFIX -component
   MDS -component IAU -component IAU_APPEND -component IAU_VIEWER -component
   OPSS -component WLS -component STB -component OAM -f < /pwd.txt
   ```

Verifying Schema Access

Verify schema access by connecting to the database as the new schema users are created by the RCU. Use SQL*Plus or another utility to connect, and provide the appropriate schema names and passwords entered in the RCU.

For example:

```
sqlplus <RCU_PREFIX>_OAM/<PASSWORD>@//<SCAN_ADDRESS>:<PORT>/<SERVICE_NAME>
```

For example:

```
sqlplus IADEDG_OAM/<password>@//db-scan.example.com:1521/oampdb_s.example.com
```

The output appears as follows:

```
SQL*Plus: Release 23.0.0.0.0 - for Oracle Cloud and Engineered Systems on Wed
Sep 11 14:20:00 2024 Version 23.5.0.24.07
Copyright (c) 1982, 2024, Oracle. All rights reserved.


Connected to:
Oracle Database 23ai EE Extreme Perf Release 23.0.0.0.0 - for Oracle Cloud
and Engineered Systems
Version 23.5.0.24.07

SQL>
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

Configuring the Oracle Access Management Domain

The following topics provide instructions for creating an Oracle Access Management domain using the Fusion Middleware Configuration wizard.

For more information on other methods available for domain creation, see Additional Tools for Creating, Extending, and Managing WebLogic Domains in *Creating WebLogic Domains Using the Configuration Wizard*.

- Starting the Configuration Wizard
- Navigating the Configuration Wizard Screens to Configure Oracle Access Management Domain

Starting the Configuration Wizard

To begin domain configuration, run the following command in the Oracle Fusion Middleware Oracle home.

```
$ORACLE_COMMON_HOME/common/bin/config.sh
```

Navigating the Configuration Wizard Screens to Configure Oracle Access Management Domain

Follow the instructions in the following sections to create and configure the domain for the topology with static clusters.

> **Note**
>
> Oracle Access Management does not support Dynamic Clusters.

Table 14-8 Navigating the Infrastructure Installation Screens

Screen	Description
Selecting the Domain Type and Domain Home Location	On the Configuration Type screen, select **Create a new domain**. In the **Domain Location** field, specify the value of the *IAD_ASERVER_HOME* variable, as defined in File System and Directory Variables Used in This Guide. For example, `/u01/oracle/products/oam` for Oracle Access Manager. **Tip** More information about the other options on this screen of the Configuration Wizard, see Configuration Type in *Creating WebLogic Domains Using the Configuration Wizard*. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-8 Navigating the Infrastructure Installation Screens

Screen	Description
Selecting the Configuration Templates	On the Templates screen, make sure **Create Domain Using Product Templates** is selected, then select the following templates: • **Oracle Access Management Suite - [idm]** • Selecting this template automatically selects the following dependencies: – Oracle Enterprise Manager - [em] – Oracle JRF - [oracle_common] – WebLogic Coherence Cluster Extension - [wlserver] **Tip** More information about the options on this screen can be found in Templates in *Creating WebLogic Domains Using the Configuration Wizard*. Click **Next**.
Selecting the Application Home Location	On the Application Location screen, specify the value of the *APPLICATION_HOME* variable, as defined in File System and Directory Variables Used in This Guide. For example, `/u01/oracle/config/applications/oam`. **Tip** More information about the options on this screen can be found in Application Location in *Creating WebLogic Domains Using the Configuration Wizard*. Click **Next**.
Configuring the Administrator Account	On the Administrator Account screen, specify the user name (Oracle recommends using a different name from “WebLogic”) and password for the default WebLogic Administrator account for the domain. Make a note of the user name and password specified on this screen; you will need these credentials later to boot and connect to the domain's Administration Server. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-8 Navigating the Infrastructure Installation Screens

Screen	Description
Specifying the Domain Mode and JDK	On the Domain Mode and JDK screen: • Select **Production** in the Domain Mode field. In the Enable or Disable Default Ports for your Domain field, use the default values provided for Production Mode: **SSL Termination** If you are using an SSL Terminated deployment, the following values must be selected: – Ensure that Enable Listen Ports (non-SSL Ports) is checked. – Ensure that Enable SSL Listen Ports is not checked. – Ensure that Enable Administration Port (SSL Port) is checked. **End to End SSL Deployment** In the Enable or Disable Default Ports for your Domain field, use the following default values provided for Production Mode: – Ensure that Enable Listen Ports (non-SSL Ports) is not checked. – Ensure that Enable SSL Listen Ports is checked. – Ensure that Enable Administration Port (SSL Port) is checked **Tip** More information about the options on this screen, including the differences between development mode and production mode can be found in Domain Mode and JDK in *Creating WebLogic Domains Using the Configuration Wizard.* When you start the Administration Server, a boot identity file can be created to bypass the need to provide a username and password in the production mode. • Select the **Oracle Hotspot** JDK in the JDK field. Selecting **Production Mode** on this screen gives your environment a higher degree of security, requiring a user name and password to deploy applications and to start the Administration Server. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-8 Navigating the Infrastructure Installation Screens

Screen	Description
Specifying the Database Configuration Type	On the Database Configuration Type screen: • Select **RCU Data** to activate the fields on this screen. The **RCU Data** option instructs the Configuration Wizard to connect to the database and Service Table (STB) schema to automatically retrieve schema information for the schemas needed to configure the domain. • Verify that **Vendor** is `Oracle` and **Driver** is `*Oracle's Driver (Thin) for Service Connections; Versions: Any`. • Verify that **Connection Parameters** is selected. **Note** If you choose to select **Manual Configuration** on this screen, you will have to manually fill in the parameters for your schema on the JDBC Component Schema screen. After you select **RCU Data**, fill in the fields as shown below:

Field	Description
Host Name	Enter the Single Client Access Name (SC Oracle RAC database, which you entered *Deployment Workbook*. For information about the Enterprise Dep Using the Enterprise Deployment Workbo
DBMS/Service	Enter the service name for the Oracle RA this domain where you will install the proc example: `iamedg.example.com` Specify the service name based on the va the Preparing the Database for an Enterp
Port	Enter the port number on which the datab `1521`.
Schema Owner	Enter the user name and password for co Service Table schema.
Schema Password	This is the schema user name and passw the Service Table component on the "Sch RCU (see Creating the Database Schema The default user name is *prefix*_STB, custom prefix that you defined in RCU.

Click **Get RCU Configuration** when you are finished specifying the database connection information. The following output in the Connection Result Log indicates that the operating succeeded:

```
Connecting to the database server...OK
Retrieving schema data from database server...OK
Binding local schema components with retrieved data...OK

Successfully Done.
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-8 Navigating the Infrastructure Installation Screens

Screen	Description
	Click **Next** if the connection to the database is successful. **Tip** More information about the **RCU Data** option can be found in Understanding the Service Table Schema in *Creating Schemas with the Repository Creation Utility*. More information about the other options on this screen can be found in Datasource Defaults in *Creating WebLogic Domains Using the Configuration Wizard*.
Specifying JDBC Component Schema Information	Verify that the values on the JDBC Component Schema screen are correct for all schemas. The schema table should be populated, because you selected **Get RCU Data** on the previous screen. As a result, the Configuration Wizard locates the database connection values for all the schemas required for this domain. At this point, the values are configured to connect to a single-instance database. However, for an enterprise deployment, you should use a highly available Real Application Clusters (RAC) database, as described in Preparing the Database for an Enterprise Deployment. In addition, Oracle recommends that you use an Active GridLink datasource for each of the component schemas. For more information about the advantages of using GridLink data sources to connect to a RAC database, see Database Considerations in the *High Availability Guide*. To convert the data sources to GridLink: 1. Select all the schemas by selecting the checkbox at in the first header row of the schema table. 2. Click **Convert to GridLink** and click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-8 Navigating the Infrastructure Installation Screens

<table>
<tr><th>Screen</th><th>Description</th></tr>
<tr><td>Providing the GridLink Oracle RAC Database Connection Details</td><td>On the GridLink Oracle RAC Component Schema screen, provide the information required to connect to the RAC database and component schemas, as shown below:

<table>
<tr><th>Element</th><th>Description and Recommended Value</th></tr>
<tr><td>Service Name</td><td>Verify that the service name for the Oracl appropriate.
For example: <code>iadedg.example.com</code>.</td></tr>
<tr><td>SCAN, Host Name, and Port</td><td>Select the SCAN check box.
In the Host Name field, enter the Single C (SCAN) Address for the Oracle RAC data
In the Port field, enter the SCAN listening example, 1521)</td></tr>
<tr><td>ONS Host and Port</td><td>Leave blank.</td></tr>
<tr><td>Enable Fan</td><td>Verify that the Enable Fan check box is s can receive and process FAN events.</td></tr>
</table>
For more information about specifying the information on this screen, as well as information about how to identify the correct SCAN address, see Configuring Active GridLink Data Sources with Oracle RAC in the High Availability Guide.
You can also click Help to display a brief description of each field on the screen.
Click Next.</td></tr>
<tr><td>Testing the JDBC Connections</td><td>Use the JDBC Component Schema Test screen to test the data source connections you have just configured.
A green check mark in the Status column indicates a successful test. If you encounter any issues, see the error message in the Connection Result Log section of the screen, fix the problem, then try to test the connection again.

Tip
More information about the other options on this screen can be found in Test Component Schema in Creating WebLogic Domains Using the Configuration Wizard

Click Next.</td></tr>
</table>

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-8 Navigating the Infrastructure Installation Screens

Screen	Description
Selecting Advanced Configuration	To complete domain configuration for the topology, select the following options on the Advanced Configuration screen: • **Administration Server** This is required to properly configure the listen address of the Administration Server. • **Node Manager** This is required to configure Node Manager. • **Topology** This is required to add, delete, or modify the Settings for Server Templates, Managed Servers, Clusters, Virtual Targets, and Coherence. **Note** When using the Advanced Configuration screen in the Configuration Wizard: • If any of the above options are not available on the screen, then return to the Templates screen, and be sure you selected the required templates for this topology. • Do not select the **Domain Frontend Host Capture** advanced configuration option. You will later configure the frontend host property for specific clusters, rather than for the domain. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-8 Navigating the Infrastructure Installation Screens

Screen	Description
Configuring the Administration Server Listen Address	On the Administration Server screen: 1. In the **Server Name** field, retain the default value - AdminServer. 2. In the **Listen Address** field, enter the virtual host name that corresponds to the VIP of the IADADMINVHN that you procured in Procuring Resources for an Enterprise Deployment and enabled in Preparing the Host Computers for an Enterprise Deployment. For more information on the reasons for using the IADADMINVHN virtual host, see Reserving the Required IP Addresses for an Enterprise Deployment. 3. In the Configure Administration Server Ports section, perform the following steps: **SSL Terminated Deployments** a. Leave the **Enable Listen Port** field checked. The Listen Port value will be 7001. b. Ensure the **Enable SSL Listen port** field is unchecked c. The Listen Port value will be disabled in grey. d. Leave the default value as 9002 in the Administration Port. **End to End SSL Deployments** a. Leave the **Enable Listen Port** field unchecked. The Listen Port value will be disabled in grey. b. Ensure the **Enable SSL Listen port** field is checked. c. Leave the default value as 7002 in the **SSL Listen Port** field. d. Leave the default value as 9002 in the Administration Port. 4. Leave the default value as Unspecified in the Server Group. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-8 Navigating the Infrastructure Installation Screens

Screen	Description
Configuring Node Manager	Select **Manual Node Manager Setup** as the Node Manager type. **Warning** You can ignore the warning in the bottom pane. This guide provides the required steps for the Manual Node Manager configuration. • Username: This is the user name used to connect to the Node Manager. For example, `admin`. • Password and Confirm Password: Enter the password you wish to associate with the Node Manager username. **Tip** For more information about the options on this screen, see Node Manager in *Creating WebLogic Domains Using the Configuration Wizard*. For more information about per domain and per host Node Manager implementations, see About the Node Manager Configuration in a Typical Enterprise Deployment. For information about Node Manager configurations, see Configuring Node Manager on Multiple Machines in *Administering Node Manager for Oracle WebLogic Server*. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-8 Navigating the Infrastructure Installation Screens

Screen	Description
Configuring Managed Servers	On the Managed Servers screen, a new Managed Server for Oracle Access Management appears in the list of servers. Perform the following tasks to modify the default Oracle Access Management Managed Server and create a second Managed Server: 1. Click **Add** to create a new Managed Server and name it `oam_server2`. **Tip** The server names recommended here will be used throughout this document; if you choose different names, be sure to replace them as needed. 2. Use the information below to fill in the rest of the columns for each Oracle Access Manager Server. **Table 14-9 SSL Terminated** **Table 14-10 End to End SSL**

Table 14-9 SSL Terminated

Server Name	Listen Address	Listen Port	Enable SSL	SSL Listen Port	Server Groups
oam_server1	OAMHOST1	14100	Unchecked	Disabled	OAM-MGD-SVR
oam_server2	OAMHOST2	14100	Unchecked	Disabled	OAM-MGD-SVR
oam_policy_mgr1	OAMHOST1	14150	Unchecked	Disabled	OAM-POLICY-M
oam_policy_mgr2	OAMHOST2	14150	Unchecked	Disabled	OAM-POLICY-M

Table 14-10 End to End SSL

Server Name	Listen Address	Listen Port	Enable SSL	SSL Listen Port	Server Groups
oam_server1	OAMHOST1	Disabled	Checked	14101	OAM-MGD-SVR
oam_server2	OAMHOST2	Disabled	Checked	14101	OAM-MGD-SVR
oam_policy_mgr1	OAMHOST1	Disabled	Checked	14151	OAM-POLICY-M
oam_policy_mgr2	OAMHOST2	Disabled	Checked	14151	OAM-POLICY-M

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-8 Navigating the Infrastructure Installation Screens

Screen	Description
Configuring a Cluster	In this task, you create clusters of Managed Servers to which you can target the Oracle Access Manager software. You must create the following clusters: **Table 14-11 SSL Terminated** **Table 14-12 End to End SSL** Use the Clusters screen to create a new cluster: 1. Click the **Add** button. 2. Specify `OAM_Cluster` in the **Cluster Name** field. 3. From the **Dynamic Server Groups** drop-down list, select `Unspecified`. 4. Specify `login.example.com` for the **Frontend Host** field. 5. Specify 443 for the **Frontend HTTPS Port** field. **Note** By default, server instances in a cluster communicate with one another using unicast. If you want to change your cluster communications to use multicast, refer to Considerations for Choosing Unicast or Multicast in *Administering Clusters for Oracle WebLogic Server*. **Tip** More information about the options on this screen can be found in Clusters in *Creating WebLogic Domains Using the Configuration Wizard*. 6. Repeat the steps to create the second cluster `POLICY_Cluster`. 7. Click **Next**.

Table 14-11 SSL Terminated

Cluster	Frontend Host	Frontend HTTP Port	Fronte
OAM_Cluster	login.example.com		443
POLICY_Cluster	iadadmin.example.com	80	

Table 14-12 End to End SSL

Cluster	Frontend Host	Frontend HTTP Port	Frontend HTTPS Port
OAM_Cluster	login.example.com		443
POLICY_Cluster	iadadmin.example.com		443

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-8 Navigating the Infrastructure Installation Screens

Screen	Description
	Tips For more information about the options on this screen, see Clusters in *Creating WebLogic Domains Using the Configuration Wizard*. Click **Next**.
Assigning Server Templates	Click **Next** to proceed to the next screen.
Configuring Dynamic Servers	Verify that all dynamic server options are disabled for clusters that are to remain as static clusters. 1. Confirm that the **Dynamic Cluster**, **Calculated Listen Port**, and **Calculated Machine Names** checkboxes on this screen are unchecked. 2. Confirm the **Server Template** selection is **Unspecified**. 3. Click **Next**.
Assigning Managed Servers to the Cluster	Use the Assign Servers to Clusters screen to assign your managed servers to the clusters you have just created. At the end of this you will have the following assignments: **Cluster** / **Managed Servers** OAM_Cluster: oam_server1, oam_server2 POLICY_Cluster: oam_policy_mgr1, oam_policy_mgr2 1. In the Clusters pane, select the cluster to which you want to assign the servers. 2. In the Servers pane, assign the managed servers to the clusters as in the table above, using one of the following methods: • Click once on the Managed Server to select it, then click on the right arrow to move it beneath the selected cluster in the Clusters pane. • Double-click on managed server to move it beneath the selected cluster in the clusters pane. 3. Repeat to assign each managed server to a cluster as shown in the table. 4. Click **Next** to proceed to the next screen. **Tip** More information about the options on this screen can be found in Assign Servers to Clusters in *Creating WebLogic Domains Using the Configuration Wizard*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-8 Navigating the Infrastructure Installation Screens

Screen	Description
Configuring Coherence Clusters	Use the Coherence Clusters screen to configure the Coherence cluster that is automatically added to the domain. In the **Cluster Listen Port**, enter *9991*. **Note** For Coherence licensing information, Oracle Coherence Products in *Oracle Fusion Middleware Licensing Information User Manual*. Click **Next**.
Creating Machines for Oracle Access Management Servers	Use the Machines screen to create new machines in the domain. A machine is required in order for the Node Manager to be able to start and stop the servers. Select the Unix Machine tab. 1. Click the **Add** button to create new UNIX machines. 2. Use the values in the table below to define the Name and Node Manager Listen Address of each machine. 3. Verify the port in the **Node Manager Listen Port** field. The port number 5556, shown in this example, may be referenced by other examples in the documentation. Replace this port number with your own port number, as needed. **Table 14-13** **Tip** More information about the options on this screen can be found in Machines in *Creating WebLogic Domains Using the Configuration Wizard*.

Table 14-13

Name	Node Manager Listen Address	No
ADMINHOST	Enter the value of the IADADMINVHN variable. For example: `iadadminvhn.example.com`	55
OAMHOST1	The value of the OAMHOST1 host name variable. For example, `oamhost1.example.com`.	55
OAMHOST2	The value of the OAMHOST2 host name variable. For example, `oamhost2.example.com`.	55

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-8 Navigating the Infrastructure Installation Screens

Screen	Description
Assigning Servers to Machines	Use the Assign Servers to Machines screen to assign the Oracle Access Manager Managed Servers you just created to the corresponding machines in the domain. You can assign the machines as follows: **Servers** / **Machines** AdminServer / ADMINHOST oam_policy_mgr1, oam_server1 / OAMHOST1 oam_policy_mgr2, oam_server2 / OAMHOST2 **Tip** More information about the options on this screen can be found in Assign Servers to Machines in *Creating WebLogic Domains Using the Configuration Wizard*. Click **Next**.
Creating Virtual Targets	Click **Next**.
Creating Partitions	Click **Next**.
Reviewing Your Configuration Specifications and Configuring the Domain	The Configuration Summary screen contains the detailed configuration information for the domain you are about to create. Review the details of each item on the screen and verify that the information is correct. You can go back to any previous screen if you need to make any changes, either by using the **Back** button or by selecting the screen in the navigation pane. Domain creation will not begin until you click **Create**. **Tip** More information about the options on this screen can be found in Configuration Summary in *Creating WebLogic Domains Using the Configuration Wizard*. Click **Next**.
Writing Down Your Domain Home and Administration Server URL	The Configuration Success screen will show the following items about the domain you just configured: • Domain Location • Administration Server URL You must make a note of both items as you will need them later; the domain location is needed to access the scripts used to start the Administration Server. Click **Finish** to dismiss the Configuration Wizard.

Enabling SSL

If you are configuring End to End SSL, you must perform additional steps.

Copyright © 2018, 2025, Oracle and/or its affiliates.

The steps are as follows:

- Adding Certificate Stores Location to the WebLogic Servers Start Scripts
- Update Server's Security Settings Using the Remote Console

Adding Certificate Stores Location to the WebLogic Servers Start Scripts

Once the Identity and Trust Stores are created for the domain some Java properties must be added to the WebLogic start scripts. These properties are added to the file `setUserOverridesLate.sh` in `$IAD_ASERVER_HOME/bin`. Any customizations you add to this file are preserved during domain upgrade operations and are carried over to remote servers when using the pack and unpack commands.

Manually create the file `setUserOverridesLate.sh` in `$IAD_ASERVER_HOME/bin`. Edit the file and add the variable `EXTRA_JAVA_PROPERTIES` to set the `javax.net.ssl.trustStore` and `javax.net.ssl.trustStorePassword` properties with the values used by your EDG system. For example:

```
EXTRA_JAVA_PROPERTIES="${EXTRA_JAVA_PROPERTIES}
 -Djavax.net.ssl.trustStore=/u01/oracle/config/keystores/idmTruststore.p12
 -Djavax.net.ssl.trustStorePassword=password"
export EXTRA_JAVA_PROPERTIES
```

The order of the extra java properties is relevant. In case that the same property is defined more than once, the later value is used. The custom values must be defined as in the example provided.

Update Server's Security Settings Using the Remote Console

- Connecting to the Remote Console Using the Administration Server’s Virtual Hostname as Provider
- Updating the WebLogic Servers Security Settings

Connecting to the Remote Console Using the Administration Server’s Virtual Hostname as Provider

The following procedure temporarily starts the Administration Server with the default start script so to enable you to perform these tasks. After you perform these tasks, you can stop this temporary session and use the Node Manager to start the Administration Server.

> **Note**
>
> For this Remote Console initial access to the Administration Server, it is required that the machine that runs the Remote Console can resolve and connect to the Admin Server's Listen Address. This can be done by starting the Remote Console directly in the node where the Admin Server runs or creating a tunnel to this address from the node where the remote Console is executed.

1. Using the following default start script to start the Administration Server:

Copyright © 2018, 2025, Oracle and/or its affiliates.

a. Change directory to the following directory:

```
cd $IAD_ASERVER_HOME/bin
```

b. Run the start script:

```
./startWebLogic.sh
```

Monitor the terminal till the following message is displayed:

```
<Server state changed to RUNNING>
```

Also you must verify that the appropriate SSL listener is available, which can be confirmed with the a message like the following displayed in output:

```
<Server> <BEA-002613> <Channel "DefaultSecure" is now listening on
XXXX:7002 for protocols iiops, t3s, ldaps, https.>
```

2. Create a new provider in the WebLogic Remote Console as follows:

a. Download the domain's trust keystore to the host or laptop where you run the WebLogic Remote Console. For example, when using the per-domain CA steps in previous sections, this would be located at `$KEYSTORE_HOME/idmTruststore.p12`.

b. Open the Remote Console and add the domain trust store to the remote console settings. Click **File** > **Settings** and enter the following values.

i. Trust Store type - pkcs12

ii. Trust Store Path - The path to the trust keystore file in the host where the Remote Console runs.

iii. Trust Store Key - Enter the password provided in the steps above for certificate creation.

iv. Check **Disable HostName verification** if you are using Demo certificates as described in the steps above.

c. Using the Providers window in the Remote Console, create a new provider by selecting **Add Admin Server Connection Provider**.

i. In the provider name, enter a name for the connection, for example `oam`.

ii. Enter the WebLogic Domain Administration username provided in the configuration wizard user name.

iii. Enter the password used for the domain creation.

iv. Use https protocol and the admin server listen address used in the configuration wizard as URL for access and specify port 9002.

For example, `https://iadadminvhn.example.com:9002`.

v. Check the **Make Insecure Connection** checkbox.

> **Note**
>
> This provider should not be used once the front end and webtier are configured.

Copyright © 2018, 2025, Oracle and/or its affiliates.

The Remote Console Home Window for the domain will be displayed.

Updating the WebLogic Servers Security Settings

Perform the following steps to update the WebLogic Servers Security Settings and Administration Port:

1. Access the Domain provider in the Remote Console and update the Administration Server and WebLogic Servers Security Settings:
   a. Click **Edit Tree**.
   b. Click **Environment** > **Servers** > **AdminServer**.
   c. Click **Security** tab.
   d. Change the keystores dropdown to **Custom Identity and Custom Trust**.
   e. In **Custom Identity Keystore**, enter the fully qualified path to the identity keystore as follows:

      `$KEYSTORE_HOME/idmcerts.p12`
      Replace `$KEYSTORE_HOME` with the value of the folder you use for storing keystore, as described in the Table 8-2.
   f. Set the **Custom Identity Keystore Type** to PKCS12.
   g. In **Custom Identity Keystore Passphrase**, enter the password `Keystore_Password` you provided in the certificate generation steps.
   h. In **Custom Trust Keystore**, enter the fully qualified path to the trust keystore.

      `$KEYSTORE_HOME/idmTrustStore.p12`
      Replace `$KEYSTORE_HOME` with the value of the folder you use for storing keystore, as described in the Table 8-2.
   i. Set the **Custom Trust Keystore Type** to PKCS12.
   j. In **Custom Trust Keystore Passphrase**, enter the password you provided as the <**keypass**> in the certificate generation steps.
   k. Click **Save**.
   l. Under **Security** settings, navigate to **SSL** tab.
   m. In the **Server Private Key Alias** filed enter the alias provided in the certificate generation steps. If you are using a SAN based certificate use the alias you used to create the certificate. If you are using host based certificates this is usually the hosts short name. For example `oamhost1` for `oamhost1.example.com`.
   n. In the **Server Private Key Pass Phrase** field, enter the password provided in the certificate generation steps.
   o. Click **Save**.

      The cart on the top right part of the screen will show **full** with a yellow bag inside.
   p. Click the Cart icon on the top right and select **Commit Changes**.

   Repeat the above steps for each managed server in the domain changing the alias to match the alias used for the certificates.

Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> For managed servers you should use $LOCAL_KEYSTORE_HOME instead of $KEYSTORE_HOME.

2. Return to the terminal window where you started the Administration Server with the start script.
3. Press **Ctrl+C** to stop the Administration Server process.

   Wait for the Administration Server process to end and for the terminal command prompt to appear.
4. Start the Administration Server again by using the following script:
   a. Change directory to the following directory:

      ```
      cd $IAD_ASERVER_HOME/bin
      ```

   b. Run the start script:

      ```
      ./startWebLogic.sh
      ```

   c. Monitor the output in the terminal till the following output is displayed.

      ```
      <Server state changed to RUNNING>
      ```

Configuring a Per Host Node Manager for an Enterprise Deployment

For specific enterprise deployments, Oracle recommends that you configure a per-host Node Manager, as opposed to the default per-domain Node Manager.

For more information about the advantages of a per host Node Manager, see *About the Node Manager Configuration in a Typical Enterprise Deployment*.

- Creating a Per Host Node Manager Configuration
- Starting the Node Manager on OAMHOST1
- Configuring the Node Manager Credentials
- Enrolling the Domain with NM
- Adding Truststore Configuration to Node Manager

Creating a Per Host Node Manager Configuration

The step in configuring a per-host Node Manager is to create a configuration directory and two new node manager configuration files. You must also edit the default `startNodeManager.sh` file.

To create a per-host Node Manager configuration, perform the following tasks, first on OAMHOST1, and then on OAMHOST2:

1. Log in to OAMHOST1 and create a directory for the Node Manager configuration files :

   For example:

Copyright © 2018, 2025, Oracle and/or its affiliates.

```
mkdir -p /u02/oracle/config/nodemanager
```

Note that this directory should be on a local disk, because it is specific to the host. This directory location is known as the Node Manager home, and it is identified by the *NM_HOME* directory variable in examples in this guide.

2. If you haven't already, copy the keystores to your local host to ensure that node manager has access to them. For example:

```
mkdir -p $LOCAL_KEYSTORE_HOME
```

```
cp -r $KEYSTORE_HOME $LOCAL_KEYSTORE_HOME
```

3. Change directory to the Node Manager home directory:

```
cd $NM_HOME
```

4. Create a new text file called `nodemanager.properties` and add the values shown in Example: Contents of the nodemanager.properties File to this new file.

   Use the pertaining identity alias for the node that you are configuring. For example, `oamhost1.example.com` in OAMHOST1 and `oamhost2.example.com` in OAMHOST2.

   For more information about the properties that you can add to the `nodemanager.properties` file, see Node Manager Properties in *Administering Node Manager for Oracle WebLogic Server*.

   In the `nodemanager.properties` file, you enable crash recovery for servers as a part of this configuration. See Node Manager and System Crash Recovery in *Administering Node Manager for Oracle WebLogic Server*.

   Example: Contents of the nodemanager.properties File

```
DomainsFile=/u02/oracle/config/nodemanager/nodemanager.domains
LogLimit=0
PropertiesVersion=14.1.2.0.0
AuthenticationEnabled=true
NodeManagerHome=/u02/oracle/config/nodemanager
#Include the specific JDK home
JavaHome=/u01/oracle/products/jdk
LogLevel=INFO
DomainsFileEnabled=true
StartScriptName=startWebLogic.sh
#Leave blank for listening on ANY
ListenAddress=
NativeVersionEnabled=true
ListenPort=5556
LogToStderr=true
SecureListener=true
LogCount=1
StopScriptEnabled=false
QuitEnabled=false
LogAppend=true
StateCheckInterval=500
CrashRecoveryEnabled=true
StartScriptEnabled=true
LogFile=/u02/oracle/config/nodemanager/nodemanager.log
LogFormatter=weblogic.nodemanager.server.LogFormatter
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
ListenBacklog=50
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityAlias=idmcerts.example.com
CustomIdentityKeyStoreFileName=/u02/oracle/config/keystores/
idmTrustStore.pkcs12
CustomIdentityKeyStorePassPhrase=password
CustomIdentityPrivateKeyPassPhrase=password
```

5. Locate the `startNodeManager.sh` file in the following directory:

   ```
   $WL_HOME/server/bin
   ```

6. Copy the `startNodeManager.sh` file to the Node Manager home directory.

   ```
   cp $WL_HOME/server/bin/startNodeManager.sh $NM_HOME
   ```

7. Edit the new `startNodeManager.sh` file and add the `NODEMGR_HOME` property as follows:

   ```
   NODEMGR_HOME="<NM_HOME>"
   ```

   For example:

   ```
   NODEMGR_HOME="/u02/oracle/config/nodemanager"
   ```

8. Locate the `stopNodeManager.sh` script in the `$WL_HOME/server/bin` directory. Copy it to the Node Manager home directory. Edit the copied file and edit the `NODEMGR_HOME` property pointing to the node manager home (as it has been done for the `startNodemanager.sh` file):

   ```
   NODEMGR_HOME="<NM_HOME>"
   ```

   In this example, replace *<NM_HOME>* with the actual path to the Node Manager home.

9. Create another new file in the Node Manager home directory, called `nodemanager.domains`.

   The `nodemanager.domains` file provides additional security by restricting Node Manager client access to the domains listed in this file.

10. Perform steps 1 through 8 on OAMHOST2.

11. Add the following entries to the new `nodemanager.domains` files:

    On OAMHOST1, add values for both the Administration Server domain home and the Managed Servers domain home:

    ```
    oam=IAD_MSERVER_HOME;IAD_ASERVER_HOME
    ```

    > **Note**
    >
    > The path that is mentioned first (*IAD_MSERVER_HOME*) is considered as the primaryDomainPath and Managed Servers are run from this location.

    On OAMHOST2, add the value for the Managed Servers domain home only:

    ```
    oam=IAD_MSERVER_HOME
    ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

In these examples, replace *IAD_ASERVER_HOME* and *IAD_MSERVER_HOME* with the values of the respective variables, as described in File System and Directory Variables Used in This Guide.

Starting the Node Manager on OAMHOST1

After you manually set up the Node Manager to use a per-host Node Manager configuration, you can start the Node Manager on OAMHOST1, by using the `startNodeManager.sh` script.

To start the Node Manager on OAMHOST1:

1. Change directory to the Node Manager home directory:

   ```
   cd $NM_HOME
   ```

2. Run the following command to start the Node Manager and send the output of the command to an output file, rather than to the current terminal shell:

   ```
   nohup ./startNodeManager.sh > ./nodemanager.out 2>&1 &
   ```

3. Monitor the the `nodemanager.out` file; make sure the NodeManager starts successfully. The output should eventually contain the following strings:

   ```
   <INFO> <Upgrade> <Encrypting NodeManager property:
   CustomIdentityKeyStorePassPhrase>
   <INFO> <Upgrade> <Encrypting NodeManager property:
   CustomIdentityPrivateKeyPassPhrase>
   <Upgrade> <Saving upgraded NodeManager properties to '/u02/oracle/config/
   nodemanager/nodemanager.properties'>
   <INFO> <Loading domains file: /u02/oracle/config/nodemanager/
   nodemanager.domains>
   <INFO> <Loading identity key store: FileName=/u02/oracle/config/keystores/
   idmTrustStore.p12, Type=pkcs12, PassPhraseUsed=true>
   <INFO> <Loaded NodeManager configuration properties from '/u02/oracle/
   config/nodemanager/nodemanager.properties'>
   <INFO> <14.1.2.0.0>
   <INFO> <Server Implementation Class:
   weblogic.nodemanager.server.NMServer$ClassicServer.>
   <INFO> <Secure socket listener started on port 5556>
   ```

   You must check that the plain text used for passwords in `nodemanager.properties` has now been encrypted by running the following command:

   ```
   cat /u02/oracle/config/nodemanager/nodemanager.properties
   ```

   The output will look similar to the following:

   ```
   #<DATE>
   #<DATE>
   DomainsFile=/u02/oracle/config/nodemanager/nodemanager.domains
   LogLimit=0
   PropertiesVersion=14.1.2.0.0
   AuthenticationEnabled=true
   NodeManagerHome=/u02/oracle/config/nodemanager
   #Include the specific JDK home
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
JavaHome=/u01/oracle/products/jdk
LogLevel=INFO
DomainsFileEnabled=true
StartScriptName=startWebLogic.sh
#Leave blank for listening on ANY
ListenAddress=
NativeVersionEnabled=true
ListenPort=5556
LogToStderr=true
SecureListener=true
LogCount=1
StopScriptEnabled=false
QuitEnabled=false
LogAppend=true
StateCheckInterval=500
CrashRecoveryEnabled=true
StartScriptEnabled=true
LogFile=/u02/oracle/config/nodemanager/nodemanager.log
LogFormatter=weblogic.nodemanager.server.LogFormatter
ListenBacklog=50
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityAlias=idmcerts.example.com
CustomIdentityKeyStoreFileName=/u02/oracle/config/keystores/
idmTrustStore.p12
CustomIdentityKeyStorePassPhrase={AES256}EMvPrOCRfN7fyv3d8JcEnttTLyneG9Su+U
VK5DGEmqmqDwLkpLz9nQFZ+fL1Bidc
CustomIdentityPrivateKeyPassPhrase={AES256}O5cEJD8WVYP3aRLp9KAbFZ3CGLyxmmIW
FX1YzVfJvPpl1dc5RbMksAcsBLquKcWW
```

Configuring the Node Manager Credentials

Perform the following steps to set the Node Manager credentials using the Remote Console:

1. Access the Domain provider in the Remote Console.
2. Click **Edit Tree**.
3. Click **Environment** > **Domain**> **Security**.
4. Check the **Show Advanced Fields** field.
5. Set **Node Manager Username** to the same as the Weblogic Administrator, since this username will be used in other tasks mentioned in this guide.
6. Change the NM password. Ensure the **Node Manager password** is set to the same as the Weblogic Administrator since this password will be used in other tasks mentioned in this guide.
7. Click **Save**. The cart on the top right part of the screen will show **full** with a yellow bag inside.
8. Click the Cart Icon on the top right and select **Commit Changes**.

Enrolling the Domain with NM

Perform the following steps in a new terminal window to enroll the domain with Node manager.

Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> You will be unable to connect to the Node Manager and use it to start the servers in the domain without performing this step.

1. Change directory to the following directory:

   ```
   cd $ORACLE_COMMON_HOME/common/bin
   ```

2. Start the WebLogic Server Scripting Tool (WLST). In order to use the certificates created for the appropriate SSL handshake, the location of the stores and password of the same need to be provided to WLST. Use the following command or add these in a script that can be easily invoked:

   ```
   export WLST_PROPERTIES="-Dweblogic.security.TrustKeyStore=CustomTrust -
   Dweblogic.security.CustomTrustKeyStoreFileName=/u02/oracle/config/
   keystores/idmTrustStore.p12 -
   Dweblogic.security.CustomTrustKeyStorePassPhrase=password -
   Dweblogic.security.SSL.ignoreHostnameVerification=true"
   ```

   > **Note**
   >
   > You must avoid including the password in the script.

3. Connect to the Administration Server by using the following WLST command:

   ```
   connect('admin_user','admin_password','admin_url')
   ```

   For example:

   ```
   connect('weblogic','<password>','t3s://iadadminvhn.example.com:9002')
   ```

4. Use the `nmEnroll` command to enable the Node Manager to manage servers in a specified WebLogic domain.

   ```
   nmEnroll('<IAD_ASERVER_HOME>')
   ```

   For example:

   ```
   nmEnroll('/u01/oracle/config/domains/oam')
   ```

5. Generate startup properties for the Admin Server using the following WLST command:

   ```
   nmGenBootStartupProps('AdminServer')
   ```

   The `startup.properties` and `boot.properties` files are created in the following directory:

   ```
   <IAD_ASERVER_HOME>/servers/AdminServer/data/nodemanager/
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

Adding Truststore Configuration to Node Manager

It is required to add the corresponding truststore configuration for Node Manager communication with the different WebLogic Server listeners. To do this, edit Node Manager's start script `startNodeManager.sh` located at `$NM_HOME` and add the variable JAVA_OPTIONS to set the `javax.net.ssl.trustStore` and `javax.net.ssl.trustStorePassword` properties with the values used by your EDG system. For example:

```
export JAVA_OPTIONS="${JAVA_OPTIONS} -Djavax.net.ssl.trustStore=/u02/oracle/
config/keystores/idmTrustStore.pkcs12 -
Djavax.net.ssl.trustStorePassword=mypassword"
```

Configuring the Domain Directories and Starting the Servers

After you have configured the domain and configured the Node Manager, you can start the Administration Server by using the Node Manager. In an enterprise deployment, the Node Manager is used to start and stop the Administration Server and all the Managed Servers in the domain.

- Create a ServerOverrides File
  Perform prerequisite tasks before starting the servers.
- Starting the Administration Server Using the Node Manager
  After you have configured the domain and configured the Node Manager, you can start the Administration Server by using the Node Manager. In an enterprise deployment, the Node Manager is used to start and stop the Administration Server and all the Managed Servers in the domain.
- Validating the Administration Server
  Before proceeding with the configuration steps, validate that the Administration Server has started successfully by making sure you have access to the Oracle Enterprise Manager Fusion Middleware Control, which is installed and configured on the Administration Servers.
- Creating a Separate Domain Directory for Managed Servers
  When you initially create the domain for enterprise deployment, the domain directory resides on a shared disk. This default domain directory will be used to run the Administration Server. You can now create a copy of the domain on the local storage for each of your managed server hosts. The domain directory on the local (or private) storage will be used to run the Managed Servers.

Create a ServerOverrides File

Perform prerequisite tasks before starting the servers.

This involves:

- Disabling the Derby Database - Disable the embedded Derby database, which is a file-based database, packaged with Oracle WebLogic Server. The Derby database is used primarily for development environments. Therefore, you must disable it when you are configuring a production-ready enterprise deployment environment. Otherwise, the Derby database process starts automatically when you start the Managed servers.
- Enabling IPv6 Networking if required - If the Managed Server is configured to use IPv6 networking, then you may encounter issues when you start the Managed Server.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Adjusting the Memory Parameters for your installation - The initial startup parameter in the IAMAccessDomain, which defines the memory usage, is insufficient. You must increase the value of this parameter and set the Java initial memory allocation pool (Xms) to 1024m, and the maximum memory allocation pool (Xmx) to 8192m.

In order to perform the above tasks, create a `$IAD_ASERVER_HOME/bin/setUserOverrides.sh` file with the following contents:

```
DERBY_FLAG=false
JAVA_OPTIONS="${JAVA_OPTIONS} -Djava.net.preferIPv4Stack=true"
MEM_ARGS="-Xms4096m -Xmx8192m"
```

Starting the Administration Server Using the Node Manager

After you have configured the domain and configured the Node Manager, you can start the Administration Server by using the Node Manager. In an enterprise deployment, the Node Manager is used to start and stop the Administration Server and all the Managed Servers in the domain.

To start the Administration Server by using the Node Manager:

1. Ensure that the Administration Server is stopped.
2. If your domain is SSL enabled then set the following environment variable so that your keystores are used:

   ```
   export WLST_PROPERTIES="-Dweblogic.security.TrustKeyStore=CustomTrust -
   Dweblogic.security.CustomTrustKeyStoreFileName=/u02/oracle/config/
   keystores/idmTrustStore.p12 -
   Dweblogic.security.CustomTrustKeyStorePassPhrase=password -
   Dweblogic.security.SSL.ignoreHostnameVerification=true"
   ```

3. Start the WebLogic Scripting Tool (WLST):

   ```
   cd $IAD_ORACLE_HOME/oracle_common/common/bin

   ./wlst.sh
   ```

4. Connect to Node Manager by using the Node Manager credentials:

   ```
   nmConnect('nodemanager_username','nodemanager_password','iadadminvhn.exampl
   e.com','5556','domain_name',
               '<IAD_SERVER_HOME>','SSL')
   ```

   For example:

   ```
   nmConnect('admin','password','iadadminvhn.example.com','5556','oam',
               '/u01/oracle/config/domains/oam','SSL')
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> This user name and password are used only to authenticate connections between Node Manager and clients. They are independent of the server administrator ID and password and are stored in the `nm_password.properties` file located in the following directory:
>
> `$IAD_ASERVER_HOME/config/nodemanager`

5. Start the Administration Server:

```
nmStart('AdminServer')
```

6. Exit WLST:

```
exit()
```

Validating the Administration Server

Before proceeding with the configuration steps, validate that the Administration Server has started successfully by making sure you have access to the Oracle Enterprise Manager Fusion Middleware Control, which is installed and configured on the Administration Servers.

To navigate to Fusion Middleware Control use the URL in URLs Used in This Chapter. Log in with the Oracle WebLogic Server administrator credentials.

You should be also able to connect to the Administration Server from the WebLogic Remote Console as before.

Creating a Separate Domain Directory for Managed Servers

When you initially create the domain for enterprise deployment, the domain directory resides on a shared disk. This default domain directory will be used to run the Administration Server. You can now create a copy of the domain on the local storage for each of your managed server hosts. The domain directory on the local (or private) storage will be used to run the Managed Servers.

> **Note**
>
> If you are creating a domain for Oracle Access Management, it is not necessary to perform this step at this time. This is because, at the time of infrastructure creation, there are no managed servers in existence yet.

Placing the *IAD_MSERVER_HOME* on local storage is recommended to eliminate the potential contention and overhead cause by servers writing logs to shared storage. It is also faster to load classes and jars need from the domain directory, so any temporary or cache data that Managed Servers use from the domain directory is processed quicker.

As described in Preparing the File System for an Enterprise Deployment, the path to the Administration Server domain home is represented by the *IAD_ASERVER_HOME* variable, and the path to the Managed Server domain home is represented by the *IAD_MSERVER_HOME* variable.

To create the Managed Server domain directory:

Copyright © 2018, 2025, Oracle and/or its affiliates.

1. Sign in to the host running the Administration Server, for example, OAMHOST1, and run the `pack` command to create a template as follows:

   ```
   cd $ORACLE_COMMON_HOME/common/bin
   ```

   ```
   ./pack.sh -managed=true \
   -domain=$IAD_ASERVER_HOME \
   -template=$SHARED_CONFIG_DIR/domains/oamdomaintemplate.jar \
   -template_name=oam_domain_template \
   -log_priority=DEBUG \
   -log=/tmp/pack.log
   ```

   In this example:

   - Replace $IAD_ASERVER_HOME with the actual path to the domain directory you created on the shared storage device.
   - Replace *full_path* with the complete path to the location where you want to create the domain template jar file. You will need to reference this location when you copy or unpack the domain template jar file. It is recommended to choose a shared volume other than ORACLE_HOME, or write to `/tmp/` and copy the files manually between servers.

     You must specify a full path for the template jar file as part of the `-template` argument to the pack command:

     ```
     $SHARED_CONFIG_DIR/domains/template_filename.jar
     ```

   - `oamdomaintemplate.jar` is a sample name for the jar file you are creating, which will contain the domain configuration files.
   - `oam_domain_template` is the label assigned to the template data stored in the template file.

2. Make a note of the location of the `oamdomaintemplate.jar` file you just created with the pack command.

   > **Tip**
   >
   > For more information about the pack and unpack commands, see Overview of the Pack and Unpack Commands in *Creating Templates and Domains Using the Pack and Unpack Commands*.

3. If you haven't already, create the recommended directory structure for the Managed Server domain on the OAMHOST1 local storage device.

   Use the examples in File System and Directory Variables Used in This Guide as a guide.

4. Run the `unpack` command to unpack the template in the domain directory onto the local storage, as follows:

   ```
   cd $ORACLE_COMMON_HOME/common/bin
   ```

   ```
   ./unpack.sh -domain=$IAD_MSERVER_HOME \
   -overwrite_domain=true \
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
-template=$SHARED_CONFIG_DIR/domains/oamdomaintemplate.jar \
-log_priority=DEBUG \
-log=/tmp/unpack.log \
-app_dir=$MS_APPLICATION_HOME
```

> **Note**
>
> The `-overwrite_domain` option in the unpack command allows unpacking a managed server template into an existing domain and existing applications directories. For any file that is overwritten, a backup copy of the original is created. If any modifications had been applied to the start scripts and ear files in the managed server domain directory, they must be restored after this unpack operation.
>
> Additionally, to customize server startup parameters that apply to all servers in a domain, you can create a file called `setUserOverridesLate.sh` and configure it to, for example, add custom libraries to the WebLogic Server classpath, specify additional JAVA command line options for running the servers, or specify additional environment variables. Any customizations you add to this file are preserved during domain upgrade operations, and are carried over to remote servers when using the pack and unpack commands.

In this example:

- `$IAD_MSERVER_HOME` is the complete path to the domain home to be created on the local storage disk. This is the location where the copy of the domain will be unpacked.
- `$SHARED_CONFIG_DIR/domains/oamdomaintemplate.jar` is the complete path and file name of the domain template jar file that you created when you ran the pack command to pack up the domain on the shared storage device.
- `$MS_APPLICATION_HOME` is the complete path to the Application directory for the domain on shared storage. See File System and Directory Variables Used in This Guide.

> **Tip**
>
> For more information about the pack and unpack commands, see Overview of the Pack and Unpack Commands in *Creating Templates and Domains Using the Pack and Unpack Commands*.

5. Change directory to the newly created Managed Server directory and verify that the domain configuration files were copied to the correct location on the OAMHOST1 local storage device.

Removing OAM Server from WebLogic Server defaultCoherenceCluster

You must exclude all Oracle Access Management (OAM) clusters (including policy manager and OAM runtime server) from the default WebLogic Server coherence cluster using the WebLogic Remote Console.

OAM server-side session management uses database and does not require coherence cluster to be established. In some environments, warnings and errors are observed due to default

Copyright © 2018, 2025, Oracle and/or its affiliates.

coherence cluster initialized by WebLogic. To avoid or fix these errors, exclude all of the OAM clusters from default WebLogic Server coherence cluster using the following steps:

1. In WebLogic Remote Console, click **Edit Tree**.
2. In the left pane of the console, expand **Environment**, select **Coherence Clusters**, and click **defaultCoherenceCluster**.

   The **defaultCoherenceCluster** page is displayed.
3. Select the **Members** tab.
4. Remove Any Items from the Chosen box.
5. Click **Save**.
6. Click the **shopping cart** and select **Commit changes**.

Tuning the WebLogic Server

Tune the WebLogic Server for optimum performance by adding the Minimum Thread Constraint and removing the Max Thread and Capacity constraints.

Create A Deployment Plan

1. Log into the WebLogic Remote Console.
2. Select **Monitoring Tree**.
3. Navigate to **Deployments** > **Application Management** > **oam_server**
4. Click **Create Plan** and provide the following plan path if it is not already defaulted:

   `$ORACLE_HOME/idm/oam/server/apps/Plan.xml`
5. Click **Done**.
6. Select **Deployment Tasks** to verify that the plan was successfully created.

Update Deployment Plan

7. Log into the WebLogic Remote Console.
8. Select **Monitoring Tree**.
9. Navigate to **Deployments** > **Application Management** > **oam_server** > **Deployment Plan (Advanced)**.
10. Select the **Variable Assignments** tab.
11. Select **Find** from the **Edit** menu and search for `/weblogic-web-app/work-manager/[name="wm/OAPOverRestWM"]/min-threads-constraint/[name="MinThreadsCount"]/count`.
12. Select the checkbox in the row and click **Edit**.
13. Set the following fields:

    Value: `400`

    Operation: `Replace`
14. Click **Done**.

Deploy Plan

15. Log into the WebLogic Remote Console.
16. Select **Monitoring Tree**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

17. Navigate to **Deployments** > **Application Management** .
18. Check the checkbox in the `oam_server` row.
19. Click **Update/Redeploy**.
20. Select the third option **Redeploy - Deployment Source and Plan on Server**.
21. Click **Done**.
22. Click **Deployment Tasks** to monitor the status of the redeploy.

Verify Change

23. To verify if the value is updated, log into the Enterprise Manager console using the URL referenced in URLs Used in This Chapter.
24. Select **System MBean Browser** and search for MBean `wm/OAPOverRestWM`.
25. Expand `MinThreadsConstraintRuntime` and then expand `MinThreadsCount`.
26. Check that the value of Count is 400.

Adding a Load Balancer Certificate to the Oracle Keystore Service

The Oracle Access Manager forgot password functionality requires that the SSL certificate used by the load balancer be added to the Oracle Keystore Service Trusted Certificates.

To add the certificate, do the following:

1. Create a directory to hold user created keystores and certificates. For example:

   ```
   mkdir -p $SHARED_CONFIG_DIR/keystores
   ```

   For example:

   ```
   mkdir -p /u01/oracle/config/keystores
   ```

2. Obtain the certificate from the load balancer. You can obtain the load balancer certificate using a browser, such as Firefox. However, the easiest way to obtain the certificate is to use the **openssl** command. The syntax of the command is as follows:

   ```
   openssl s_client -connect <LOADBALANCER>:<PORT> -showcerts </dev/null
   2>/dev/null| sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p'
   > $SHARED_CONFIG_DIR/keystores/<LOADBALANCER>.pem
   ```

   For example:

   ```
   openssl s_client -connect <LOADBALANCER>:<PORT> -showcerts </dev/null
   2>/dev/null| sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p'
   > $SHARED_CONFIG_DIR/keystores/login.example.com.pem
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> This command saves the certificate to a file called `login.example.com.pem` in `$SHARED_CONFIG_DIR/keystores`.

3. Load the certificate into the Oracle Keystore Service using WLST. Connect to WLST by using the following command:

```
$IAD_ORACLE_HOME/oracle_common/common/bin/wlst.sh
```

> **Note**
>
> If using End to End SSL then specify the location of your certificates first:
>
> ```
> export WLST_PROPERTIES="-
> Dweblogic.security.TrustKeyStore=CustomTrust \
> -Dweblogic.security.CustomTrustKeyStoreFileName=/u02/oracle/config/
> keystores/idmTrustStore.p12 \
> -Dweblogic.security.CustomTrustKeyStorePassPhrase=password \
> -Dweblogic.security.SSL.ignoreHostnameVerification=true"
> ```

4. Inside the WLST shell, connect to the Administration Server using the following command:

```
connect('<OAM_WEBLOGIC_USER>','<OAM_WEBLOGIC_PWD>','t3://
iadadminvhn.example.com:<OAM_ADMIN_PORT>')
```

For example, for SSL Terminated:

```
connect('weblogic','<password>','t3://iadadminvhn.example.coml:7001')
```

For example, for End to End SSL:

```
connect('weblogic','<password>','t3s://iadadminvhn.example.coml:7002')
```

5. Download the access artifacts by using the following command:

```
downloadAccessArtifacts(domain_home="/u01/oracle/config/domains/OAM",
propsFile="/u01/oracle/config/db.props"
```

> **Note**
>
> For information about the contents of the properties file, see Doc ID 2318818.1 on My Oracle Support.

6. Load the certificate using the following commands:

```
svc = getOpssService(name='KeyStoreService')
svc.importKeyStoreCertificate(appStripe='system',name='trust',password='',
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
keypassword='',alias='<CertificateName>',type='TrustedCertificate',
filepath='/<SHARED_CONFIG_DIR>/keystores/<LOADBALANCER>.pem')
```

For example:

```
svc = getOpssService(name='KeyStoreService')
svc.importKeyStoreCertificate(appStripe='system',name='trust',password='',
keypassword='',alias='login.example.com',type='TrustedCertificate',
filepath='/u01/oracle/config/keystores/login.example.com.pem')
```

7. Synchronize the keystore service with the file system by using the following command:

```
syncKeyStores(appStripe='system', keystoreFormat='KSS')
```

8. Save the access artifacts by using the following command:

```
saveAccessArtifacts(domain_home="/u01/oracle/config/domains/OAM",
propsFile="/u01/oracle/config/db.props"
```

9. Exit the WLST shell:

```
exit()
```

10. You will need to restart the domain for the changes to take effect.

Tuning the oamDS Data Source

For optimium performance, increase the number of connections allowed by the OAM data source.

To tune oamDS, complete the following steps:

1. Log into the WebLogic Remote Console.
2. Click **Edit Tree**.
3. In **Domain Structure**, expand **Services**, and then click **Data Sources**.
4. Click **oamDS**.
5. In **Settings for oamDS**, click the **Connection Pool** tab and change the following values:
   - **Initial Capacity** to 800
   - **Maximum Capacity** to 800
   - **Minimum Capacity** to 800
6. Click **Save**.
7. Click the **Shopping cart** and select **Commit changes**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Configuring the WebLogic Proxy Plug-In

Before you can validate that requests are routed correctly through the Oracle HTTP Server instances, you must set the `WebLogic Plug-In Enabled` parameter.

It is recommended to set the `WebLogic Plug-In Enabled` parameter at the domain level. Any clusters or servers not using the plugin via the web-tier can have their `WebLogic Plug-In Enabled` parameter value set to `no` on an exception basis as needed.

1. Log in to the Oracle WebLogic Remote Console.
2. Click **Edit Tree**.
3. Click **Domain**.

   The domain settings page is displayed.
4. Click on the Domain Name.
5. Click on the **Web Applications** tab.
6. Locate and select the **WebLogic PlugIn Enabled** option.
7. Click **Save**.
8. Click the **Shopping Cart** and select **Commit Changes**.

Configuring and Integrating with LDAP

Configure OAM to use the LDAP directory.

- Configuring Access Manager to Use the LDAP Directory

Configuring Access Manager to Use the LDAP Directory

After completing the initial installation and setting the security model, you have to associate Oracle Access Manager with the LDAP directory. You can use Oracle Unified Directory (OUD) as the LDAP directory.

To associate Access Manager and the LDAP directory, perform the following tasks:

- Creating a Configuration File
- Integrating Access Manager and LDAP Using the idmConfigTool
- Validating the OAM LDAP Configuration

Creating a Configuration File

Configuring Oracle Access Management to use LDAP requires running the `idmConfigTool` utility. Therefore, you must create a configuration file called `oam.props` to use during the configuration. The contents of this file will be the same as the Configuration file created in Creating a Configuration File with some additions.

For a full description of these properties and values, see Variables Used When Creating Infrastructure for Oracle Access Management.

```
#LDAP Properties
IDSTORE_HOST: idstore.example.com
IDSTORE_PORT: 1636
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
IDSTORE_KEYSTORE_FILE: /u01/oracle/config/keystores/idmTrustStore.p12
IDSTORE_SSL_ENABLED: true
IDSTORE_NEW_SETUP: true
IDSTORE_BINDDN: cn=oudadmin
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=example,dc=com
IDSTORE_SEARCHBASE: dc=example,dc=com
IDSTORE_USERNAMEATTRIBUTE: cn
IDSTORE_LOGINATTRIBUTE: uid
OAM_SERVER_LOGIN_ATTRIBUTE: uid
IDSTORE_USERSEARCHBASE: cn=Users,dc=example,dc=com
IDSTORE_NEW_SETUP: true
IDSTORE_DIRECTORYTYPE: OUD
IDSTORE_OAMADMINUSER: oamadmin
IDSTORE_OAMSOFTWAREUSER: oamLDAP
OAM_IDSTORE_ROLE_SECURITY_ADMIN: OAMAdministrators
IDSTORE_SYSTEMIDBASE: cn=SystemIDs,dc=example,dc=com
IDSTORE_OIMADMINGROUP: OIMAdministrators
IDSTORE_WLSADMINUSER : weblogic_iam
IDSTORE_WLSADMINGROUP : WLSAdministrators
OAM_SERVER_LOGIN_ATTRIBUTE: uid
OAM_IDSTORE_NAME: OAMIDSTORE

#OAM Properties
PRIMARY_OAM_SERVERS: oamhost1.example.com:5575
WEBGATE_TYPE: ohsWebgate14c
ACCESS_GATE_ID: Webgate_IDM
COOKIE_DOMAIN: .example.com
COOKIE_EXPIRY_INTERVAL: 120
OAM_WG_DENY_ON_NOT_PROTECTED: true
OAM_IDM_DOMAIN_OHS_HOST: login.example.com
OAM_IDM_DOMAIN_OHS_PORT: 443
OAM_IDM_DOMAIN_OHS_PROTOCOL: https
OAM_SERVER_LBR_HOST: login.example.com
OAM_SERVER_LBR_PORT: 443
OAM_SERVER_LBR_PROTOCOL: https
OAM_OAM_SERVER_TRANSFER_MODE: open
OAM_OAM_SSLENABLED: true
OAM_TRANSFER_MODE: open
OAM_SSO_ONLY_FLAG: false
OAM_IMPERSONATION_FLAG: false
OAM_IDM_DOMAIN_LOGOUT_URLS: /console/jsp/common/logout.jsp,/em/targetauth/
emaslogout.jsp
OAM_OIM_INTEGRATION_REQ: true
OAM_OIM_OHS_URL: https://oig.example.com:443/
# WebLogic Properties
WLSHOST: iadadminvhn.example.com
WLSPORT: 9002
WLSADMIN: weblogic
WLS_IS_SSLENABLED: true
WLS_TRUSTSTORE: /u01/oracle/config/keystores/idmTrustStore.p12

WLS_SSL_HOST_VERIFICATION: true

# Logger Properties
LOG_FILE: /home/oracle/automation_integ.log
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
LOG_LEVEL: ALL
SSL_DEBUG_ENABLE: FALSE
```

> **Note**
>
> You can also specify the passwords directly in the file, if required. If you do not specify the passwords, you will be prompted for them at runtime. The parameters are:
>
> - IDSTORE_KEYSTORE_PASSWORD
> - IDSTORE_PASSWD
> - IDSTORE_PWD_OAMADMINUSER
> - IDSTORE_PWD_OAMSOFTWAREUSER
> - IDSTORE_PASSWD
> - OAM_OIM_WEBGATE_PASSWD
> - WLSPASSWD
> - WLS_TRUSTSTORE_PASSWORD

Integrating Access Manager and LDAP Using the idmConfigTool

This section describes how to integrate Oracle Access Manager and LDAP using the `idmConfigTool`.

> **Note**
>
> Before running the `idmconfigTool`, ensure that the oam_server1 and oam_server2 Managed Servers and policy manager server are shut down.

Perform the following tasks on OAMHOST1:

1. Set the following environment variables:

   ```
   export ORACLE_HOME=$IAD_ORACLE_HOME/idm
   export DOMAIN_HOME=$IAD_ASERVER_HOME
   ```

2. Run the idmConfigTool utility to perform the integration.

   The syntax of the command on Linux is:

   ```
   cd $IAD_ORACLE_HOME/idm/idmtools/bin


   ./idmConfigTool.sh -configOAM input_file=configfile
   ```

   For example:

   ```
   ./idmConfigTool.sh -configOAM input_file=oam.props./ infant
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

When the command runs you are prompted to enter the password of the account you are connecting to the Identity Store with if you have not specified them in the property file. You are also asked to specify the passwords you want to assign to these accounts:

- IDSTORE_PWD_OAMSOFTWAREUSER
- IDSTORE_PWD_OAMADMINUSER
- OAM_WLS_ADMIN_PASSWD

3. Check the log file for any errors or warnings and correct them. A file named automation.log is created in the directory where you run the tool.
4. Restart the Administration console along with the policy manager server.

> **Note**
>
> After you run `idmConfigTool`, several files are created that you need for subsequent tasks. Keep these in a safe location.
>
> The following files exist in the following directory:
>
> ```
> $IAD_ASERVER_HOME/output/Webgate_IDM
> ```
>
> You need these when you install the WebGate software.
>
> - `cwallet.sso`
> - `ObAccessClient.xml`
> - `password.xml`
> - `aaa_cert.pem`
> - `aaa_key.pem`

> **Note**
>
> If the `oam_policy_mgr` servers were running when `configOAM` was run, then the `WebGate_IDM` artifacts may have been created in `$IAD_MSERVER_HOME/output`. If this is the case, move them back to `SIAD_ASERVER_HOME/output`.

Validating the OAM LDAP Configuration

To validate that this has completed correctly:

1. Log in to the OAM Console using the URL in URLs Used in This Chapter.
2. Click **Agents** from the Application Security screen.
3. When the Search SSO Agents screen appears, click **Search**.
4. You should see the Web Gate agent Webgate_IDM.
5. Log into the WebLogic Remote Console.
6. Click **Security Data** tree.
7. Navigate to **Realms** > **myrealm** > **Authentication Providers** > **OUD Authenticator**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

8. Click **Authenticator Name**. For example, **OUDAuthenticator**.

   A summary screen is displayed. You must verify that the status is OK.

9. Select **Users** or **Groups** to view the Users and Groups returned from the Authenticator.

> **Note**
>
> The list of users and groups will be visible only after you restart the domain.

Updating WebGate Agents

Update the WebGate SSO Agents to use the new security model.

When the `idmConfigTool` is run, it changes the default OAM security model and creates a new WebGate SSO Agent. However, it does not change the existing WebGate SSO Agents to the new security model. After running the `idmConfigTool`, you must update any WebGate agents that previously existed. This involves the following steps:

- Change the security mode to match that of the OAM servers. Failure to do so results in a security mismatch error.
- When WebGates are created at first install, they are unaware that a highly available (HA) installation is performed. After enabling HA, you must ensure that all of the OAM servers are included in the agent configuration, to ensure system continuity.
- You must check that any logout URLs are redirected to the hardware load balancer than one of the local OAM servers.
- Update the REST points for Oracle 14c WebGate HTTP OAM APIs.
- A WebGate agent called **IAMSuiteAgent** is created out of the box. This is created without any password protection and needs to have one added.

To perform these actions, complete the following steps:

1. Access the OAM Console using the URL in URLs Used in This Chapter. Login using the OAM Administration user (`oamadmin`).
2. Click **Agents** pad on the Application Security screen.
3. Ensure that the **WebGates** tab is selected.
4. Click **Search**.
5. Click an Agent, for example: **IAMSuiteAgent**.
6. Set the Security value to the same value defined to **OAM Transfer Mode** on the Access Manager Configuration screen during response file creation.

   If you have changed the OAM security model using the `idmConfigTool`, change the security model used by any existing Webgates to reflect this change.

   Click **Apply**.
7. In the **Primary Server** list, click **+** and add any missing Access Manager Servers.
8. If a password has not already been assigned, enter a password into the **Access Client Password** field and click **Apply**.
9. Set **Maximum Connections** to 20. This is the total maximum number of connections for the primary servers, which is 10 x oam_server1 connections plus 10 x oam_server2 connections.

Copyright © 2018, 2025, Oracle and/or its affiliates.

10. If `OAMRestEndPointHostName` is present or missing for the WebGates `IAMSuiteAgent` and `accessgate-oic`, ensure that it is set to: `login.example.com`.

    If OAMRestEndPointPort is present or missing for any of the WebGates ensure that it is set to: `443`.

    Without setting these two values, the 14c WebGate will not be able to use the new OAP REST APIs for authentication.

11. Click **Apply**.
12. Repeat Steps through for each WebGate.
13. Check that the security setting matches that of your Access Manager servers.

Updating Host Identifiers

When you access your domain you enter using different load balancer entry points. Each of these entry points (virtual hosts) need to be added to the Policy list.

This ensures that if you request access to a resource using `login.example.com` or `oig.example.com`, you have access to the same set of policy rules.

1. Access the OAM Console using the URL in URLs Used in This Chapter. Login using the OAM Administration user (`oamadmin`).
2. Select **Launch Pad** if not already displayed.
3. Click on **Host Identifiers** under **Access Manager**.
4. Click **Search**.
5. Click on **IAMSuiteAgent**.
6. Click **+** in the operations box.
7. Enter the following information.

Table 14-14 Host Name Port Values

Host Name	SSL Terminated Port	End to End SSL Port
`iadadmin.example.com`	80	443
`igdadmin.example.com`	80	443
`igdinternal.example.com`	7777	443
`oig.example.com`	443	443
`login.example.com`	443	443

8. Click **Apply**.

Updating Idle Timeout Value

The default timeout value set in Oracle Access Manager is often too long and can cause issues such as not logging a session out after that session has timed out. Therefore, it is recommended that this value is reduced to 15 minutes.

To update the idle timeout value:

1. Access the OAM Console using the URL in URLs Used in This Chapter. Login using the OAM Administration user (`oamadmin`).

Copyright © 2018, 2025, Oracle and/or its affiliates.

2. Click **Configuration**.
3. Select **Common Settings** under **Settings**.
4. Change **Idle Time out (minutes)** to 15.
5. Click **Apply**.

Validating the Authentication Providers

Set the order of identity assertion and authentication providers in the WebLogic Remote Console.

1. Log in to the WebLogic Remote Console, if not already logged in.
2. Click on the **Edit Tree**.
3. Select **Security Realms**.
4. Click the **myrealm** default realm entry.
5. Click the **Authentication Providers** tab.
6. From the table of providers, click the **DefaultAuthenticator**.
7. Set the Control Flag to `SUFFICIENT`.
8. Click **Save** to save the settings.
9. From the navigation breadcrumbs, click **Authentication Providers** to return to the list of providers.
10. Select a provider and reorder as per the table below by clicking **Move Up** or **Move Down**

Table 14-15 Sort order

Sort Order	Provider	Control Flag
1	OAMIDAsserter	`REQUIRED`
2	LDAP Authentication Provider	`SUFFICIENT`
3	DefaultIdentityAsserter	`N/A`
4	Trust Service Identity Asserter	`N/A`
5	DefaultAuthenticator	`SUFFICIENT`

11. Click **Shopping Cart** > **Commit Changes**.
12. Shut down the Administration Server, Managed Servers, and any system components, as applicable.
13. If you are going to configure ADF consoles with SSO, you can keep the managed servers down and restart them later. If not, you need to restart managed servers now.

Configuring Oracle ADF and OPSS Security with Oracle Access Manager

Some Oracle Fusion Middleware management consoles use Oracle Application Development Framework (Oracle ADF) security, which can integrate with Oracle Access Manager Single Sign-on (SSO). These applications can take advantage of Oracle Platform Security Services

Copyright © 2018, 2025, Oracle and/or its affiliates.

(OPSS) SSO for user authentication, but you must first configure the domain-level `jps-config.xml` file to enable these capabilities.

The domain-level `jps-config.xml` file is located in the following location after you create an Oracle Fusion Middleware domain:

```
$IAD_ASERVER_HOME/config/fmwconfig/jps-config.xml
```

> **Note**
>
> The domain-level `jps-config.xml` should not be confused with the `jps-config.xml` that is deployed with custom applications.

1. Change to the following directory:

   ```
   cd $ORACLE_COMMON_HOME/common/bin
   ```

2. If your WebLogic domain is running in secure mode set the environment variable `WLST_PROPERTIES` to include the location of your trust store. For example:

   ```
   export WLST_PROPERTIES="-Dweblogic.security.TrustKeyStore=CustomTrust -
   Dweblogic.security.CustomTrustKeyStoreFileName=/u01/oracle/config/
   keystores/idmTrustStore.p12 -
   Dweblogic.security.CustomTrustKeyStorePassPhrase=password -
   Dweblogic.security.SSL.ignoreHostnameVerification=true"
   ```

3. Start the WebLogic Server Scripting Tool (WLST):

   ```
   ./wlst.sh
   ```

   > **Note**
   >
   > If you are using an SSL enabled Admin Server, you must set the following environment variable before invoking WLST:
   >
   > ```
   > export WLST_PROPERTIES="-
   > Dweblogic.security.TrustKeyStore=CustomTrust -
   > Dweblogic.security.CustomTrustKeyStoreFileName=/u01/oracle/config/
   > keystores/idmTrustStore.p12 -
   > Dweblogic.security.CustomTrustKeyStorePassPhrase=password -
   > Dweblogic.security.SSL.ignoreHostnameVerification=true"
   > ```

4. Connect to the Administration Server, by using the following WLST command:

   ```
   connect('admin_user','admin_password','admin_url')
   ```

   For example SSL Terminated

   ```
   connect('weblogic','Manager1','t3://iadadminvhn.example.com:7001')
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

For example End to End SSL

```
connect('weblogic','password','t3s://iadadminvhn.example.com:9002')
```

5. Run the `addOAMSSOProvider` command, as shown:

```
addOAMSSOProvider(loginuri="/${app.context}/adfAuthentication",
logouturi="/oam/logout.html")
```

The following table defines the expected value for each argument in the `addOAMProvider` command.

> **Note**
>
> Perform this action for each domain in your configuration.

Table 14-16 Expected Values for the Argument in the `addOAMProvider` command

Argument	Definition
loginuri	Specifies the URI of the login page **Note** For ADF security enabled applications, "/*context-root*/adfAuthentication" should be provided for the 'loginuri' parameter. For example: `/${app.context}/adfAuthentication` **Note** `${app.context}` must be entered as shown. At runtime, the application replaces the variable appropriately. Here is the flow: a. User accesses a resource that has been protected by authorization policies in OPSS, fox example. b. If the user is not yet authenticated, ADF redirects the user to the URI configured in *loginuri*. c. Access Manager, should have a policy to protect the value in *loginuri*: for example, "/*context-root*/ adfAuthentication". d. When ADF redirects to this URI, Access Manager displays a Login Page (depending on the authentication scheme configured in Access Manager for this URI).

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-16 (Cont.) Expected Values for the Argument in the `addOAMProvider` command

Argument	Definition
logouturi	Specifies the URI of the logout page. The value of the *loginurl* is usually `/oam/logout.html`.
autologinuri	Specifies the URI of the autologin page. This is an optional parameter.

6. Disconnect from the Administration Server by entering the following command:

   ```
   disconnect()
   ```

7. Restart the Administration Server and the managed servers.

Propagating the Domain to OAMHOST2

After you start and validate the Administration Server and Managed Servers on OAMHOST1, you can then perform the following tasks on OAMHOST2.

- Unpacking the Domain Configuration on OAMHOST2
- Starting the Node Manager on OAMHOST2

Unpacking the Domain Configuration on OAMHOST2

Now that you have the Administration Server and Managed Servers running on OAMHOST1, you can configure the domain on OAMHOST2.

1. Log into OAMHOST2.
2. If you haven't already, create the recommended directory structure for the Managed Server domain on the OAMHOST2 storage device.

   Use the examples in File System and Directory Variables Used in This Guide as a guide.
3. Make sure the `oamdomaintemplate.jar` accessible to OAMHOST2.

   For example, if you are using a separate shared storage volume or partition for OAMHOST2, then copy the template to the volume or partition mounted to OAMHOST2.
4. Run the `unpack` command to unpack the template in the domain directory onto the local storage, as follows:

   ```
   cd $ORACLE_COMMON_HOME/common/bin


   ./unpack.sh -domain=$IAD_MSERVER_HOME \
   -overwrite_domain=true \
   -template=$SHARED_CONFIG_DIR/domains/oamdomaintemplate.jar \
   -log_priority=DEBUG
   -log=/tmp/unpack.log
   -app_dir=$MS_APPLICATION_HOME
   ```

   In this example:

   - `$IAD_MSERVER_HOME` is the complete path to the domain home to be created on the local storage disk. This is the location where the copy of the domain will be unpacked.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Replace *full_path* with the complete path and file name of the domain template jar file that you created when you ran the pack command to pack up the domain on the shared storage device.
- Replace `$MS_APPLICATION_HOME` with the complete path to the Application directory for the domain on local storage. See File System and Directory Variables Used in This Guide.

> **Tip**
>
> For more information about the pack and unpack commands, see Overview of the Pack and Unpack Commands in *Creating Templates and Domains Using the Pack and Unpack Commands*.

5. Change directory to the newly created `$IAD_MSERVER_HOME` directory and verify that the domain configuration files were copied to the correct location on the OAMHOST2 local storage device.

Starting the Node Manager on OAMHOST2

After you manually set up the Node Manager to use a per host Node Manager configuration, you can start the Node Manager by using the following commands on OAMHOST2.

1. After you manually set up the Node Manager to use a per host Node Manager configuration, you can start the Node Manager by using the following commands on OAMHOST2:

   ```
   cd $NM_HOME
   ```

2. Use the following command to start the Node Manager on OAMHOST2:

   ```
   nohup ./startNodeManager.sh > nodemanager.out 2>&1 &
   ```

For information about additional Node Manager configuration options, see *Administering Node Manager for Oracle WebLogic Server*.

Starting the Managed Servers in the Domain

Start the Managed Servers.

- Starting the oam_server1 Managed Server

Starting the oam_server1 Managed Server

From the Remote Console, start the oam_server1 Managed Server:

1. Log into Enterprise Manager using the URL in URLs Used in This Chapter.
2. Sign into the Fusion Middleware Control by using the administrator's account.
3. Select the Servers pane to view the Managed Servers in the domain.
4. Select the Managed Servers that you choose to start.
5. Click **Control** > **Start on the tool bar to start the selected Managed Server**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

6. Verify that the server status is reported as `Running`. If the server is shown as `Starting or Resuming`, wait for the server status to change to `Started`. If another status is reported (such as `Admin` or `Failed`), check the server output log files for errors.
7. Repeat for each Managed Server you choose to start.

Validating Access Manager

You can validate Oracle Access Manager by using the `oamtest` tool.

To do this, perform the following steps:

1. Ensure that the oam_server1 managed server is up and running.
2. Naviage to the following directory:

   ```
   cd $IAD_ORACLE_HOME/idm/oam/server/tester
   ```

3. Start the test tool in a terminal window using the command:

   ```
   java -jar oamtest.jar
   ```

4. When the OAM test tool starts, enter the following information in the Server Connection section of the page:
   - **Primary IP Address**: `OAMHOST1.example.com`
   - **Port**: `5575` (*`OAM_PROXY_PORT`*)
   - **Agent ID**: `Webgate_IDM`
   - **Agent Password**: `webgate password`
   - **Mode**: `Open`
   - **Global Passphrase**: Enter the value you set as the global password in Setting a Global Passphrase.
5. Click **Connect**.

   In the status window you'll see: `response] Connected to primary access server.`
6. In the Protected Resource URI section, enter the following information:
   - **Scheme**: http
   - **Host**: `iadadmin.example.com`
   - **Port**: `80` (*IAD_HTTP_PORT*)
   - **Resource**: `/oamconsole`

     Click **Validate**.

     In the status window you see: `[request] [validate] yes`.
7. In the User Identity window, enter:
   - **Username**: oamadmin
   - **Password**: password
   - Click **Authenticate**.
   - In the status window, you see: `[request] [authenticate] yes`
   - Click **Authorize**.
   - In the status window you see. `[request] [authorize] yes`

Copyright © 2018, 2025, Oracle and/or its affiliates.

Enabling Forgotten Password

You can set up the One Time Pin forgotten password functionality which is provided with Oracle Access Manager

If you want to configure the Challenge Question forgotten password functionality, as provided by Oracle Identity Governance, see Configuring and Integrating with LDAP and Integrating Oracle Identity Governance and Oracle Access Manager.

This section contains the following topics:

- Prerequisites for Enabling Forgotten Password
- Add Permissions to oamLDAP user
- Create an OTP Administrative Group in LDAP
- Enabling Adaptive Authentication Service
- Configuring Adaptive Authentication Plug-in
- Enabling Password Management in the Directory
- Storing User Messaging Credentials in CSF
- Setting Up the Forgot Password Link on the Login Page
- Restarting the Domain
- Validating the Forgotten Password Functionality

Prerequisites for Enabling Forgotten Password

Forgotten Password Management in Oracle Access Manager takes the form of sending an Email or SMS message with a link to reset the password.

Email or SMS is sent using the Oracle User Messaging Service. Before enabling the Oracle Forgotten Password functionality, you first need to have an Oracle User Messaging deployment. This is often located inside the Oracle Governance Domain but can be located inside the Access Domain if that is all you are installing. Alternatively, it could be a completely independent domain.

Forgotten Password functionality works only if you have successfully configured Single Sign-On as described in Configuring Single Sign-On for an Enterprise Deployment.

Adding the User Messaging Service to the Access domain or creating a User Messaging Service domain is outside of the scope of the this EDG. For more information about installing and configuring the Oracle User Messaging Service, see Installing User Messaging Service and Configuring Oracle User Messaging Service in *Administering Oracle User Messaging Service*.

Add Permissions to oamLDAP user

When created out of the box the `oamLDAP` user (the user used to link OAM to LDAP) is granted privileges to read the LDAP directory. It is not however granted permission to update those users. You need to add these privileges for the OAM forgotten password functionality to work.

To do this you perform the following steps on LDAPHOST1:

Copyright © 2018, 2025, Oracle and/or its affiliates.

1. Create a file called `add_aci.ldif` with the following contents:

```
dn: cn=oamLDAP,cn=systemids,dc=example,dc=com  changetype: modify add: ds-
privilege-name ds-privilege-name: password-reset
dn: cn=Users,dc=example,dc=com changetype: modify add: aci aci:
(targetattr = "*")(targetfilter= "(objectclass=inetorgperson)")
(targetscope = "subtree") (version 3.0; acl "iam admin changepwd"; allow
(compare,search,read,selfwrite,add,write,delete) userdn = "ldap:///
cn=oamLDAP,cn=systemids,dc=example,dc=com";)
```

2. Use the `ldapmodify` command to add the privileges to LDAP. For example:

On LDAPHOST1 action the file using the command:

```
export OUD_ORACLE_INSTANCE=/u02/oracle/config/instances/oud1


$OUD_ORACLE_INSTANCE/bin/ldapmodify -D cn=oudadmin -h LDAPHOST1 -p 1389 -f ./
add_aci.ldif
```

Create an OTP Administrative Group in LDAP

In order for the oamadmin group to be able to invoke forgotten password system calls it needs to be a member of the group `OTPRestUserGroup`. This group is not created by `idmConfigTool` and must therefore be created manually.

To do this you perform the following steps on LDAPHOST1:

1. Create a file called `create_otp_group.ldif` with the following contents:

```
dn: cn=OTPRestUserGroup,cn=Groups,dc=example,dc=com
changetype: add
objectClass: top
objectClass: orclgroup
objectClass: groupofuniquenames
cn: OTPRestUserGroup
description: Forgotten Password Admin group
displayName: OTPRestUserGroup
uniquemember: cn=oamadmin,cn=Users,dc=example,dc=com
```

2. Use the `ldapmodify` command to add the group to LDAP. For example:

```
export OUD_ORACLE_INSTANCE=/u02/oracle/config/instances/oud1


$OUD_ORACLE_INSTANCE/bin/ldapmodify -D cn=oudadmin -h LDAPHOST1 -p 1389 -f
create_otp_group.ldif
```

Enabling Adaptive Authentication Service

Forgotten password requires you to enable the Adaptive Authentication Service.

To enable this service:

Copyright © 2018, 2025, Oracle and/or its affiliates.

1. Access the OAM Console using the URL in URLs Used in This Chapter. Login using the OAM Administration user (`oamadmin`).
2. Click **Configuration**.
3. Click **Available Services**.
4. Click **Enable Service** next to Adaptive Authentication Service.
5. When prompted, confirm that you want to enable the service.

Configuring Adaptive Authentication Plug-in

Now that the Authentication service is enabled, it needs to be informed about your User Messaging service.

To configure Adaptive Authentication Plug-In, perform the following steps:

1. Access the OAM Console using the URL in URLs Used in This Chapter. Login using the OAM Administration user (`oamadmin`).
2. From the Application Security Launch Pad, click **Authentication Plug-ins** in the Plug-ins panel. From the Authentication Plug-in tab, type **Adaptive** in the quick search box above the Plug-in Name column and hit **Enter**.

   The **AdaptiveAuthenticationPlugin** is displayed.
3. Enter the following plug in properties:

Table 14-17 AdaptiveAuthentication Plug-In Properties

Attribute	Value
UmsAvailable	True

UmsClientURL	Specify the entry point of your User Messaging service. If you have configured Oracle Identity Manager, then this will be either:
	
	`http://igdinternal.example.com:7777/ucs/messaging/webservice`
	
	or
	
	`https://igdinternal.example.com:443/ucs/messaging/webservice`

4. Click **Save**.

Enabling Password Management in the Directory

By default, OAM is not set to enable password management. You have to enable it through the OAM Console.

To enable Password Management in the directory:

Copyright © 2018, 2025, Oracle and/or its affiliates.

1. Access the OAM Console using the URL in URLs Used in This Chapter. Login using the OAM Administration user (`oamadmin`).
2. Click **Configuration**.
3. Click **User Identity Stores**.
4. Click the LDAP identity store in the OAM Identity Store section. For example: `OAMIDSTORE`.
5. Click **Edit**.
6. Select **Enable Password Management**.
7. Enter the details in the **User Information** field.

Table 14-18 User Information Details

Attribute	Description
Global Common ID Attribute	The unique identifier in LDAP for the user. For example: **uid**.
First Name	The LDAP attribute which holds the users name. For example: **cn**.
Last Name	The LDAP attribute which holds the users last name. For example: **sn**.
Email Address	The LDAP attribute which holds the user's email address. For example: **mail**.

8. Click **Apply**.

Storing User Messaging Credentials in CSF

Before you can access the User Messaging Service, you need to store the credentials in the WebLogic credential store.

1. Run the following command to start `wlst`:

```
$ORACLE_COMMON_HOME/bin/wlst.sh
```

> **Note**
>
> If your administration server is SSL enabled, set the following environment variable before starting wlst.
>
> ```
> export WLST_PROPERTIES="-
> Dweblogic.security.TrustKeyStore=CustomTrust -
> Dweblogic.security.CustomTrustKeyStoreFileName=/u02/oracle/config/
> keystores/idmTrustStore.p12 -
> Dweblogic.security.CustomTrustKeyStorePassPhrase=Manager1 -
> Dweblogic.security.SSL.ignoreHostnameVerification=true"
> ```

2. Run the following command to Store the messaging credentials.

Copyright © 2018, 2025, Oracle and/or its affiliates.

For SSL Terminated:

```
connect('weblogic','password','t3://iadadminvhn.example.com:7001')


createCred(map="OAM_CONFIG", key="umsKey", user="weblogic",
password="password")


createCred(map="OAM_CONFIG", key="oam_rest_cred", user="oamadmin",
password="password")


exit ()
```

For End to End SSL:

```
connect('weblogic','password','t3s://iadadminvhn.example.com:9002')


createCred(map="OAM_CONFIG", key="umsKey", user="weblogic",
password="password")


createCred(map="OAM_CONFIG", key="oam_rest_cred", user="oamadmin",
password="password")


exit ()
```

Setting Up the Forgot Password Link on the Login Page

The following REST API command enables the OTP forgot password link on the default login page in OAM.

```
 curl -X -k PUT \
  https://login.example.com/oam/services/rest/access/api/v1/config/
otpforgotpassword/ \
  -u oamadmin:Password \
  -H 'content-type: application/json' \
  -d
'{"displayOTPForgotPassworLink":"true","defaultOTPForgotPasswordLink":"false",
"localToOAMServer":"true","forgotPasswordURL":"https://login.example.com/
otpfp/pages/fp.jsp", "mode":"userselectchallenge"}'
```

Enter the required attributes and values:

Table 14-19 Forgot Password Link on Login Page

Attributes	Value
`ForgotPasswordURL`	The OAM Forgotten Password URL. For example, https://login.example.com/otpfp/pages/fp.jsp

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 14-19 (Cont.) Forgot Password Link on Login Page

Attributes	Value
`mode`	distribution_mode The distribution mode determines how the password reset URL is sent to the end user. Valid values are: **email, sms, userchoose, userselectchallenge**. The last entry enables the user to choose from masked values. • **Email** - An OTP is sent to the email configured in the mail field. • **SMS** - An OTP is sent to the mobile number configured in the mobile field. • **Userchoose** - An OTP is sent by letting the user choose either the email or the mobile option, without the exact values. • **Userselectchallenge** - User can see the masked values either as email or mobile and select one of the options.

> **Note**
>
> If you are using self signed certificates in the load balancer the curl command may object with a message similar to:
> curl performs SSL certificate verification by default, using a **bundle** of Certificate Authority (CA) public keys (CA certs). If the default bundle file isn't adequate, you can specify an alternate file using the --cacert option. If this HTTPS server uses a certificate signed by a CA represented in the bundle, the certificate verification probably failed due to a problem with the certificate (it might be expired, or the name might not match the domain name in the URL). If you like to turn off curl's verification of the certificate, use the **-k (or --insecure)** option.
>
> If you see this message and are sure, add **-k after -u oamadmin:Password**.

Verify that this has succeeded by accessing the followig URL in a browser:

```
https://login.example.com/oam/services/rest/access/api/v1/config/
otpforgotpassword
```

When prompted, enter your `oamadmin` account and password.

> **Note**
>
> One of the OAM Managed Servers must be running for this command to succeed.

Restarting the Domain

Shutdown the **Administration Server** and any running managed servers and restart the Administration Server and all of the managed servers (oam_policy_mgr1, oam_policy_mgr2, oam_server1, oam_server2).

Copyright © 2018, 2025, Oracle and/or its affiliates.

Validating the Forgotten Password Functionality

If you have set up the OAM Forgotten Password functionality, rather than off-loading to OIG, you can validate the forgotten password using the curl command, which shows you the password policies in force.

To validate the Forgotten Password functionality, run the following `curl` command:

```
curl -X GET https://login.example.com/oam/services/rest/access/api/v1/pswdmanagement/
UserPasswordPolicyRetriever/oamadmin?description=true  -u oamadmin:<password> -k
```

This command displays the password policies.

If this command works, access the Enterprise Manager URL using the URL in URLs Used in This Chapter. After you enable single sign-on, you see a link for the forgotten password on the login page. Click this link and enter the user name for which you want to reset the password. Click **Generate Pin** to receive an email, which enables you to change the password.

> **Note**
>
> Before validating, ensure that you enable SSO as described in Configuring Single Sign-On for an Enterprise Deployment. Else, validation fails.

Replacing Connect Strings with the Appropriate TNS Alias

Oracle recommends using TNS Alias in the connection strings used by FMW components instead of repeating long JDBC strings across multiple connections pools.

For more information about how to use TNS alias in your Datasources, see Using TNS Alias in Connect Strings in the *Common Configuration and Management Tasks for an Enterprise Deployment* chapter.

Backing Up the Configuration

It is an Oracle best practices recommendation to create a backup after you successfully extended a domain or at another logical point. Create a backup after you verify that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

The backup destination is the local disk. You can discard this backup when the enterprise deployment setup is complete. After the enterprise deployment setup is complete, you can initiate the regular deployment-specific Backup and Recovery process.

For information about backing up your configuration, see Performing Backups and Recoveries for an Enterprise Deployment.

Copyright © 2018, 2025, Oracle and/or its affiliates.

15

Configuring Oracle Identity Governance

Install and configure Oracle Identity Governance (OIG).

The following topics describe how to install and configure an initial domain, which can be used as the starting point for an enterprise deployment. Later chapters in this guide describe how to extend this initial domain with the various products and components that comprise the enterprise topology you are deploying.

A complete Oracle Identity and Access Management uses a split domain deployment, where there is a single domain for Oracle Access Management and a different one for Oracle Identity Governance. You must create a separate infrastructures for Access and Governance.

- Synchronizing the System Clocks
  Before you deploy Oracle Identity Governance, verify that the system clocks on each host computer are synchronized. You can do this by running the `date` command simultaneously on all the hosts in each cluster.
- About the Initial Infrastructure Domain
  Before you create the initial Infrastructure domain, ensure that you review the key concepts.
- Variables Used When Creating the Infrastructure Domain
  As you perform the tasks in this chapter, you will be referencing the variables listed in this section.
- Setting Environment Variables
  Set environment variables used in this chapter.
- URLs Used in This Chapter
  This section describes the URLs used in this chapter.
- Installing the Oracle Fusion Middleware
  Install the Oracle Fusion Middleware software in preparation for configuring a new domain for Oracle Identity Governance.
- Verifying the Installation
  After you complete the installation, you nust verify it.
- Downloading the Oracle Connector Bundle
  Download the Oracle Connector bundle.
- Installing the Oracle Identity Governance Connector
  After you download the Oracle Connector for LDAP, install it into the ORACLE_HOME directory.
- Creating the Database Schemas for Oracle Identity Governance
  Oracle Fusion Middleware components require the existence of schemas in a database before you configure a Fusion Middleware Infrastructure domain. Install the schemas listed in this topic in a certified database for use with this release of Oracle Fusion Middleware.
- Configuring the Oracle Identity Governance Domain
  The following topics provide instructions for creating an Oracle Identity Governance domain using the Fusion Middleware Configuration wizard.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Creating Oracle Identity Manager Authenticator
  Before you start the domain, you have to run a script which creates the Oracle Identity Manager (OIM) Authenticator in the domain.
- Enabling SSL
  If you are configuring End to End SSL, you must perform additional steps.
- Configuring a Per Host Node Manager for an Enterprise Deployment
  For specific enterprise deployments, Oracle recommends that you configure a per-host Node Manager, as opposed to the default per-domain Node Manager.
- Tuning JMS Queues
  You need to tune the JMS queues to ensure maximum throughput.
- Configuring the Domain Directories and Starting the Servers
  After you have configured the domain and configured the Node Manager, you can start the Administration Server by using the Node Manager. In an enterprise deployment, the Node Manager is used to start and stop the Administration Server and all the Managed Servers in the domain.
- Creating the New Authentication Provider
  After creating the new domain, if you are using LDAP and want to log in using LDAP, then you must create an authentication provider for the directory inside the OIG domain.
- Adding a Load Balancer Certificate to Oracle Keystore Service
  Some OIG Products require that the SSL certificate used by the load balancer be added to the Oracle Keystore Service Trusted Certificates.
- Configuring the WebLogic Proxy Plug-In
  Before you can validate that requests are routed correctly through the Oracle HTTP Server instances, you must set the `WebLogic Plug-In Enabled` parameter.
- Modifying the Upload and Stage Directories to an Absolute Path
- Starting and Validating the Oracle Identity Governance Managed Servers
  Now that you have extended the domain, started the Administration Server, and propagated the domain to the other hosts, you can start the newly configured Oracle Identity Governance Managed Servers.
- Analyzing the Bootstrap Report
  When you start the Oracle Identity Governance server, the bootstrap report is generated at `$IGD_ASERVER_HOME/servers/oim_server1/logs/BootStrapReportPreStart.html`.
- Creating a Separate Domain Directory for Managed Servers
  When you initially create the domain for enterprise deployment, the domain directory resides on a shared disk. This default domain directory is used to run the Administration Server. You can now create a copy of the domain on the local storage for each of your managed server hosts. The domain directory on the local (or private) storage is used to run the Managed Servers.
- Validating the Fusion Middleware Control Application
  After the bootstrap process has been executed and validated, access to the Fusion Middleware Control application should be available.
- Configuring the Web Tier for the Domain
  Configure the web server instances on the web tier so that the instances route requests for both public and internal URLs to the proper clusters in the extended domain.
- Integrating Oracle Identity Governance with Oracle SOA Suite
  Oralce Identity Governance invokes Oracle SOA Suite using a number of URLs, which out of the box are wired to individual managed servers. In a High Availibility environment you need to update these URLs to use a load balancer.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Managing the Notification Service
  An event is an operation that occurs in Oracle Identity Manager, such as user creation, request initiation, or any custom event created by the user. These events are generated as part of the business operations or through the generation of errors. Event definition is the metadata that describes the event.
- Configuring the Messaging Drivers
  Each messaging driver needs to be configured. You have to configure this service if you want to enable OAM's forgotten password functionality.
- Forcing Oracle Identity Governance to use Correct Multicast Address
  Oracle Identity Governance uses multicast for certain functions. By default, the managed servers communicate using the multi cast address assigned to the primary host name. If you wish multicast to use a different network, for example, of the internal network, you must complete additional steps.
- Integrating Oracle Identity Governance with LDAP
  Integrate Oracle Identity Governance with LDAP.
- Integrating Oracle Identity Governance and Oracle Access Manager
  You have to complete several tasks to integrate Oracle Identity Governance and Oracle Access Manager. These tasks include creating the WLS authentication providers, deleting OIMSignatureAuthenticator and recreating OUDAuthenticator, adding the administration role to the new administration group, and so on.
- Propagating the Domain and Starting the Servers on OIGHOST2
  After you start and validate the Administration Server on OIGHOST1, you can then you must propagate the domain and start the servers on OIGHOST2.
- Configuring OIG Workflow Notifications to be Sent by Email
  OIG uses the human workflow, which is integrated with the SOA workflow. The SOA server configures email to receive the notifications that are delivered to the user mailbox. The user can accept or reject the notifications.
- Adding the wsm-pm Role to the Administrators Group
  After you configure a new LDAP-based Authorization Provider and restart the Administration Server, add the enterprise deployment administration LDAP group (OIMAdministrators) as a member to the `policy.Updater` role in the `wsm-pm` application stripe.
- Adding the Oracle Access Manager Load Balancer Certificate to the Oracle Keystore Service
  The Oracle Identity Governance to Business Intelligence Reports link inside of the Self Service application requires that the SSL certificate used by the load balancer be added to the Oracle Keystore Service Trusted Certificates.
- Restarting the IAMGovernanceDomain
  Restart the domain for the configuration steps to take effect.
- Setting Challenge Questions
  If you have integrated OAM and OIG, then after the environment is ready, you need to set up the challenge questions for your system users.
- Replacing Connect Strings with the Appropriate TNS Alias
  Oracle recommends using TNS Alias in the connection strings used by FMW components instead of repeating long JDBC strings across multiple connections pools.
- Integrating Oracle Identity Manager with Oracle Business Intelligence Publisher
  Oracle Identity Manager comes with a number of prebuilt reports that can be used to provide information about Oracle Identity and Access Management.
- Verification of Manual Failover of the Administration Server
  After you configure the domain, you must test failover is working correctly.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Backing Up the Configuration
  It is an Oracle best practices recommendation to create a backup after you successfully extended a domain or at another logical point. Create a backup after you verify that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

Synchronizing the System Clocks

Before you deploy Oracle Identity Governance, verify that the system clocks on each host computer are synchronized. You can do this by running the `date` command simultaneously on all the hosts in each cluster.

Alternatively, there are third-party and open-source utilities you can use for this purpose.

About the Initial Infrastructure Domain

Before you create the initial Infrastructure domain, ensure that you review the key concepts.

- About the Infrastructure Distribution
- Characteristics of the Domain

About the Infrastructure Distribution

You create the initial Infrastructure domain for an enterprise deployment by using the Oracle Fusion Middleware Infrastructure distribution. This distribution contains both the Oracle WebLogic Server software and the Oracle JRF software.

The Oracle JRF software consists of Oracle Web Services Manager, Oracle Application Development Framework (Oracle ADF), Oracle Enterprise Manager Fusion Middleware Control, the Repository Creation Utility (RCU), and other libraries and technologies that are required to support the Oracle Fusion Middleware products.

> **Note**
>
> The Access infrastructure does not use the Web Services Manager.

See Understanding Oracle Fusion Middleware Infrastructure in *Understanding Oracle Fusion Middleware*.

Characteristics of the Domain

The following table lists some of the key characteristics of the domain that you are about to create. Reviewing these characteristics helps you to understand the purpose and context of the procedures that are used to configure the domain.

Many of these characteristics are described in more detail in Understanding a Typical Enterprise Deployment.

Characteristic of the Domain	More Information
Uses a separate virtual IP (VIP) address for the Administration Server.	Configuration of the Administration Server and Managed Servers Domain Directories

Copyright © 2018, 2025, Oracle and/or its affiliates.

Characteristic of the Domain	More Information
Uses separate domain directories for the Administration Server and the Managed Servers in the domain.	Configuration of the Administration Server and Managed Servers Domain Directories
Uses a per domain Node Manager configuration.	About the Node Manager Configuration in a Typical Enterprise Deployment
Requires a separately installed LDAP-based authentication provider.	Understanding OPSS and Requests to the Authentication and Authorization Stores

Variables Used When Creating the Infrastructure Domain

As you perform the tasks in this chapter, you will be referencing the variables listed in this section.

The following table explains the configuration file property values required in this section.

Table 15-1 LDAP Variables Used in This Chapter

Variable	Sample Value	Description
IDSTORE_HOST	`idstore.example.com`	The host and port of your Identity Store directory. When preparing the Identity Store these should point to one of the LDAP instances. When configuring components such as OAM or OIG they should point to the load balancer entry point.
IDSTORE_PORT	`1636`	The port of your Identity Store directory. When preparing the Identity Store these should point to one of the LDAP instances. When configuring components such as OAM or OIG they should point to the load balancer entry point.
IDSTORE_DIRECTORYTYPE	`OUD`	The type of directory you are using. Valid value is OUD.
IDSTORE_BINDDN	`cn=oudadmin`	An administrative user in the Identity Store Directory.
IDSTORE_BINDDN_PWD	`password`	The the password of IDSTORE_BINDDN. You will be prompted for the value if not supplied.
IDSTORE_SEARCHBASE	`dc=example,dc=com`	The location in the directory where Users and Groups are stored.
IDSTORE_SSL_ENABLED	`true`	Whether SSL to the identity store is enabled. Valid values: `true | false`.
OAM_IDSTORE_NAME	`OAMIDSTORE`	Name of the IDStore to create.
IDSTORE_USERSEARCHBASE	`cn=Users,dc=example,dc=com`	The location in the directory where Users are Stored.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-1 (Cont.) LDAP Variables Used in This Chapter

Variable	Sample Value	Description
IDSTORE_GROUPSEARCHBASE	`cn=Groups,dc=example,dc=com`	The location in the directory where Groups are Stored.
IDSTORE_SYSTEMIDBASE	`cn=SystemIDs,dc=example,dc=com`	The location of a container in the directory where system users can be placed when you do not want them in the main user container.
IDSTORE_KEYSTORE_FILE	`/u01/oracle/config/keystores/idmTrustStore.p12`	The location of the Truststore for OIG.
KEYSTORE_HOME	`/u01/oracle/config/keystores`	The location of the keystore home on shared storage for OIG.
LOCAL_KEYSTORE_HOME	`/u02/oracle/config/keystores`	The location of the keystore home on shared storage for OIG.
IDSTORE_KEYSTORE_PASSWORD	`password`	The password of the IDSTORE_KEYSTORE_FILE.

Table 15-2 Variables Used in This Chapter

Variable	Sample Value	Description
IGD_ORACLE_HOME	`/u01/oracle/products/oig`	The read-only location for the OIG product binaries stored on shared disk.
ASERVER_HOME	`/u01/oracle/config/domains/oig`	The Administration Server domain home, which is installed on a shared disk.
IGD_ASERVER_HOME	`/u01/oracle/config/domains/oig`	Should be the same value as *IGD_ASERVER_HOME*.
IGD_MSERVER_HOME	`/u02/oracle/config/domains/oig`	The Managed Server domain home, which is created by using the unpack command on the local disk of each application tier host.
MSERVER_HOME	`/u02/oracle/config/domains/oig`	Should be the same value as *IGD_MSERVER_HOME*.
APPLICATION_HOME	`/u01/oracle/config/applications/oig`	The Application home directory, which is installed on shared disk, so the directory is accessible by all the application tier host computers.
MS_APPLICATION_HOME	`/u02/oracle/config/applications/oig`	The Application home directory, which is installed on local disk, so the directory is accessible by all the application tier host computers.
JAVA_HOME	`/u01/oracle/products/jdk`	The location where you install the supported Java Development Kit (JDK).

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-2 (Cont.) Variables Used in This Chapter

Variable	Sample Value	Description
ADMINVHN	`igdadminvhn.example.com`	The virtual host name used as the listen address for the Administration Server used by the IAMGovernanceDomain and fails over with manual failover of the Administration Server. It is enabled on the node where the Administration Server process is running.
OIGHOST1	`oighost1.example.com`	The hostname of OIGHOST1.
OIGHOST2	`oighost2.example.com`	The hostname of OIGHOST2.
SCAN_ADDRESS	`db-scan.example.com`	Address for the Oracle RAC Database.
WEBGATE_TYPE	`ohsWebgate14c`	The type of webgate profile to create. This should always be `ohsWebgate14c`.
COOKIE_DOMAIN	`.example.com`	The domain you wish to associate the OAM cookie with this is normally the same as the IDSTORE_SEARCH_BASE in domain format. The search base can be found in the worksheet (REALM_DN).
OAM_TRANSFER_MODE	`open`	The type of OAM security transport to be used. This should be the same as OAM_OAM_SERVER_TRANSPORT_MODE.
OAM_OIM_OHS_URL	`https://oig.example.com:443/`	If you are planning on using OIM for Forgotten Password functionality then you need to specify the external entry point for OIG. This is the OIG URL to which OAM directs the requests. This URL is made up of the following values from the worksheet: `https://oig.example.com:IAG_HTTPS_PORT/`.
IDSTORE_PWD_OAMADMINUSER	`password`	The password of the Admin user account (`oamadmin`) you are connecting to the Identity Store with.
IDSTORE_PWD_OAMSOFTWAREUSER	`password`	The password of the account (`oamLDAP`) you are connecting to the Identity Store with.
OAM_WLS_ADMIN_PASSWD	`password`	The password of the WLS Admin account (`weblogic_iam`) .
OIM_WLSHOST	`IGDADMINVHN.example.com`	Virtual Host name for Admin Server on OIGHOST1.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-2 (Cont.) Variables Used in This Chapter

Variable	Sample Value	Description
OIM_WLSPORT	`9102`	Corresponding port number for Admin Server on OIGHOST1.
OIM_WLSADMIN	`weblogic`	The weblogic administrator user in OIM domain.
OIM_WLSADMIN_PWD	`password`	Password for OIM_WLSADMIN.
OIM_IDSTORE_ROLE_SECURITY_ADMIN	`WLSAdministrators`	The group in your LDAP directory which has WebLogic administration access.
WLS_IS_SSLENABLED	`true`	Whether SSL to the identity store is enabled. Valid values: `true` | `false`.
WLS_SSL_HOST_VERIFICATION	`true`	Whether the WLS SSL host verfication is enabled or not. Valid values: `true | false`.
WLS_TRUSTSTORE	`/u01/oracle/config/keystores/idmTrustStore.p12`	The location of the WLS truststore on shared storage.
LOCAL_WLS_TRUSTSTORE	`/u02/oracle/config/keystores/idmTrustStore.p12`	The location of the WLS truststore on local storage.
WLS_TRUSTSTORE_PASSWORD	`password`	Password for the WLS truststore.

Setting Environment Variables

Set environment variables used in this chapter.

To help navigate this guide, to be able to copy sample commands without modification you can set the following environment variables, replacing the values with values appropriate to your environment.

```
export IGD_ORACLE_HOME=/u01/oracle/products/oig
export ORACLE_HOME=$IGD_ORACLE_HOME
export ORACLE_COMMON_HOME=$IGD_ORACLE_HOME/oracle_common
export JAVA_HOME=/u01/oracle/products/jdk
export PATH=$JAVA_HOME/bin:$PATH
export IGD_ASERVER_HOME=/u01/oracle/config/domains/oig
export DOMAIN_HOME=$IGD_ASERVER_HOME
export IGD_MSERVER_HOME=/u02/oracle/config/domains/oig
export WL_HOME=$IGD_ORACLE_HOME/wlserver
export NM_HOME=/u02/oracle/config/nodemanager
export APPLICATION_HOME=/u01/oracle/config/applications/oig
export MS_APPLICATION_HOME=/u02/oracle/config/applications/oig
export DB_HOST=db-scan.example.com
export DB_PORT=1521
export DB_SERVICE=oigsvc.example.com
export SHARED_CONFIG_DIR=/u01/oracle/config
export LOCAL_CONFIG_DIR=/u02/oracle/config
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

URLs Used in This Chapter

This section describes the URLs used in this chapter.

Table 15-3 SSL Termination

Function	Component URL	Load Balancer URL
Remote Console connection	*http://igdadminvhn.example.com:7001/*	*http://igdadmin.example.com*
Enterprise Manager	*http://igdadminvhn.example.com:7001/em*	*http://igdadmin.example.com/em*
OIG System Administration console	*http://igdadminvhn.example.com:14000/sysadmin*	*http://igdadmin.example.com/sysadmin*
Identity Console	*http://igdadminvhn.example.com:14000/identity*	*https://oig.example.com/identity*

Table 15-4 End to End SSL Termination

Function	Component URL	Load Balancer URL
Remote Console connection	*https://igdadminvhn.example.com:9002/*	*https://igdadmin.example.com*
Enterprise Manager	*https://igdadminvhn.example.com:9002/em*	*https://igdadmin.example.com/em*
OIG System Administration console	*https://igdadminvhn.example.com:14001/sysadmin*	*https://igdadmin.example.com/sysadmin*
Identity Console	*https://igdadminvhn.example.com:14001/identity*	*https://oig.example.com/identity*

Installing the Oracle Fusion Middleware

Install the Oracle Fusion Middleware software in preparation for configuring a new domain for Oracle Identity Governance.

- Installing the Oracle Fusion Middleware Infrastructure, SOA Suite, and Oracle Identity Governance on OIGHOST1
  Use the following sections to install the Oracle Fusion Middleware Infrastructure software in preparation for configuring a new domain for an enterprise deployment.

Installing the Oracle Fusion Middleware Infrastructure, SOA Suite, and Oracle Identity Governance on OIGHOST1

Use the following sections to install the Oracle Fusion Middleware Infrastructure software in preparation for configuring a new domain for an enterprise deployment.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Installing a Supported JDK
- Starting the Oracle Identity Management Quick Installer
- Navigating the Installation Screens
- Installing the Stack Bundle Patch
- Installing Oracle Fusion Middleware Infrastructure on the Other Host Computers

Installing a Supported JDK

Oracle Fusion Middleware requires that a certified Java Development Kit (JDK) is installed on your system.

- Locating and Downloading the JDK Software
- Installing the JDK Software
  Oracle Fusion Middleware requires you to install a certified Java Development Kit (JDK) on your system.

Locating and Downloading the JDK Software

To find a certified JDK, see the certification document for your release on the Oracle Fusion Middleware Supported System Configurations page.

After you identify the Oracle JDK for the current Oracle Fusion Middleware release, you can download an Oracle JDK from the following location on Oracle Technology Network:

```
https://www.oracle.com/java/technologies/downloads/
```

Be sure to navigate to the download for the Java SE JDK.

Installing the JDK Software

Oracle Fusion Middleware requires you to install a certified Java Development Kit (JDK) on your system.

For more information about the recommended location for the JDK software, see Understanding the Recommended Directory Structure for an Enterprise Deployment.

To install JDK 21.0:

1. Change directory to the location where you downloaded the JDK archive file.

   ```
   cd download_dir
   ```

2. Unpack the archive into the JDK home directory, and then run the following commands:

   ```
   tar -xzvf jdk-21.0.4+8_linux-x64_bin.tar.gz
   ```

   Note that the JDK version listed here was accurate at the time this document was published. For the latest supported JDK, see the *Oracle Fusion Middleware System Requirements and Specifications* for the current Oracle Fusion Middleware release.

3. Move the JDK directory to the recommended location in the directory structure.

   For example:

   ```
   ln -s jdk-21.0.4 /u01/oracle/products/jdk
   ```

   See File System and Directory Variables Used in This Guide.

Copyright © 2018, 2025, Oracle and/or its affiliates.

4. Run the following command to verify that the appropriate `java` executable is in the path and your environment variables are set correctly:

   ```
   java -version
   ```

   The Java version in the output should be displayed as “21.0.4”.

Starting the Oracle Identity Management Quick Installer

The Quick Installer installs the Oracle Fusion Middleware Infrastructure, Oracle SOA Suite, and Oracle Identity and Access Management.

To start the installation program, perform the following steps.

1. Log in to OIGHOST1.
2. Go to the directory where you downloaded the installation program.
3. Launch the installation program by invoking the `java` executable from the JDK directory on your system, as shown in the example below.

   ```
   $JAVA_HOME/bin/java  -jar fmw_14.1.2.1.0_idmquickstart_generic.jar
   ```

   In this example:

   - Replace *`distribution_file_name`* with the actual name of the distribution JAR file.

     If you download the distribution from the Oracle Technology Network (OTN), then the JAR file is typically packaged inside a downloadable ZIP file.

     To install the software required for the initial Infrastructure domain, the distribution you want to install is:

     fmw_14.1.2.1.0_idmquickstart_generic.jar

     For more information about the actual file names of each distribution, see Identifying and Obtaining Software Downloads for an Enterprise Deployment.

When the installation program appears, you are ready to begin the installation. See Navigating the Installation Screens for a description of each installation program screen.

Navigating the Installation Screens

The installation program displays a series of screens, in the order listed in the following table.

If you need additional help with any of the installation screens, click the screen name or click the **Help** button on the screen.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-5 Navigating the Infrastructure Installation Screens

Screen	Description
Installation Inventory Setup	On UNIX operating systems, this screen appears if you are installing any Oracle product on this host for the first time. Specify the location where you want to create your central inventory. Ensure that the operating system group name selected on this screen has write permissions to the central inventory location. See Understanding the Oracle Central Inventory in *Installing Software with the Oracle Universal Installer*. **Note** Oracle recommends that you configure the central inventory directory on the products shared volume. Example: `/u01/oracle/products/oraInventory` You may also need to execute the `createCentralinventory.sh` script as root from the `oraInventory` folder after the installer completes.
Welcome	This screen introduces you to the product installer.
Auto Updates	Use this screen to search My Oracle Support automatically for available patches or automatically search a local directory for patches that you have already downloaded for your organization.
Installation Location	Use this screen to specify the location of your Oracle home directory. For the purposes of an enterprise deployment, enter the value of the $ORACLE_HOME variable for the product listed in Table 8-2. For example, `/u01/oracle/products/oig`
Prerequisite Checks	This screen verifies that your system meets the minimum requirements. If there are any warning or error messages, refer to the Oracle Fusion Middleware System Requirements and Specifications document on the Oracle Technology Network (OTN).
Installation Summary	Use this screen to verify the installation options that you have selected. If you want to save these options to a response file, click **Save Response File** and provide the location and name of the response file. Response files can be used later in a silent installation situation. For more information about silent or command-line installation, see Using the Oracle Universal Installer in Silent Mode in *Installing Software with the Oracle Universal Installer*.
Installation Progress	This screen allows you to see the progress of the installation.
Installation Complete	This screen appears when the installation is complete. Review the information on this screen, then click **Finish** to dismiss the installer.

Installing the Stack Bundle Patch

After installing the software binaries, you must apply the latest Stack Bundle Patch.

For an enterprise deployment you must download and apply the July 2025 Stack Bundle Patch or later. For more details, see Identifying and Obtaining Software Distributions for an Enterprise Deployment.

Copyright © 2018, 2025, Oracle and/or its affiliates.

To apply the patch run the following commands:

1. After downloading the patch, unzip it to your preferred location. For example:

   ```
   unzip p38184742_141210_Linux-x86-64.zip
   ```

   This location will be known as $PATCH_DIR.

2. Navigate to the $PATCH_DIR:

   ```
   cd $PATCH_DIR/tools/spbat/generic/SPBAT/
   ```

3. Apply the patch using the command:

   ```
   ./spbat.sh -type oig -phase downtime -mw_home $IGD_ORACLE_HOME -
   spb_download_dir $PATCH_DIR
   ```

Installing Oracle Fusion Middleware Infrastructure on the Other Host Computers

If you have configured a separate shared storage volume or partition for secondary hosts, then you must install the Infrastructure on one of those hosts.

See Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment.

To install the software on the other host computers in the topology, log in to each host, and use the instructions in Starting the Oracle Identity Management Quick Installer and Navigating the Installation Screens to create the Oracle home on the appropriate storage device.

You must install the Stack Bundle Patch and any other mandatory patches outlined in Identifying and Obtaining Software Distributions for an Enterprise Deployment.

Verifying the Installation

After you complete the installation, you nust verify it.

Perform the following tasks:

- Reviewing the Installation Log Files
- Checking the Directory Structure
- Viewing the Contents of Your Oracle Home

Reviewing the Installation Log Files

Review the contents of the installation log files to make sure that no problems were encountered. For a description of the log files and where to find them, see Understanding Installation Log Files in *Installing Software with the Oracle Universal Installer*.

Checking the Directory Structure

The contents of your installation vary based on the options that you select during the installation.

The addition of Oracle Identity Governance adds the following directory and sub-directories. Use the following command to verify the directory structure:

Copyright © 2018, 2025, Oracle and/or its affiliates.

1. Navigate to the $ORACLE_HOME:

   ```
   cd $ORACLE_HOME
   ```

2. Enter the following command:

   ```
   ls --format=single-column
   ```

   The directory structure on your system must match the structure shown in the following example:

   ```
   bin
   cfgtoollogs
   coherence
   domain-registry.xml
   em
   envPropertiesCache
   idm
   install
   inventory
   jdeveloper
   jlib
   lib
   mft
   OPatch
   opmn
   oracle_common
   oraInst.loc
   osb
   oui
   root.sh
   soa
   wlserver
   ```

For more information about the directory structure you should see after installation, see What are the Key Oracle Fusion Middleware Directories? in *Understanding Oracle Fusion Middleware*.

Viewing the Contents of Your Oracle Home

You can also view the contents of your Oracle home by using the `viewInventory` script. See Viewing the contents of an Oracle home in *Installing Software with the Oracle Universal Installer*.

Downloading the Oracle Connector Bundle

Download the Oracle Connector bundle.

Download the Oracle Connector bundle from the following location:

http://www.oracle.com/technetwork/middleware/id-mgmt/downloads/connectors-101674.html

Copyright © 2018, 2025, Oracle and/or its affiliates.

Copy the connector bundle for Oracle Internet Directory (it covers OUD as well) to the following directory:

```
$IGD_ORACLE_HOME/idm/server/ConnectorDefaultDirectory
```

Installing the Oracle Identity Governance Connector

After you download the Oracle Connector for LDAP, install it into the ORACLE_HOME directory.

To do this perform the following steps:

1. Go to the following directory:

   ```
   cd $IGD_ORACLE_HOME/idm/server/ConnectorDefaultDirectory
   ```

2. Unzip the LDAP directory using the following command:

   ```
   unzip oid_<version>.zip
   ```

Creating the Database Schemas for Oracle Identity Governance

Oracle Fusion Middleware components require the existence of schemas in a database before you configure a Fusion Middleware Infrastructure domain. Install the schemas listed in this topic in a certified database for use with this release of Oracle Fusion Middleware.

- Oracle Identity Manager

  This automatically selects Oracle SOA Suite schemas along with the following ones:

  - Metadata Services (MDS)
  - Audit Services (IAU)
  - Audit Services Append (IAU_APPEND)
  - Audit Services Viewer (IAU_VIEWER)
  - Oracle Platform Security Services (OPSS)
  - User Messaging Service (UMS)
  - WebLogic Services (WLS)
  - Common Infrastructure Services (STB)

Use the Repository Creation Utility (RCU) to create the schemas. This utility is installed in the Oracle home for each Oracle Fusion Middleware product. For more information about RCU and how the schemas are created and stored in the database, see Preparing for Schema Creation in *Creating Schemas with the Repository Creation Utility*.

Complete the following steps to install the required schemas:

- Installing and Configuring a Certified Database
- Verifying Schema Access
- Creating the Database Schemas Using GUI
- Creating the Database Schemas Using CLI

Copyright © 2018, 2025, Oracle and/or its affiliates.

Installing and Configuring a Certified Database

Make sure that you have installed and configured a certified database, and that the database is up and running.

See the Preparing the Database for an Enterprise Deployment.

Verifying Schema Access

Verify schema access by connecting to the database as the new schema users created by the RCU. Use SQL*Plus or another utility to connect, and provide the appropriate schema names and passwords entered in the RCU.

For example:

```
sqlplus <RCU_PREFIX>_OIM/<PASSWORD>@//<SCAN_ADDRESS>:<PORT>/<SERVICE_NAME>
```

For example:

```
sqplus IGDEDG_OIM/<password>@//db-scan.example.com:1521/oigpdb_s.example.com
```

The output appears as follows:

```
SQL*Plus: Release 23.0.0.0.0 - for Oracle Cloud and Engineered Systems on Wed
Sep 11 14:20:00 2024 Version 23.5.0.24.07
Copyright (c) 1982, 2024, Oracle. All rights reserved.


Connected to:
Oracle Database 23ai EE Extreme Perf Release 23.0.0.0.0 - for Oracle Cloud
and Engineered Systems
Version 23.5.0.24.07

SQL>
```

Creating the Database Schemas Using GUI

- Starting the Repository Creation Utility (RCU)
- Navigating the RCU Screens to Create the Schemas

Starting the Repository Creation Utility (RCU)

To start the Repository Creation Utility (RCU):

1. Navigate to the following directory on OIGHOST1:

   ```
   cd $ORACLE_COMMON_HOME/bin
   ```

2. Start RCU:

   ```
   ./rcu
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> If your database has Transparent Data Encryption (TDE) enabled, and you want to encrypt your tablespaces created by the RCU, provide the `-encryptTablespace true` option when you start the RCU.
>
> This will default the appropriate RCU GUI Encrypt Tablespace checkbox selection on the Map Tablespaces screen without further effort during the RCU execution. See Encrypting Tablespaces in *Creating Schemas with the Repository Creation Utility*.

Navigating the RCU Screens to Create the Schemas

Schema creation involves the following tasks:

Screen	Description
Introducing RCU	Review the Welcome screen and verify the version number for RCU. Click **Next** to begin.
Selecting a Method of Schema Creation	If you have the necessary permission and privileges to perform DBA activities on your database, select **System Load and Product Load** on the Create Repository screen. The procedure in this document assumes that you have the necessary privileges. If you do not have the necessary permission or privileges to perform DBA activities in the database, you must select **Prepare Scripts for System Load** on this screen. This option will generate a SQL script, which can be provided to your database administrator. See Understanding System Load and Product Load in *Creating Schemas with the Repository Creation Utility*. Click **Next**. **Tip** For more information about the options on this screen, see Create repository in *Creating Schemas with the Repository Creation Utility*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Screen	Description
Providing Database Connection Details	Provide the database connection details for RCU to connect to your database. 1. As **Database Type**, select **Oracle Database** enabled for edition based redefinition. 2. In the **Host Name** field, enter the SCAN address of the Oracle RAC Database. 3. Enter the **Port** number of the RAC database scan listener, for example 1521. 4. Enter the RAC **Service Name** of the database. 5. Enter the **User Name** of a user that has permissions to create schemas and schema objects, for example SYS. 6. Enter the **Password** of the user name that you provided in step 5 7. If you have selected the SYS user, ensure that you set the role to SYSDBA. 8. Click **Next** to proceed, then click **OK** on the dialog window confirming that connection to the database was successful. **Tip** For more information about the options on this screen, see Database Connection Details in *Creating Schemas with the Repository Creation Utility*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Screen	Description
Specifying a Custom Prefix and Selecting Schemas	1. Specify the custom prefix you want to use to identify the Oracle Fusion Middleware schemas. The custom prefix is used to logically group these schemas together for use in this domain. For Oracle Identity Governance, use the prefix `IGD`. **Tip** Make a note of the custom prefix you choose to enter here; you will need this later, during the domain creation process. For more information about custom prefixes, see Understanding Custom Prefixes in *Creating Schemas with the Repository Creation Utility*. 2. Expand the group IDM Schemas, and then select the **Oracle Identity Manager** schema. All the relative schemas will be selected: • **Common infrastructure Services** • **Oracle Platform Security Services** • **User Messaging Service** • **Audit Services** • **Audit Services Append** • **Audit Services Viewer** • **Metadata Services** • **SOA Infrastructure** • **Weblogic Services** There are two mandatory schemas that are selected by default. You cannot deselect them: **Common Infrastructure Services** (the STB schema) and **WebLogic Services** (the WLS schema). The **Common Infrastructure Services** schema enables you to retrieve information from RCU during domain configuration. See Understanding the Service Table Schema in *Creating Schemas with the Repository Creation Utility*. **Tip** For more information about how to organize your schemas in a multi-domain environment, see Planning Your Schema Creation in *Creating Schemas with the Repository Creation Utility*. Click **Next** to proceed, then click **OK** on the dialog window confirming that prerequisite checking for schema creation was successful.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Screen	Description
Specifying Schema Passwords	Specify how you want to set the schema passwords on your database, then specify and confirm your passwords. Ensure that the complexity of the passwords meet the database security requirements before you continue. RCU will proceed at this point even if you do not meet the password polices. Hence, perform this check outside RCU itself.. Click **Next**. **Tip** You must make a note of the passwords you set on this screen; you will need them later on during the domain creation process.
Verifying the Tablespaces for the Required Schemas	You can accept the default settings on the remaining screens, or you can customize how RCU creates and uses the required tablespaces for the Oracle Fusion Middleware schemas. **Note** You can configure a Fusion Middleware component to use JDBC stores for JMS servers and Transaction Logs, by using the Configuration Wizard. These JDBC stores are placed in the WebLogic Services component tablespace. If your environment expects to have a high level of transactions and/or JMS activity, you can increase the default size of the <*PREFIX*>_WLS tablespace to better suit the environment load. Click **Next** to continue, and then click **OK** on the dialog window to confirm the tablespace creation. For more information about RCU and its features and concepts, see About the Repository Creation Utility in *Creating Schemas with the Repository Creation Utility*.
Creating Schemas	Review the summary of the schemas to be loaded and click **Create** to complete schema creation. **Note** If failures occurred, review the listed log files to identify the root cause, resolve the defects, and then use RCU to drop and re-create the schemas before you continue.
Reviewing Completion Summary and Completing RCU Execution	When you reach the Completion Summary screen, verify that all schema creations have been completed successfully, and then click **Close** to dismiss RCU.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Creating the Database Schemas Using CLI

Run the following commands to create the database schemas using the Repository Creation Assistant CLI in silent mode:

1. Create a password file `pwd.txt` that contains the the password for the database sysdba account and password assigned for the database schemas. This file should contain two lines as shown in the following example:

   ```
   sysdba_password
   schema_password
   ```

2. Run the following command to execute the RCU in silent mode:

   ```
   cd $ORACLE_COMMON_HOME/bin
   ```

   ```
   ./rcu -silent -createRepository -databaseType ORACLE -
   connectString $DB_HOST:$DB_PORT/$DB_SERVICE -dbUser sys -dbRole sysdba -
   selectDependentsForComponents true -useSamePasswordForAllSchemaUsers true -
   schemaPrefix $RCU_PREFIX --component MDS -component IAU -component
   SOAINFRA -component IAU_APPEND -component IAU_VIEWER -component OPSS -
   component WLS -component STB -component OIM -component UCSUMS -f < /pwd.txt
   ```

Configuring the Oracle Identity Governance Domain

The following topics provide instructions for creating an Oracle Identity Governance domain using the Fusion Middleware Configuration wizard.

For more information on the other methods that are available for creating a domain, see Additional Tools for Creating, Extending, and Managing WebLogic Domains in *Creating WebLogic Domains Using the Configuration Wizard*.

- Starting the Configuration Wizard
- Navigating the Configuration Wizard Screens to Configure the Oracle Identity Governance Domain

Starting the Configuration Wizard

To begin domain configuration, run the following command in the Oracle Fusion Middleware Oracle home.

```
$ORACLE_COMMON_HOME/common/bin/config.sh
```

Navigating the Configuration Wizard Screens to Configure the Oracle Identity Governance Domain

Follow the instructions in the following sections to create and configure the domain for the topology, with static clusters.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

Screen	Description
Selecting the Domain Type and Domain Home Location	On the Configuration Type screen, select **Create a new domain**. In the **Domain Location** field, specify the value of the *IGD_ASERVER_HOME* variable, as defined in File System and Directory Variables Used in This Guide. For example, `/u01/oracle/products/oig` for Oracle Identity Governance. **Tip** More information about the other options on this screen of the Configuration Wizard, see Configuration Type in *Creating WebLogic Domains Using the Configuration Wizard*. Click **Next**.
Selecting the Configuration Templates	On the Templates screen, make sure **Create Domain Using Product Templates** is selected, then select the following templates: • **Oracle Identity Manager - [idm]** • Selecting this template automatically selects the following dependencies: – Oracle Enterprise Manager - [em] – Oracle WSM Policy Manager - [wsm] – Oracle JRF - [oracle_common] – WebLogic Coherence Cluster Extension - [wlserver] – Oracle SOA Suite - [soa] **Tip** More information about the options on this screen can be found in Templates in *Creating WebLogic Domains Using the Configuration Wizard*. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

Screen	Description
Configuring High Availability Options	This screen appears for the first time when you create a cluster that uses Automatic Service Migration or JDBC stores or both. After you select HA Options for a cluster, all subsequent clusters that are added to the domain by using the Configuration Wizard, automatically apply HA options (that is, the Configuration Wizard creates the JDBC stores and configures ASM for them). On the High Availability Options screen: • Select **Enable Automatic Service Migration** with **Database Leasing**. • Set **JTA Transaction Log Persistence** to **JDBC TLog Store**. • Set **JMS Server Persistence** to **JMS JDBC Store**. • Click **Next**. **Note** Oracle recommends that you use JDBC stores, which leverage the consistency, data protection, and high availability features of an oracle database and makes resources available for all the servers in the cluster. So, the Configuration Wizard steps assume that the JDBC persistent stores are used along with Automatic Service Migration. When you choose JDBC persistent stores, additional unused File Stores are automatically created but are not targeted to your clusters. Ignore these File Stores.
Selecting the Application Home Location	On the Application Location screen, specify the value of the *APPLICATION_HOME* variable, as defined in File System and Directory Variables Used in This Guide. For example, `/u01/oracle/config/applications/oig`. **Tip** More information about the options on this screen can be found in Application Location in *Creating WebLogic Domains Using the Configuration Wizard*. Click **Next**.
Configuring the Administrator Account	On the Administrator Account screen, specify the user name (Oracle recommends using a different name from “WebLogic”) and password for the default WebLogic Administrator account for the domain. Make a note of the user name and password specified on this screen; you will need these credentials later to boot and connect to the domain's Administration Server. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

Screen	Description
Specifying the Domain Mode and JDK	On the Domain Mode and JDK screen: • Select **Production** in the Domain Mode field. In the Enable or Disable Default Ports for your Domain field, use the default values provided for Production Mode: **SSL Termination** If you are using an SSL Terminated deployment, the following values must be selected: – Ensure that **Enable Listen Ports (non-SSL Ports)** is checked. – Ensure that **Enable SSL Listen Ports** is not checked. – Ensure that **Enable Administration Port (SSL Port)** is checked. **End to End SSL Deployment** In the Enable or Disable Default Ports for your Domain field, use the following default values provided for Production Mode: – Ensure that **Enable SSL Listen Ports (non-SSL Ports)** is not checked. – Ensure that **Enable SSL Listen Ports** is checked. – Ensure that **Enable Administration Port (SSL Port)** is checked. **Tip** More information about the options on this screen, including the differences between development mode and production mode can be found in Domain Mode and JDK in *Creating WebLogic Domains Using the Configuration Wizard.* When you start the Administration Server, a boot identity file can be created to bypass the need to provide a username and password in the production mode. • Select the **Oracle Hotspot** JDK in the JDK field. Selecting **Production Mode** on this screen gives your environment a higher degree of security, requiring a user name and password to deploy applications and to start the Administration Server. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

Screen	Description
Specifying the Database Configuration Type	On the Database Configuration Type screen: • Select **RCU Data** to activate the fields on this screen. The **RCU Data** option instructs the Configuration Wizard to connect to the database and Service Table (STB) schema to automatically retrieve schema information for the schemas needed to configure the domain. • Verify that **Vendor** is `Oracle` and **Driver** is `*Oracle's Driver (Thin) for Service Connections; Versions: Any`. • Verify that **Connection Parameters** is selected. **Note** If you choose to select **Manual Configuration** on this screen, you will have to manually fill in the parameters for your schema on the JDBC Component Schema screen. After you select **RCU Data**, fill in the fields as shown below:

Field	Description
Host Name	Enter the Single Client Access Name (SC Oracle RAC database, which you entered *Deployment Workbook*. For information about the Enterprise Dep Using the Enterprise Deployment Workbo
DBMS/Service	Enter the service name for the Oracle RA this domain where you will install the proc example: `iamedg.example.com` Specify the service name based on the va the Preparing the Database for an Enterp
Port	Enter the port number on which the datab `1521`.
Schema Owner	Enter the user name and password for co Service Table schema.
Schema Password	This is the schema user name and passw the Service Table component on the "Sch RCU (see Creating the Database Schema The default user name is *prefix*_STB, custom prefix that you defined in RCU.

Click **Get RCU Configuration** when you are finished specifying the database connection information. The following output in the Connection Result Log indicates that the operating succeeded:

```
Connecting to the database server...OK
Retrieving schema data from database server...OK
Binding local schema components with retrieved data...OK

Successfully Done.
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

Screen	Description
	Click **Next** if the connection to the database is successful. **Tip** More information about the **RCU Data** option can be found in Understanding the Service Table Schema in *Creating Schemas with the Repository Creation Utility*. More information about the other options on this screen can be found in Datasource Defaults in *Creating WebLogic Domains Using the Configuration Wizard*.
Specifying JDBC Component Schema Information	Verify that the values on the JDBC Component Schema screen are correct for all schemas. The schema table should be populated, because you selected **Get RCU Data** on the previous screen. As a result, the Configuration Wizard locates the database connection values for all the schemas required for this domain. At this point, the values are configured to connect to a single-instance database. However, for an enterprise deployment, you should use a highly available Real Application Clusters (RAC) database, as described in Preparing the Database for an Enterprise Deployment. In addition, Oracle recommends that you use an Active GridLink datasource for each of the component schemas. For more information about the advantages of using GridLink data sources to connect to a RAC database, see Database Considerations in the *High Availability Guide*. To convert the data sources to GridLink: 1. Select all the schemas by selecting the checkbox at in the first header row of the schema table. 2. Click **Convert to GridLink** and click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

<table>
<tr><th>Screen</th><th>Description</th></tr>
<tr><td>Providing the GridLink Oracle RAC Database Connection Details</td><td>On the GridLink Oracle RAC Component Schema screen, provide the information required to connect to the RAC database and component schemas, as shown below:

<table>
<tr><th>Element</th><th>Description and Recommended Value</th></tr>
<tr><td>Service Name</td><td>Verify that the service name for the Oracl appropriate.
For example,<code>oigsvc.example.com</code>.</td></tr>
<tr><td>SCAN, Host Name, and Port</td><td>Select the SCAN check box.
In the Host Name field, enter the Single C (SCAN) Address for the Oracle RAC data
In the Port field, enter the SCAN listening example, 1521)</td></tr>
<tr><td>ONS Host and Port</td><td>Leave blank.</td></tr>
<tr><td>Enable Fan</td><td>Verify that the Enable Fan check box is s can receive and process FAN events.</td></tr>
</table>

For more information about specifying the information on this screen, as well as information about how to identify the correct SCAN address, see Configuring Active GridLink Data Sources with Oracle RAC in the High Availability Guide.
You can also click Help to display a brief description of each field on the screen.
Click Next.</td></tr>
<tr><td>Testing the JDBC Connections</td><td>Use the JDBC Component Schema Test screen to test the data source connections you have just configured.
A green check mark in the Status column indicates a successful test. If you encounter any issues, see the error message in the Connection Result Log section of the screen, fix the problem, then try to test the connection again.

Tip
More information about the other options on this screen can be found in Test Component Schema in Creating WebLogic Domains Using the Configuration Wizard

Click Next.</td></tr>
<tr><td>Entering Credentials</td><td>Enter the credentials you wish to use for the Oracle Identity Governance components. You have the choice of choosing both a username and a password for the various objects.
• keystore: Set the username to "keystore" and the password to the password you wish to use for all automatically created keystores.
• OIMSchemaPassword: Set the username to the OIG schema which you created in the earlier sections. For example, <code>IGD_OIM</code> (username) and its associated password.
• Sysadmin: This is the administrative user you will use for OIG. This is typically <code>xelsysadm</code>, but can be anything. Set the password to a value you wish to use for this account.
• WebLogicAdminKey: This is the domain admin username and password. For example, <code>weblogic</code>.
Click Next.</td></tr>
</table>

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

Screen	Description
Keystore	Use this screen to specify details about the keystore to be used in the domain. For a typical enterprise deployment, you can leave the default values. See Keystore in *Creating WebLogic Domains Using the Configuration Wizard*. Click **Next**.
Selecting Advanced Configuration	To complete domain configuration for the topology, select the following options on the Advanced Configuration screen: • **Administration Server** This is required to properly configure the listen address of the Administration Server. • **Node Manager** This is required to configure Node Manager. • **Topology** This is required to add, delete, or modify the Settings for Server Templates, Managed Servers, Clusters, Virtual Targets, and Coherence. **Note** When using the Advanced Configuration screen in the Configuration Wizard, if any of the above options are not available on the screen, then return to the Templates screen, and be sure you selected the required templates for this topology. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

Screen	Description
Configuring the Administration Server Listen Address	On the Administration Server screen: 1. In the **Server Name** field, retain the default value - AdminServer. 2. In the **Listen Address** field, enter the virtual host name that corresponds to the VIP of the IGDADMINVHN that you procured in Procuring Resources for an Enterprise Deployment and enabled in Preparing the Host Computers for an Enterprise Deployment. For more information on the reasons for using the IGDADMINVHN virtual host, see Reserving the Required IP Addresses for an Enterprise Deployment. 3. In the Configure Administration Server Ports section, perform the following steps: **SSL Terminated Deployment** a. Set the value of the **listen port** to 7101. b. Ensure the **Enable SSL Listen port** field is unchecked. c. Set the value of **Administration Port** to 9102. **End to End SSL Deployment** a. Leave the **Enable Listen Port** field unchecked. The **Listen Port** value will be disabled in grey. b. Ensure the **Enable SSL Listen port** field is checked. c. Set the value of **SSL Listen Port** to 7102 d. Set the value **Administration Port** to 9102. **Note** The default values are not used to ensure that there is no port conflict if Oracle Access Manager and Oracle Identity Governance run on the same host . 4. In the **Listen Port** field, enter the port number to access the administration server. This guide recommends you to use the default port `7101` for Oracle Identity Governance. Leave the other fields at their default values. In particular, be sure that no server groups are assigned to the Administration Server. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

Screen	Description
Configuring Node Manager	Select **Manual Node Manager Setup** as the Node Manager type. **Warning** You can ignore the warning in the bottom pane. This guide provides the required steps for the Manual Node Manager configuration. **Tip** For more information about the options on this screen, see Node Manager in *Creating WebLogic Domains Using the Configuration Wizard*. For more information about per domain and per host Node Manager implementations, see About the Node Manager Configuration in a Typical Enterprise Deployment. For information about Node Manager configurations, see Configuring Node Manager on Multiple Machines in *Administering Node Manager for Oracle WebLogic Server*. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

Screen	Description
Configuring Managed Servers	Use the Managed Servers screen to create two new Managed Servers: 1. Update the existing managed server oim_server1 and set the **Listen address** to oighost1.example.com. 2. Update the existing managed server soa_server1 and set the **Listen address** to oighost1.example.com. 3. Click the **Add** button to create a new Managed Server. 4. Specify `oim_server2` in the **Server name** column. 5. In the **SSL Listen Port** column, enter the same port used by oim_server1. For example, 14001. Be sure to enter the host name that corresponds to oighost2.example.com; do not use the IP address. 6. In the **Listen Port** column, enter the same port used by oim_server1. For example, 14000. 7. In the **Server Groups** drop down, ensure the same server groups as oim_server1 are selected. 8. Repeat this process for `soa_server2`. The following table summarises the managed servers to be created.

Table 15-7 SSL Terminated Deployments

Server Name	Listen Address	Enable Listen Port	Listen Port	Enable SSL	SSL Listen Port	Server Groups
oim_server1	oighost1.example.com	Yes	14000	No	14001	OIM-MGD-SVRS
oim_server2	oighost2.example.com	Yes	14000	No	14001	OIM-MGD-SVRS
soa_server1	oighost1.example.com	Yes	7003	No	7004	SOA-MGD-SVRS
soa_server2	oighost2.example.com	Yes	7003	No	7004	SOA-MGD-SVRS

Table 15-8 End to End SSL Deployments

Server Name	Listen Address	Enable Listen Port	Listen Port	Enable SSL	SSL Listen Port	Server Groups
oim_server1	oighost1.example.com	No	14000	Yes	14001	OIM-MGD-SVRS
oim_server2	oighost2.example.com	No	14000	Yes	14001	OIM-MGD-SVRS
soa_server1	oighost1.example.com	No	7003	Yes	7004	SOA-MGD-SVRS
soa_server2	oighost2.example.com	No	7003	Yes	7004	SOA-MGD-SVRS

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

Screen	Description
	The Managed Server names suggested in this procedure are referenced throughout this document, if you choose different names then ensure to replace them as needed. Click **Next**.
Configuring a Cluster	In this task, you create a cluster for each set of Managed Servers. You can then target the Oracle Identity Governance and Oracle SOA Suite components to the relevant cluster. Create the following clusters: • oim_cluster • soa_cluster Use the Clusters screen to create a new cluster: 1. Click the **Add** button. 2. Specify the cluster name in the **Cluster Name** field. 3. Repeat the steps to create all of the clusters. **Note** By default, server instances in a cluster communicate with one another using unicast. If you want to change your cluster communications to use multicast, refer to Considerations for Choosing Unicast or Multicast in *Administering Clusters for Oracle WebLogic Server*. Click **Next**. For more information about the options on this screen, see Clusters in *Creating WebLogic Domains Using the Configuration Wizard*.
Assigning Server Templates	Click **Next** to proceed to the next screen.
Configuring Dynamic Servers	Verify that all dynamic server options are disabled for clusters that are to remain as static clusters. 1. Confirm that the **Dynamic Cluster**, **Calculated Listen Port**, and **Calculated Machine Names** checkboxes on this screen are unchecked. 2. Confirm the **Server Template** selection is **Unspecified**. 3. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

<table>
<tr><th>Screen</th><th>Description</th></tr>
<tr><td>Assigning Managed Servers to the Cluster</td><td>Use the Assign Servers to Clusters screen to assign your managed servers to the clusters you have just created. At the end of this you will have the following assignments:

<table><tr><th>Cluster</th><th>Managed Servers</th></tr><tr><td>oim_cluster</td><td>oim_server1
oim_server2</td></tr><tr><td>soa_cluster</td><td>soa_server1
soa_server1</td></tr></table>
1. In the Clusters pane, select the cluster to which you want to assign the servers.
2. In the Servers pane, assign the managed servers to the clusters as in the table above, using one of the following methods:
• Click once on the Managed Server to select it, then click on the right arrow to move it beneath the selected cluster in the Clusters pane.
• Double-click on managed server to move it beneath the selected cluster in the clusters pane.
3. Repeat to assign each managed server to a cluster as shown in the table.
4. Click Next to proceed to the next screen.

Tip
More information about the options on this screen can be found in Assign Servers to Clusters in Creating WebLogic Domains Using the Configuration Wizard.</td></tr>
<tr><td>Configuring Coherence Clusters</td><td>Use the Coherence Clusters screen to configure the Coherence cluster that is automatically added to the domain.
In the Cluster Listen Port, enter 9991.

Note
For Coherence licensing information, Oracle Coherence Products in Oracle Fusion Middleware Licensing Information User Manual.

Click Next.</td></tr>
</table>

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

Screen	Description
Creating Machines	Use the Machines screen to create new machines in the domain. A machine is required in order for the Node Manager to be able to start and stop the servers. You must create a machine even if your topology contains just the Administration Server. 1. Select the **Unix Machine** tab. 2. Click the **Add** button to create new UNIX machines. Use the values in Values to Use When Creating Unix Machines to define the Name and Node Manager Listen Address of each machine. 3. Verify the port in the Node Manager Listen Port field. The port number 5556, shown in this example, may be referenced by other examples in the documentation. Replace this port number with your own port number as needed.

Table 15-9 Values to Use When Creating Unix Machines

Name	Node Manager Listen Address	NNoodeMaanagerType
ADMINHOST	Enter the value of the IGDADMINVHN variable.	S55SL
OIGHOST1	The hostname of the server that will run the managed servers. For example, `oighost1.example.com`.	S55SL
OIGHOST2	The hostname of the server that will run the managed servers. For example, `oighost2.example.com`.	S55SL

> **Note**
>
> If you are installing OIG on the same host as Oracle Access Management (OAM), ensure that the Node Manager ports are unique to each deployment.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

<table>
<tr><th>Screen</th><th>Description</th></tr>
<tr><td></td><td>
Tip
More information about the options on this screen can be found in Machines in Creating WebLogic Domains Using the Configuration Wizard.

Click Next.
</td></tr>
<tr><td>Assigning Servers to Machines</td><td>
Use the Assign Servers to Machines screen to assign the Oracle Identity Governance Managed Servers to the corresponding machines in the domain.
Assign the machines as shown in the following table:

<table>
<tr><th>Servers</th><th>Machines</th></tr>
<tr><td>AdminHost</td><td>AdminServer</td></tr>
<tr><td>oim_server1</td><td>OIGHOST1</td></tr>
<tr><td>soa_server1</td><td></td></tr>
<tr><td>oim_server2</td><td>OIGHOST2</td></tr>
<tr><td>soa_server2</td><td></td></tr>
</table>
1. In the Machines pane, select the machine to which you want to assign the servers.
2. In the Servers pane, assign the Managed Servers to the machine as in the table above, using one of the following methods:
• Click on the Managed Server to select it, and then click on the right arrow to move it beneath the selected machines in the machines pane.
• Double-click on the Managed Server to move it beneath the selected machine in the machines pane.
3. Repeat to assign each of the Managed Server to the respective machine.
4. Click Next.

For more information about the options on this screen, see Assign Servers to Machines in Creating WebLogic Domains Using the Configuration Wizard.
</td></tr>
<tr><td>Creating Virtual Targets</td><td>Click Next.</td></tr>
<tr><td>Creating Partitions</td><td>Click Next.</td></tr>
</table>

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-6 Navigating the Infrastructure Installation Screens

Screen	Description
Configuring Domain Front End Host	In the Domain Front End host screen you specify the main entry point for OIG. This will equate to the name on the load balancer. For example, set **Plain** to `http://oig.example.com`. **Note** Even though you are specifying this value it will never be used. **SSL**: `https://oig.example.com` **Default**: SSL Click **Next**.
Reviewing Your Configuration Specifications and Configuring the Domain	The Configuration Summary screen contains the detailed configuration information for the domain you are about to create. Review the details of each item on the screen and verify that the information is correct. You can go back to any previous screen if you need to make any changes, either by using the **Back** button or by selecting the screen in the navigation pane. Domain creation will not begin until you click **Create**. **Tip** More information about the options on this screen can be found in Configuration Summary in *Creating WebLogic Domains Using the Configuration Wizard*. Click **Next**.
Writing Down Your Domain Home and Administration Server URL	The Configuration Success screen will show the following items about the domain you just configured: • Domain Location • Administration Server URL You must make a note of both items as you will need them later; the domain location is needed to access the scripts used to start the Administration Server. Click **Finish** to dismiss the Configuration Wizard.

Creating Oracle Identity Manager Authenticator

Before you start the domain, you have to run a script which creates the Oracle Identity Manager (OIM) Authenticator in the domain.

To do this, complete the following steps:

1. Navigate to the `$IGD_ORACLE_HOME/idm/server/bin` directory.

```
cd $IGD_ORACLE_HOME/idm/server/bin
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

2. Run the following command:

```
./offlineConfigManager.sh
```

> **Note**
>
> If you do not have execute permissions for this file, add it using the following command:
>
> ```
> chmod 750 offlineConfigManager.sh
> ```

Enabling SSL

If you are configuring End to End SSL, you must perform additional steps.

The steps are as follows:

- Adding Certificate Stores Location to the WebLogic Servers Start Scripts
- Update Server's Security Settings Using the Remote Console

Adding Certificate Stores Location to the WebLogic Servers Start Scripts

Once the Identity and Trust Stores are created for the domain some Java properties must be added to the WebLogic start scripts. These properties are added to the file `setUserOverridesLate.sh` in `$IGD_ASERVER_HOME/bin`. Any customizations you add to this file are preserved during domain upgrade operations and are carried over to remote servers when using the pack and unpack commands.

Manually create the file `setUserOverridesLate.sh` in `$IGD_ASERVER_HOME/bin`. Edit the file and add the variable `EXTRA_JAVA_PROPERTIES` to set the `javax.net.ssl.trustStore` and `javax.net.ssl.trustStorePassword` properties with the values used by your EDG system. For example:

```
EXTRA_JAVA_PROPERTIES="${EXTRA_JAVA_PROPERTIES}
 -Djavax.net.ssl.trustStore=/u01/oracle/config/keystores/idmTruststore.p12
 -Djavax.net.ssl.trustStorePassword=password"
export EXTRA_JAVA_PROPERTIES
```

The order of the extra java properties is relevant. In case that the same property is defined more than once, the later value is used. The custom values must be defined as in the example provided.

Update Server's Security Settings Using the Remote Console

- Connecting to the Remote Console Using the Administration Server’s Virtual Hostname as Provider
- Updating the WebLogic Servers Security Settings

Copyright © 2018, 2025, Oracle and/or its affiliates.

Connecting to the Remote Console Using the Administration Server's Virtual Hostname as Provider

The following procedure temporarily starts the Administration Server with the default start script so to enable you to perform these tasks. After you perform these tasks, you can stop this temporary session and use the Node Manager to start the Administration Server.

> **Note**
>
> For this Remote Console initial access to the Administration Server, it is required that the machine that runs the Remote Console can resolve and connect to the Admin Server's Listen Address. This can be done by starting the Remote Console directly in the node where the Admin Server runs or creating a tunnel to this address from the node where the remote Console is executed.

1. Using the following default start script to start the Administration Server:
   a. Change directory to the following directory:

      ```
      cd $IGD_ASERVER_HOME/bin
      ```

   b. Run the start script:

      ```
      ./startWebLogic.sh
      ```

      Monitor the terminal till the following message is displayed:

      ```
      <Server state changed to RUNNING>
      ```

      Also you must verify that the appropriate SSL listener is available, which can be confirmed with the a message like the following displayed in output:

      ```
      <Server> <BEA-002613> <Channel "DefaultSecure" is now listening on
      XXXX:7002 for protocols iiops, t3s, ldaps, https.>
      ```

2. Create a new provider in the WebLogic Remote Console as follows:
   a. Download the domain's trust keystore to the host or laptop where you run the WebLogic Remote Console. For example, when using the per-domain CA steps in previous sections, this would be located at `$KEYSTORE_HOME/idmTrustStore.p12`.
   b. Open the Remote Console and add the domain trust store to the remote console settings. Click **File** > **Settings** and enter the following values.
      i. Trust Store type - pkcs12
      ii. Trust Store Path - The path to the trust keystore file in the host where the Remote Console runs.
      iii. Trust Store Key - Enter the password provided in the steps above for certificate creation.
      iv. Check **Disable HostName verification** if you are using Demo certificates as described in the steps above.

Copyright © 2018, 2025, Oracle and/or its affiliates.

c. Using the Providers window in the Remote Console, create a new provider by selecting **Add Admin Server Connection Provider**.

   i. In the provider name, for example `oig`.

   ii. Enter the WebLogic Domain Administration username provided in the configuration wizard user name.

   iii. Enter the password used for the domain creation.

   iv. Use https protocol and the admin server listen address used in the configuration wizard as URL for access and specify port 9002.

   For example, `https://igdadminvhn.example.com:9002`.

   v. Check the **Make Insecure Connection** checkbox.

> **Note**
>
> This provider should not be used once the front end and webtier are configured.

The Remote Console Home Window for the domain will be displayed.

Updating the WebLogic Servers Security Settings

Perform the following steps to update the WebLogic Servers Security Settings and Administration Port:

1. Access the Domain provider in the Remote Console and update the Administration Server and WebLogic Servers Security Settings:

   a. Click **Edit Tree**.

   b. Click **Environment** > **Servers** > **AdminServer**.

   c. Click **Security** tab.

   d. Change the keystores dropdown to **Custom Identity and Custom Trust**.

   e. In **Custom Identity Keystore**, enter the fully qualified path to the identity keystore as follows:

   `$KEYSTORE_HOME/idmcerts.pkcs12`
   Replace `$KEYSTORE_HOME` with the value of the folder you use for storing keystore, as described in the Table 8-2.

   f. Set the **Custom Identity Keystore Type** to PKCS12.

   g. In **Custom Identity Keystore Passphrase**, enter the password `Keystore_Password` you provided in the certificate generation steps.

   h. In **Custom Trust Keystore**, enter the fully qualified path to the trust keystore.

   `$KEYSTORE_HOME/idmTrustStore.pkcs12`
   Replace `$KEYSTORE_HOME` with the value of the folder you use for storing keystore, as described in the Table 8-2.

   i. Set the **Custom Trust Keystore Type** to PKCS12.

   j. In **Custom Trust Keystore Passphrase**, enter the password you provided as the <**keypass**> in the certificate generation steps.

   k. Click **Save**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

l. Under **Security** settings, navigate to **SSL** tab.

m. In the **Server Private Key Alias** filed enter the alias provided in the certificate generation steps. If you are using a SAN based certificate use the alias you used to create the certificate. If you are using host based certificates this is usually the hosts short name. For example `oighost1` for `oighost1.example.com`..

n. In the **Server Private Key Pass Phrase** field, enter the password provided in the certificate generation steps.

o. Click **Save**.

The cart on the top right part of the screen will show **full** with a yellow bag inside.

p. Click the Cart icon on the top right and select **Commit Changes**.

Repeat the above steps for each managed server in the domain changing the alias to match the alias used for the certificates.

> **Note**
>
> For managed servers you should use $LOCAL_KEYSTORE_HOME instead of $KEYSTORE_HOME.

2. Return to the terminal window where you started the Administration Server with the start script.

3. Press **Ctrl+C** to stop the Administration Server process.

Wait for the Administration Server process to end and for the terminal command prompt to appear.

4. Start the Administration Server again by using the following script:

a. Change directory to the following directory:

```
cd $IGD_ASERVER_HOME/bin
```

b. Run the start script:

```
./startWebLogic.sh
```

c. Monitor the output in the terminal till the following output is displayed.

```
<Server state changed to RUNNING>
```

Configuring a Per Host Node Manager for an Enterprise Deployment

For specific enterprise deployments, Oracle recommends that you configure a per-host Node Manager, as opposed to the default per-domain Node Manager.

For more information about the advantages of a per host Node Manager, see About the Node Manager Configuration in a Typical Enterprise Deployment

- Creating a Per Host Node Manager Configuration
- Starting the Node Manager on OIGHOST1

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Configuring the Node Manager Credentials
- Enrolling the Domain with NM
- Adding Truststore Configuration to Node Manager

Creating a Per Host Node Manager Configuration

The step in configuring a per-host Node Manager is to create a configuration directory and two new node manager configuration files. You must also edit the default `startNodeManager.sh` file.

To create a per-host Node Manager configuration, perform the following tasks, first on OIGHOST1, and then on OIGHOST2:

1. Log in to OIGHOST1 and create a directory for the Node Manager configuration files :

   For example:

   ```
   mkdir -p /u02/oracle/config/nodemanager
   ```

   Note that this directory should be on a local disk, because it is specific to the host. This directory location is known as the Node Manager home, and it is identified by the *NM_HOME* directory variable in examples in this guide.

2. Change directory to the Node Manager home directory:

   ```
   cd $NM_HOME
   ```

3. If you haven't already, copy the keystores to your local host to ensure that node manager has access to them. For example:

   ```
   mkdir -p $LOCAL_KEYSTORE_HOME
   ```

   ```
   cp -r $KEYSTORE_HOME $LOCAL_KEYSTORE_HOME
   ```

4. Change directory to the Node Manager home directory:

   ```
   cd $NM_HOME
   ```

5. Create a new text file called `nodemanager.properties` and add the values shown in Example: Contents of the nodemanager.properties File to this new file.

   Use the pertaining identity alias for the node that you are configuring. For example, `oighost1.example.com` in OIGHOST1 and `oighost2.example.com` in OIGHOST2.

   For more information about the properties that you can add to the `nodemanager.properties` file, see Node Manager Properties in *Administering Node Manager for Oracle WebLogic Server*.

   In the `nodemanager.properties` file, you enable crash recovery for servers as a part of this configuration. See Node Manager and System Crash Recovery in *Administering Node Manager for Oracle WebLogic Server*.

   Example: Contents of the nodemanager.properties File

   ```
   DomainsFile=/u02/oracle/config/nodemanager/nodemanager.domains
   LogLimit=0
   PropertiesVersion=14.1.2.0.0
   AuthenticationEnabled=true
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
NodeManagerHome=/u02/oracle/config/nodemanager
#Include the specific JDK home
JavaHome=/u01/oracle/products/jdk
LogLevel=INFO
DomainsFileEnabled=true
StartScriptName=startWebLogic.sh
#Leave blank for listening on ANY
ListenAddress=
NativeVersionEnabled=true
ListenPort=5556
LogToStderr=true
SecureListener=true
LogCount=1
StopScriptEnabled=false
QuitEnabled=false
LogAppend=true
StateCheckInterval=500
CrashRecoveryEnabled=true
StartScriptEnabled=true
LogFile=/u02/oracle/config/nodemanager/nodemanager.log
LogFormatter=weblogic.nodemanager.server.LogFormatter
ListenBacklog=50
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityAlias=idmcerts.example.com
CustomIdentityKeyStoreFileName=/u02/oracle/config/keystores/
idmTrustStore.pkcs12
CustomIdentityKeyStorePassPhrase=password
CustomIdentityPrivateKeyPassPhrase=password
```

6. Locate the `startNodeManager.sh` file in the following directory:

   ```
   $WL_HOME/server/bin
   ```

7. Copy the `startNodeManager.sh` file to the Node Manager home directory.

   ```
   cp $WL_HOME/server/bin/startNodeManager.sh $NM_HOME
   ```

8. Edit the new `startNodeManager.sh` file and add the `NODEMGR_HOME` property as follows:

   ```
   NODEMGR_HOME="<NM_HOME>"
   ```

   For example:

   ```
   NODEMGR_HOME="/u02/oracle/config/nodemanager"
   ```

9. Locate the `stopNodeManager.sh` script in the `$WL_HOME/server/bin` directory. Copy it to the Node Manager home directory. Edit the copied file and edit the NODEMGR_HOME property pointing to the node manager home (as it has been done for the `startNodemanager.sh` file):

   ```
   NODEMGR_HOME="<NM_HOME>"
   ```

   In this example, replace *<NM_HOME>* with the actual path to the Node Manager home.

10. Create another new file in the Node Manager home directory, called `nodemanager.domains`.

    The `nodemanager.domains` file provides additional security by restricting Node Manager client access to the domains listed in this file.

Copyright © 2018, 2025, Oracle and/or its affiliates.

11. Perform steps 1 through 8 on OIGHOST2.
12. Add the following entries to the new `nodemanager.domains` files:

    On OIGHOST1, add values for both the Administration Server domain home and the Managed Servers domain home:

    ```
    oig=IGD_MSERVER_HOME;IGD_ASERVER_HOME
    ```

    > **Note**
    >
    > The path that is mentioned first (*IGD_MSERVER_HOME*) is considered as the primaryDomainPath and Managed Servers are run from this location.

    On OIGHOST2, add the value for the Managed Servers domain home only:

    ```
    oig=IGD_MSERVER_HOME
    ```

    In these examples, replace *IGD_ASERVER_HOME* and *IGD_MSERVER_HOME* with the values of the respective variables, as described in File System and Directory Variables Used in This Guide.

Starting the Node Manager on OIGHOST1

After you manually set up the Node Manager to use a per-host Node Manager configuration, you can start the Node Manager on OIGHOST1, by using the `startNodeManager.sh` script.

To start the Node Manager on OIGHOST1:

1. Change directory to the Node Manager home directory:

   ```
   cd $NM_HOME
   ```

2. Run the following command to start the Node Manager and send the output of the command to an output file, rather than to the current terminal shell:

   ```
   nohup ./startNodeManager.sh > ./nodemanager.out 2>&1 &
   ```

3. Monitor the the `nodemanager.out` file; make sure the NodeManager starts successfully. The output should eventually contain the following strings:

   ```
   <INFO> <Upgrade> <Encrypting NodeManager property:
   CustomIdentityKeyStorePassPhrase>
   <INFO> <Upgrade> <Encrypting NodeManager property:
   CustomIdentityPrivateKeyPassPhrase>
   <Upgrade> <Saving upgraded NodeManager properties to '/u02/oracle/config/
   nodemanager/nodemanager.properties'>
   <INFO> <Loading domains file: /u02/oracle/config/nodemanager/
   nodemanager.domains>
   <INFO> <Loading identity key store: FileName=/u02/oracle/config/keystores/
   idmTrustStore.p12, Type=pkcs12, PassPhraseUsed=true>
   <INFO> <Loaded NodeManager configuration properties from '/u02/oracle/
   config/nodemanager/nodemanager.properties'>
   <INFO> <14.1.2.0.0>
   <INFO> <Server Implementation Class:
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
weblogic.nodemanager.server.NMServer$ClassicServer.>
<INFO> <Secure socket listener started on port 5556>
```

You must check that the plain text used for passwords in `nodemanager.properties` has now been encrypted by running the following command:

```
cat /u02/oracle/config/nodemanager/nodemanager.properties
```

The output will look similar to the following:

```
#<DATE>
#<DATE>
DomainsFile=/u02/oracle/config/nodemanager/nodemanager.domains
LogLimit=0
PropertiesVersion=14.1.2.0.0
AuthenticationEnabled=true
NodeManagerHome=/u02/oracle/config/nodemanager
#Include the specific JDK home
JavaHome=/u01/oracle/products/jdk
LogLevel=INFO
DomainsFileEnabled=true
StartScriptName=startWebLogic.sh
#Leave blank for listening on ANY
ListenAddress=
NativeVersionEnabled=true
ListenPort=5556
LogToStderr=true
SecureListener=true
LogCount=1
StopScriptEnabled=false
QuitEnabled=false
LogAppend=true
StateCheckInterval=500
CrashRecoveryEnabled=true
StartScriptEnabled=true
LogFile=/u02/oracle/config/nodemanager/nodemanager.log
LogFormatter=weblogic.nodemanager.server.LogFormatter
ListenBacklog=50
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityAlias=idmcerts.example.com
CustomIdentityKeyStoreFileName=/u02/oracle/config/keystores/
idmTrustStore.p12
CustomIdentityKeyStorePassPhrase={AES256}EMvPrOCRfN7fyv3d8JcEnttTLyneG9Su+U
VK5DGEmqmqDwLkpLz9nQFZ+fL1Bidc
CustomIdentityPrivateKeyPassPhrase={AES256}O5cEJD8WVYP3aRLp9KAbFZ3CGLyxmmIW
FX1YzVfJvPpl1dc5RbMksAcsBLquKcWW
```

Configuring the Node Manager Credentials

Perform the following steps to set the Node Manager credentials using the Remote Console:

1. Access the Domain provider in the Remote Console.
2. Click **Edit Tree**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

3. Click **Environment** > **Domain**> **Security**.
4. Check the **Show Advanced Fields** field.
5. Set **Node Manager Username** to the same as the Weblogic Administrator, since this username will be used in other tasks mentioned in this guide.
6. Change the NM password. Ensure the **Node Manager password** is set to the same as the Weblogic Administrator since this password will be used in other tasks mentioned in this guide.
7. Click **Save**. The cart on the top right part of the screen will show **full** with a yellow bag inside.
8. Click the Cart Icon on the top right and select **Commit Changes**.

Enrolling the Domain with NM

Perform the following steps in a new terminal window to enroll the domain with Node manager.

> **Note**
>
> You will be unable to connect to the Node Manager and use it to start the servers in the domain without performing this step.

1. Change directory to the following directory:

   ```
   cd $ORACLE_COMMON_HOME/common/bin
   ```

2. Start the WebLogic Server Scripting Tool (WLST). In order to use the certificates created for the appropriate SSL handshake, the location of the stores and password of the same need to be provided to WLST. Use the following command or add these in a script that can be easily invoked:

   ```
   export WLST_PROPERTIES="-Dweblogic.security.TrustKeyStore=CustomTrust -
   Dweblogic.security.CustomTrustKeyStoreFileName=/u02/oracle/config/
   keystores/idmTrustStore.p12 -
   Dweblogic.security.CustomTrustKeyStorePassPhrase=password -
   Dweblogic.security.SSL.ignoreHostnameVerification=true"
   ```

   > **Note**
   >
   > You must avoid including the password in the script.

3. Connect to the Administration Server by using the following WLST command:

   ```
   connect('admin_user','admin_password','admin_url')
   ```

   For example:

   ```
   connect('weblogic','<password>','t3s://igdadminvhn.example.com:9002')
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

4. Use the `nmEnroll` command to enable the Node Manager to manage servers in a specified WebLogic domain.

   ```
   nmEnroll('<IGD_ASERVER_HOME>')
   ```

   For example:

   ```
   nmEnroll('/u01/oracle/config/domains/oig')
   ```

5. Generate startup properties for the Admin Server using the following WLST command:

   ```
   nmGenBootStartupProps('AdminServer')
   ```

   The `startup.properties` and `boot.properties` files are created in the following directory:

   ```
   <IGD_ASERVER_HOME>/servers/AdminServer/data/nodemanager/
   ```

Adding Truststore Configuration to Node Manager

It is required to add the corresponding truststore configuration for Node Manager communication with the different WebLogic Server listeners. To do this, edit Node Manager's start script `startNodeManager.sh` located at `$NM_HOME` and add the variable JAVA_OPTIONS to set the `javax.net.ssl.trustStore` and `javax.net.ssl.trustStorePassword` properties with the values used by your EDG system. For example:

```
export JAVA_OPTIONS="${JAVA_OPTIONS} -Djavax.net.ssl.trustStore=/u02/oracle/
config/keystores/idmTrustStore.pkcs12 -
Djavax.net.ssl.trustStorePassword=mypassword"
```

Tuning JMS Queues

You need to tune the JMS queues to ensure maximum throughput.

Perform the following steps from the WebLogic Remote Console to tune the JMS queues:

1. Click **Edit Tree**.
2. Navigate to **Services** > **JMS Servers**.
3. Click **OIMJMSServer**.

   Set the **Message Buffer Size** to 1073741824 (1GB).
4. Click **Save**.
5. Click **Thresholds** tab.

   Set **Messages Maximum** to 1000000.
6. Click **Save**.
7. Click **Shopping Cart** and select **Commit Changes**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Configuring the Domain Directories and Starting the Servers

After you have configured the domain and configured the Node Manager, you can start the Administration Server by using the Node Manager. In an enterprise deployment, the Node Manager is used to start and stop the Administration Server and all the Managed Servers in the domain.

- Create a ServerOverrides File
- Starting the Administration Server Using the Node Manager
  After you have configured the domain and configured the Node Manager, you can start the Administration Server by using the Node Manager. In an enterprise deployment, the Node Manager is used to start and stop the Administration Server and all the Managed Servers in the domain.
- Validating the Administration Server
  Before you proceed with the configuration steps, validate that the Administration Server has started successfully by making sure that you have access to the Oracle Enterprise Manager Fusion Middleware Control; this is installed and configured on the Administration Servers.
- Starting and Validating the soa_server1 Managed Server on OIGHOST1
  After you have configured Node Manager and created the Managed Server domain directory, you can use WebLogic Administration Console to start the soa_server1 Managed Server on OIGHOST1.

Create a ServerOverrides File

The following prerequisites must be completed before starting the Servers:

- Disable the Derby Database - Disable the embedded Derby database, which is a file-based database, packaged with Oracle WebLogic Server. The Derby database is used primarily for development environments. Therefore, you must disable it when you are configuring a production-ready enterprise deployment environment. Otherwise, the Derby database process starts automatically when you start the Managed servers.
- Enable IPv6 Networking if required - If the Managed Server is configured to use IPv6 networking, then you may encounter issues when you start the Managed Server.
- Adjust the Memory Parameters for your installation - The initial startup parameter in the IAMGovernanceDomain, which defines the memory usage, is insufficient. You must increase the value of this parameter and set the Java initial memory allocation pool (Xms) to 1024m, and the maximum memory allocation pool (Xmx) to 8192m.

In order to perform the above tasks, create a `$IGD_ASERVER_HOME/bin/setUserOverrides.sh` file with the following contents:

```
DERBY_FLAG=false
JAVA_OPTIONS="${JAVA_OPTIONS} -Djava.net.preferIPv4Stack=true"
MEM_ARGS="-Xms4096m -Xmx8192m"
```

Starting the Administration Server Using the Node Manager

After you have configured the domain and configured the Node Manager, you can start the Administration Server by using the Node Manager. In an enterprise deployment, the Node

Copyright © 2018, 2025, Oracle and/or its affiliates.

Manager is used to start and stop the Administration Server and all the Managed Servers in the domain.

To start the Administration Server by using the Node Manager:

1. Ensure that the Administration Server is stopped.
2. If your domain is SSL enabled then set the following environment variable so that your keystores are used:

   ```
   export WLST_PROPERTIES="-Dweblogic.security.TrustKeyStore=CustomTrust -
   Dweblogic.security.CustomTrustKeyStoreFileName=/u02/oracle/config/
   keystores/idmTrustStore.p12 -
   Dweblogic.security.CustomTrustKeyStorePassPhrase=password -
   Dweblogic.security.SSL.ignoreHostnameVerification=true"
   ```

3. Start the WebLogic Scripting Tool (WLST):

   ```
   cd $IGD_ORACLE_HOME/oracle_common/common/bin

   ./wlst.sh
   ```

4. Connect to Node Manager by using the Node Manager credentials:

   ```
   nmConnect('nodemanager_username',password','igdadminvhn.example.com','5556'
   ,'domain_name','<IGD_ASERVER_HOME>','SSL')
   ```

   For example:

   ```
   nmConnect('admin','password','igdadminvhn.example.com','5556','oig','/u01/
   oracle/config/domains/oig','SSL')
   ```

   > **Note**
   >
   > This user name and password are used only to authenticate connections between Node Manager and clients. They are independent of the server administrator ID and password and are stored in the `nm_password.properties` file located in the following directory:
   >
   > *$IAD_ASERVER_HOME*`/config/nodemanager`

5. Start the Administration Server:

   ```
   nmStart('AdminServer')
   ```

6. Exit WLST:

   ```
   exit()
   ```

Validating the Administration Server

Before you proceed with the configuration steps, validate that the Administration Server has started successfully by making sure that you have access to the Oracle Enterprise Manager Fusion Middleware Control; this is installed and configured on the Administration Servers.

To navigate to Fusion Middleware Control use the URL in URLs Used in This Chapter. Log in with the Oracle WebLogic Server administrator credentials.

Copyright © 2018, 2025, Oracle and/or its affiliates.

You should be able to connect to the Admin Server from the Remote Console as before.

Starting and Validating the soa_server1 Managed Server on OIGHOST1

After you have configured Node Manager and created the Managed Server domain directory, you can use WebLogic Administration Console to start the soa_server1 Managed Server on OIGHOST1.

1. Access Fusion Middleware Control using the URL in URLs Used in This Chapter. Log in with the Oracle WebLogic Server administrator credentials.
2. Select the Servers pane to view the Managed Servers in the domain.
3. Select only the **soa_server1** Managed Server, and note the assigned port number.
4. Click **Control** > **Start** on the tool bar to start the selected Managed Server.
5. To verify that the Managed Server is working correctly, open your browser and enter the following URL:

   For End to End SSL:

   ```
   https://oighost1.example.com:7004/soa-infra
   ```

   For SSL Terminated:

   ```
   http://oighost1.example.com:7003/soa-infra
   ```

Creating the New Authentication Provider

After creating the new domain, if you are using LDAP and want to log in using LDAP, then you must create an authentication provider for the directory inside the OIG domain.

To create a new LDAP-based authentication provider:

1. Change directory to `$IGD_ORACLE_HOME/idm/server/ssointg/config`
2. Edit the `configureWLSAuthnProviders.config` file as shown below. For more details on these parameters, see Variables Used When Creating the Infrastructure Domain:

   ```
   OIM_WLSHOST: IGDADMINVHN.example.com
   OIM_WLSPORT: 9102
   OIM_WLSADMIN: weblogic
   OIM_IDSTORE_ROLE_SECURITY_ADMIN: WLSAdministrators
   WLS_IS_SSLENABLED: true
   WLS_TRUSTSTORE: /u01/oracle/config/keystores/idmTrustStore.p12
   WLS_SSL_HOST_VERIFICATION: true
   IDSTORE_DIRECTORYTYPE: OUD
   IDSTORE_HOST: idstore.example.com
   IDSTORE_PORT: 1636
   IDSTORE_SSL_ENABLED: true
   IDSTORE_KEYSTORE_FILE: /u01/oracle/config/keystores/idmTrustStore.p12
   IDSTORE_BINDDN: cn=oimLDAP,cn=systemids,dc=example,dc=com
   IDSTORE_USERSEARCHBASE: cn=Users,dc=example,dc=com
   IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=example,dc=com
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> You can also specify the passwords directly in the file, if required. If you do not specify the passwords, you will be prompted for them at runtime. The parameters are:
>
> - IDSTORE_BINDDN_PWD
> - IDSTORE_KEYSTORE_PASSWORD
> - OIM_WLSADMIN_PWD
> - WLS_TRUSTSTORE_PASSWORD

Save the file when done.

3. Before you can use the integration scripts, you must grant execute permissions on the file by running the following commands:

```
chmod 750 $IGD_ORACLE_HOME/idm/server/ssointg/bin/OIGOAMIntegration.sh


chmod 750 $IGD_ORACLE_HOME/idm/server/ssointg/bin/OIGOAMIntegration.sh
```

4. Execute the `OIGOAMIntegration.sh` script for creating the authenticator. For example:

```
cd $IGD_ORACLE_HOME/idm/server/ssointg/bin
```

5. Make sure the environment variables are set as per Setting Environment Variables
6. Execute the `OIGOAMIntegration.sh` script for configuring the authentication provider. For example:

```
./OIGOAMIntegration.sh -configureWLSAuthnProviders
```

7. Verify that there are no errors.

Adding a Load Balancer Certificate to Oracle Keystore Service

Some OIG Products require that the SSL certificate used by the load balancer be added to the Oracle Keystore Service Trusted Certificates.

To add the certificate, do the following:

1. Obtain the certificate from the load balancer. You can obtain the load balancer certificate using a browser, such as Firefox. However, the easiest way to obtain the certificate is to use the **openssl** command. The syntax of the command is as follows:

```
openssl s_client -connect <LOADBALANCER>:<PORT> -showcerts </dev/null
2>/dev/null| sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p'
> $SHARED_CONFIG_DIR/keystores/<LOADBALANCER>.pem
```

For example:

Copyright © 2018, 2025, Oracle and/or its affiliates.

```
openssl s_client -connect <LOADBALANCER>:<PORT> -showcerts </dev/null
2>/dev/null| sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p'
> $SHARED_CONFIG_DIR/keystores/login.example.com.pem
```

> **Note**
>
> This command saves the certificate to a file called `login.example.com.pem` in `$SHARED_CONFIG_DIR/keystores`.

2. Load the certificate into the Oracle Keystore Service using WLST. Connect to WLST by using the following command:

   ```
   $IGD_ORACLE_HOME/oracle_common/common/bin/wlst.sh
   ```

   > **Note**
   >
   > If using End to End SSL then specify the location of your certificates first:
   >
   > ```
   > export WLST_PROPERTIES="-
   > Dweblogic.security.TrustKeyStore=CustomTrust \
   > -Dweblogic.security.CustomTrustKeyStoreFileName=/u02/oracle/config/
   > keystores/idmTrustStore.p12 \
   > -Dweblogic.security.CustomTrustKeyStorePassPhrase=password \
   > -Dweblogic.security.SSL.ignoreHostnameVerification=true"
   > ```

3. Inside the WLST shell, connect to the Administration Server using the following command:

   ```
   connect('<OIG_WEBLOGIC_USER>','<OIG_WEBLOGIC_PWD>','t3://
   igdadminvhn.example.com:<OIG_ADMIN_PORT>')
   ```

   For example, for SSL Terminated:

   ```
   connect('weblogic','<password>','t3://igdadminvhn.example.coml:7101')
   ```

   For example, for End to End SSL:

   ```
   connect('weblogic','<password>','t3s://igdadminvhn.example.coml:9201')
   ```

4. Download the access artifacts by using the following command:

   ```
   downloadAccessArtifacts(domain_home="/u01/oracle/config/domains/OIG",
   propsFile="/u01/oracle/config/db.props"
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> For information about the contents of the properties file, see Doc ID 2318818.1 on My Oracle Support.

5. Load the certificate using the following commands:

```
svc = getOpssService(name='KeyStoreService')
svc.importKeyStoreCertificate(appStripe='system',name='trust',password='',
keypassword='',alias='<CertificateName>',type='TrustedCertificate',
filepath='/<SHARED_CONFIG_DIR>/keystores/<LOADBALANCER>.pem')
```

   For example:

```
svc = getOpssService(name='KeyStoreService')
svc.importKeyStoreCertificate(appStripe='system',name='trust',password='',
keypassword='',alias='login.example.com',type='TrustedCertificate',
filepath='/u01/oracle/config/keystores/login.example.com.pem')
```

6. Synchronize the keystore service with the file system by using the following command:

```
syncKeyStores(appStripe='system', keystoreFormat='KSS')
```

7. Save the access artifacts by using the following command:

```
saveAccessArtifacts(domain_home="/u01/oracle/config/domains/OIG",
propsFile="/u01/oracle/config/db.props"
```

8. Exit the WLST shell:

```
exit()
```

9. You will need to restart the domain for the changes to take effect.

Configuring the WebLogic Proxy Plug-In

Before you can validate that requests are routed correctly through the Oracle HTTP Server instances, you must set the `WebLogic Plug-In Enabled` parameter.

It is recommended to set the `WebLogic Plug-In Enabled` parameter at the domain level. Any clusters or servers not using the plugin via the web-tier can have their `WebLogic Plug-In Enabled` parameter value set to `no` on an exception basis as needed.

1. Log in to the Oracle WebLogic Remote Console.
2. Click **Edit Tree**.
3. Click **Domain**.

   The domain settings page is displayed.
4. Click on the Domain Name.
5. Click on the **Web Applications** tab.
6. Locate and select the **WebLogic PlugIn Enabled** option.

Copyright © 2018, 2025, Oracle and/or its affiliates.

7. Click **Save**.
8. Click the **Shopping Cart** and select **Commit Changes**.

Modifying the Upload and Stage Directories to an Absolute Path

After you configure the domain and unpack it to the Managed Server domain directories on all the hosts, verify and update the upload and stage directories for Managed Servers in the new clusters. See Modifying the Upload and Stage Directories to an Absolute Path in an Enterprise Deployment.

Starting and Validating the Oracle Identity Governance Managed Servers

Now that you have extended the domain, started the Administration Server, and propagated the domain to the other hosts, you can start the newly configured Oracle Identity Governance Managed Servers.

This process involves three tasks as described in the following sections.

- Starting the Oracle Identity Governance Managed Servers and Bootstrapping the Domain
  You are required to boot strap the OIG domain to deploy the OIG artifacts into the domain.
- Starting the soa_server1 and oim_server1 Managed Servers
- Validating the Managed Server by Logging in to the Identity Console
- Starting and Validating soa_server2 and oim_server2 Managed Servers

Starting the Oracle Identity Governance Managed Servers and Bootstrapping the Domain

You are required to boot strap the OIG domain to deploy the OIG artifacts into the domain.

Bootstrapping the domain is largely automatic and is performed by starting and stopping the managed servers in the domain in the following order:

1. Start the Oracle SOA Suite Managed Server soa_server1 if not already started.
2. Start the Oracle Identity Governance Managed Server oim_server1.

   The bootstrap process starts the Managed Server, and then stops it again automatically. You may see a Failed status in the WebLogic console, which can be ignored.
3. Stop the Oracle SOA Suite Managed Server soa_server1.
4. Stop oim_server1.
5. Stop the WebLogic Administration Server.
6. Start the WebLogic Administration Server.
7. Start the Oracle SOA Suite Managed Servers soa_server1 and soa_server2.
8. Start the Oracle Identity Governance Managed Servers oim_server1 and oim_server2.

In order for the bootstrapping process to successfully complete, it must occur when the OIM server is started from the *IGD_ASERVER_HOME* directory. However, the Node Manager that runs out of the *IGD_ASERVER_HOME* communicates using the `igdadmin` address. Rather than temporarily reconfiguring the Managed Servers to use this address, the Managed Servers can

Copyright © 2018, 2025, Oracle and/or its affiliates.

be started outside of Node Manager for the bootstrap process. Once the process is complete, the Managed Servers will be moved to local storage and Node Manager configured will be able to start and stop them.

Starting the soa_server1 and oim_server1 Managed Servers

To start the soa_server1 and oim_server1 Managed Servers:

1. Access Fusion Middleware Control using the URLs in URLs Used in This Chapter. Log in using the Administration Server credentials.
2. In the **Target Navigation** pane, expand the domain to view the Managed Servers in the domain.
3. Select only the **soa_server1** Managed Server and click **Start Up** on the Oracle WebLogic Server toolbar.
4. When the startup operation is complete, navigate to the Domain home page and verify that the **soa_server1** Managed Server is up and running.
5. Repeat for **oim_server1**.

Validating the Managed Server by Logging in to the Identity Console

Validate the Oracle Identity Manager Server instance by accessing the Oracle Identity Manager Consoles in a Web browser. See URLs Used in This Chapter. Log in using the **xelsysadm** username and password.

Validate the SOA configuration:

For SSL Terminated:

- Component URL: `http://oighost1.example.com:7003/soa-infra`
- Load Balancer URL: `http://igdinternal.example.com:7777/soa-infra`

For End to End SSL:

- Component URL: `https://oighost1.example.com:7004/soa-infra`
- Load Balancer URL: `https://igdinternal.example.com/soa-infra`

Starting and Validating soa_server2 and oim_server2 Managed Servers

After validating the successful configuration and startup of the soa_server1 and oim_server1 Managed Servers, you can start and validate the soa_server2 and oim_server2 Managed Servers.

To start and validate the soa_server2 Managed Server, use the procedure in Starting and Validating the soa_server1 Managed Server on OIGHOST1 for soa_server2 Managed Server. Use the procedure to start and validate the oim_server2 Managed Servers too.

For the validation URL, enter the following URL into the Identity your web browser and log in using the enterprise deployment administrator user:

For SSL Terminated:

`http://oighost2.example.com:14000/identity`

For End to End SSL:

`https://oighost2.example.com:14001/identity`

Copyright © 2018, 2025, Oracle and/or its affiliates.

Analyzing the Bootstrap Report

When you start the Oracle Identity Governance server, the bootstrap report is generated at `$IGD_ASERVER_HOME/servers/oim_server1/logs/BootStrapReportPreStart.html`.

The bootstrap report `BootStrapReportPreStart.html` is an html file that contains information about the topology that you have deployed, the system level details, the connection details like the URLs to be used, the connectivity check, and the task execution details. You can use this report to check if the system is up, and also to troubleshoot the issues, post-configuration. Every time you start the Oracle Identity Governance server, the bootstrap report is updated.

Sections in the Bootstrap Report

- **Topology Details**

  This section contains information about your deployment. It shows whether you have configured a cluster setup, SSL enabled, or upgraded an Oracle Identity Manager environment from an earlier release.

- **System Level Details**

  This section contains information about the JDK version, Database version, JAVA_HOME, DOMAIN_HOME and OIG_HOME.

- **Connection Details**

  This section contains information about the connect details like the Administration URL, OIM Front End URL, SOA URL, and RMI URL.

  This also shows whether the Administration Server, Database, and SOA server is up or not.

- **Execution Details**

  This section lists the various tasks and their statuses.

Creating a Separate Domain Directory for Managed Servers

When you initially create the domain for enterprise deployment, the domain directory resides on a shared disk. This default domain directory is used to run the Administration Server. You can now create a copy of the domain on the local storage for each of your managed server hosts. The domain directory on the local (or private) storage is used to run the Managed Servers.

- Stopping the Managed Servers
- Packing the Domain
- Unpacking the Domain

Stopping the Managed Servers

Before you move the managed servers to a separate directory you must first stop any managed servers which are currently running.

Perform the following tasks:

1. Login to Fusion Middleware Control as an Administrator using the URL outlined in URLs Used in This Chapter.

Copyright © 2018, 2025, Oracle and/or its affiliates.

2. In the Target Navigation pane, expand the domain to view the Managed Servers in the domain.
3. Select any running Managed Servers and click **Shutdown** on the Oracle WebLogic Server toolbar.
4. Wait for the shutdown operation to complete.
5. Repeat the above for all other managed servers.

Packing the Domain

Placing the *IGD_MSERVER_HOME* on local storage is recommended to eliminate the potential contention and overhead caused by servers writing logs to shared storage. It is also faster to load classes and jars need from the domain directory, so any temporary or cache data that the Managed Servers use from the domain directory is processed quicker.

As described in Preparing the File System for an Enterprise Deployment, the path to the Administration Server domain home is represented by the *IGD_ASERVER_HOME* variable, and the path to the Managed Server domain home is represented by the *IGD_MSERVER_HOME* variable.

To create the Managed Server domain directory:

1. Sign in to the host running the Administration Server, for example, OIGHOST1, and run the `pack` command to create a template as follows:

```
cd $ORACLE_COMMON_HOME/common/bin

./pack.sh -managed=true \
-domain=$IGD_ASERVER_HOME \
-template=/full_path/edgdomaintemplate.jar \
-template_name=edg_domain_template \
-log_priority=DEBUG \
-log=/tmp/pack.log
```

   In this example:

   - Replace $IGD_ASERVER_HOME with the actual path to the domain directory you created on the shared storage device.
   - Replace *full_path* with the complete path to the location where you want to create the domain template jar file. You need to reference this location when you copy or unpack the domain template jar file. It is recommended to choose a shared volume other than ORACLE_HOME, or write to `/tmp/` and copy the files manually between servers.

     You must specify a full path for the template jar file as part of the `-template` argument to the pack command:

     ```
     $SHARED_CONFIG_DIR/domains/template_filename.jar
     ```

   - The `edgdomaintemplate.jar` file is a sample name for the jar file that you create, which contains the domain configuration files.
   - The `edg_domain_template` label is the label is assigned to the template data stored in the template file.

Copyright © 2018, 2025, Oracle and/or its affiliates.

For Example:

```
cd $ORACLE_COMMON_HOME/common/bin


./pack.sh -managed=true \
-domain=$IGD_ASERVER_HOME \
-template=/tmp/edgdomaintemplate.jar \
-template_name=edg_domain_template \
-log_priority=DEBUG \
-log=/tmp/pack.log
```

2. Make a note of the location of the `edgdomaintemplate.jar` file that you just created with the pack command.

> **Tip**
>
> For more information about the `pack` and `unpack` commands, see Overview of the Pack and Unpack Commands in *Creating Templates and Domains Using the Pack and Unpack Commands*.

3. If you have not already, create the recommended directory structure for the Managed Server domain on the OIGHOST1 local storage device.

   Use the examples in File System and Directory Variables Used in This Guide as a guide.

Unpacking the Domain

To unpack the domain on local storage:

1. Run the `unpack` command to unpack the template in the domain directory onto the local storage, as follows:

```
cd $ORACLE_COMMON_HOME/common/bin


./unpack.sh -domain=$IGD_MSERVER_HOME \
-overwrite_domain=true \
-template=/full_path/edgdomaintemplate.jar \
-log_priority=DEBUG \
-log=/tmp/unpack.log \
-app_dir=$MS_APPLICATION_HOME
```

For Example:

```
cd $ORACLE_COMMON_HOME/common/bin


./unpack.sh -domain=$IGD_MSERVER_HOME \
-overwrite_domain=true \
-template=/tmp/edgdomaintemplate.jar \
-log_priority=DEBUG \
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
-log=/tmp/unpack.log \
-app_dir=$MS_APPLICATION_HOME
```

> **Note**
>
> The `-overwrite_domain` option in the `unpack` command allows you to unpack a managed server template into an existing domain and existing applications directories. For any file that is overwritten, a backup copy of the original is created. If any modifications had been applied to the start scripts and ear files in the managed server domain directory, they must be restored after this unpack operation.
>
> Additionally, to customize server startup parameters that apply to all servers in a domain, you can create a file called `setUserOverridesLate.sh` and configure it to, for example, add custom libraries to the WebLogic Server classpath, specify additional JAVA command-line options for running the servers, or specify additional environment variables. Any customizations that you add to this file are preserved during domain upgrade operations, and are carried over to remote servers when you use the `pack` and `unpack` commands.

In this example:

- Replace IGD_MSERVER_HOME with the complete path to the domain home to be created on the local storage disk. This is the location where the copy of the domain is unpacked.
- Replace `/full_path/edgdomaintemplate.jar` with the complete path and file name of the domain template jar file that you created when you ran the pack command to pack the domain on the shared storage device.
- Replace $MS_APPLICATION_HOME with the complete path to the Application directory for the domain on shared storage. See File System and Directory Variables Used in This Guide.

> **Tip**
>
> For more information about the `pack` and `unpack` commands, see Overview of the Pack and Unpack Commands in *Creating Templates and Domains Using the Pack and Unpack Commands*.

2. Change directory to the newly created Managed Server directory and verify that the domain configuration files were copied to the correct location on the OIGHOST1 local storage device.

Validating the Fusion Middleware Control Application

After the bootstrap process has been executed and validated, access to the Fusion Middleware Control application should be available.

To navigate to the Fusion Middleware Control application, use the URL in the URLs Used in This Chapter section.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Configuring the Web Tier for the Domain

Configure the web server instances on the web tier so that the instances route requests for both public and internal URLs to the proper clusters in the extended domain.

For additional steps in preparation for possible scale-out scenarios, see Updating Cross Component Wiring Information.

- Validating the Oracle SOA Suite URLs Through the Load Balancer

Validating the Oracle SOA Suite URLs Through the Load Balancer

To validate the configuration of the Oracle HTTP Server virtual hosts and to verify that the external load balancer can route requests through the Oracle HTTP Server instances to the application tier:

1. Verify that the server status is reported as **Running** in the Administration Console.

   If the server is shown as **Starting** or **Resuming**, wait for the server status to change to **Started**. If another status is reported (such as **Admin** or **Failed**), check the server output log files for errors.

2. Verify that you can access these URLs:

   > **Note**
   >
   > It is not necessary at this stage to attempt to login to the individual pages. All you are checking is that the pages can be accessed through the load balancer and the web server.

   SSL Terminated Deployments

   - `http://igdinternal.example.com:7777/soa-infra`
   - `http://igdinternal.example.com:7777/integration/worklistapp`
   - `http://igdinternal.example.com:7777/soa/composer`

   End to End SSL Deployments

   - `https://igdinternal.example.com/soa-infra`
   - `https://igdinternal.example.com/integration/worklistapp`
   - `https://igdinternal.example.com/soa/composer`

Integrating Oracle Identity Governance with Oracle SOA Suite

Oralce Identity Governance invokes Oracle SOA Suite using a number of URLs, which out of the box are wired to individual managed servers. In a High Availibility environment you need to update these URLs to use a load balancer.

To do this perform the following steps:

1. Change directory to `$IGD_ORACLE_HOME/idm/server/ssointg/config`.

   ```
   cd $IGD_ORACLE_HOME/idm/server/ssointg/config
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

2. Edit the file `configureSOAConnector.config` as shown below. Save when complete:

```
OIM_HOST: oighost1.example.com
OIM_PORT: 14001
OIM_WLSADMIN: weblogic
OIM_WLSADMIN_PWD: password
WLS_TRUSTSTORE: /u02/oracle/config/keystores/idmTrustStore.p12
WLS_TRUSTSTORE_PASSWORD: password
WLS_IS_SSLENABLED: true

OIM_FRONTEND_URL: https://igdinternal.example.com:443/
OIM_EXTERNAL_FE_URL: https://oig.edg.com:443/
SOA_SOAP_URL: https://igdinternal.example.com:443/
SOA_RMI_URL: cluster:t3s://soa_cluster
UMS_WS_URL: https://igdinternal.example.com:443/ucs/messaging/webservice/
```

3. Run the following command:

```
./OIGOAMIntegration.sh -configureSOAIntegration
```

> **Note**
>
> If you do not specify the passwords, you will be prompted for them at runtime.

Managing the Notification Service

An event is an operation that occurs in Oracle Identity Manager, such as user creation, request initiation, or any custom event created by the user. These events are generated as part of the business operations or through the generation of errors. Event definition is the metadata that describes the event.

To define the metadata for events, you must identify all event types supported by a functional component. For example, as a part of the scheduler component, metadata is defined for a scheduled job execution failure and shutting down of the scheduler. Every time a job fails or the scheduler shuts down, the associated events get triggered, and the notifications associated with the event get sent.

The data available in the event is used to create the content of the notification. The different parameters defined for an event help the system to select the appropriate notification template. The various parameters defined for an event help the system decide which event variables should be made available at template design time.

A notification template is used to send notifications. These templates contain variables that refer to available data to provide more context to the notifications. The notification is sent through a notification provider. Examples of such channels are e-mail, Instant Messaging (IM), Short Message Service (SMS), and voice. To use these notification providers, Oracle Identity Manager uses Oracle User Messaging Service (UMS).

At the back end, the notification engine is responsible for generating the notification and utilizing the notification provider to send the notification.

- Using SMTP for Notification
- Adding a CSF Key

Copyright © 2018, 2025, Oracle and/or its affiliates.

Using SMTP for Notification

Using SMTP for notification involves configuring the SMTP email notification provider properties and adding the CSF key.

Configuring the SMTP Email Notification Provider Properties

To configure SMTP Email Notification Provider properties by using the **EmailNotificationProviderMBean MBean** :

1. Log into the Oracle Fusion Middleware Control using the URL in the URLs Used in This Chapter section.

   The Administration Server host and port number were in the URL on the End of Configuration screen (Writing Down Your Domain Home and Administration Server URL). The default Administration Server port number is `7001`.

   The login credentials were provided on the Administrator Account screen (Configuring the Administrator Account).
2. Click **weblogic_domain**, and then click **System Mbean Browser**.
3. In the search box, enter `EmailNotificationProviderMBean`, and click **Search**. The mbean is displayed.

   > **Note**
   >
   > If Oracle Identity Governance still starting (coming up) or is just started (RUNNING MODE), the Enterprise Manager does not show any Mbeans defined by OIM. Wait for two minutes for the server to start, and then try searching for the Mbean in **System Mbean Browser** of the Enterprise Manager.
4. Ensure that the correct information is entered for your email server in particular:

Table 15-10 SMTP Email Notification Provider Properties

Attribute	Value
CSFKey	Set this to a name of a CSF credential, this can be any name and will be used while adding a CSF key. For example; **mailUser**
Enabled	Set to true.
MailServerName	Set to the host name of your email server.
WSUrl	`http://igdinternal.example.com/ucs/messaging/webservice` (SSL Terminated) `https://igdinternal.example.com/ucs/messaging/webservice` (End to End SSL)

5. Click **Apply** to save the changes.

Adding a CSF Key

To add a CSF key:

1. Login to Oracle Enterprise Manager.

Copyright © 2018, 2025, Oracle and/or its affiliates.

2. Click WebLogic Domain and select **Security>Credentials**.
3. Expand **oracle.wsm.security** and click **Create Key**.
4. Enter the following information.

Table 15-11 CSF Key Properties

Attribute	Value
Key name	Enter the value of the credential Key, this must be the same value as defined in Using SMTP for Notification for example; mailUser.
Username	Enter the name of the user you use to authenticate with your email server.
Password/Confirm Password	Enter the password of the user you use to authenticate with your email server.
Description	Provide a description of the key being created. For example, Mail Server Credentials

5. Click **OK**.

Configuring the Messaging Drivers

Each messaging driver needs to be configured. You have to configure this service if you want to enable OAM's forgotten password functionality.

- Configuring the Email Driver

Configuring the Email Driver

To configure the driver to send and emails then you need to perform the following steps:

1. Log in to the Oracle Fusion Middleware Control.
2. Click the **Target Navigation** icon next to the Domain name.
3. Click **usermessagingserver (soa_server1)** under User Messaging Service. A list of all the drivers will be shown.
4. Click **Configure Driver** next to the User Messaging Email Driver.
5. If a configuration does not exist then click **Create**. If the configuration exists, click **Edit**.
6. Update the attributes with the required details.

Table 15-12 Configuring the Email Driver Attributes

Attributes	Values
Name	MyemailServer
Sender Address	Enter the From email address for the emails you wish to send in the format: EMAIL: `myuser@example.com`

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-12 (Cont.) Configuring the Email Driver Attributes

Attributes	Values
Capability	Choose whether you are going to send or receive emails. Complete the following Email Properties using the values specific to your organisation. Contact your email administrator for details, the details below are for Sending only. Refer to the documentation for receiving email details. • Outdoing Mail server. • Outgoing Mail server port • Outgoing email Server Security • Outgoing User name and password, if your email server requires it.

7. Click **Test** to validate the information.
8. Click **OK** to save the information.

Forcing Oracle Identity Governance to use Correct Multicast Address

Oracle Identity Governance uses multicast for certain functions. By default, the managed servers communicate using the multi cast address assigned to the primary host name. If you wish multicast to use a different network, for example, of the internal network, you must complete additional steps.

1. Log into the Oracle WebLogic Remote console using the URL in the URLs Used in This Chapter section.
2. Click **Edit Tree**.
3. Under **Domain Structure**, click **Environment** and then expand **Servers**. The Summary of Servers page is displayed.
4. Click the OIM Managed Server name, for example, `oim_server1` on the list of servers. The Settings for oim_server1 are displayed.
5. Click the **Advanced** tab followed by the **Node Manager** tab.
6. Go to the **Server Start** tab.
7. Add the following line to the arguments field:

   ```
   -Dmulticast.bind.address=oighost1.example.com
   ```

8. Click **Save**.
9. Repeat for the Managed Server `oim_server2`. When doing so, make sure you add the following line to the arguments field:

   ```
   -Dmulticast.bind.address=oighost2.example.com
   ```

10. Click the **Shopping Cart** and select **Activate Changes**.
11. Restart the managed servers `oim_server1` and `oim_server2`.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Integrating Oracle Identity Governance with LDAP

Integrate Oracle Identity Governance with LDAP.

This section includes the following topics:

- Variables Used in OIG Integration with LDAP
- Installing the Connector Bundle
- Configuring the Oracle Connector for LDAP
- Restarting the Domains

Variables Used in OIG Integration with LDAP

As you perform the tasks in this section, you will be referencing the variables listed below.

The following table explains the configuration file property values required in this section.

Table 15-13 OIG Integration with LDAP Variables Used in This Chapter

Variable	Sample Value	Description
IDSTORE_DIRECTORY TYPE	`OUD`	The type of directory you are using. Valid value is OUD.
IDSTORE_HOST	`idstore.example.com`	The host and port of your Identity Store directory. When preparing the Identity Store these should point to one of the LDAP instances. When configuring components such as OAM or OIG they should point to the load balancer entry point.
IDSTORE_PORT	`1636`	The port of your Identity Store directory. When preparing the Identity Store these should point to one of the LDAP instances. When configuring components such as OAM or OIG they should point to the load balancer entry point.
IDSTORE_BINDDN	`cn=oudadmin`	An administrative user in the Identity Store Directory.
IDSTORE_BINDDN_PA SSWD	`password`	The the password of IDSTORE_BINDDN. You will be prompted for the value if not supplied.
IDSTORE_SSL_ENABL ED	`true`	Whether SSL to the identity store is enabled. Valid values: `true` | `false`.
IDSTORE_KEYSTORE_ FILE	`/u01/oracle/config/keystores/ idmTrustStore.p12`	The location of the LDAP Truststore for LDAP connections.
IDSTORE_KEYSTORE_ PASSWORD	`password`	The password of the IDSTORE_KEYSTORE_FILE.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-13 (Cont.) OIG Integration with LDAP Variables Used in This Chapter

Variable	Sample Value	Description
IDSTORE_OIMADMINUSERDN	`cn=oimLDAP,cn=systemids,dc=example,dc=com`	The location of a container in the directory where system-operations users are stored. There are only a few system-operations users and are kept separate from enterprise users stored in the main user container. For example, the Oracle Identity Governance reconciliation user which is also used for the bind DN user in Oracle Virtual Directory adapters.
IDSTORE_SEARCHBASE	`dc=example,dc=com`	The location in the directory where Users and Groups are stored.
IDSTORE_USERSEARCHBASE	`cn=Users,dc=example,dc=com`	The location in the directory where Users are Stored.
IDSTORE_GROUPSEARCHBASE	`cn=Groups,dc=example,dc=com`	The location in the directory where Groups are Stored.
IDSTORE_USERSEARCHBASE_DESCRIPTION	`Default user container`	Description for the directory user search base.
IDSTORE_GROUPSEARCHBASE_DESCRIPTION	`Default group container`	Description for the directory group search base.
IIDSTORE_EMAIL_DOMAIN	`example.com`	The domain used for e-mail For example, `example.com`.
OIM_HOST	`oighost1.example.com`	Host name for oim_server1 on OIGHOST1.
OIM_PORT	`14000`	Port number for oim_server1 on OIGHOST1.
WLS_OIM_SYSADMIN_USER	`xelsysadm`	The system admin user to be used to connect to OIG while configuring SSO. This user needs to have system admin role.
WLS_OIM_SYSADMIN_USER_PWD	`password`	Enter the password for OIG system administrator user.
OIM_WLSHOST	`igdadminvhn.example.com`	Virtual Host name for Admin Server on OIGHOST1.
OIM_WLSPORT	`9102`	Corresponding port number for Admin Server on OIGHOST1.
WLS_IS_SSLENABLED	`true`	Whether SSL to the identity store is enabled. Valid values: `true` | `false`.
WLS_TRUSTSTORE	`/u01/oracle/config/keystores/idmTrustStore.p12`	The location of the WLS truststore.
WLS_TRUSTSTORE_PASSWORD	`password`	Password for the WLS truststore.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-13 (Cont.) OIG Integration with LDAP Variables Used in This Chapter

Variable	Sample Value	Description
OIM_WLSADMIN	`weblogic`	The weblogic administrator user in OIM domain.
OIM_SERVER_NAME	`oim_server1`	The OIG server name.
CONNECTOR_MEDIA_PATH	OID/OUD =*IGD_ORACLE_HOME*`/idm/server/ConnectorDefaultDirectory/OID-12.2.1.3.0` AD = *IGD_ORACLE_HOME*`/idm/server/ConnectorDefaultDirectory/activedirectory-12.2.1.3.0`	The location of the Connector bundle downloaded and unzipped. Oracle Identity Governance would use this location to pick the Connector bundle to be installed.
IS_LDAP_SECURE	`false`	It indicates the usage of SSL for LDAP Communication. Use yes or no for ActiveDirectory.
LOG_FILE	`/home/oracle/workdir/logs/configure_ldap.log`	Location of log file to be created.
SSO_TARGET_APPINSTANCE_NAME	`SSOTarget`	The Target application instance name used for provisioning account to target LDAP.

Table 15-14 Active Directory Variables

Variable	Sample Value	Description
AD_CONNECTORSERVER_HOST	`192.0.2.1`	the host name or IP address of the computer hosting the connector server.
AD_CONNECTORSERVER_KEY	`<connectorserverkey>`	Enter the key for the connector server.
AD_CONNECTORSERVER_PORT	`8759`	Enter the number of the port at which the connector server is listening.
AD_CONNECTORSERVER_TIMEOUT	`0`	Enter an integer value that specifies the number of milliseconds after which the connection between the connector server and the Oracle Identity Governance times out. A value of 0 means that the connection never times out.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-14 (Cont.) Active Directory Variables

Variable	Sample Value	Description
AD_CONNECTORSERVER_USESSL	`true (or false)`	Enter true to specify that you will configure SSL between Oracle Identity Governance or Oracle Unified Directory and the Connector Server. Otherwise, enter false. For Active Directory, the value should be yes or no. The default value is false **Note** It is recommended that you configure SSL to secure communication with the connector server.
AD_DOMAIN_NAME	`example.com`	Enter the domain name configured in Microsoft Active Directory.

Installing the Connector Bundle

1. Download the Connector bundle from the artifactory: Download Connector Bundle
   - For OID or OUD, download the Connector bundle corresponding to Oracle Internet Directory.

> **Note**
>
> For all directory types, the required Connector version for OIG-OAM integration is 12.2.1.3.0.

Copyright © 2018, 2025, Oracle and/or its affiliates.

2. Unzip the Connector bundle to the desired connector path under `$ORACLE_HOME/idm/server/ConnectorDefaultDirectory`.

   For example:

   ```
   $IGD_ORACLE_HOME/idm/server/ConnectorDefaultDirectory
   ```

Configuring the Oracle Connector for LDAP

The Oracle Connector for LDAP allows you to store users and passwords in a certified LDAP directory. Configure the connector before using it. Perform the following steps to configure the connector:

1. Change directory to `$IGD_ORACLE_HOME/idm/server/ssointg/config`.
2. Edit the file `configureLDAPConnector.config` as shown below. For an explanation of the parameters, see Variables Used in OIG Integration with LDAP:

```
##-------------------------------------------------------------##
## [configureLDAPConnector]
IDSTORE_DIRECTORYTYPE=OUD
IDSTORE_HOST=idstore.example.com
IDSTORE_PORT=1636
IDSTORE_BINDDN=cn=oudadmin
IDSTORE_SSL_ENABLED: true
IDSTORE_KEYSTORE_FILE: /u02/oracle/config/keystores/idmTrustStore.p12
IDSTORE_OIMADMINUSERDN=cn=oimLDAP,cn=systemids,dc=example,dc=com
IDSTORE_SEARCHBASE=dc=example,dc=com
IDSTORE_USERSEARCHBASE=cn=Users,dc=example,dc=com
IDSTORE_GROUPSEARCHBASE=cn=Groups,dc=example,dc=com
IDSTORE_USERSEARCHBASE_DESCRIPTION=Default user container
IDSTORE_GROUPSEARCHBASE_DESCRIPTION=Default group container
IDSTORE_EMAIL_DOMAIN=example.com
OIM_HOST=oighost1.example.com
OIM_PORT=14000
WLS_OIM_SYSADMIN_USER=xelsysadm
WLS_IS_SSLENABLED: true
WLS_TRUSTSTORE: /u02/oracle/config/keystores/idmTrustStore.p12
OIM_WLSHOST=igdadminvhn.example.com
OIM_WLSPORT=9102
OIM_WLSADMIN=weblogic
OIM_SERVER_NAME=oim_server1
CONNECTOR_MEDIA_PATH=IGD_ORACLE_HOME/idm/server/ConnectorDefaultDirectory/
OID-12.2.1.3.0
IS_LDAP_SECURE: true
LOG_FILE: /home/oracle/workdir/logs/configure_ldap.log
SSO_TARGET_APPINSTANCE_NAME: SSOTarget
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> You can also specify the passwords directly in the file, if required. If you do not specify the passwords, you will be prompted for them at runtime. The parameters are:
>
> - IDSTORE_BINDDN_PWD
> - IDSTORE_KEYSTORE_PASSWORD
> - IDSTORE_OIMADMINUSER_PWD
> - WLS_OIM_SYSADMIN_USER_PWD
> - OIM_WLSADMIN_PWD
> - WLS_TRUSTSTORE_PASSWORD

Save the file when done.

3. Make sure the environment variables are set as per Setting Environment Variables.
4. Execute the `OIGOAMIntegration.sh` script for configuring the connector. For example:

```
cd $IGD_ORACLE_HOME/idm/server/ssointg/bin


./OIGOAMIntegration.sh -configureLDAPConnector
```

Restarting the Domains

You must restart the IAMAccessDomain and the IAMGovernanceDomain domains.

Integrating Oracle Identity Governance and Oracle Access Manager

You have to complete several tasks to integrate Oracle Identity Governance and Oracle Access Manager. These tasks include creating the WLS authentication providers, deleting OIMSignatureAuthenticator and recreating OUDAuthenticator, adding the administration role to the new administration group, and so on.

- Variables Used in OIG and OAM Integration
- Configuring SSO Integration in the IAMGovernanceDomain
- Restarting the Domains
- Enable OAM Notifications

Variables Used in OIG and OAM Integration

As you perform the tasks in this section, you will be referencing the variables listed below.

The following tables explains the configuration file property values required in this section.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-15 OIG and OAM Integration Variables Used in This Section

Variable	Sample Value	Description
OAM_HOST	`login.example.com`	The listen address of the front end load balancer for the OAM cluster.
OAM_PORT	`443`	The port of the front end load balancer for the OAM cluster.
OAM_SSL_FLAG	`true`	Select `true` if the OAM domain is SSL or Secure Mode enabled. Otherwise set to `false`.
ACCESS_SERVER_HOST	`oamhost1.example.com`	Should be set to the value as OAM_HOST.
ACCESS_SERVER_PORT	`5575`	It is the port number for OAM PROXY PORT.
WEBGATE_PWD	`password`	Password for the WebGate.
COOKIE_DOMAIN	`.example.com`	The value assigned for the domain in Creating a Configuration File.
OAM_TRANSFER_MODE	`open`	The value assigned for the transfer mode in Creating a Configuration File.
OIM_LOGINATTRIBUTE	`uid`	The LDAP field containing the users login attribute, usually uid or cn.
SSO_INTEGRATION_MODE	`CQR`	The integration mode with OAM. With Challenge Question Response (CQR) mode, OIG will handle the password policy and password operations. With One Time Password (OTP) mode, any password operations will be handled by OAM itself and there will be no password change or reset in OIG.
SSO_ENABLED_FLAG	`true`	Set it to true to enable OIG-OAM integration is enabled.
OAM_WLS_ADMIN_HOST	`iadadminvhn.example.com`	The listen address of the Administration Server in the domain IAMAccessDomain.
OAM_WLS_ADMIN_PORT	`9002`	The listen port of the OAM_WLS_ADMIN_HOST. For example: 9002 for End to End SSL, or 7001 for SSL Terminated.
OAM_WLSHOST	`iadadminvhn.example.com`	The virtual hostname for the Admin Server on OAMHOST1.
OAM_WLSPORT	`7002`	The listen port for the Admin Server on OAMHOST1.
OAM_WLS_ADMIN_USER	`weblogic`	The Administration User of the IAD Administration Server.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-15 (Cont.) OIG and OAM Integration Variables Used in This Section

Variable	Sample Value	Description
OAM_WLS_ADMIN_PASSWD	`password`	Password for OAM_WLS_ADMIN_USER.
OAM_WLS_IS_SSLENABLED	`true`	Whether the OAM domain is SSL enabled.
OAM_IDSTORE_NAME	`OAMIDSTORE`	The value assigned for the idstore in Creating a Configuration File .
OIM_WLSHOST	`igdadminvhn.example.com`	Virtual Host name for Admin Server on OIGHOST1.
OIM_WLSPORT	`7102`	Corresponding port number for OIM_WLSHOST. If you have administration ports enabled in your OIG domain then set this to the administration port of the admin server 9102, otherwise set it to the Admin Server port 7101 or 7102 (SSL).
OIM_WLS_IS_SSLENABLED	`true`	Specifies whether the OIM domain is SSL enabled.
OIM_WLSADMIN	`weblogic`	The weblogic administrator user in OIM domain.
OIM_WLSADMIN_PWD	`password`	The password for the OIM_WLSADMIN user.
OIM_SERVER_NAME	`oim_server1`	The server name of the oim server.
IDSTORE_OAMADMINUSER	`oamadmin`	The user you use to access your Oracle Access Management Console.
IDSTORE_OAMADMINUSER_PWD	`oim_server1`	The name of the oim_server1.
IDSTORE_OAMSOFTWAREUSER	`oamLDAP`	The OAM software user account in LDAP.
IDSTORE_PWD_OAMSOFTWAREUSER	`password`	The password of the account (`oamLDAP`) you are connecting to the Identity Store with.
IDSTORE_HOST	`idstore.example.com`	The host and port of your Identity Store directory. When preparing the Identity Store these should point to one of the LDAP instances. When configuring components such as OAM or OIG they should point to the load balancer entry point.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-15 (Cont.) OIG and OAM Integration Variables Used in This Section

Variable	Sample Value	Description
IDSTORE_PORT	`1636`	The port of your Identity Store directory. When preparing the Identity Store these should point to one of the LDAP instances. When configuring components such as OAM or OIG they should point to the load balancer entry point.
IDSTORE_DIRECTORYTYPE	`OUD`	The type of directory you are using. Valid value is OUD.
IDSTORE_BINDDN	`cn=oudadmin`	An administrative user in the Identity Store Directory.
IDSTORE_BINDDN_PASSWD	`password`	The the password of IDSTORE_BINDDN. You will be prompted for the value if not supplied.
IDSTORE_USERSEARCHBASE	`cn=Users,dc=example,dc=com`	The location in the directory where Users are Stored.
IDSTORE_GROUPSEARCHBASE	`cn=Groups,dc=example,dc=com`	The location in the directory where Groups are Stored.
IDSTORE_SYSTEMIDBASE	`cn=SystemIDs,dc=example,dc=com`	The location of a container in the directory where system users can be placed when you do not want them in the main user container.
IS_LDAP_SECURE	`true`	Whether the connection to LDAP uses SSL or not.
WLS_IS_SSLENABLED	`true`	Whether SSL to the identity store is enabled. Valid values: `true` | `false`.
WLS_TRUSTSTORE	`/u01/oracle/config/keystores/idmTrustStore.p12`	The location of the WLS truststore.
WLS_TRUSTSTORE_PASSWORD	`password`	Password for the WLS truststore.
WLS_SSL_HOST_VERIFICATION	`true`	Whether the WLS SSL host verfication is enabled or not. Valid values: `true | false`.
OIM_TRUST_LOC	`/u01/oracle/config/keystores/idmTrustStore.p12`	The location of the OIG truststore.
OIM_TRUST_PWD	`password`	Password for the OIG truststore.
OIM_TRUST_TYPE	`PKCS12`	JKS or PKCS12.
OIM_WLS_SSL_IGNORE_HOST_VERIFICATION	`true`	Specify whether to ignore SSL Host name verification, this is often the case where wildcard certificates are used.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Configuring SSO Integration in the IAMGovernanceDomain

Having deployed the connector the next step in the process is the configuration of SSO in the domain. In order to do this you need to perform the following steps:

1. Change directory to `$IGD_ORACLE_HOME/idm/server/ssointg/config`:
2. Edit the file `configureSSOIntegration.config` updating the properties in the section **configureSSOIntegration** as shown below. For an explanation of the parameters, see Variables Used in OIG and OAM Integration:

```
##-------------------------------------------------------------##
## [configureSSOIntegration]
OAM_HOST: login.example.com
OAM_PORT: 443
OAM_SSL_FLAG: true
ACCESS_SERVER_HOST: oamhost1.example.com
ACCESS_SERVER_PORT: 5575
COOKIE_DOMAIN: example.com
OAM_TRANSFER_MODE: open
OIM_LOGINATTRIBUTE: uid
SSO_INTEGRATION_MODE: CQR
SSO_ENABLED_FLAG: true
OAM_WLS_ADMIN_HOST: igdadminvhn.example.com
OAM_WLS_ADMIN_PORT: 9002
OAM_WLSPORT: 7002

OAM_WLS_ADMIN_USER: weblogic
OAM_WLS_IS_SSLENABLED: true
OAM_IDSTORE_NAME: OAMIDSTORE
OIM_WLSHOST: igdadminvhn.example.com
OIM_WLS_IS_SSLENABLED: true
OIM_WLSPORT: 7102
OIM_WLSADMIN: weblogic
OIM_SERVER_NAME: oim_server1
IDSTORE_OAMADMINUSER: oamadmin
OIM_TRUST_LOC:/u01/oracle/config/keystores/idmTrustStore.p12
OIM_TRUST_TYPE:PKCS12
OIM_WLS_SSL_IGNORE_HOST_VERIFICATION:true
```

> **Note**
>
> You can also specify the passwords directly in the file, if required. If you do not specify the passwords, you will be prompted for them at runtime. The parameters are:
>
> - WEBGATE_PWD
> - OAM_WLS_ADMIN_PASSWD
> - OIM_WLSADMIN_PWD
> - IDSTORE_OAMADMINUSER_PWD
> - OIM_TRUST_PWD

Copyright © 2018, 2025, Oracle and/or its affiliates.

Save the file when done.

> **Note**
>
> If required you can also specify the passwords directly in the file. If you do not specify the passwords, you will be prompted for them at runtime.

3. Make sure the environment variables are set as per Setting Environment Variables.
4. Execute the `OIGOAMIntegration.sh` script for configuring the connector. For example:

```
cd $IGD_ORACLE_HOME/idm/server/ssointg/bin


./OIGOAMIntegration.sh -configureSSOIntegration
```

Restarting the Domains

You must restart the IAMAccessDomain and the IAMGovernanceDomain domains.

Enable OAM Notifications

Having deployed the connector the next step in the process is to tell OIG how to interact with OAM for terminating a user session after a user has been expired or terminated. In order to do this you need to perform the following steps:

1. Navigate to the following directory:

```
$IGD_ORACLE_HOME/idm/server/ssointg/config
```

2. Edit the file `enableOAMSessionDeletion.config` updating the properties in the section **enableOAMNotifications** as shown below:

```
##----------------------------------------------------------------##

## [enableOAMNotifications]
OIM_WLSHOST: oighost1.example.com
OIM_WLSPORT: 7002
WLS_IS_SSLENABLED: true
WLS_TRUSTSTORE: /u01/oracle/config/keystores/idmTrustStore.p12
WLS_SSL_HOST_VERIFICATION: true
OIM_WLSADMIN: weblogic
IDSTORE_DIRECTORYTYPE: OUD
IDSTORE_HOST:idstore.example.com
IDSTORE_PORT: 1636
IDSTORE_KEYSTORE_FILE: /u01/oracle/config/keystores/idmTrustStore.p12
IDSTORE_KEYSTORE_PASSWORD: password
IDSTORE_SSL_ENABLED: true
IS_LDAP_SECURE: true
IDSTORE_BINDDN: cn=oudadmin
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=example,dc=com
IDSTORE_SYSTEMIDBASE: cn=systemids,dc=example,dc=com
IDSTORE_OAMADMINUSER: oamadmin
IDSTORE_OAMSOFTWAREUSER: oamLDAP
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
IDSTORE_USERSEARCHBASE: cn=Users,dc=example,dc=com
OIM_SERVER_NAME: oim_server1
```

> **Note**
>
> You can also specify the passwords directly in the file, if required. If you do not specify the passwords, you will be prompted for them at runtime. The parameters are:
>
> - WLS_TRUSTSTORE_PASSWORD
> - OIM_WLSADMIN_PWD
> - IDSTORE_KEYSTORE_PASSWORD
> - IDSTORE_BINDDN_PWD
> - IDSTORE_OIMADMINUSER_PWD
> - IDSTORE_OAMSOFTWAREUSER_PWD

Save the file when done.

3. Make sure the environment variables are set as per Setting Environment Variables. Execute the `OIGOAMIntegration` script for enabling notifications. For example:

```
cd $IGD_ORACLE_HOME/idm/server/ssointg/bin
```

```
./OIGOAMIntegration.sh -enableOAMSessionDeletion
```

Propagating the Domain and Starting the Servers on OIGHOST2

After you start and validate the Administration Server on OIGHOST1, you can then you must propagate the domain and start the servers on OIGHOST2.

- Unpacking the Domain Configuration on OIGHOST2
- Starting the Node Manager in the Managed Server Domain Directory OIGHOST2
- Starting and Validating the soa_server2 and oim_server2 Managed Servers on OIGHOST2

Unpacking the Domain Configuration on OIGHOST2

Now that you have the Administration Server running on OIGHOST1, you can configure the domain on OIGHOST2.

1. Log in to OIGHOST2.
2. If you haven't already, create the recommended directory structure for the Managed Server domain on the OIGHOST2 storage device.

   Use the examples in File System and Directory Variables Used in This Guide as a guide.
3. Make sure the `edgdomaintemplate.jar` accessible to OIGHOST2.

Copyright © 2018, 2025, Oracle and/or its affiliates.

For example, if you are using a separate shared storage volume or partition for OIGHOST2, then copy the template to the volume or partition mounted to OIGHOST2.

4. Run the `unpack` command to unpack the template in the domain directory onto the local storage, as follows:

```
cd $ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=$IGD_MSERVER_HOME
-overwrite_domain=true
-template=/full_path/edgdomaintemplate.jar
-log_priority=DEBUG
-log=/tmp/unpack.log
-app_dir=$MS_APPLICATION_HOME
```

For Example:

```
cd $ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=$IGD_MSERVER_HOME \
-overwrite_domain=true \
-template=/tmp/edgdomaintemplate.jar \
-log_priority=DEBUG \
-log=/tmp/unpack.log \
-app_dir=$MS_APPLICATION_HOME
```

In this example:

- Replace *IGD_MSERVER_HOME* with the complete path to the domain home to be created on the local storage disk. This is the location where the copy of the domain will be unpacked.
- Replace *full_path* with the complete path and file name of the domain template jar file that you created when you ran the pack command to pack up the domain on the shared storage device.
- Replace $MS_APPLICATION_HOME with the complete path to the Application directory for the domain on shared storage. See File System and Directory Variables Used in This Guide.

> **Tip**
>
> For more information about the pack and unpack commands, see Overview of the Pack and Unpack Commands in *Creating Templates and Domains Using the Pack and Unpack Commands*.

5. Change directory to the newly created $IGD_MSERVER_HOME directory and verify that the domain configuration files were copied to the correct location on the OIGHOST2 local storage device.

Starting the Node Manager in the Managed Server Domain Directory OIGHOST2

After you manually set up the Node Manager to use a per host Node Manager configuration, you can start the Node Manager by using the following commands onOIGHOST2:

Copyright © 2018, 2025, Oracle and/or its affiliates.

1. Change directory to the Node Manager home directory:

   ```
   cd $NM_HOME
   ```

2. Run the following command to start the Node Manager and send the output of the command to an output file, rather than to the current terminal shell:

   ```
   nohup ./startNodeManager.sh > nodemanager.out 2>&1 &
   ```

Starting and Validating the soa_server2 and oim_server2 Managed Servers on OIGHOST2

Use the procedure in Starting and Validating the soa_server1 Managed Server on OIGHOST1 to start and validate the Managed Servers on OIGHOST2

1. Login to the WebLogic Remote Console.
2. Click **Monitoring Tree** and expand the **Environment** node, then select **Servers**.
3. From the Servers page, select **soa_server2** from the Servers column of the table and click **Start**.
4. Select **soa_server2** from the Servers column of the table and click **Start**.

Configuring OIG Workflow Notifications to be Sent by Email

OIG uses the human workflow, which is integrated with the SOA workflow. The SOA server configures email to receive the notifications that are delivered to the user mailbox. The user can accept or reject the notifications.

Both incoming and outgoing email addresses and mailboxes dedicated to the portal workflow are required for the full functionality. See Configuring Human Workflow Notification Properties in *Administering Oracle SOA Suite and Oracle Business Process Management Suite*.

To configure the OIG workflow notifications:

1. Log in to the Fusion Middleware Control by using the administrators account. For example, `weblogic_iam`.
2. Expand the Target Navigation panel and navigate to **SOA** > **soa-infra (soa_server1)** service.
3. From the SOA infrastructure drop-down, select **SOA Administration** > **Workflow Properties**.
4. Set the Notification mode to **Email**. Provide the correct e-mail address for the notification service.
5. Click **Apply** and confirm when prompted.
6. Verify the changes.
7. Expand **Target Navigation**, select **User Messaging Service**, and then **usermessagingdriver-email (soa_servern)**. Each SOA Managed Server that is running will have a driver. Only one of these entries should be selected.
8. From the **User Messaging Email Driver** drop-down list, select **Email Driver Properties**.
9. Click **Create** if the email driver does not exist already.
10. Click **Test** and verify the changes.

Copyright © 2018, 2025, Oracle and/or its affiliates.

11. Click **OK** to save the email driver configuration.
12. Restart the SOA cluster. No configuration or restart is required for OIG.

Adding the wsm-pm Role to the Administrators Group

After you configure a new LDAP-based Authorization Provider and restart the Administration Server, add the enterprise deployment administration LDAP group (OIMAdministrators) as a member to the `policy.Updater` role in the `wsm-pm` application stripe.

1. Sign in to the Fusion Middleware Control by using the administrator's account. For example: `weblogic_iam`.
2. From the **WebLogic Domain** menu, select **Security**, and then **Application Roles**.
3. Select the **wsm-pm** application stripe from the Application Stripe drop-down menu.
4. Click the triangular icon next to the role name text box to search for all role names in the wsm-pm application stripe.
5. Select the row for the **policy.Updater** role to be edited.
6. Click the Application Role **Edit** icon to edit the role.
7. Click the Application Role **Add** icon on the Edit Application Role page.
8. In the Add Principal dialog box, select **Group** from the **Type** drop-down menu.
9. To search for the enterprise deployment administrators group, enter the group name `WLSAdministrators` in the **Principal Name Starts With** field and click the right arrow to start the search.
10. Select the appropriate administrators group in the search results and click **OK**.
11. Click **OK** on the Edit Application Role page.

Adding the Oracle Access Manager Load Balancer Certificate to the Oracle Keystore Service

The Oracle Identity Governance to Business Intelligence Reports link inside of the Self Service application requires that the SSL certificate used by the load balancer be added to the Oracle Keystore Service Trusted Certificates.

To add the certificate, do the following:

1. Create a directory to hold user created keystores and certificates if it doesn't already exist.

   For example:

   ```
   mkdir $SHARED_CONFIG_DIR/keystores
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

2. Obtain the certificate from the load balancer. You can obtain the load balancer certificate from using a browser, such as Firefox. However, the easiest way to obtain the certificate is to use the `openssl` command. The syntax of the command is as follows:

   ```
   openssl s_client -connect <LOADBALANCER>:<PORT> -showcerts </dev/null
   2>/dev/null| sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p' >
   LOADBALANCER.pem
   ```

   For example:

   ```
   openssl s_client -connect login.example.com:443 -showcerts </dev/null
   2>/dev/null| sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p' >
   login.example.com.pem
   ```

   The `openssl` command saves the certificate to a file called `login.example.com.pem` in `$SHARED_CONFIG_DIR/keystores`.

3. Load the certificate into the Oracle Keystore Service using WLST.

   a. Connect to WLST using the following command:

      ```
      $ORACLE_COMMON_HOME/common/bin/wlst.sh
      ```

   b. Connect to the Administration Server using the following command:

      ```
      connect('<AdminUser>','<AdminPwd>','t3://<Adminserverhost>:<Adminserver
      port>')
      ```

   c. Load the certificate using the following commands:

      ```
      svc = getOpssService(name='KeyStoreService')
      svc.importKeyStoreCertificate(appStripe='system',name='trust',password='
      ', keypassword='',alias='<CertificateName>',type='TrustedCertificate',
      filepath='/<SHARED_CONFIG_DIR>/keystores/<LOADBALANCER>.pem')
      ```

   d. Synchronize the Keystore Service with the file system using the following command:

      ```
      syncKeyStores(appStripe='system', keystoreFormat='KSS')
      ```

      For example:

      ```
      connect('weblogic','password','t3://igdadminvhn.example.com:7101')
      svc = getOpssService(name='KeyStoreService')
      svc.importKeyStoreCertificate(appStripe='system',name='trust',password='
      ', keypassword='',alias='login.example.com',type='TrustedCertificate',
      filepath='/u01/oracle/config/keystores/login.example.com.pem')
      syncKeyStores(appStripe='system',keystoreFormat='KSS')
      exit()
      ```

You will need to restart the domain for the changes to take effect. The default password for the Node Manager keystores is *COMMON_IAM_PASSWORD*. You will be prompted to confirm that the certificate is valid.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Restarting the IAMGovernanceDomain

Restart the domain for the configuration steps to take effect.

1. Shut down the Managed Servers oim_server1 and oim_server2.
2. Shut down the Managed Servers soa_server1 and soa_server2.
3. Shut down the Administration Server.
4. Restart the Administration Server.
5. Start the Managed Servers soa_server1 and soa_server2.
6. Start the Managed Servers oim_server1 and oim_server2.

Setting Challenge Questions

If you have integrated OAM and OIG, then after the environment is ready, you need to set up the challenge questions for your system users.

To set up the challenge questions, log in to Identity Self Service using the URL outlined in URLs Used in This Chapter.

Log in with your user name and when prompted, add the challenge questions. You should set up these questions for the following users:

- `xelsysadm`
- `weblogic_iam`
- `oamadmin`

Replacing Connect Strings with the Appropriate TNS Alias

Oracle recommends using TNS Alias in the connection strings used by FMW components instead of repeating long JDBC strings across multiple connections pools.

For more information about how to use TNS alias in your Datasources, see Using TNS Alias in Connect Strings in the *Common Configuration and Management Tasks for an Enterprise Deployment* chapter.

Integrating Oracle Identity Manager with Oracle Business Intelligence Publisher

Oracle Identity Manager comes with a number of prebuilt reports that can be used to provide information about Oracle Identity and Access Management.

Oracle Identity Manager reports are classified based on the functional areas such as Access Policy Reports, Request and Approval Reports, Password Reports, and so on. It is no longer named Operational and Historical. These reports are not generated through Oracle Identity Manager but by the Oracle Business Intelligence Publisher (BIP). Oracle Identity Manager reports provide a restriction for Oracle BI Publisher.

Copyright © 2018, 2025, Oracle and/or its affiliates.

The setup of a highly available enterprise deployment of Oracle BI Publisher is beyond the scope of this document. For more information, see Understanding the Business Intelligence Enterprise Deployment Topology in the *Enterprise Deployment Guide for Business Intelligence*.

> **Note**
>
> During BI configuration for Oracle Identity Manager, you must configure only Business Intelligence Publisher. If you select other components during BI Publisher configuration, such as Business Intelligence Enterprise Edition and Essbase, the integration with Oracle Identity Manager may not work. See Configuring Reports in *Developing and Customizing Applications for Oracle Identity Manager*

- Creating a User to Run BI Reports
- Configuring Oracle Identity Manager to Use BI Publisher
- Assigning the BIServiceAdministrator Role to idm_report
- Storing the BI Credentials in Oracle Identity Governance
- Creating OIM and BPEL Data Sources in BIP
- Deploying Oracle Identity Governance Reports to BI
- Enable Certification Reports
- Validating the Reports

Creating a User to Run BI Reports

You may ignore this section if you already have a user to run reports in your Business Intelligence domain.

If you need to create a user in your BI Publisher domain to run reports, use the following `LDIF` command to create a user in the LDAP directory.

1. Create a file called `report_user.ldif` with the following contents:

```
dn: cn=idm_report,cn=Users,dc=example,dc=com
changetype: add
orclsamaccountname: idm_report
givenname: idm_report
sn: idm_report
userpassword: <password>
mail: idm_report
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetorgperson
objectclass: orcluser
objectclass: orcluserV2
uid: idm_report
cn: idm_report
```

2. Save the file.

Copyright © 2018, 2025, Oracle and/or its affiliates.

3. Load the file into the LDAP directory using the following command:

```
ldapmodify -D cn=oudadmin -h idstore.example.com -p 1389 report_user.ldif
```

Configuring Oracle Identity Manager to Use BI Publisher

You can set up Oracle BI Publisher to generate Oracle Identity Manager reports.

To configure Oracle Identity Manager to use the BI Publisher:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control using the URL outlined in URLs Used in This Chapter.
2. Click WebLogic Domain, and then select **System MBean Browser**.
3. Enter `XMLConfig.DiscoveryConfig` as the search criteria and click **Search**.

   The XMLConfig.DiscoveryConfig MBean is displayed.
4. Update the value of the **Discovery Config BI publisher URL** to the BIP URL. For example, `http://bi.example.com`
5. Click **Apply**.

Assigning the BIServiceAdministrator Role to idm_report

If you are using LDAP as your identity store in the Business Intelligence (BI) domain, you must have created an LDAP authenticator in the BI domain. You can view the user and group names stored within LDAP.

The Oracle Identity Manager (OIM) system administration account (for example, `idm_report`) needs to be assigned the `BIServiceAdministrator` role, to generate reports.

To assign this role:

1. Ensure that the OIM administrator user is visible in the domain by logging in to the BI publisher WebLogic Console using the following URL:

   `http://biadmin.example.com/console`
2. Click **Security Realms**, and then click **myrealm**.
3. Go to the **Users and Groups** tab.
4. Look at the list of users and ensure that the user OIM Administration User (`idm_report`) is in the list of users.
5. Sign in to the BI Fusion Middleware Control by using the URL `http://biadmin.example.com/em` and the administrator's account. For example: *weblogic_bi*.
6. From the **WebLogic Domain** menu, select **Security**, and then **Application Roles**.
7. From the **Application Stripe** drop-down list, select **obi**.
8. Click the triangular icon next to the role name text box to search for all role names in the **obi** application stripe.
9. Select the row for the **BIServiceAdministrator** role to edit.
10. Click the Application Role Edit icon to edit the role.
11. Click the Application Role Add icon on the Edit Application Role page.
12. In the Add Principal dialog box, select **User** from the **Type** drop-down menu.

Copyright © 2018, 2025, Oracle and/or its affiliates.

13. To search for the `idm_report` user, enter the user name `idm_report` in the **Principal Name Starts With** field and click the right arrow to start the search.
14. Select the appropriate user in the search results and click **OK**.
15. Click **OK** on the Edit Application Role page.

Storing the BI Credentials in Oracle Identity Governance

To configure BIP credentials in Oracle Identity Governance:

1. Log in to the Oracle Enterprise Manager using the using the URL outlined in URLs Used in This Chapter.
2. In the left pane, expand the **Weblogic Domain**. The domain name is displayed.
3. Right-click the domain name, and navigate to **Security**, and then **Credentials**. A list of maps in the credential store, including the oim map, is displayed.
4. Expand the oim map. A list of entries of type Password is displayed.
5. Edit the `BIPWSKey` key if it already exists, or create a new one with the following values:

Table 15-16 Properties of a new CSF entry

Attribute	Value
Select Map	oim
Key	BIPWSKey
Type	Password
Username	`idm_report`
Password	`idm_report` password
Description	Login credentials for BI Publisher web service

Creating OIM and BPEL Data Sources in BIP

Create OIM Datasource

Oracle BIP must be connected to the OIM and SOA database schemas to run a report.

In order to do this you need to create BIP datasources using the following procedure:

1. Login to the BI Publisher Home page using the URL `https://bi.example.com/xmlpserver`
2. Click the **Administration** link on the top of the BI Publisher Home page. The BI Publisher Administration page is displayed.
3. Under Data Sources, click **JDBC Connection** link. The Data Sources page is displayed.
4. In the JDBC tab, click **Add Data Source** to create a JDBC connection to your database. The Add Data Source page is displayed.
5. Enter values in the following fields:

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-17 OIM Add Data Source Attributes

Attributes	Value
Data Source Name	Specify the Oracle Identity Governance JDBC connection name. For example, OIM JDBC.
Driver Type	Select the driver type appropriate to the database version you are using
Database Driver Class	Specify a driver class to suit your database, such as `oracle.jdbc.OracleDriver`
Connection String	Specify the database connection details in the format `jdbc:oracle:thin:@HOST_NAME:PORT_NUMBER/SID`. For example, `jdbc:oracle:thin:@igddbscan:1521/oim.example.com`
User name	Specify the Oracle Identity Governance database user name for example IGD_OIM
Password	Specify the Oracle Identity Governance database user password.

6. Click **Test Connection** to verify the connection.
7. Click **Apply** to establish the connection.
8. If the connection to the database is established, a confirmation message is displayed indicating the success.
9. Click **Apply**.

In the JDBC page, you can see the newly defined Oracle Identity Governance JDBC connection in the list of JDBC data sources.

Create BPEL Datasource

1. Login to the BI Publisher Home page using the URL `https://bi.example.com/xmlpserver`.
2. Click the **Administration** link on the **BI Publisher** home page. The **BI Publisher Administration** page is displayed.
3. Under Data Sources, click **JDBC Connection** link. The Data Sources page is displayed.
4. In the JDBC tab, click **Add Data Source** to create a JDBC connection to your database. The Add Data Source page is displayed.
5. Enter values in the following fields:

Table 15-18 JDBC Add Data Source Attributes

Attributes	Value
Data Source Name	Specify the Oracle Identity Governance JDBC connection name. For example, BPEL JDBC.
Driver Type	Select the Driver for the database version you are using.
Database Driver Class	Specify a driver class to suit your database, such as `oracle.jdbc.OracleDriver`

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 15-18 (Cont.) JDBC Add Data Source Attributes

Attributes	Value
Connection String	Specify the database connection details in the format `jdbc:oracle:thin:@HOST_NAME:PORT_NUMBER/SID`. For example, `jdbc:oracle:thin:@igddbscan:1521/oim.example.com`
User name	Specify the Oracle Identity Governance database user name for example IGD_SOAINFRA.
Password	Specify the Oracle Identity Governance database user password.

6. Click **Test Connection** to verify the connection.
7. Click **Apply** to establish the connection.
8. If the connection to the database is established, a confirmation message is displayed indicating the success.
9. Click **Apply**.

In the JDBC page, you can see the newly defined Oracle Identity Governance JDBC connection in the list of JDBC data sources.

Deploying Oracle Identity Governance Reports to BI

After **BI Publisher** is integrated with Oracle Identity Governance, you can deploy the predefined reports for using them. To deploy Oracle Identity Manager reports:

1. Copy and unzip the predefined report `$IGD_ORACLE_HOME/idm/server/reports/oim_product_BIPReports_12c.zip` located on OIGHOST1 file to the directory `$SHARED_CONFIG_DIR/biconfig/bidata`.

   > **Note**
   >
   > The $SHARED_CONFIG_DIR is defined in the `$IGD_ASERVER_HOME/config/fmwconfig/bienv/core/bi-environment.xml` file.

2. Add folder level permission to the **BIServiceAdministrator** BI application role to view and run the predefined Oracle Identity Governance reports. To do so:
   - Login to Oracle BI Publisher `https://bi.example.com/xmlpserver` by using the WebLogic admin credentials.
   - Click the **Catalog** link at the top. The Oracle Identity Manager named folder under shared folders is displayed in the left pane. Select the Oracle Identity Manager named folder.
   - Click **Permissions** option under the **Tasks** window on the bottom left.
   - Click the plus sign and perform a blank search on the available role.
   - Select the **BI Service Administrator** role, and add to the right panel.
   - Click **Ok**.
3. Logout as WebLogic user.

Copyright © 2018, 2025, Oracle and/or its affiliates.

4. Login as the Oracle Identity Manager system administrator user to **BI Publisher console**.
5. Run the Oracle Identity Manager reports.

Enable Certification Reports

Select or deselect the **Enable Certification Reports** option to enable or disable the certification reports. To enable the generation of certification reports, after configuring the **BI Publisher** credentials and URL, perform the following:

1. Log in to the Oracle Identity Self Service using the URL outlined in URLs Used in This Chapter.
2. Click the **Compliance** tab.
3. Click the **Identity Certification** box.
4. Select **Certification Configuration**. The Certification Configuration page is displayed.
5. Select the **Enable Certification Reports**.
6. Click **Save**.

> **Note**
>
> By default, the **Compliance** tab is not shown. If you want to enable compliance functionality, you must fist set the `OIGIsIdentityAuditorEnabled` property to `true` in the Sysadmin Console (located in the **Configuration Properties** section).

Validating the Reports

We need to create the sample data source to generate reports against the sample data source.

Creating the Sample Reports

To view an example report data without running a report against the production JDBC Data Source, generate a sample report against the sample data source. Create the sample data source before you can generate the sample reports.

- Generating Reports Against the Sample Data Source
- Generating Reports Against the Oracle Identity Manager JDBC Data Source
- Generating Reports Against the BPEL-Based JDBC Data Source

Generating Reports Against the Sample Data Source

After you create the sample data source, you can generate sample reports against it by performing the following steps:

1. Login to Oracle BI Publisher using the url : `https://bi.example.com/xmlpserver`.
2. Click **Shared Folders**.
3. In the Templates screen, ensure **Create Domain Using Product Templates** is selected and then select the **Oracle Identity Manager template**.
4. Select **Sample Reports**.
5. Click **View** for the sample report you want to generate.

Copyright © 2018, 2025, Oracle and/or its affiliates.

6. Select an output format for the sample report and click **View**.

The sample report is generated.

Generating Reports Against the Oracle Identity Manager JDBC Data Source

To generate reports against the OIM JDBC data source, navigate to the Oracle Identity Manager reports by logging in to the Oracle BI Publisher, and select an output format for the report you want to generate.

To generate reports against the Oracle Identity Manager JDBC data source:

1. Log in to Oracle BI Publisher using the URL: `https://bi.example.com/xmlpserver`.
2. Navigate to Oracle Identity Manager reports. To do so:
   - In the **BI Publisher** home page, under Browse or Manage, click **Catalog Folders**. Alternatively, you can click **Catalog** at the top of the page.

     The Catalog page is displayed with a tree structure on the left side of the page and the details on the right.
   - On the left pane, expand **Shared Folders**, and navigate to the Oracle Identity Manager. All the objects in the Oracle Identity Manager folder are displayed.

     You are ready to navigate to BI Publisher and use the Oracle Identity Manager BI Publisher reports.
3. Click **View** under the report you want to generate.
4. Select an output format for the report and click **View**.

The report is generated.

Generating Reports Against the BPEL-Based JDBC Data Source

Some reports have a secondary data source, which is BPEL-based JDBC data source. This section describes how to generate reports against the BPEL-based JDBC data source.

Reports With Secondary Data Source

The following four reports have a secondary data source, which connects to the BPEL database to retrieve the BPEL data:

- Task Assignment History
- Request Details
- Request Summary
- Approval Activity

These reports have a secondary data source (BPEL-based JDBC data source) called BPEL JDBC. To generate reports against the BPEL-based JDBC data source:

1. Log in to Oracle BI Publisher using the url: `https://bi.example.com/xmlpserver`.
2. Navigate to the Oracle Identity Manager reports. To do so:
   - In the **BI Publisher** home page, under Browse or Manage, click **Catalog Folders**. Alternatively, you can click **Catalog** at the top of the page.

     The catalog page is displayed with a tree structure on the left side of the page and the details on the right.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- On the left pane, expand **Shared Folders**, and navigate to the Oracle Identity Manager. All the objects in the Oracle Identity Manager folder is displayed.

  Navigate to the BI Publisher 12*c* and use the Oracle Identity Manager BI Publisher reports.

3. Select the report you want to generate and click **Open**.
4. Select an output format for the report, and click **Apply**.

The report is generated based on the BPEL-based JDBC data source.

- Adding the Business Intelligence Load Balancer Certificate to Oracle Keystore Trust Service

Adding the Business Intelligence Load Balancer Certificate to Oracle Keystore Trust Service

The Oracle Identity Governance to Business Intelligence Reports link inside of the Self Service application requires that the SSL certificate used by the load balancer be added to the Oracle Keystore Service Trusted Certificates.

To add the certificate:

1. Create a directory to hold user created keystores and certificates if it doesn't already exist.

   For example:

   ```
   mkdir $SHARED_CONFIG_DIR/keystores
   ```

2. Obtain the certificate from the load balancer. You can obtain the load balancer certificate from using a browser, such as Firefox. However, the easiest way to obtain the certificate is to use the `openssl` command. The syntax of the command is as follows:

   ```
   openssl s_client -connect LOADBALANCER -showcerts </dev/null 2>/dev/null|
   openssl x509 -outform PEM>SHARED_CONFIG_DIR/keystores/LOADBALANCER.pem
   ```

   For example:

   ```
   openssl s_client -connect bi.example.com:443 -showcerts </dev/null 2>/dev/
   null|openssl x509 -outform PEM>$SHARED_CONFIG_DIR/keystores/
   bi.example.com.pem
   ```

   The `openssl` command saves the certificate to a file called `bi.example.com.pem` in `$SHARED_CONFIG_DIR/keystores`.

3. Load the certificate into the Oracle Keystore Service using WLST.
   a. Connect to WLST using the following command:

      ```
      $ORACLE_COMMON_HOME/common/bin/wlst.sh
      ```

   b. Connect to the Administration Server using the following command:

      ```
      connect('<AdminUser>','<AdminPwd>','t3://<Adminserverhost>:<Adminserver
      port>')
      ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

c. Load the certificate using the following commands:

```
svc = getOpssService(name='KeyStoreService')
svc.importKeyStoreCertificate(appStripe='system',name='trust',password='
', keypassword='',alias='<CertificateName>',type='TrustedCertificate',
filepath='/<SHARED_CONFIG_DIR>/keystores/<LOADBALANCER>.pem')
```

d. Synchronize the Keystore Service with the file system using the following command:

```
syncKeyStores(appStripe='system', keystoreFormat='KSS')
```

For example:

```
connect('weblogic','password','t3://IGDADMINVHN.example.coml:7101')
svc = getOpssService(name='KeyStoreService')
svc.importKeyStoreCertificate(appStripe='system',name='trust',password='
', keypassword='',alias='bi.example.com',type='TrustedCertificate',
filepath='/u01/oracle/config/keystores/bi.example.com.pem')
syncKeyStores(appStripe='system',keystoreFormat='KSS')
exit()
```

You will need to restart the domain for the changes to take effect. The default password for the JDK is *changeit*. The default password for the Node Manager keystores is *COMMON_IAM_PASSWORD*. You will be prompted to confirm that the certificate is valid.

Verification of Manual Failover of the Administration Server

After you configure the domain, you must test failover is working correctly.

Perform the steps described in Verifying Manual Failover of the Administration Server.

Backing Up the Configuration

It is an Oracle best practices recommendation to create a backup after you successfully extended a domain or at another logical point. Create a backup after you verify that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

The backup destination is the local disk. You can discard this backup when the enterprise deployment setup is complete. After the enterprise deployment setup is complete, you can initiate the regular deployment-specific Backup and Recovery process.

For information about backing up your configuration, see Performing Backups and Recoveries for an Enterprise Deployment.

Copyright © 2018, 2025, Oracle and/or its affiliates.

16

Configuring Oracle HTTP Server for an Enterprise Deployment

Install and configure Oracle HTTP Server (OHS) to direct HTTP(S) requests from the hardware load balancer to specific Managed Servers in the application tier.

For an enterprise deployment, Oracle HTTP Server must be installed on each of the web tier hosts and configured as Oracle HTTP standalone domains on each host.

In an SSL Terminated deployment, the load balancer communicates with OHS over the HTTP protocol and the OHS instances are also communicate over the HTTP Protocol to the backend WebLogic servers.

In end-to-end SSL deployments, the Load balancer communicates with OHS over SSL protocol for a more secure configuration. The OHS instances also communicate over SSL protocol with the specific Managed Servers in the application tier. SSL is configured all the way from the LBR to the backend WebLogic servers.

Before you configure Oracle HTTP Server, be sure to review Understanding the Web Tier.

- About the Oracle HTTP Server Domains
  In an enterprise deployment, each Oracle HTTP Server instance is configured on a separate host and in its own standalone domain. This allows for a simple configuration that requires a minimum amount of configuration and a minimum amount of resources to run and maintain.
- Variables Used When Configuring the Oracle HTTP Server
  As you perform the tasks in this chapter, you will be referencing the variables listed in this section.
- Setting Environment Variables
  Set environment variables used in this chapter.
- Installing Oracle HTTP Server on WEBHOST1
  Install the Oracle HTTP Server software on the web tier by using the Oracle Universal Installer. Verify the installation after you complete the procedure.
- Creating an Oracle HTTP Server Domain on WEBHOST1
  You can create a new Oracle HTTP Server standalone domain on the first web tier host by using the Configuration Wizard.
- Installing and Configuring an Oracle HTTP Server Domain on WEBHOST2
  After you install Oracle HTTP Server and configure a domain on WEBHOST1, then you must also perform the same tasks on WEBHOST2.
- Starting the Node Manager and Oracle HTTP Server Instances on WEBHOST1 and WEBHOST2
  It is important to understand how to start the Oracle HTTP Server instances on WEBHOST1 and WEBHOST2.
- Backing Up the Configuration
  It is an Oracle best practices recommendation to create a backup after you successfully extended a domain or at another logical point. Create a backup after you verify that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Generate Required Certificates for OHS SSL Listeners
  In an End to End SSL deployment the OHS listeners use SSL and you must create appropriate certificates for them.
- Configuring Oracle HTTP Server to Route Requests to the Application Tier
  Update the Oracle HTTP Server configuration files so that the web server instances route requests to the servers in the domain.
- Validating the Virtual Server Configuration and Access to the Consoles
  Validate the virtual server configuration on the load balancer, and the access to the management console and the Administration Server.
- Sample Virtual Host Files
  The sample list includes the complete examples of all the virtual host files used in an Oracle Identity and Access Management deployment.

About the Oracle HTTP Server Domains

In an enterprise deployment, each Oracle HTTP Server instance is configured on a separate host and in its own standalone domain. This allows for a simple configuration that requires a minimum amount of configuration and a minimum amount of resources to run and maintain.

> **Note**
>
> Oracle Fusion Middleware requires that a certified Java Development Kit (JDK) is installed on your system and JAVA_HOME is set on the web tier hosts.

For more information about the role and configuration of the Oracle HTTP Server instances in the web tier, see Understanding the Web Tier.

Variables Used When Configuring the Oracle HTTP Server

As you perform the tasks in this chapter, you will be referencing the variables listed in this section.

The following table explains the configuration file property values required in this section.

Table 16-1 OHS Variables Used in This Chapter

Variable	Sample Value	Description
WEB_ORACLE_HOME	`/u02/oracle/products/ohs`	The read-only location for the Oracle HTTP Server product binaries. For the web tier host computers, this directory is stored on the local disk.
WEB_DOMAIN_HOME	`/u02/oracle/config/domains/ohsDomain`	The Domain home for the standalone Oracle HTTP Server domain, which is created when you install Oracle HTTP Server on the local disk of each web tier host.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 16-1 (Cont.) OHS Variables Used in This Chapter

Variable	Sample Value	Description
JAVA _HOME	`/u02/oracle/products/jdk`	The location where you install the supported Java Development Kit (JDK).
IADADMINVHN	`iadadminvhn.example.com`	The virtual host name used as the listen address for the Administration Server used by the IAMAccessDomain and fails over with manual failover of the Administration Server. It is enabled on the node where the Administration Server process is running.
IGDADMINVHN	`igdadminvhn.example.com`	The virtual host name used as the listen address for the Administration Server used by the IAMGovernanceDomain and fails over with manual failover of the Administration Server. It is enabled on the node where the Administration Server process is running.
WEBHOST1	`webhost1.example.com`	The hostname of WEBHOST1.
WEBHOST2	`webhost2.example.com`	The hostname of WEBHOST2.

Setting Environment Variables

Set environment variables used in this chapter.

To help navigate this guide, to be able to copy sample commands without modification you can set the following environment variables, replacing the values with values appropriate to your environment.

```
export WEB_ORACLE_HOME=/u02/oracle/products/ohs
export JAVA_HOME=/u02/oracle/products/jdk
export PATH=$JAVA_HOME/bin:$PATH
```

Installing Oracle HTTP Server on WEBHOST1

Install the Oracle HTTP Server software on the web tier by using the Oracle Universal Installer. Verify the installation after you complete the procedure.

- Starting the Installer on WEBHOST1
- Navigating the Oracle HTTP Server Installation Screens
- Verifying the Oracle HTTP Server Installation
- Installing a Supported JDK

Copyright © 2018, 2025, Oracle and/or its affiliates.

Starting the Installer on WEBHOST1

To start the installation program, perform the following steps.

1. Log in to WEBHOST1.
2. Go to the directory in which you downloaded the installation program.
3. Enter the following command to launch the installation program:

```
./fmw_14.1.2.0.0_ohs_linux64.bin
```

When the installation program appears, you are ready to begin the installation.

Navigating the Oracle HTTP Server Installation Screens

The following table lists the screens in the order that the installation program displays them.

If you need additional help with any of the installation screens, click the Help button on the screen.

Table 16-2 Oracle HTTP Server Installation Screens

Screen	Description
Installation Inventory Setup	On UNIX operating systems, this screen appears if you install any Oracle product on this host for the first time. Specify the location where you want to create your central inventory. Ensure that the operating system group name selected on this screen has write permissions to the central inventory location. See Understanding the Oracle Central Inventory in *Installing Software with the Oracle Universal Installer*. **Note** Oracle recommends that you configure the central inventory directory within the products directory. Example: `/u02/oracle/products/oraInventory` You may also need to execute the `createCentralinventory.sh` script as root from the `oraInventory` folder after the installer completes.
Welcome	This screen introduces you to the product installer.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 16-2 (Cont.) Oracle HTTP Server Installation Screens

Screen	Description
Auto Updates	Use this screen to automatically search My Oracle Support for available patches or automatically search the local directory for patches that you have already downloaded for your organization.
Installation Location	Use this screen to specify the location of your Oracle home directory. For the purposes of an enterprise deployment, enter the value of the *WEB_ORACLE_HOME* variable listed in Table 8-3. For Example: `/u02/oracle/products/ohs`
Installation Type	Select **Standalone HTTP Server (Managed independently of WebLogic server)**. This installation type allows you to configure the Oracle HTTP Server instances independently from any other existing Oracle WebLogic Server domains.
JDK Selection	For the value of JDK Home, enter the value of *JAVA_HOME* that you set when installing the JDK software.
Prerequisite Checks	This screen verifies that your system meets the minimum necessary requirements. If there are any warning or error messages, verify that your host computers and the required software meet the system requirements and certification information described in Host Computer Hardware Requirements and Operating System Requirements for the Enterprise Deployment Topology.
Installation Summary	Use this screen to verify the installation options that you selected. If you want to save these options to a response file, click **Save Response File** and provide the location and name of the response file. Response files can be used later in a silent installation situation. See Using the Oracle Universal Installer in Silent Mode in *Installing Software with the Oracle Universal Installer*.
Installation Progress	This screen allows you to see the progress of the installation.
Installation Complete	This screen appears when the installation is complete. Review the information on this screen, then click **Finish** to close the installer.

Verifying the Oracle HTTP Server Installation

Verify that the Oracle HTTP Server installation completed successfully by validating the `WEB_ORACLE_HOME` folder contents.

Run the following command to compare the installed folder structure with the following list:

1. Navigate to the $ORACLE_HOME:

```
cd $ORACLE_HOME
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

2. Run the following command:

```
ls --format=single-column $WEB_ORACLE_HOME
```

The following files and directories are listed in theOracle HTTP Server Oracle Home:

```
assistants
bin
cfgtoollogs
clone
crs
crypto
css
cv
deinstall
drdaas
env.ora
has
hs
install
instantclient
inventory
javavm
jdbc
jlib
jpub
ldap
lib
network
nls
odbc
ohs
olap
OPatch
opmn
oracle_common
oracore
oraInst.loc
ord
oss
oui
perl
plsql
plugins
precomp
QOpatch
racg
rdbms
root.sh
schagent.conf
sdk
slax
sqlcl
sqlj
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
sqlplus
srvm
suptools
ucp
unixODBC
usm
utl
webgate
wlserver
xdk
```

Installing a Supported JDK

Oracle Fusion Middleware requires that a certified Java Development Kit (JDK) is installed on your system.

- Locating and Downloading the JDK Software
- Installing the JDK Software
  Oracle Fusion Middleware requires you to install a certified Java Development Kit (JDK) on your system.

Locating and Downloading the JDK Software

To find a certified JDK, see the certification document for your release on the Oracle Fusion Middleware Supported System Configurations page.

After you identify the Oracle JDK for the current Oracle Fusion Middleware release, you can download an Oracle JDK from the following location on Oracle Technology Network:

```
https://www.oracle.com/java/technologies/downloads/
```

Be sure to navigate to the download for the Java SE JDK.

Installing the JDK Software

Oracle Fusion Middleware requires you to install a certified Java Development Kit (JDK) on your system.

You must install the JDK in the following locations:

On the local storage device for each of the Web tier host computers. The Web tier host computers, which reside in the DMZ, do not necessarily have access to the shared storage on the application tier.

For more information about the recommended location for the JDK software, see Understanding the Recommended Directory Structure for an Enterprise Deployment.

To install JDK 21.0:

1. Change directory to the location where you downloaded the JDK archive file.

   ```
   cd download_dir
   ```

2. Unpack the archive into the JDK home directory, and then run the following commands:

   ```
   tar -xzvf jdk-21.0.4+8_linux-x64_bin.tar.gz
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

Note that the JDK version listed here was accurate at the time this document was published. For the latest supported JDK, see the *Oracle Fusion Middleware System Requirements and Specifications* for the current Oracle Fusion Middleware release.

3. Move the JDK directory to the recommended location in the directory structure.

   For example:

   ```
   mv ./jdk-21.0.4 /u02/oracle/products/jdk
   ```

   See File System and Directory Variables Used in This Guide.

4. Run the following command to verify that the appropriate `java` executable is in the path and your environment variables are set correctly:

   ```
   java -version
   ```

   The Java version in the output should be displayed as `21.0.4`.

Creating an Oracle HTTP Server Domain on WEBHOST1

You can create a new Oracle HTTP Server standalone domain on the first web tier host by using the Configuration Wizard.

- Starting the Configuration Wizard on WEBHOST1
- Navigating the Configuration Wizard Screens for an Oracle HTTP Server Domain

Starting the Configuration Wizard on WEBHOST1

To start the Configuration Wizard, navigate to the following directory and start the WebLogic Server Configuration Wizard, as follows:

```
cd $WEB_ORACLE_HOME/oracle_common/common/bin
```

```
./config.sh
```

Navigating the Configuration Wizard Screens for an Oracle HTTP Server Domain

Oracle recommends that you create a standalone domain for the Oracle HTTP Server instances on each web tier host.

The following topics describe how to create a new standalone Oracle HTTP Server domain:

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 16-3 Navigating the Infrastructure Installation Screens

Screen	Description
Selecting the Domain Type and Domain Home Location	On the Configuration Type screen, select **Create a new domain**. In the **Domain Location** field, specify the value of the <WEB_DOMAIN_HOME> variable, as defined in File System and Directory Variables Used in This Guide. For example, `/u02/oracle/products/ohs`. **Note** • The Configuration Wizard creates the new directory that you specify here. • Create the directory on local storage, so the web servers do not have any dependencies on storage devices outside the DMZ. **Tip** • More information about the Domain home directory can be found in About the Domain Home Directory in *Planning an Installation of Oracle Fusion Middleware*. • More information about the other options on this screen can be found in Configuration Type in *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard*. • For more information about the web tier and the DMZ, see Understanding the Firewalls and Zones of a Typical Enterprise Deployment. • For more information about the *<WEB_DOMAIN_HOME>* directory variable, see File System and Directory Variables Used in This Guide. Click **Next**.
Selecting the Configuration Templates	On the Templates screen, make sure **Create Domain Using Product Templates** is selected, then select the following templates: • **Oracle HTTP Server (Standalone) - [ohs]** **Tip** More information about the options on this screen can be found in Templates in *Creating WebLogic Domains Using the Configuration Wizard*. Click **Next**.
Selecting the JDK for the Web Tier Domain	Select the Oracle HotSpot JDK installed in the `/u02/oracle/products/jdk` directory prior to the Oracle HTTP Server installation. Click **Next**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 16-3 (Cont.) Navigating the Infrastructure Installation Screens

Screen	Description
Configuring System Components	On the System Components screen, configure one Oracle HTTP Server instance. The screen should, by default, have a single instance defined. This is the only instance that you need to create. 1. The default instance name in the **System Component** field is `ohs1`. Use this default name when you configure `WEBHOST1`. 2. Make sure that `OHS` is selected in the **Component Type** field. 3. If an application is not responding, use the **Restart Interval Seconds** field to specify the number of seconds to wait before you attempt a restart if an application is not responding. 4. Use the **Restart Delay Seconds** field to specify the number of seconds to wait between restart attempts. Click **Next**.
Configuring OHS Server	Use the OHS Server screen to configure the OHS servers in your domain: 1. Select **ohs1** from the **System Component** drop-down menu. 2. In the **Listen Address** field, enter *webhost1.example.com*. All the remaining fields are prepopulated, but you can change the values as required for your organization. See OHS Server in *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard*. 3. In the **Server Name** field, verify the value of the listen address and listen port. It should appear as follows: `http://webhost1.example.com:7777`
Configuring Node Manager	Select **Per Domain Default Location** as the Node Manager type, and specify the user name and password for the Node Manager. **Note** For more information about the options on this screen, see Node Manager in *Creating WebLogic Domains Using the Configuration Wizard*. For information about Node Manager configuration, see Configuring Node Manager on Multiple Machines in *Administering Node Manager for Oracle WebLogic Server*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 16-3 (Cont.) Navigating the Infrastructure Installation Screens

Screen	Description
Reviewing Your Configuration Specifications and Configuring the Domain	The Configuration Summary screen contains the detailed configuration information for the domain you are about to create. Review the details of each item on the screen and verify that the information is correct. You can go back to any previous screen if you need to make any changes, either by using the **Back** button or by selecting the screen in the navigation pane. Domain creation will not begin until you click **Create**. **Tip** More information about the options on this screen can be found in Configuration Summary in *Creating WebLogic Domains Using the Configuration Wizard*. Click **Next**.
Writing Down Your Domain Home and Administration Server URL	The Configuration Success screen shows the domain home location. Make a note of the information provided here, as you need it to start the servers and access the Administration Server. Click **Finish** to close the Configuration Wizard.

Installing and Configuring an Oracle HTTP Server Domain on WEBHOST2

After you install Oracle HTTP Server and configure a domain on WEBHOST1, then you must also perform the same tasks on WEBHOST2.

1. Log in to WEBHOST2 and install Oracle HTTP Server by using the instructions in Installing Oracle HTTP Server on WEBHOST1.
2. Configure a new standalone domain on WEBHOST2 by using the instructions in Creating a Web Tier Domain on WEBHOST1.

   Use the name `ohs2` for the instance on WEBHOST2, and be sure to replace all occurrences of WEBHOST1 with WEBHOST2 and all occurrences of `ohs1` with `ohs2` in each of the examples.

Starting the Node Manager and Oracle HTTP Server Instances on WEBHOST1 and WEBHOST2

It is important to understand how to start the Oracle HTTP Server instances on WEBHOST1 and WEBHOST2.

- Starting the Node Manager on WEBHOST1 and WEBHOST2
- Starting the Oracle HTTP Server Instances

Copyright © 2018, 2025, Oracle and/or its affiliates.

Starting the Node Manager on WEBHOST1 and WEBHOST2

Before you can start the Oracle HTTP Server instances, you must start the Node Manager on WEBHOST1 and WEBHOST2:

1. Log into WEBHOST1 and navigate to the following directory:

   ```
   cd $WEB_DOMAIN_HOME/nodemanager
   ```

2. Start the Node Manager as shown in the following sections by using `nohup` and `nodemanager.out` as an example output file:

   ```
   nohup $WEB_DOMAIN_HOME/bin/startNodeManager.sh > $WEB_DOMAIN_HOME/nodemanager/
   nodemanager.out 2>&1 &
   ```

3. Log in to WEBHOST2 and perform steps 1 and 2.

See Advanced Node Manager Configuration in *Administering Node Manager for Oracle WebLogic Server*.

Starting the Oracle HTTP Server Instances

To start the Oracle HTTP Server instances:

1. Navigate to the following directory on WEBHOST1:

   ```
   cd $WEB_DOMAIN_HOME/bin
   ```

   For more information about the location of the WEB_DOMAIN_HOME directory, see File System and Directory Variables Used in This Guide.

2. Enter the following command:

   ```
   ./startComponent.sh ohs1
   ```

   > **Note**
   >
   > Every time you start the Oracle HTTP server, you will be asked for the Node Manager password. If you do not wish this behaviour, then use the following command the first time you start the Oracle HTTP server:
   >
   > ```
   > ./startComponent.sh ohs1 storeUserConfig
   > ```
   >
   > This time when you enter the Node Manager password, it will be encrypted and stored. Future start and stop of the Oracle HTTP server will not require you to enter the Node Manager password.

   > **Note**
   >
   > For more information, see Storing Your Node Manager Password.

3. When prompted, enter the Node Manager password.
4. Repeat steps 1 through 3 to start the `ohs2` instance on WEBHOST2. See Starting Oracle HTTP Server Instances in *Administering Oracle HTTP Server*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Backing Up the Configuration

It is an Oracle best practices recommendation to create a backup after you successfully extended a domain or at another logical point. Create a backup after you verify that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

The backup destination is the local disk. You can discard this backup when the enterprise deployment setup is complete. After the enterprise deployment setup is complete, you can initiate the regular deployment-specific Backup and Recovery process.

For information about backing up your configuration, see Performing Backups and Recoveries for an Enterprise Deployment.

Generate Required Certificates for OHS SSL Listeners

In an End to End SSL deployment the OHS listeners use SSL and you must create appropriate certificates for them.

The OHS certificates must be SAN certificates and include the virtual server name and the server names of the server they are running on. It is required to have certificates for each WEBHOST address, adding as SAN the different ServerNames that are used in them.

This enterprise deployment uses igdinternal.example.com, igdadmin.example.com, oig.example.com, login.example.com and iadadmin.example.com as frontend addresses. These addresses are used in the WLS domain configuration as frontend addresses for different clusters and servers.

Oracle recommends using the same Identity and Trust store files for all the Certificate Authorities (CAs) and certificates used in the application tier. The OHS nodes, do not use shared storage so the stores need to be copied to their private folders. Certificates in a production system should come from formal CA's.

For more information about certificates, see Obtaining SSL Certificates.

Perform the following steps to create OHS wallets from existing certificates:

1. Create a Wallet Trust Store and add in the certificate CA.

```
$WEB_ORACLE_HOME/bin/orapki wallet create -wallet /u02/oracle/config/
keystores/orapki -auto_login_only


$WEB_ORACLE_HOME/bin/orapki wallet add -wallet /u02/oracle/config/
keystores/orapki -auto_login_only -trusted_cert -cert /u02/oracle/config/
keystores/idmCA.crt
```

2. Create a Wallet for each virtual host.

```
export vHost=login.example.com
export wHost=webhost1.example.com
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
export ksHome=/u02/oracle/config/keystores
export kspwd=password


$WEB_ORACLE_HOME/bin/orapki wallet create -wallet $ksHome/orapki/
wallet_$vHost -auto_login_only


$WEB_ORACLE_HOME/bin/orapki wallet import_pkcs12 -wallet $ksHome/orapki/
wallet_$vHost -auto_login_only -pkcs12file $ksHome/$vHost.p12 -
pkcs12pwd $kspwd
```

3. Repeat all steps for each virtual host.
4. Repeat all steps for the WEBHOST2.

Configuring Oracle HTTP Server to Route Requests to the Application Tier

Update the Oracle HTTP Server configuration files so that the web server instances route requests to the servers in the domain.

- About the Oracle HTTP Server Configuration for an Enterprise Deployment
- Modifying the httpd.conf File to Include Virtual Host Configuration Files
- Modifying the httpd.conf File to Set Server Runtime Parameters
- Creating the Virtual Host Configuration Files
- Routing Requests to WebLogic Servers
- Copying Configuration Files to WEBHOST2
- Restarting the OHS Instances on WEBHOST1 and WEBHOST2
- Validating Access Through the Load Balancer

About the Oracle HTTP Server Configuration for an Enterprise Deployment

The following topics provide overview information about the changes that are required to the Oracle HTTP Server configuration files in an enterprise deployment.

- Purpose of the Oracle HTTP Server Virtual Hosts
- About the WebLogicCluster Parameter of the <VirtualHost> Directive
- Recommended Structure of the Oracle HTTP Server Configuration Files

Purpose of the Oracle HTTP Server Virtual Hosts

The reference topologies in this guide require that you define a set of virtual servers on the hardware load balancer. You can then configure Oracle HTTP Server to recognize requests to specific virtual hosts (that map to the load balancer virtual servers) by adding `<VirtualHost>` directives to the Oracle HTTP Server instance configuration files.

For each Oracle HTTP Server virtual host, you define a set of specific URLs (or context strings) that route requests from the load balancer through the Oracle HTTP Server instances

Copyright © 2018, 2025, Oracle and/or its affiliates.

to the appropriate Administration Server or Managed Server in the Oracle WebLogic Server domain.

About the WebLogicCluster Parameter of the <VirtualHost> Directive

A key parameter of the Oracle HTTP Server `<VirtualHost>` directive is the `WebLogicCluster` parameter, which is part of the WebLogic Proxy Plug-In for Oracle HTTP Server. When you configure Oracle HTTP Server for an enterprise deployment, consider the following information when you add this parameter to the Oracle HTTP Server configuration files.

The servers specified in the `WebLogicCluster` parameter are important only at startup time for the plug-in. The list needs to provide at least one running cluster member for the plug-in to discover other members of the cluster. When you start the Oracke HTTP server, the listed cluster member must be running. Oracle WebLogic Server and the plug-in work together to update the server list automatically with new, failed, and recovered cluster members.

Some example scenarios:

- Example 1: If you have a two-node cluster and then add a third member, you do not need to update the configuration to add the third member. The third member is discovered on the fly at runtime.
- Example 2: You have a three-node cluster but only two nodes are listed in the configuration. However, if both listed nodes are down when you start Oracle HTTP Server, then the plug-in would fail to route to the cluster. You must ensure that at least one of the listed nodes is running when you start Oracle HTTP Server.

  If you list all members of the cluster, then you guarantee you can route to the cluster, assuming at least one member is running when Oracle HTTP Server is started.

Recommended Structure of the Oracle HTTP Server Configuration Files

Rather than adding multiple virtual host definitions to the `httpd.conf` file, Oracle recommends that you create separate, smaller, and more specific configuration files for each of the virtual servers required for the products that you are deploying. This avoids populating an already large `httpd.conf` file with additional content, and it can make troubleshooting configuration problems easier.

For example, in a typical Oracle Fusion Middleware Infrastructure domain, you can add a specific configuration file called `admin_vh.conf` that contains the virtual host definition for the Administration Server virtual host (ADMINVHN). If you are using an End to End SSL deployment all virtual hosts in this Enterprise Deployment Guide will use SSL. This Enterprise Deployment Guide segregates the listeners and certificates that are used by the different endpoints exposed through OHS. It uses different certificates and listeners for the external, internal and administration virtual hosts. This permits segregating the traffic and the encryption quality for each type of access and provides a well-structured mapping of front ends, Virtual Hosts and listeners.

Modifying the httpd.conf File to Include Virtual Host Configuration Files

Perform the following tasks to prepare the `httpd.conf` file for the additional virtual hosts required for an enterprise topology:

1. Log in to WEBHOST1.
2. Locate the `httpd.conf` file for the first Oracle HTTP Server instance (`ohs1`) in the domain directory:

   ```
   cd $WEB_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

3. Verify if the `httpd.conf` file has the appropriate configuration as follows:

   a. Run the following command to verify the `ServerName` parameter, be sure that it is set correctly, substituting the correct value for the current WEBHOST*n*:

   ```
   grep "ServerName http" httpd.conf
   ServerName http://webhost1.example.com:7777
   ```

   b. Run the following command to verify there is an include statement that includes all `*.conf` files from the moduleconf subdirectory:

   ```
   grep moduleconf httpd.conf
   IncludeOptional "moduleconf/*.conf"
   ```

   c. If either validation fails to return results, or returns results that are commented out, open the `httpd.conf` file in a text editor and make the required changes in the appropriate locations.

   ```
   #
   # ServerName gives the name and port that the server uses to identify
   itself.
   # This can often be determined automatically, but we recommend you
   specify
   # it explicitly to prevent problems during startup.
   #
   # If your host doesn't have a registered DNS name, enter its IP address
   here.
   #
   ServerName http://webhost1.example.com:7777
   #  and at the end of the file:
   # Include the admin virtual host (Proxy Virtual Host) related
   configuration
   include "admin.conf"
   IncludeOptional "moduleconf/*.conf"
   ```

   d. Save the `httpd.conf` file.

4. Ensure `ssl.conf` is included in the httpd configuration.

   ```
   grep ssl.conf httpd.conf
   ```

   Include `ssl.conf` and copy the `ssl.conf` file to a different file name.

   > **Note**
   >
   > This is used as a template for other module conf files.

   ```
   cp $WEB_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/
   ssl.conf $WEB_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/moduleconf/
   ssl.template
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

5. Log in to WEBHOST2 and perform steps 2 and 3 for the httpd.conf file, replacing any occurrences of webhost1 or ohs1 with webhost2 or ohs2 in the instructions as necessary.
6. Edit the ssl.conf file to include only the following lines (remove other content from the file):

```
<IfModule ossl_module>
#
# Some MIME-types for downloading Certificates and CRLs
    AddType application/x-x509-ca-cert .crt
    AddType application/x-pkcs7-crl .crl

# Inter-Process Session Cache:
# Configure the SSL Session Cache: First the mechanism
# to use, second the expiring timeout (in seconds) and third
# the mutex to be used.
    SSLSessionCache "shmcb:${ORACLE_INSTANCE}/servers/${COMPONENT_NAME}/
logs/ssl_scache(512000)"
    SSLSessionCacheTimeout 300

</IfModule>

   # Client Authentication (Type):
   # Client certificate verification type and depth. Types are
   # none, optional and require.
   SSLVerifyClient None

   # SSL Protocol Support:
   # Configure usable SSL/TLS protocol versions.
   SSLProtocol TLSv1.2 TLSv1.3

   # Option to prefer the server's cipher preference order
   SSLHonorCipherOrder on

   # SSL Cipher Suite:
   # List the ciphers that the client is permitted to negotiate.
   SSLCipherSuite
TLS_AES_128_GCM_SHA256,TLS_AES_256_GCM_SHA384,TLS_CHACHA20_POLY1305_SHA256,
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SH
A384,TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_RSA_WITH_AES_256_GCM_S
HA384

   <FilesMatch "\.(cgi|shtml|phtml|php)$">
      SSLOptions +StdEnvVars
   </FilesMatch>

   <Directory "${ORACLE_INSTANCE}/config/fmwconfig/components/$
{COMPONENT_TYPE}/instances/${COMPONENT_NAME}/cgi-bin">
      SSLOptions +StdEnvVars
   </Directory>

   BrowserMatch "MSIE [2-5]" \
         nokeepalive ssl-unclean-shutdown \
         downgrade-1.0 force-response-1.0
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
   # Add the following directive to add HSTS
   <IfModule mod_headers.c>
   Header always set Strict-Transport-Security "max-age=63072000; preload;
includeSubDomains"
```

7. Modify the `$WEB_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/mod_wl_ohs.conf` to include the appropriate WLSSWallet file (required to route on SSL to the WLS backends) as follows:

> **Note**
>
> This is a template to configure `mod_weblogic`.

```
LoadModule weblogic_module "${PRODUCT_HOME}/modules/mod_wl_ohs.so"
```

This empty block is needed to save `mod_wl` related configuration from EM to this file when changes are made at the Base Virtual Host Level.

```
<IfModule weblogic_module>

    WLIOTimeoutSecs 900
    KeepAliveSecs 290
    FileCaching OFF
    WLSocketTimeoutSecs 15
    ErrorPage http://www.oracle.com/splash/cloud/index.html
    WLRetryOnTimeout NONE
    WLForwardUriUnparsed On
    SecureProxy On
    WLSSLWallet "/u02/oracle/config/keystores/orapki/"
</IfModule>
```

Modifying the httpd.conf File to Set Server Runtime Parameters

Out of the box, the Oracle HTTP Server comes configured with a number of values which effect how the server behaves when it is running. For most of the deployments, these values are sufficient. However, in an Oracle Identity and Access Management deployment, it is recommended that you update these values by doing the following:

1. Log in to WEBHOST1.
2. Locate the `httpd.conf` file for the first Oracle HTTP Server instance (`ohs1`) in the domain directory:

   ```
   cd $WEB_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/
   ```

3. Locate the section of the file with the following line:

   ```
   <IfModule mpm_worker_module>
   ```

4. Update the entries in this section to reflect the following:

   ```
   <IfModule mpm_worker_module>
     ServerLimit             20
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
  StartServers            10
  MaxClients              1500
  MinSpareThreads         200
  MaxSpareThreads         800
  ThreadsPerChild         250
  ThreadLimit             250
  MaxRequestsPerChild     1000
  MaxRequestWorkers       400
  MaxConnectionsPerChild  0
 </IfModule>
```

5. Update the following values:
   - `MaxKeepAliveRequests 0`
   - `Timeout 300`
   - `KeepAliveTimeout 10`
6. Save the `httpd.conf` file.
7. Log in to `WEBHOST2` and perform steps 2 and 3 for the `httpd.conf` file, replacing any occurrences of `WEBHOST1` or `ohs1` with `WEBHOST2` or `ohs2` in the instructions as necessary.

Creating the Virtual Host Configuration Files

- SSL Terminated Deployments
- End to End SSL Deployments

SSL Terminated Deployments

To create the virtual host configuration files:

> **Note**
>
> Before you create the virtual host configuration files, be sure that you have configured the virtual servers on the load balancer, as described in Purpose of the Oracle HTTP Server Virtual Hosts.

1. Log in to WEBHOST1 and change directory to the configuration directory for the first Oracle HTTP Server instance (`ohs1`):

   ```
   cd $WEB_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/moduleconf
   ```

2. If you are configuring Oracle Access Management, create the `iadadmin_vh.conf` file and add the following directive:

   ```
   <VirtualHost webhost1.example.com:7777>
       ServerName http://iadadmin.example.com:80
       ServerAdmin you@your.address
       RewriteEngine On
       RewriteOptions inherit
       UseCanonicalName On
   </VirtualHost>
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

3. If you are configuring Oracle Access Management, create the `login_vh.conf` file and add the following directive:

```
<VirtualHost webhost1.example.com:7777>
    ServerName https://login.example.com:443
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
    UseCanonicalName On
</VirtualHost>
```

4. If you are configuring Oracle Identity Governance, create the `igdadmin_vh.conf` file, and add the following directive:

```
<VirtualHost webhost1.example.com:7777>
    ServerName http://igdadmin.example.com:80
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
    UseCanonicalName On
</VirtualHost>
```

5. If you are configuring Oracle Identity Governance, create the `prov_vh.conf` file, and add the following directive:

```
<VirtualHost webhost1.example.com:7777>
    ServerName https://oig.example.com:443
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
    UseCanonicalName On
</VirtualHost>
```

6. If you are configuring Oracle Identity Governance, create the `igdinternal_vh.conf` file, and add the following directive:

```
<VirtualHost webhost1.example.com:7777>
    ServerName http://igdinternal.example.com:7777
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
</VirtualHost>
```

End to End SSL Deployments

To create the virtual host files for end to end ssl deployments:

In an end to end SSL deployment, the LBR communicates with OHS over SSL protocol for a more secure configuration. The OHS instances also communicate over SSL protocol with the specific Managed Servers in the application tier. SSL is configured all the way from the LBR to the backend WLS servers.

Each virtual host listens on a different port to allow different certificates to be used for each. This simplifies load balancing and allows different quality of certificates to be used for each server if required.

Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> Before you create the virtual host configuration files, ensure that you have configured the virtual servers on the load balancer as described in Purpose of the Oracle HTTP Server Virtual Hosts.

1. Log into WEBHOST1 and change directory to the configuration directory for the first Oracle HTTP Server instance (`ohs1`):

   ```
   cd $WEB_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/moduleconf
   ```

2. If you are configuring Oracle Access Management, create the `iadadmin_vh.conf` file and add the following directive:

   ```
   Listen 4445
   <VirtualHost webhost1.example.com:4445>

       ServerName http://iadadmin.edg.com:443
       ServerAdmin you@your.address
       RewriteEngine On
       RewriteOptions inherit
       UseCanonicalName On
       AllowEncodedSlashes On

       SSLEngine on
       SSLWallet "/u02/oracle/config/keystores/orapki/wallet_iadadmin.edg.com"

   </VirtualHost>
   ```

3. If you are configuring Oracle Access Management, create the `login_vh.conf` file and add the following directive:

   ```
   Listen 4447
   <VirtualHost webhost1.example.com:4447>

       ServerName http://login.edg.com:443
       ServerAdmin you@your.address
       RewriteEngine On
       RewriteOptions inherit
       UseCanonicalName On
       AllowEncodedSlashes On

       SSLEngine on
       SSLWallet "/u02/oracle/config/keystores/orapki/wallet_login.edg.com"
   </VirtualHost>
   ```

4. If you are configuring Oracle Identity Governance, create the `igdadmin_vh.conf` file, and add the following directive:

   ```
   Listen 4446
   <VirtualHost webhost1.example.com:4446>

       ServerName https://igdadmin.edg.com:443
       ServerAdmin you@your.address
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
    RewriteEngine On
    RewriteOptions inherit
    UseCanonicalName On
    AllowEncodedSlashes On

    SSLEngine on
    SSLWallet "/u02/oracle/config/keystores/orapki/wallet_igdadmin.edg.com"

</VirtualHost>
```

5. If you are configuring Oracle Identity Governance, create the `oig_vh.conf` file, and add the following directive:

```
Listen 4448
<VirtualHost webhost1.example.com:4448>

    ServerName https://oig.example.com:443
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
    UseCanonicalName On
    AllowEncodedSlashes On

    SSLEngine on
    SSLWallet "/u02/oracle/config/keystores/orapki/wallet_oig.example.com"

</VirtualHost>
```

6. If you are configuring Oracle Identity Governance, create the `igdinternal_vh.conf` file, and add the following directive:

```
Listen 4449
<VirtualHost webhost1.example.com:4449>

    ServerName http://igdinternal.edg.com:443
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
    UseCanonicalName On
    AllowEncodedSlashes On

    SSLEngine on
    SSLWallet "/u02/oracle/config/keystores/orapki/
wallet_igdinternal.edg.com"

</VirtualHost>
```

Routing Requests to WebLogic Servers

You need to configure each virtual host to redirect requests to the back end WebLogic servers.

Copyright © 2018, 2025, Oracle and/or its affiliates.

For SSL Terminated environments all `Location` directives must contain `WLSRequest ON`, `WLProxySSL ON` and `WLProxySSLPassThrough ON`, for example:

```
<Location /console>
    WLSRequest ON
    WLProxySSL ON
    WLProxySSLPassThrough ON
    WebLogicHost iadadminvhn.example.com
    WeblogicPort 7001
</Location>

<Location /oam>
    WLSRequest ON
    WLProxySSL ON
    WLProxySSLPassThrough ON
    WebLogicCluster oamhost1.example.com:14100,oamhost2.example.com:14100
</Location>
```

For End to End SSL environments all `Location` directives must contain `WLSRequest ON`, for example:

```
<Location /console
    WLSRequest ON
    WebLogicHost iadadminvhn.example.com
    WeblogicPort 9002
</Location>

<Location /oam>
    WLSRequest ON
    WebLogicCluster oamhost1.example.com:14101,oamhost2.example.com:14101
</Location>
```

The following tables lists the configuration files the you must edit on WEBHOST1 in `$WEB_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/moduleconf/`. Add the Location directives as above with the corresponding values from the table.

The following sample files are available to reference:

- SSL Terminated Deployments
- End to End SSL Deployments

Table 16-4 Directives Required for iadadmin_vh.conf

Location	Type (Host/Cluster)	Back End(s)	SSL Terminated Port	End to End SSL Port
`/console`	`WebLogicHost`	`iadadminvhn.example.com`	`7001`	`9002`
`/em`	`WebLogicHost`	`iadadminvhn.example.com`	`7001`	`9002`
`/management`	`WebLogicHost`	`iadadminvhn.example.com`	`7001`	`9002`

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 16-4 (Cont.) Directives Required for iadadmin_vh.conf

Location	Type (Host/Cluster)	Back End(s)	SSL Terminated Port	End to End SSL Port
/oamconsole	WebLogicHost	iadadminvhn.example.com	7001	9002
/access	WebLogicCluster	oamhost1.example.com,oamhost2.example.com	14100	14101
/oam/services	WebLogicHost	iadadminvhn.example.com	7001	7002
/oam/services/rest	WebLogicHost	iadadminvhn.example.com	7001	7002
/iam/admin	WebLogicHost	iadadminvhn.example.com	7001	7002
/oam/services/rest/11.1.2.0.0	WebLogicHost	iadadminvhn.example.com	7001	7002
/oam/services/rest/ssa	WebLogicHost	iadadminvhn.example.com	7001	7002
/oam/services/rest/ssa	WebLogicHost	iadadminvhn.example.com	7001	7002
/dms	WebLogicHost	iadadminvhn.example.com	7001	7002
/oam	WebLogicCluster	oamhost1.example.com,oamhost2.example.com	14100	14101

Table 16-5 Directives Required for igdadmin_vh.conf

Location	Cookie Name	Type (Host/ Cluster)	Back Ends	SSL Terminated Port	End to End SSL Port
/console		WebLogicHost	igdadminvhn.example.com	7101	9201
/em		WebLogicHost	igdadminvhn.example.com	7101	9201
/management		WebLogicHost	igdadminvhn.example.com	7101	9201
/oim	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001
/iam	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001
/sysadmin	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001
/admin	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 16-5 (Cont.) Directives Required for igdadmin_vh.conf

Location	Cookie Name	Type (Host/ Cluster)	Back Ends	SSL Terminated Port	End to End SSL Port
/identity	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001
/OIGUI	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001
/FacadeWebApp	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001
/SchedulerService-web	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001
/dms		WebLogicHost	igdadminvhn.example.com	7101	9201

Table 16-6 Directives Required for login_vh.conf

Location	Cookie Name	Type (Host/ Cluster)	Back Ends	SSL Terminated Port	End to End SSL Port	Comment
/oam	OAMJSESSIONID	WebLogicCluster	oamhost1.example.com,oamhost2.example.com	14100	14101	
/oamfed	OAMJSESSIONID	WebLogicCluster	oamhost1.example.com,oamhost2.example.com	14100	14101	
/otpfp/	OAMJSESSIONID	WebLogicCluster	oamhost1.example.com,oamhost2.example.com	14100	14101	
/ms_oauth	OAMJSESSIONID	WebLogicCluster	oamhost1.example.com,oamhost2.example.com	14100	14101	
/oamservices/rest/auth	OAMJSESSIONID	WebLogicCluster	oamhost1.example.com,oamhost2.example.com	14100	14101	
/oamservices/rest/access	OAMJSESSIONID	WebLogicCluster	oamhost1.example.com,oamhost2.example.com	14100	14101	

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 16-6 (Cont.) Directives Required for login_vh.conf

Location	Cookie Name	Type (Host/ Cluster)	Back Ends	SSL Terminated Port	End to End SSL Port	Comment
/iam/access	OAMJSESSIONID	WebLogicCluster	oamhost1.example.com,oamhost2.example.com	14100	14101	
/oauth2	OAMJSESSIONID	WebLogicCluster	oamhost1.example.com,oamhost2.example.com	14100	14101	
/.well-known/openid-configuration	OAMJSESSIONID	WebLogicCluster	oamhost1.example.com,oamhost2.example.com	14100	14101	Also add: • PathTrim /.well-known • PathPrepend /oauth2/rest :
/.well-known/oidc-configuration	OAMJSESSIONID	WebLogicCluster	oamhost1.example.com,oamhost2.example.com	14100	14101	Also add: • PathTrim /.well-known • PathPrepend /oauth2/rest :
/CustomConsent	OAMJSESSIONID	WebLogicCluster	oamhost1.example.com,oamhost2.example.com	14100	14101	

Table 16-7 Directives Required for oig_vh.conf

Location	Cookie Name	Type (Host/ Cluster)	Back Ends	SSL Terminated Port	End to End SSL Port
/identity	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001
/HTTPClnt	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001
/reqsvc	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 16-7 (Cont.) Directives Required for oig_vh.conf

Location	Cookie Name	Type (Host/ Cluster)	Back Ends	SSL Terminated Port	End to End SSL Port
/FacadeWebApp	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001
/iam	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001
/OIGUI	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001

Table 16-8 Directives Required for igdinternal_vh.conf

Location	Cookie Name	Type (Host/ Cluster)	Back Ends	SSL Terminated Port	End to End SSL Port	Comments
/sodcheck	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	7003	7004	
/role-sod	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001	
/workflowservice	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001	
/callbackResponseService	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001	
/spml-xsd	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001	
/spmlws	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001	Also add: • PathTrim /weblogic :
/reqsvc	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001	
/soa-infra	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	7003	7004	

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 16-8 (Cont.) Directives Required for igdinternal_vh.conf

Location	Cookie Name	Type (Host/ Cluster)	Back Ends	SSL Terminated Port	End to End SSL Port	Comments
/ucs	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	7003	14001	
/provisioning-callback	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001	
/CertificationCallbackService	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001	
/IdentityAuditCallbackService	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001	
/soa/composer	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	7003	7004	
/integration	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	7003	7004	
/sdpmessaging/userprefs-ui	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	7003	7004	
/iam	oimjsessionid	WebLogicCluster	oighost1.example.com,oighost2.example.com	14000	14001	

Copying Configuration Files to WEBHOST2

Once the Location directives are added to the OHS configuration files on WEBHOST1, you must copy the files to WEBHOST2 and edit appropriately.

1. Copy the `iadadmin_vh.conf`, `igdadmin_vh.conf`, `login_vh.conf`, `oig_vh.conf` and`igdinternal_vh.conf` to the configuration directory for the second Oracle HTTP Server instance (ohs2) on WEBHOST2:

   ```
   scp $WEB_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/moduleconf/*
   oracle@webhost2:$WEB_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs2/
   moduleconf
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

2. Edit each of the files copied in the previous step on WEBHOST2 and change any references of `webhost1.example.com` to `webhost2.example.com`.

Restarting the OHS Instances on WEBHOST1 and WEBHOST2

Restart the Oracle HTTP Server (OHS) instances on both WEBHOST1 and WEBHOST2.

1. Restart the ohs1 instance on WEBHOST1 by performing the following steps:
   a. Change directory to the following location:

      ```
      cd $WEB_DOMAIN_HOME/bin
      ```

   b. Enter the following commands to stop and start the instance:

      ```
      ./stopComponent.sh ohs1


      ./startComponent.sh ohs1
      ```

2. Restart the ohs2 instance on WEBHOST2 by performing the following steps:
   a. Change directory to the following location:

      ```
      cd $WEB_DOMAIN_HOME/bin
      ```

   b. Enter the following commands to stop and start the instance:

      ```
      ./stopComponent.sh ohs2


      ./startComponent.sh ohs2
      ```

Validating Access Through the Load Balancer

You should verify URLs to ensure that appropriate routing and failover is working from Oracle HTTP Server to OAM_Cluster.

- Verifying the URLs

Verifying the URLs

To verify the URLs:

1. While oam_server2 is running, stop oam_server1 using the Oracle Fusion Middleware Control.
2. Access `https://login.example.com/oam/server/logout`.
3. Start oam_server1 from the Oracle Fusion Middleware Control.
4. Stop oam_server2 from the Oracle Fusion Middleware Control.
5. Access `https://login.example.com/oam/server/logout`.

You can verify the cluster node to which you were directed after the traffic balancing provided through your load balancer and then again through the web tier.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Validating the Virtual Server Configuration and Access to the Consoles

Validate the virtual server configuration on the load balancer, and the access to the management console and the Administration Server.

From the load balancer, access the following URLs to ensure that your load balancer and Oracle HTTP Server are configured properly. These URLs should show the initial Oracle HTTP Server 14*c* web page.

- `https://login.example.com/index.html`
- `https://oig.example.com/index.html`
- `http://iadadmin.example.com/index.html`
- `http://igdadmin.example.com/index.html`

Use the following URLs to the hardware load balancer to display the Oracle Fusion Middleware Control, and log in using the Oracle WebLogic Server administrator credentials:

- SSL Terminated: `http://iadadmin.example.com/em`
- End to End SSL: `https://iadadmin.example.com/em`

This validates that the `iadadmin.example.com` virtual host on the load balancer is able to route requests to the Oracle HTTP Server instances on the web tier, which in turn can route requests for the WebLogic Remote Console to the Administration Server in the application tier.

Similarly, you should be able to access the Fusion Middleware Control for the `igdadmin` virtual host using the following URLs:

- SSL Terminated: `http://igdadmin.example.com/em`
- End to End SSL: `https://igdadmin.example.com/em`

Sample Virtual Host Files

The sample list includes the complete examples of all the virtual host files used in an Oracle Identity and Access Management deployment.

- SSL Terminated Deployments
- End to End SSL Deployments

SSL Terminated Deployments

- `login_vh.conf`
- `oig_vh.conf`
- `iadadmin_vh.conf`
- `igdadmin_vh.conf`
- `igdinternal_vh.conf`

Copyright © 2018, 2025, Oracle and/or its affiliates.

login_vh.conf

```
Listen 7777

<VirtualHost webhost1.example.com:7777>
    ServerName https://login.example.com:443
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit

#OAM Entries
    <Location /oam>
        WLSRequest ON
        WLProxySSL ON
        WLProxySSLPassThrough ON
        WLCookieName OAMJSESSIONID
        WebLogicCluster oamhost1.example.com:14100,oamhost2.example.com:14100
    </Location>

    <Location /oamfed>
        WLSRequest ON
        WebLogicCluster oamhost1.example.com:14100,oamhost2.example.com:14100
        WLCookieName OAMJSESSIONID
        WLProxySSL ON
        WLProxySSLPassThrough ON
    </Location>

# OAM Forgotten Password Page
    <Location /otpfp/>
        WLSRequest ON
        WebLogicCluster
oamhost1.example.com:14100,oamhost2.example.com:14100
        WLCookieName OAMJSESSIONID
        WLProxySSL ON
        WLProxySSLPassThrough ON
     </Location>

     <Location /ms_oauth>
         WLSRequest ON
         WebLogicCluster oamhost1.example.com:14100,oamhost2.example.com:14100
         WLCookieName OAMJSESSIONID
         WLProxySSL ON
         WLProxySSLPassThrough ON
     </Location>

     <Location /oamservices/rest/auth>
         WLSRequest ON
         WLCookieName OAMJSESSIONID
         WebLogicCluster oamhost1.example.com:14100,oamhost2.example.com:14100
     </Location>

     <Location /oamservices/rest/access>
         WLSRequest ON
         WLCookieName OAMJSESSIONID
         WebLogicCluster oamhost1.example.com:14100,oamhost2.example.com:14100
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
    </Location>

    <Location /iam/access>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        WebLogicCluster oamhost1.example.com:14100,oamhost2.example.com:14100
    </Location>

    <Location /oauth2>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        WebLogicCluster oamhost1.example.com:14100,oamhost2.example.com:14100
    </Location>

    <Location /.well-known/openid-configuration>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        PathTrim /.well-known
        PathPrepend /oauth2/rest
        WebLogicCluster oamhost1.example.com:14100,oamhost2.example.com:14100
    </Location>

    <Location /.well-known/oidc-configuration>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        PathTrim /.well-known
        PathPrepend /oauth2/rest
        WebLogicCluster oamhost1.example.com:14100,oamhost2.example.com:14100
    </Location>

    <Location /CustomConsent>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        WebLogicCluster oamhost1.example.com:14100,oamhost2.example.com:14100
    </Location>
</VirtualHost>
```

oig_vh.conf

```
Listen 7777

<VirtualHost webhost1.example.com:7777>
    ServerName https://oig.example.com:443
    RewriteEngine On
    RewriteOptions inherit

    <Location /identity>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    WLProxySSL ON
    WLProxySSLPassThrough ON
    </Location>

    <Location /HTTPClnt>
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    WLProxySSL ON
    WLProxySSLPassThrough ON
    </Location>

# Requests webservice URL
    <Location /reqsvc>
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    WLProxySSL ON
    WLProxySSLPassThrough ON
    </Location>

    <Location /FacadeWebApp>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    WLProxySSL ON
    WLProxySSLPassThrough ON
    </Location>

    <Location /iam>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    WLProxySSL ON
    WLProxySSLPassThrough ON
    </Location>

    <Location /OIGUI>
    SetHandler weblogic-handler
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    WLProxySSL ON
    WLProxySSLPassThrough ON
    </Location>
</VirtualHost>
```

iadadmin_vh.conf

```
Listen 7777

<VirtualHost webhost1.example.com:7777>
    ServerName iadadmin.example.com:80
    RewriteEngine On
    RewriteOptions inherit
    UseCanonicalName On

# Admin Server and EM
    <Location /console>
    WLSRequest ON
    WebLogicHost iadadminvhn.example.com
    WeblogicPort 7001
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
    </Location>

    <Location /management>
    WLSRequest ON
    WebLogicHost iadadminvhn.example.com
    WeblogicPort 7001
    </Location>

    <Location /em>
    WLSRequest ON
    WebLogicHost iadadminvhn.example.com
    WeblogicPort 7001
    </Location>

    <Location /oamconsole>
    WLSRequest ON
    WebLogicHost iadadminvhn.example.com
    WeblogicPort 7001
    </Location>

    <Location /access>
    WLSRequest ON
    WebLogicCluster oamhost1.example.com:14150,oamhost2.example.com:14150
    WLCookieName OAMJSESSIONID
    </Location>

# Required for Multi-Datacenter
    <Location /oam/services>
    WLSRequest ON
    WebLogicHost iadadminvhn.example.com
    WeblogicPort 7001
    </Location>

    <Location /oam/admin/api>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 7001
    </Location>

    <Location /oam/services/rest>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 7001
    </Location>

    <Location /iam/admin>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 7001
    </Location>

    <Location /oam/services/rest/11.1.2.0.0>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 7001
    </Location>
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
    <Location /oam/services/rest/ssa>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 7001
    </Location>

    <Location /dms>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 7001
    </Location>

    <Location /oam>
        WLSRequest ON
        WebLogicCluster oamhost1.example.com:14100,oamhost2.example.com:14100
    </Location>

</VirtualHost>
```

igdadmin_vh.conf

```
Listen 7777

<VirtualHost webhost1.example.com:7777>
    ServerName igdadmin.example.com:80
    RewriteEngine On
    RewriteOptions inherit
    UseCanonicalName On

# Admin Server and EM
    <Location /console>
    WLSRequest ON
    WebLogicHost igdadminvhn.example.com
    WeblogicPort 7101
    </Location>

    <Location /management>
    WLSRequest ON
    WebLogicHost igdadminvhn.example.com
    WeblogicPort 7101
    </Location>

    <Location /em>
    WLSRequest ON
    WebLogicHost igdadminvhn.example.com
    WeblogicPort 7101
    </Location>

    <Location /oim>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    </Location>
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
    <Location /iam>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    </Location>

    <Location /sysadmin>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    </Location>

    <Location /admin>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    </Location>

# OIM self service console
    <Location /identity>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    </Location>

    <Location /OIGUI>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    </Location>

    <Location /FacadeWebApp>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    </Location>

# Scheduler webservice URL
    <Location /SchedulerService-web>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    </Location>

    <Location /dms>
      WLSRequest ON
      WebLogicHost igdadminvhn.example.com
      WeblogicPort 7101
    </Location>
</VirtualHost>
```

igdinternal_vh.conf

```
Listen 7777
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
<VirtualHost webhost1.example.com:7777>
    ServerName igdinternal.example.com:7777
    RewriteEngine On
    RewriteOptions inherit

    <Location /sodcheck>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:7003,oighost2.example.com:7003
    </Location>

# OIM, role-sod profile
    <Location /role-sod>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    </Location>

# Callback webservice for SOA. SOA calls this when a request is approved/
rejected
# Provide the SOA Managed Server Port
    <Location /workflowservice>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    </Location>

# used for FA Callback service.
    <Location /callbackResponseService>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000

    </Location>

# spml xsd profile
    <Location /spml-xsd>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    </Location>

# OIM, spml dsml profile
    <Location /spmlws>
    WLSRequest ON
    PathTrim /weblogic
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    </Location>

    <Location /reqsvc>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
    </Location>
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
# SOA Infra
    <Location /soa-infra>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:7003,oighost2.example.com:7003
    </Location>

# UMS Email Support
    <Location /ucs>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:7003,oighost2.example.com:7003
    </Location>

    <Location /provisioning-callback>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000

    </Location>

   <Location /CertificationCallbackService>
   WLSRequest ON
   WLCookieName oimjsessionid
   WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000

   </Location>

   <Location /IdentityAuditCallbackService>
   WLSRequest ON
   WLCookieName oimjsessionid
   WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000

   </Location>

# SOA Callback webservice for SOD - Provide the SOA Managed Server Ports
    <Location /soa/composer>
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:7003,oighost2.example.com:7003
    </Location>

    <Location /integration>
    WebLogicCluster oighost1.example.com:7003,oighost2.example.com:7003
    WLCookieName oimjsessionid
    </Location>

    <Location /sdpmessaging/userprefs-ui>
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:7003,oighost2.example.com:7003

    </Location>

    <Location /iam>
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14000,oighost2.example.com:14000
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
    </Location>
</VirtualHost>
```

End to End SSL Deployments

- login_vh.conf
- oig_vh.conf
- iadadmin_vh.conf
- igdadmin_vh.conf
- igdinternal_vh.conf

login_vh.conf

```
Listen 4447

<VirtualHost webhost1.example.com:4447>
    ServerName https://login.example.com:443
    ServerAdmin you@your.address
    SSLWallet "/u01/oracle/config/keystores/orapki/wallet_login.example.com"
    RewriteEngine On
    RewriteOptions inherit
    UseCanonicalName On

#OAM Entries
    <Location /oam>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        WebLogicCluster oamhost1.example.com:14101,oamhost2.example.com:14101
    </Location>

   <Location /oamfed>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        WebLogicCluster oamhost1.example.com:14101,oamhost2.example.com:14101
    </Location>


# OAM Forgotten Password Page
   <Location /otpfp>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        WebLogicCluster oamhost1.example.com:14101,oamhost2.example.com:14101
    </Location>

    <Location /ms_oauth>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        WebLogicCluster oamhost1.example.com:14101,oamhost2.example.com:14101
    </Location>

    <Location /oamservices/rest/auth>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
        WebLogicCluster oamhost1.example.com:14101,oamhost2.example.com:14101
    </Location>

    <Location /oamservices/rest/access>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        WebLogicCluster oamhost1.example.com:14101,oamhost2.example.com:14101
    </Location>

    <Location /iam/access>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        WebLogicCluster oamhost1.example.com:14101,oamhost2.example.com:14101
    </Location>

    <Location /oauth2>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        WebLogicCluster oamhost1.example.com:14101,oamhost2.example.com:14101
    </Location>

    <Location /.well-known/openid-configuration>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        PathTrim /.well-known
        PathPrepend /oauth2/rest
        WebLogicCluster oamhost1.example.com:14101,oamhost2.example.com:14101
    </Location>

    <Location /.well-known/oidc-configuration>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        PathTrim /.well-known
        PathPrepend /oauth2/rest
        WebLogicCluster oamhost1.example.com:14101,oamhost2.example.com:14101
    </Location>

    <Location /CustomConsent>
        WLSRequest ON
        WLCookieName OAMJSESSIONID
        WebLogicCluster oamhost1.example.com:14101,oamhost2.example.com:14101
    </Location>

</VirtualHost>
```

oig_vh.conf

```
Listen 4448

<VirtualHost webhost1.example.com:4448>
    ServerName https://oig.example.com:443
    AllowEncodedSlashes On
    SSLEngine on
    SSLWallet "/u01/oracle/config/keystores/orapki/wallet_oig.example.com"
    RewriteEngine On
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
    RewriteOptions inherit
    UseCanonicalName On

    <Location /identity>
        WLSRequest ON
        WLCookieName oimjsessionid
        WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
    </Location>

    <Location /HTTPClnt>
        WLSRequest ON
        WLCookieName oimjsessionid
        WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
    </Location>

# Requests webservice URL
    <Location /reqsvc>
        WLSRequest ON
        WLCookieName oimjsessionid
        WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
    </Location>

    <Location /FacadeWebApp>
        WLSRequest ON
        WLCookieName oimjsessionid
        WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
    </Location>

    <Location /iam>
        WLSRequest ON
        WLCookieName oimjsessionid
        WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
    </Location>

    <Location /OIGUI>
        WLSRequest ON
        WLCookieName oimjsessionid
        WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
    </Location>

</VirtualHost>
```

iadadmin_vh.conf

```
Listen webhost1.example.com:4445

<VirtualHost webhost1.example.com:4445>
    ServerName https://iadadmin.example.com:443
    AllowEncodedSlashes On
    SSLEngine on
    SSLWallet "/u02/oracle/config/keystores/orapki/
wallet_iadadmin.example.com"
    RewriteEngine On
    RewriteOptions inherit
    UseCanonicalName On
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
    <Location /console>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 9002
    </Location>

    <Location /management>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 9002
    </Location>

    <Location /em>
        WLSRequest ON
        WebLogicHost oamhost1..example.com
        WebLogicPort 9002
    </Location>

    <Location /oamconsole>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 9002
    </Location>

    <Location /access>
        WLSRequest ON
        WebLogicCluster oamhost1.example.com:14101,oamhost2.example.com:14101
    </Location>


# Required for Multi-Datacenter
    <Location /oam/services>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WeblogicPort 7002
    </Location>

    <Location /oam/admin/api>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 7002
    </Location>

    <Location /oam/services/rest>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 7002
    </Location>

    <Location /iam/admin>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 7002
    </Location>
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
    <Location /oam/services/rest/11.1.2.0.0>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 7002
    </Location>

    <Location /oam/services/rest/ssa>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 7002
    </Location>

    <Location /dms>
        WLSRequest ON
        WebLogicHost iadadminvhn.example.com
        WebLogicPort 7002
    </Location>

    <Location /oam>
        WLSRequest ON
        WebLogicCluster oamhost1.example.com:14101,oamhost2.example.com:14101
    </Location>

</VirtualHost>
```

igdadmin_vh.conf

```
Listen webhost1.example.com:4446

<VirtualHost webhost1.example.com:4446>
    ServerName https://igdadmin.example.com:443
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
    UseCanonicalName On
    RequestHeader set "X-Forwarded-Host" "igdadmin.example.com"

    # Admin Server and EM
    <Location /console>
        WLSRequest ON
        WebLogicHost igdadminvhn.example.com
        WeblogicPort 9201
    </Location>

    <Location /management>
        WLSRequest ON
        WebLogicHost igdadminvhn.example.com
        WeblogicPort 9201
    </Location>

    <Location /em>
        WLSRequest ON
        WebLogicHost igdadminvhn.example.com
        WeblogicPort 9201
    </Location>
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
<Location /oim>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
</Location>

<Location /iam>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
</Location>

<Location /sysadmin>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
</Location>

<Location /admin>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
</Location>

# OIM self service console
<Location /identity>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
</Location>

<Location /OIGUI>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
</Location>

<Location /FacadeWebApp>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
</Location>

# Scheduler webservice URL
<Location /SchedulerService-web>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
</Location>

<Location /dms>
    WLSRequest ON
    WebLogicHost igdadminvhn.example.com
    WeblogicPort 9201
</Location>
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
</VirtualHost>
```

igdinternal_vh.conf

```
Listen 4449

<VirtualHost webhost1.example.com:4449>
    ServerName https://igdinternal.example.com:443
    ServerAdmin you@your.address
    RewriteEngine On
    RewriteOptions inherit
    UseCanonicalName On
    RequestHeader set "X-Forwarded-Host" "igdinternal.example.com"

    <Location /sodcheck>
       WLSRequest ON
       WLCookieName oimjsessionid
       WebLogicCluster oighost1.example.com:7004,oighost2.example.com:7004
    </Location>

    # OIM, role-sod profile
    <Location /role-sod>
       WLSRequest ON
       WLCookieName oimjsessionid
       WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
    </Location>

    # Callback webservice for SOA. SOA calls this when a request is approved/
rejected
    # Provide the SOA Managed Server Port
    <Location /workflowservice>
        WLSRequest ON
        WLCookieName oimjsessionid
        WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
    </Location>

    # used for FA Callback service.
    <Location /callbackResponseService>
        WLSRequest ON
        WLCookieName oimjsessionid
        WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
    </Location>

    # spml xsd profile
    <Location /spml-xsd>
        WLSRequest ON
        WLCookieName oimjsessionid
        WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
    </Location>

    # OIM, spml dsml profile
    <Location /spmlws>
        WLSRequest ON
        PathTrim /weblogic
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
</Location>

<Location /reqsvc>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
</Location>

# SOA Infra
<Location /soa-infra>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:7004,oighost2.example.com:7004
</Location>

# UMS Email Support
<Location /ucs>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:7004,oighost2.example.com:7004
</Location>

<Location /provisioning-callback>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
</Location>

<Location /CertificationCallbackService>
    WLSRequest ON
    WLCookieName oimjsessioni
    WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
</Location>

<Location /IdentityAuditCallbackService>
    WLSRequest ON
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
</Location>

# SOA Callback webservice for SOD - Provide the SOA Managed Server Ports
<Location /soa/composer>
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:7004,oighost2.example.com:7004
</Location>

<Location /integration>
    WebLogicCluster oighost1.example.com:7004,oighost2.example.com:7004
    WLCookieName oimjsessionid
</Location>

<Location /sdpmessaging/userprefs-ui>
    WLCookieName oimjsessionid
    WebLogicCluster oighost1.example.com:7004,oighost2.example.com:7004
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
    </Location>

    <Location /iam>
        WLCookieName oimjsessionid
        WebLogicCluster oighost1.example.com:14001,oighost2.example.com:14001
    </Location>

</VirtualHost>
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

Part IV

Common Configuration and Management Procedures for an Enterprise Deployment

There are certain configuration and management procedures that are recommended for a typical enterprise deployment.

The following topics contain configuration and management procedures that are required for a typical enterprise deployment.

- Common Configuration and Management Tasks for an Enterprise Deployment
  The configuration tasks include a few that are common to all enterprise deployments, such as verifying sizing information, performing backups and recoveries, and so on. Patching an enterprise deployment and cross wiring components are the other common tasks.
- Scaling Procedures for an Enterprise Deployment
  The scaling procedures for an enterprise deployment include scale out, scale in, scale up, and scale down. During a scale-out operation, you add managed servers to new nodes. You can remove these managed servers by performing a scale in operation. During a scale-up operation, you add managed servers to existing hosts. You can remove these servers by performing a scale-down operation.
- Configuring Single Sign-On for an Enterprise Deployment
  You need to configure the Oracle HTTP Server WebGate in order to enable single sign-on with Oracle Access Manager.
- Troubleshooting
  You can troubleshoot the common issues that may arise with the Identity and Access Management enterprise deployment. The solutions provided for the common problems help you resolve them quickly.

Copyright © 2018, 2025, Oracle and/or its affiliates.

17

Common Configuration and Management Tasks for an Enterprise Deployment

The configuration tasks include a few that are common to all enterprise deployments, such as verifying sizing information, performing backups and recoveries, and so on. Patching an enterprise deployment and cross wiring components are the other common tasks.

This chapter includes the following topics:

- Configuration and Management Tasks for All Enterprise Deployments
  These are some of the typical configuration and management tasks you are likely need to perform on an Oracle Fusion Middleware enterprise deployment.

Configuration and Management Tasks for All Enterprise Deployments

These are some of the typical configuration and management tasks you are likely need to perform on an Oracle Fusion Middleware enterprise deployment.

- Verifying Appropriate Sizing and Configuration for the WLSSchemaDataSource
  In Oracle FMW 14.1.2, `WLSRuntimeSchemaDataSource` is the common datasource that is reserved for use by the FMW components for JMS JDBC Stores, JTA JDBC stores, and Leasing services. `WLSRuntimeSchemaDataSource` is used to avoid contention in critical WLS infrastructure services and to guard against dead-locks.
- Verifying Manual Failover of the Administration Server
  In case a host computer fails, you can fail over the Administration Server to another host. The steps to verify the failover and failback of the Administration Server from OIGHOST1 and OIGHOST2 are detailed in the following sections.
- Modifying the Upload and Stage Directories to an Absolute Path in an Enterprise Deployment
  After you configure the domain and unpack it to the Managed Server domain directories on all the hosts, verify and update the upload and stage directories for Managed Servers in the new clusters. Also, update the upload directory for the AdminServer to have the same absolute path instead of relative, otherwise deployment issues can occur.
- Setting the Front End Host and Port for a WebLogic Cluster
  You must set the front-end HTTP host and port for the Oracle WebLogic Server cluster that hosts the Oracle Identity and Access Management servers. You can specify these values in the Configuration Wizard while you are specifying the properties of the domain.
- Using Persistent Stores for TLOGs and JMS in an Enterprise Deployment
  The persistent store provides a built-in, high-performance storage solution for WebLogic Server subsystems and services that require persistence.
- About JDBC Persistent Stores for Web Services
  By default, web services use the WebLogic Server default persistent store for persistence. This store provides high-performance storage solution for web services.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Best Configuration Practices When Using RAC and Gridlink Datasources
  Oracle recommends that you use GridLink data sources when you use an Oracle RAC database. If you follow the steps described in the Enterprise Deployment guide, the datasources will be configured as GridLink.
- Using TNS Alias in Connect Strings
  You can create an alias to map the URL information instead of specifying long database connection strings in the jdbc connection pool of a datasource. The connection string information is stored in a `tnsnames.ora` file with an associated alias name. This alias is used in the connect string of the connection pool.
- Performing Backups and Recoveries for an Enterprise Deployment
  It is recommended that you follow the below mentioned guidelines to make sure that you back up the necessary directories and configuration data for an Oracle Identity and Access Management enterprise deployment.

Verifying Appropriate Sizing and Configuration for the WLSSchemaDataSource

In Oracle FMW 14.1.2, `WLSRuntimeSchemaDataSource` is the common datasource that is reserved for use by the FMW components for JMS JDBC Stores, JTA JDBC stores, and Leasing services. `WLSRuntimeSchemaDataSource` is used to avoid contention in critical WLS infrastructure services and to guard against dead-locks.

To reduce the `WLSRuntimeSchemaDataSource` connection usage, you can change the JMS JDBC and TLOG JDBC stores connection caching policy from *Default* to *Minimal* by using the respective connection caching policy settings. When there is a need to reduce connections in the back-end database system, Oracle recommends that you set the caching policy to *Minimal* . Avoid using the caching policy *None* because it causes a potential degradation in performance. For a detailed tuning advice about connections that are used by JDBC stores, see Configuring a JDBC Store Connection Caching Policy in *Administering the WebLogic Persistent Store*.

The default `WLSRuntimeSchemaDataSource` connection pool size is 75 (size is double in the case of a GridLink DataSource). You can tune this size to a higher value depending on the size of the different FMW clusters and the candidates that are configured for migration. For example, consider a typical SOA EDG deployment with the default number of worker threads per store. If more than 25 JDBC Stores or TLOG-in-DB instances or both can fail over to the same Weblogic server, and the Connection Caching Policy is not changed from *Default* to *Minimal*, possible connection contention issues could arise. In these cases, increasing the default `WLSRuntimeSchemaDataSource` pool size (maximum capacity) becomes necessary (each JMS store uses a minimum of two connections, and leasing and JTA are also added to compete for the pool).

Verifying Manual Failover of the Administration Server

In case a host computer fails, you can fail over the Administration Server to another host. The steps to verify the failover and failback of the Administration Server from OIGHOST1 and OIGHOST2 are detailed in the following sections.

Assumptions:

- The Administration Server is configured to listen on IGDADMINVHN or any custom virtual host that maps to a floating IP/VIP. It should not listen on ANY (blank listen address), localhost or any host name that uniquely identifies a single node.

Copyright © 2018, 2025, Oracle and/or its affiliates.

For more information about the IGDADMINVHN virtual IP address, see Reserving the Required IP Addresses for an Enterprise Deployment.

- These procedures assume that the Administration Server domain home (*IGD_ASERVER_HOME*) has been mounted on both host computers. This ensures that the Administration Server domain configuration files and the persistent stores are saved on the shared storage device.
- The Administration Server is failed over from OIGHOST1 to OIGHOST2, and the two nodes have these IPs:
  - OIGHOST1: 100.200.140.165
  - OIGHOST2: 100.200.140.205
  - IGDADMINVHN : 100.200.140.206. This is the Virtual IP where the Administration Server is running, assigned to a virtual sub-interface (for example, eth0:1), to be available on OIGHOST1 or OIGHOST2.
- Oracle WebLogic Server and Oracle Fusion Middleware components have been installed in OIGHOST2 as described in the specific configuration chapters in this guide.

  Specifically, both host computers use the exact same path to reference the binary files in the Oracle home.

The following topics provide details on how to perform a test of the Administration Server failover procedure.

- Failing Over the Administration Server When Using a Per Host Node Manager
  The following procedure shows how to fail over the Administration Server to a different node (OAMHOST2). Note that even after failover, the Administration Server will still use the same Oracle WebLogic Server *machine* (which is a logical machine, not a physical machine).
- Validating Access to the Administration Server on OIGHOST2 Through Oracle HTTP Server
  If you have configured the web tier to access AdminServer, it is important to verify that you can access the Administration Server after you perform a manual failover of the Administration Server, by using the standard administration URLs.
- Failing the Administration Server Back to OIGHOST1 When Using a Per Host Node Manager
  After you have tested a manual Administration Server failover, and after you have validated that you can access the administration URLs after the failover, you can then migrate the Administration Server back to its original host.

Failing Over the Administration Server When Using a Per Host Node Manager

The following procedure shows how to fail over the Administration Server to a different node (OAMHOST2). Note that even after failover, the Administration Server will still use the same Oracle WebLogic Server *machine* (which is a logical machine, not a physical machine).

This procedure assumes you've configured a per host Node Manager for the enterprise topology, as described in Creating a Per Host Node Manager Configuration. For more information, see About the Node Manager Configuration in a Typical Enterprise Deployment.

To fail over the Administration Server to a different host:

1. Stop the Administration Server on OAMHOST1.
2. Stop the Node Manager on OAMHOST1.

   You can use the script `stopNodeManager.sh` that was created in NM_HOME.

Copyright © 2018, 2025, Oracle and/or its affiliates.

3. Migrate the IADADMINVHN virtual IP address to the second host:
   a. Run the following command as root on OAMHOST1 (where X is the current interface used by IADADMINVHN) to check the virtual IP address at its CIDR:

      ```
      ip addr show dev ethX
      ```

      For example:

      ```
      ip addr show dev eth0
      ```

   b. Run the following command as root on OAMHOST1 (where *X* is the current interface used by IADADMINVHN):

      ```
      ip addr del IADADMINVHN/CIDR dev ethX
      ```

      For example:

      ```
      ip addr del 100.200.140.206/24 dev eth0
      ```

   c. Run the following command as root on OAMHOST2:

      ```
      ip addr add IADADMINVHN/CIDR dev ethX label ethX:Y
      ```

      For example:

      ```
      ip addr add 100.200.140.206/24 dev eth0 label eth0:1
      ```

      > **Note**
      >
      > Ensure that the CIDR and interface to be used match the available network configuration in OAMHOST2.

4. Update the routing tables by using `arping`, for example:

   ```
   arping -b -A -c 3 -I eth0 100.200.140.206
   ```

5. From OAMHOST1, change directory to the Node Manager home directory:

   ```
   cd $NM_HOME
   ```

6. Edit the `nodemanager.domains` file and remove the reference to *IAD_ASERVER_HOME*.

   The resulting entry in the OAMHOST1 `nodemanager.domains` file should appear as follows:

   ```
   iamedg_domain=IAD_MSERVER_HOME;
   ```

7. From OAMHOST2, change directory to the Node Manager home directory:

   ```
   cd $NM_HOME
   ```

8. Edit the `nodemanager.domains` file and add the reference to *IAD_ASERVER_HOME*.

   The resulting entry in the OAMHOST2 `nodemanager.domains` file should appear as follows:

   ```
   iamedg_domain=IAD_MSERVER_HOME;IAD_ASERVER_HOME
   ```

9. Start the Node Manager on OAMHOST1 and restart the Node Manager on OAMHOST2.
10. Start the Administration Server on OAMHOST2.
11. Check that you can access the Administration Server on OAMHOST2 and verify the status of components in Fusion Middleware Control using the following URL:

    ```
    https://iadadminvhn.example.com:9002/em
    ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

Validating Access to the Administration Server on OIGHOST2 Through Oracle HTTP Server

If you have configured the web tier to access AdminServer, it is important to verify that you can access the Administration Server after you perform a manual failover of the Administration Server, by using the standard administration URLs.

From the load balancer, access the following URLs to ensure that you can access the Administration Server when it is running on OIGHOST2:

- For SSL Terminated: `http://igdadmin.example.com/em`
- For End to End SSL: `https://igdadmin.example.com/em`

This URL should display Oracle Enterprise Manager Fusion Middleware Control.

Failing the Administration Server Back to OIGHOST1 When Using a Per Host Node Manager

After you have tested a manual Administration Server failover, and after you have validated that you can access the administration URLs after the failover, you can then migrate the Administration Server back to its original host.

This procedure assumes that you have configured a per host Node Manager for the enterprise topology, as described in Creating a Per Host Node Manager Configuration. For more information, see About the Node Manager Configuration in a Typical Enterprise Deployment.

1. Stop the Administration Server on OIGHOST2.
2. Stop the Node Manager on OIGHOST2.
3. Run the following command as root on OIGHOST2.

   ```
   ip addr del IGDADMINVHN/CIDR dev ethX
   ```

   For example:

   ```
   ip addr del 100.200.140.206/24 dev eth0
   ```

4. Run the following command as root on OIGHOST1:

   ```
   ip addr add IGDADMINVHN/CIDR dev ethX label ethX:Y
   ```

   For example:

   ```
   ip addr add 100.200.140.206/24 dev eth0 label eth0:1
   ```

   > **Note**
   >
   > Ensure that the CIDR and interface to be used match the available network configuration in OIGHOST1.

5. Update the routing tables by using `arping` on OIGHOST1:

   ```
   arping -b -A -c 3 -I eth0 100.200.140.206
   ```

6. From OIGHOST2, change directory to the Node Manager home directory:

   ```
   cd $NM_HOME
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

7. Edit the `nodemanager.domains` file and remove the reference to *IGD_ASERVER_HOME*.
8. From OIGHOST1, change directory to the Node Manager home directory:

   ```
   cd $NM_HOME
   ```

9. Edit the `nodemanager.domains` file and add the reference to *IGD_ASERVER_HOME*.
10. Start the Node Manager on OIGHOST2 and restart the Node Manager on OIGHOST1.
11. Start the Administration Server on OIGHOST1.
12. Test that you can use the WebLogic Remote Console to access the provider defined for this domain.
13. Check that you can access and verify the status of components in the Oracle Enterprise Manager by using the URL outlined in URLs Used in This Chapter.

Modifying the Upload and Stage Directories to an Absolute Path in an Enterprise Deployment

After you configure the domain and unpack it to the Managed Server domain directories on all the hosts, verify and update the upload and stage directories for Managed Servers in the new clusters. Also, update the upload directory for the AdminServer to have the same absolute path instead of relative, otherwise deployment issues can occur.

> **Note**
>
> This task is not required for Oracle Access Management Infrastructure.

This step is necessary to avoid potential issues when you perform remote deployments and for deployments that require the stage mode.

To update the directory paths for the Deployment Stage and Upload locations, complete the following steps:

1. Log into the WebLogic Remote Console to access the provider of this domain.
2. Open the **Edit Tree**.
3. Expand **Environment**.
4. Expand **Servers**.
5. Click the name of the Managed Server you want to edit. Perform the following steps for each of the Managed Server:
   a. Click the **Advanced** tab.
   b. Click the **Deployment** tab.
   c. Verify that the Staging Directory Name is set to the following:
      IGD_MSERVER_HOME`/servers/server_name/stage`

      Replace *IGD_MSERVER_HOME* with the full path for the *IGD_MSERVER_HOME* directory.

      Update with the correct name of the Managed Server that you are editing.
   d. Update the Upload Directory Name to the following value:
      IGD_ASERVER_HOME`/servers/AdminServer/upload`

      Replace *IGD_ASERVER_HOME* with the directory path for the *IGD_ASERVER_HOME* directory.

Copyright © 2018, 2025, Oracle and/or its affiliates.

e. Click **Save**.

f. Return to the **Summary of Servers** screen.

Repeat the same steps for each of the new managed servers.

6. Navigate to and update the Upload Directory Name value for the **AdminServer**:
   a. Navigate to **Servers** and select the **AdminServer**.
   b. Click the **Advanced** tab.
   c. Click the **Deployment** tab
   d. Verify that the Staging Directory Name is set to the following absolute path: *IGD_ASERVER_HOME*/servers/AdminServer/stage
   e. Update the Upload Directory Name to the following absolute path: *IGD_ASERVER_HOME*/servers/AdminServer/upload

      Replace *IGD_ASERVER_HOME* with the directory path for the *ASERVER_HOME* directory.
   f. Click **Save**.
7. When you have modified all the appropriate objects, commit the changes in the shopping cart.
8. Restart all the Servers for the changes to take effect. If you are following the EDG steps in-order and are not going to make any deployments immediately, you can wait until the next restart.

> **Note**
>
> If you continue directly with further domain configurations, a restart to enable the stage and upload directory changes is not strictly necessary at this time.

Setting the Front End Host and Port for a WebLogic Cluster

You must set the front-end HTTP host and port for the Oracle WebLogic Server cluster that hosts the Oracle Identity and Access Management servers. You can specify these values in the Configuration Wizard while you are specifying the properties of the domain.

However, when you add a SOA cluster as part of an Oracle Identity and Access Management enterprise deployment, Oracle recommends that you perform this task after you verify the SOA Managed Servers.

To set the frontend host and port from the WebLogic Server Administration Console:

1. Log in to the WebLogic Remote Console.
2. In the **Edit Tree**, click **Environment** then **Clusters**.
3. From the **Customize Table** menu, select **Frontend Host**, **Frontend HTTP Port** and **Frontend HTTPS Port** from the **Available Columns** and move to the **Selected Columns**.
4. Select the cluster as per the table below and then the **Http** tab.
5. Use the information in Table 17-1 to add the required frontend hostname and port to each cluster.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 17-1 The Frontend Hostname and Port for Each Cluster

Name	Frontend Host	Frontend HTTP Port	Frontend HTTPS
OAM_Cluster	login.example.com		443
AMA_Cluster	iadadmin.example.com	80	
OIM_Cluster	N/A	N/A	N/A
SOA_Cluster	igdinternal.example.com	7777	

6. Click **Save**.
7. Repeat the above for all clusters referenced in the table.
8. Click the **Shopping Cart** then **Commit Changes**.
9. Restart the managed servers of the cluster.

Using Persistent Stores for TLOGs and JMS in an Enterprise Deployment

The persistent store provides a built-in, high-performance storage solution for WebLogic Server subsystems and services that require persistence.

For example, the JMS subsystem stores persistent JMS messages and durable subscribers, and the JTA Transaction Log (TLOG) stores information about the committed transactions that are coordinated by the server but may not have been completed. The persistent store supports persistence to a file-based store or to a JDBC-enabled database. Persistent stores' high availability is provided by server or service migration. Server or service migration requires that all members of a WebLogic cluster have access to the same transaction and JMS persistent stores (regardless of whether the persistent store is file-based or database-based).

For an enterprise deployment, Oracle recommends using JDBC persistent stores for transaction logs (TLOGs) and JMS.

About JDBC Persistent Stores for Web Services

By default, web services use the WebLogic Server default persistent store for persistence. This store provides high-performance storage solution for web services.

The default web service persistence store is used by the following advanced features:

- Reliable Messaging
- Make Connection
- SecureConversation
- Message buffering

You also have the option to use a JDBC persistence store in your WebLogic Server web service, instead of the default store. For information about web service persistence, see Managing Web Service Persistence.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Best Configuration Practices When Using RAC and Gridlink Datasources

Oracle recommends that you use GridLink data sources when you use an Oracle RAC database. If you follow the steps described in the Enterprise Deployment guide, the datasources will be configured as GridLink.

GridLink datasources provide dynamic load balancing and failover across the nodes in an Oracle Database cluster, and also receive notifications from the RAC cluster when nodes are added or removed. For more information about GridLink datasources, see Using Active GridLink Data Sources in *Administering JDBC Data Sources for Oracle WebLogic Server*.

Here is a summary of the best practices when using GridLink to connect to the RAC database:

- Use a database service (defined with srvctl) different from the default database service
  In order to receive and process notifications from the RAC database, the GridLink needs to connect to a database service (defined with `srvctl`) instead to a default database service. These services monitor the status of resources in the database cluster and generate notifications when the status changes. A database service is used in Enterprise Deployment guide, created and configured as described in Creating Database Services.
- Use the long format database connect string in the datasources
  When Gridlink datasources are used, the long format database connect string must be used. The Configuration Wizard does not set the long format string, it sets the short format instead. You can modify it manually later to set the long format. To update the datasources:
  1. Connect to the WebLogic Server Console and navigate to **Domain Structure** > **Services** > **Datasources**.
  2. Select a datasource, click the **Configuration** tab, and then click the **Connection Pool** tab.
  3. Within the JDBC URL, change the **URL** from `jdbc:oracle:thin:[SCAN_VIP]:[SCAN_PORT]/[SERVICE_NAME]` to
     `jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=[SCAN_VIP])(PORT=[SCAN_PORT])))(CONNECT_DATA=(SERVICE_NAME=[SERVICE_NAME])))`
     For example:

     ```
     jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(LOAD_BALANCE=ON)(ADDRESS=(PROTOCOL=TCP)(HOST=db-scan-address)(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=iadedg.example.com)))
     ```

- Use auto-ons
  The ONS connection list is automatically provided from the database to the driver. You can leave the ONS Nodes list empty in the datasources configuration.
- Test Connections On Reserve
  Verify that the **Test Connections On Reserve** is checked in the datasources.

  Eventhough the GridLink datasources receive FAN events when a RAC instances becomes unavailable, it is a best practice to enable the **Test Connections On Reserve** in the datasource and ensure that the connection returned to the application is good.
- Seconds to Trust an Idle Pool Connection
  For a maximum efficiency of the test, you can also set **Seconds to Trust an Idle Pool Connection** to `0`, so the connections are always verified. Setting this value to zero means that all the connections returned to the application will be tested. If this parameter is set to `10`, the result of the previous test will be valid for 10 seconds and if a connection is reused before the lapse of 10 seconds, the result will still be valid.

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Test Frequency
  Verify that the **Test Frequency** parameter value in the datasources is not `0`. This is the number of seconds a WebLogic Server instance waits between attempts when testing unused connections. The default value of `120` is normally enough.

Using TNS Alias in Connect Strings

You can create an alias to map the URL information instead of specifying long database connection strings in the jdbc connection pool of a datasource. The connection string information is stored in a `tnsnames.ora` file with an associated alias name. This alias is used in the connect string of the connection pool.

The following example is of a connect string using tns alias.

```
jdbc:oracle:thin:@edg_alias
```

The `tnsnames.ora` file contains the following details.

```
edg_alias =
        (DESCRIPTION=
        (ADDRESS_LIST=
            (LOAD_BALANCE=ON)
            (ADDRESS=(PROTOCOL=TCP)(HOST=db-scan.example.com)(PORT=1521)))
            (CONNECT_DATA=(SERVICE_NAME=edg.example.com))
        )
```

You must specify the `oracle.net.tns_admin` property in the datasource configuration to point to a specific `tnsnames.ora` file. For example, `<property><name>oracle.net.tns_admin</name><value>/u01/oracle/config/domains/domain_name/config/tnsadmin</value></property></properties>`

This is the Maximum Availability and Enterprise Deployment recommended approach for JDBC urls. It simplifies JDBC configurations, facilitates DB configuration aliasing in disaster protection scenarios, and makes database connection changes more dynamic. For more information, see Use a TNS Alias Instead of a DB Connection String in *Administering JDBC Data Sources for Oracle WebLogic Server*.

In Oracle Fusion Middleware 14.1.2, you can use a new type of deployment module to manage the `tnsnames.ora` files, wallet files, and keystore and truststore files associated with a database connection. These are called DBClientData modules. For more information, see What Are DBClientData Modules in *Administering JDBC Data Sources for Oracle WebLogic Server*. In this EDG, DBClientData type of module is used to maintain the database client information. However, wallets and SSL configuration is not used to access the database so the DBClientData module contains only the appropriate `tnsnames.ora`.

The following steps are required to use a TNS alias in the different Datasources used by FMW and WLS schemas:

1. Create a `tnsnames.ora` with the pertaining alias and mapping URLs used in the connection pools. Copy the connect string from one of the existing datasource configuration files. For example,

Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> This is an example using the short jdbc URL.

```
grep url /u01/oracle/config/domains/oam/config/jdbc/opss-datasource-
jdbc.xml
    <url>jdbc:oracle:thin:@db-scan.example.com:1521/edg.example.com</url>
```

Use the information in the connect string to add a long URL entry to a `tnsnames.ora` file. Use an alias name that identifies your connection. Notice that in order to deploy the `tsnnames.ora` as DBCLient module the location of the deployment module needs to be two levels down under the domain config directory if it resides on the WLS Administration Server node. The file can also be created in the node that runs the WebLogic Remote Console and can also be uploaded (as an application ear or war file).

```
cat /u01/oracle/config/tnsadmin/tnsnames.ora
```

The output will look similar to the following:

```
edg_alias =
        (DESCRIPTION=
        (ADDRESS_LIST=
            (LOAD_BALANCE=ON)
            (ADDRESS=(PROTOCOL=TCP)(HOST= db-scan.example.com)(PORT=1521)))
            (CONNECT_DATA=(SERVICE_NAME=edg.example.com))
        )
```

2. Deploy the directory containing the `tnsnames.ora` as a DBClientData module.
   a. Access the domain provider in the WebLogic Remote Console.
   b. Click **Edit Tree**.
   c. Click **Environment** > **Deployments** >**Database Client Data Directories**.
   d. Click **New**.
   e. Enter a name for the dbclient directory deployment. For example, `dbclientdata_modulename`.

      If the directory containing the `tnsnames.ora` file resides on your local computer, uncheck the **Upload** checkbox.
   f. Click **Create**.
   g. Click **Save**.

      The cart on the top right part of the screen will display full with a yellow bag inside.
   h. Click the **Cart** icon and select **Commit Changes**.

      This will create a `tnsnames/dbclient` module under domain dir `/u01/oracle/config/domains/domain_name/config/dbclientdata/dbclientdata_modulename`.

      You can also perform the deployment of a database client module using the `deploy` command in wlst.

Copyright © 2018, 2025, Oracle and/or its affiliates.

3. Update the different Datasources and fmwconfig files to use the alias instead of the explicit URLS.

> **Note**
>
> To update a datasource to use the tns alias, the datasource configuration needs to include both a pointer to the `tsnames.ora` file and the alias itself in the jdbc URL.

You must perform the following steps to include a pointer to the `tnsnames.ora` file include the property `oracle.net.tns_admin` in the datasource properties.

a. Access the domain provider in the WebLogic Remote Console.
b. Click **Edit Tree**.
c. Click **Services** > **Datasources** > **Datasource_name**.
d. In the navigation tree on the left, select **Properties** for the precise Datasource.
e. Click **New**.
f. Enter `oracle.net.tns_admin` as the property name.
g. Click **Create**.
h. In the next screen with the property details, enter as **value** the directory for the `dbclientdata_modulename` that is `/u01/oracle/config/domains/domain_name/config/dbclientdata/dbclientdata_modulename` in the example above.
i. Click **Save**.

   The cart on the top right part of the screen will display full with a yellow bag inside.
j. In the navigation tree on the left, click the **Datasource** name.
k. Select the **Connection Pool** tab.
l. In the URL, replace the URL with the alias syntax as shown below:

   ```
   jdbc:oracle:thin:@edg_alias
   ```
m. Click **Save**.
n. Click the **Cart** icon and select **Commit Changes**.

   If you check the datasource configuration file, it should reflect the following under the `<jdbc-driver-params> <properties>` entries:

   ```
   <property>
   <name>oracle.net.tns_admin</name>
   <value>/u01/oracle/config/domains/domain_name/config/dbclientdata/
   dbclientdata_modulename</value>
   </property>
   ```

   The datasource configuration file should reflect as JDBC URL under `<jdbc-driver-params>` as shown below:

   ```
   <url>jdbc:oracle:thin:@edg_alias</url>
   ```

4. To update the FMW jps config to use the tns alias, the `domain_path/config/fmwconfig/jps-config.xml` and `domain_path/config/fmwconfig/jps-config-jse.xml` files need to be updated and both a pointer to the `tsnames.ora` file and the alias itself must be included

Copyright © 2018, 2025, Oracle and/or its affiliates.

in the jdbc url, that is replace the information in the propertySet for the DB with the updated URL and the tnsadmin pointer.

```
<property name="oracle.net.tns_admin" value="/u01/oracle/config/domains/
domain_name/config/dbclientdata/dbclientdata_modulename "/>
<property name="jdbc.url" value="jdbc:oracle:thin:@edg_alias "/>
```

Restart the Administration Server for all the changes to be applied.

Alternatively, you can use the `https://github.com/oracle-samples/maa/tree/main/1412EDG/fmw1412_change_to_tns_alias.sh` script instead of the steps 1, 2, 3 and 4 to deploy the corresponding DBClientData module and replace all urls in the jdbc and jps configuration with the pertaining alias.

The recommended approach is to configure your domain to suit your functional needs as described in About the IAM Enterprise Deployment. After your domain is complete and working, use the script to make the TNS alias change. Ensure you read the script's instructions in its header for its correct execution.

Performing Backups and Recoveries for an Enterprise Deployment

It is recommended that you follow the below mentioned guidelines to make sure that you back up the necessary directories and configuration data for an Oracle Identity and Access Management enterprise deployment.

> **Note**
>
> Some of the static and runtime artifacts listed in this section are hosted from Network Attached Storage (NAS). If possible, backup and recover these volumes from the NAS filer directly rather than from the application servers.

For general information about backing up and recovering Oracle Fusion Middleware products, see the following sections in *Administering Oracle Fusion Middleware*:

- Backing Up Your Environment
- Recovering Your Environment

Table 17-2 lists the static artifacts to back up in a typical Oracle Identity and Access Management enterprise deployment.

Table 17-2 Static Artifacts to Back Up in the Oracle Identity and Access Management Enterprise Deployment

Type	Host	Tier
Database Oracle home	DBHOST1 and DBHOST2	Data Tier
Oracle Fusion Middleware Oracle home	WEBHOST1 and WEBHOST2	Web Tier
Oracle Fusion Middleware Oracle home	OIGHOST1 and OIGHOST2 (or NAS Filer)	Application Tier
Installation-related files	WEBHOST1, WEHOST2, and shared storage	N/A

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 17-3 lists the runtime artifacts to back up in a typical Oracle Identity and Access Management enterprise deployment.

Table 17-3 Run-Time Artifacts to Back Up in the Oracle Identity and Access Management Enterprise Deployment

Type	Host	Tier
Administration Server domain home (ASERVER_HOME)	OIGHOST1 (or NAS Filer)	Application Tier
Application home (APPLICATION_HOME)	OIGHOST1 (or NAS Filer)	Application Tier
Oracle RAC databases	DBHOST1 and DBHOST2	Data Tier
Scripts and Customizations	Per host	Application Tier
Deployment Plan home (DEPLOY_PLAN_HOME)	OIGHOST1 (or NAS Filer)	Application Tier
OHS Configuration directory	WEBHOST1 and WEBHOST2	Web Tier

Copyright © 2018, 2025, Oracle and/or its affiliates.

18

Scaling Procedures for an Enterprise Deployment

The scaling procedures for an enterprise deployment include scale out, scale in, scale up, and scale down. During a scale-out operation, you add managed servers to new nodes. You can remove these managed servers by performing a scale in operation. During a scale-up operation, you add managed servers to existing hosts. You can remove these servers by performing a scale-down operation.

This chapter describes the procedures to scale out/in and scale up/down static and dynamic clusters.

- Scaling Out the Topology
  When you scale out the topology, you add new managed servers to new nodes.

Scaling Out the Topology

When you scale out the topology, you add new managed servers to new nodes.

This section describes the procedures to scale out the Identity Management topology.

- Prerequisites for Scaling Out
- Scaling Out a Cluster
- Verifying the Scale Out
- Scaling in the Topology

Prerequisites for Scaling Out

Before you perform a scale out of the topology, you must ensure that you meet the following requirements:

- The starting point is a cluster with managed servers already running.
- The new node can access the existing home directories for WebLogic Server and Governance. Use the existing installations in shared storage. You do not need to install WebLogic Server or IDM binaries in a new location. However, you do need to run `pack` and `unpack` commands to bootstrap the domain configuration in the new node.
- It is assumed that the cluster syntax is used for all internal RMI invocations, JMS adapter, and so on.

Scaling Out a Cluster

The steps provided in this procedure use the IDM EDG topology as a reference. Initially there are two application tier hosts (OAMHOST1 and OAMHOST2, or OIGHOST1 and OIGHOST2), each running one managed server of each cluster. A new host HOST3 is added to scale up the clusters with a third managed server. `,<server>`*n* is the generic name given to the new managed server that you add to the cluster. Depending on the cluster that is being extended

Copyright © 2018, 2025, Oracle and/or its affiliates.

and the number of existing nodes, the actual names will be `oam_server3`, `oam_policy_mgr3`, `oim_server3`, and so on.

To scale out the cluster, complete the following steps:

1. On the new node, mount the existing FMW Home, which should include the IDM installation and the domain directory. Ensure that the new node has access to this directory, similar to the rest of the nodes in the domain. Also, ensure that the FMW Home you mount is the one associated with the domain your are extending.
2. Locate the inventory in the shared directory (for example, `/u01/oracle/products/oraInventory`), per Oracle's recommendation. So you do not need to attach any home, but you may want to execute the script: `/u01/oracle/products/oraInventory/createCentralInventory.sh`.

   This command creates and updates the local file `/etc/oraInst.loc` in the new node to point it to the oraInventory location.

   If there are other inventory locations in the new host, you can use them, but `/etc/oraInst.loc` file must be updated accordingly for updates in each case.
3. Update the `/etc/hosts` files to add the name of the new host (unless you are using DNS), as described in Verifying IP Addresses and Host Names in DNS or Hosts File. If you are using host aliases such as OIGHOST, then ensure that you add an entry for that host.

   For example:

   ```
   10.229.188.203 iadadminvhn.example.com iadadminvhn
   10.229.188.204 igdadminvhn.example.com igdadminvhn
   10.229.188.205 oighost1.example.com oighost1
   10.229.188.206 oighost2.example.com oighost2
   10.229.188.207 oighost3.example.com oighost3
   10.229.188.208 oighost4.example.com oighost4
   10.229.188.209 webhost1.example.com webhost1
   10.229.188.210 webhost2.example.com webhost2
   10.229.188.211 oamhost1.example.com oamhost1
   10.229.188.212 oamhost2.example.com oamhost2
   10.229.188.213 oamhost3.example.com oamhost3
   10.229.188.214 oamhost4.example.com oamhost4
   ```
4. Log into the WebLogic Remote Console to create a new machine:
   a. Go to **Environment** and select **Machines**.
   b. Click **New** to create a new machine for the new node.
   c. Set **Name** to oighost*n*.
   d. Click the **Node Manager** tab.
   e. For End to End SSL set **Type** to **SSL**.
   f. Set **Listen Address** to `oighostn.example.com`
   g. Click **Save and Commit changes** in the **Shopping Cart**.
5. Use the Oracle WebLogic Remote Console to clone the first managed server in the cluster into a new managed server.
   a. Go to **Environment** and select **Servers**.
   b. Click **Create** and in the **Copy settings from another server**, select the first managed server in the cluster to scale out and click **Create**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

c. Select the first managed server in the cluster to scale out and click **Create**.

d. Use Details of the Cluster to be Scaled Out to set the correspondent name, listen address, and SSL listen port depending on the cluster that you want to scale out.

e. Click the new managed server, select **Configuration**, and then click **General**.

f. Update the **Machine** from OIGHOST1 to OIGHOST*n*.

g. Update the Administration port for the server also to be consistent with other server in the cluster. Refer to the existing servers for their appropriate Administration Port.

h. Click **Save and Commit changes** in the **Shopping Cart**.

Table 18-1 Details of the Cluster to be Scaled Out

Cluster to Scale Out	Server to Clone	New Server Name	Server Listen Address	Server Listen Port	SSL Server Listen Port	Server Admin Port
SOA_Cluster	soa_server1	soa_server*n*	oighost*n*.example.com	7003	7004	9004
OIM_Cluster	oim_server1	oim_server*n*	oighost*n*.example.com	14000	14001	9010
OAM_Cluster	oam_server1	oam_server*n*	oamhost*n*.example.com	14100	14101	9508
AMA_Cluster	oam_policy_mgr1	oam_policy_mgr*n*	oamhost*n*.example.com	14150	14151	9509

6. Update the deployment Staging Directory Name of the new server, as described in Modifying the Upload and Stage Directories to an Absolute Path in an Enterprise Deployment.

7. Create a new certificate as described in Obtaining SSL Certificates and update your domain certificate store as described in Update Server's Security Settings Using the Remote Console.

8. Your new server's keystore location and ssl configuration is carried over from the server copied (`oim_server1`) but you must update the password again (since it will be encrypted again for the new server) and the "Server private key alias" entry for this new server.

> **Note**
>
> This is only required for End to End SSL.

a. Navigate to **Environment** > **Servers**.

b. Click the new server.

c. Navigate to **Security** > **Keystores**.

d. Update the **Custom Identity Key Store Pass Phrase** and **Custom Trust Key Store Pass Phrase** with the password you providied when you created the SSL identity and trust stores.

e. Click the **SSL** tab under **Security**.

f. Update the **Server Private Key Pass Phrase** with the password with the password you provided when you created the certificate.

Copyright © 2018, 2025, Oracle and/or its affiliates.

g. Add the listen address that you used in the previous step (certificate generation for the new server) as **Server Private Key Alias**.

9. Update the TLOG JDBC persistent store of the new managed server:

   a. Log into the **WebLogic Remote Console**.

   b. Go to **Environment** and expand the **Servers** link on the navigation tree on the left.

   c. Click the new server `<server>`*n*.

   d. Click the **Services** > **JTA** tab.

   e. Ensure **Transaction Log Store** in JDBC is selected and change the **Transaction Log Prefix** name to **TLOG**_`<server>`*n*.

   f. The rest of the fields are carried over from the server copied (including the Datasource used for the JDBC store) **WLSRuntimeSchemaDataSource**.

   g. Click **Save** and **Commit changes** in the **Shopping Cart**.

   Use the following table to identify the clusters that use JDBC TLOGs by default:

Table 18-2 The Name of Clusters that Use JDBC TLOGs by Default

Cluster to Scale Out	New Server Name	TLOG Persistent Store
SOA_Cluster	soa_server*n*	JDBC
OIM_Cluster	oim_server*n*	Default JDBC
OAM_Cluster	oam_server*n*	Not Applicable
AMA_Cluster	oam_policy_mgr*n*	Not Applicable

10. If the cluster that you are scaling out is configured for automatic service migration, update the **JTA Migration Policy** to the required value.

    a. Go to **Environment** and expand **Servers**. From the list of servers, select **`<server>`*n*** , click the **JTA Migratable Target**.

    b. Use #unique_528/unique_528_Connect_42_GUID-888AED4E-E432-4DAA-AEFB-17E20927838B to ensure the recommended JTA Migration Policy depending on the cluster that you want to scale out is set.

Table 18-3 The Recommended JTA Migration Policy for the Cluster to be Scaled Out

Cluster to Scale Out	New Server Name	JTA Migration Policy
SOA_Cluster	soa_server*n*	Failure Recovery
OAM_Cluster	oam_server*n*	Not Applicable
AMA_Cluster	oam_policy_mgr*n*	Not Applicable
OIM_Cluster	oim_server*n*	Failure Recovery

11. If the cluster you are scaling out is configured for automatic service migration, use the Oracle WebLogic Remote Console to update the automatically created `<server>`*n* (migratable) target with the recommended migration policy, because by default it is set to **Manual Service Migration Only**.

    Use the following table for the list of migratable targets to update:

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 18-4 The Recommended Migratable Targets to Update

Cluster to Scale Out	Migratable Target to Update	Migration Policy
SOA_Cluster	soa_server*n* (migratable)	Auto-Migrate Failure Recovery Services
OIM_Cluster	oim_server*n* (migratable)	Auto-Migrate Failure Recovery Services
OAM_Cluster	Not applicable	Not applicable
AMA_Cluster	Not applicable	Not applicable

a. Go to **Environment** then **Migratable Targets**.

b. Click `<server>`3 (migratable), for example `oim_server3`.

c. Change the **Service Migration Policy** to the value listed in the table.

d. Leave the **Constrained Candidate Server** list blank in case there are chosen servers. If no servers are selected, you can migrate this migratable target to any server in the cluster.

e. Click **Save** and **Commit Changes** in the **Shopping Cart**. Notice that a change from the default migration policy (manual) requires a restart of the existing servers in the cluster.

12. Verify that the **Constrained Candidate Server** list in the existing migratable servers in the cluster is empty. It should be empty out-of-the-box because the Configuration Wizard leaves it empty. An empty candidate list means that all the servers in the cluster are candidates, which is the best practice.

    a. Go to each migratable server.

    b. Click the **Migration** tab and check the **Constrained Candidate Servers** list.

    c. Ensure that "Chosen" server list is empty. It should be empty out-of-the-box.

    d. If the server list is not empty, you should modify the list to make it blank. Or, if your list is not empty because you explicitly decided to constrain the migration to some specific servers only, modify it as per your preferences to accommodate the new server. Click **Save** and **Commit Changes** in the **Shopping Cart**. Notice that a change in the candidate list requires a restart of the existing servers in the cluster.

13. Create the required persistent stores for the JMS servers used in the new server.

    a. Log into the WebLogic Remote Console. In the **Edit Tree** expand **Services** and select **JDBC stores**.

    b. Click **New**.

    Use the following table to create the required persistent stores:

Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> The number in names and prefixes in the existing resources were assigned automatically by the Configuration Wizard during the domain creation. For example:
>
> - BPMJMSJDBCStore_auto_1 - soa_1
> - BPMJMSJDBCStore_auto_2 - soa_2
> - JDBCStore-OIM_auto_1 - oim1
> - JDBCStore-OIM_auto_2 - oim2
> - SOAJMSJDBCStore_auto_1 - soa_1
> - SOAJMSJDBCStore_auto_2 - soa_2
> - UMSJMSJDBCStore_auto_1 - soa_1
> - UMSJMSJDBCStore_auto_2 - soa_2
>
> So review the existing prefixes and select a new and unique prefix and name for each new persistent store.
>
> To avoid naming conflicts and simplify the configuration, new resources are qualified with the *scaled* tag and are shown here as an example.

Table 18-5 The New Resources Qualified with the Scaled Tag

Cluster to Scale Out	Persistent Store	Prefix Name	Data Source	Target
SOA_Cluster	UMSJMSJDBCStore_soa_scaled_3	soaums_scaled_3	WLSSchemaDataSourc	soa_server3 (migratable)
	SOAJMSJDBCStore_ soa_scaled_3	soajms_scaled_3	WLSSchemaDataSourc	soa_server3 (migratable)
	BPMJMSJDBCStore_ soa_scaled_3	soabpm_scaled_3	WLSSchemaDataSourc	soa_server3 (migratable)
OIM_Cluster	N/A	JDBCStore-OIM_scaled_3	WLSSchemaDataSource	oim_server3 (migratable)

14. Create the required JMS Servers for the new managed server.
    a. Go to **WebLogic Remote Console**. In the Edit Tree, select **Services**, and then click **JMS Servers**.
    b. Click **New**.

Use the following table to create the required JMS Servers. Assign to each JMS Server the previously created persistent stores:

Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> The number in the names of the existing resources are assigned automatically by the Configuration Wizard during domain creation.
>
> So review the existing JMS server names and select a new and unique name for each new JMS server.
>
> To avoid naming conflicts and simplify the configuration, new resources are qualified with the *product_scaled_N* tag and are shown here as an example.

Cluster to Scale Out	JMS Server Name	Persistent Store	Target
SOA_Cluster	UMSJMSServer_soa_scaled_3	UMSJMSJDBCStore_soa_scaled_3	soa_server3 (migratable)
	SOAJMSServer_soa_scaled_3	SOAJMSJDBCStore_soa_scaled_3	soa_server3 (migratable)
	BPMJMSServer_soa_scaled_3	BPMJMSJDBCStore_soa_scaled_3	soa_server3 (migratable)
	(only when you use Insight) ProcMonJMSServer_soa_scaled_3	ProcMonJMSJDBCStore_soa_scaled_3	soa_server3 (migratable)
OIM_Cluster	OIMJMSServer_scaled_3	JDBCStore-OIM_scaled_3	oim_server3 (migratable)

15. Update the SubDeployment Targets for JMS Modules (if applicable) to include the recently created JMS servers.

    a. Expand **Services** and select **JMS Modules**.

    b. Click the JMS module. For example: `BPMJMSModule`.

    Expand the **Sub Deployments** and select the corresponding one to update the targets, use the following table to identify the JMS modules to update, depending on the cluster that you are scaling out:

Table 18-6 The JMS Modules to Update

Cluster to Scale Out	JMS Module to Update	JMS Server to Add to the Subdeployment
SOA_Cluster	UMSJMSSystemResource *	UMSJMSServer_soa_scaled_3
	SOAJMSModule	SOAJMSServer_soa_scaled_3
	BPMJMSModule	BPMJMSServer_soa_scaled_3
	(Only if you have configured Insight) ProcMonJMSModule *	ProcMonJMSServer_soa_scaled_3
OIM_Cluster	OIMJMSModule	OIMJMSServer_scaled_3

(*) Some modules (UMSJMSystemResource) may be targeted to more than one cluster. Ensure that you update the appropriate subdeployment in each case.

    c. Add the corresponding JMS Server to the existing subdeployment.

Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> The subdeployment module name is a random name in the form of `SOAJMSServerXXXXXX`, `UMSJMSServerXXXXXX`, or `BPMJMSServerXXXXXX`, resulting from the Configuration Wizard JMS configuration for the first two servers (`soa_server1` and `soa_server2`).

d. Click **Save** and **Commit Changes** in the **Shopping Cart**.

16. Start the Node Manager in the Managed Server Domain Directory on OIGHOST3. Follow these steps to update and start the Node Manager from the Managed Server home

    a. Verify that the listen address in the `nodemanager.properties` file is set correctly, by completing the following steps

    - Change directory to the *MSERVER_HOME* binary directory:

      `cd $IGD_MSERVER_HOME/nodemanager`

    - Open the `nodemanager.properties` file for editing.
    - Validate the **ListenAddress** property to the correct hostname as follows:

      `oighost3: ListenAddress=oighost3`

    - Update the **ListenPort** property with the correct Listen Port details.
    - Make sure that **QuitEnabled** is set to **true**. If this line is not present in the **nodemanager.properties** file, add the following line:

      QuitEnabled=true

    b. Change directory to the *MSERVER_HOME* binary directory:

    `cd $IGD_MSERVER_HOME/bin`

    c. Use the following command to start the Node Manager:

    nohup ./startNodeManager.sh > $*MSERVER_HOME*/nodemanager/nodemanager.out 2>&1 &

17. The configuration is finished. Now sign in to the new host and run the *pack* command to create a template pack, as follows:

```
cd $ORACLE_COMMON_HOME/common/bin


./pack.sh -managed=true \
-domain=$IGD_ASERVER_HOME \
-template=/full_path/scaleout_domain.jar \
-template_name=scaleout_domain_template \
-log=/tmp/pack_scaleout.log \
-log_priority=debug
```

    In this example:

    - Replace *$IGD_ASERVER_HOME* with the actual path to the domain directory that you created on the shared storage device.
    - Replace *full_path* with the complete path to the location where you want to create the domain template jar file. You need to reference this location when you copy or unpack the domain template jar file. Oracle recommends that you choose a shared volume

Copyright © 2018, 2025, Oracle and/or its affiliates.

other than *ORACLE_HOME*, or write to `/tmp/` and copy the files manually between servers.

You must specify a full path for the template jar file as part of the `-template` argument to the `pack` command:

```
$SHARED_CONFIG_DIR/domains/template_filename.jar
```

- `scaleout_domain.jar` is a sample name for the jar file that you are creating, which contains the domain configuration files.
- `scaleout_domain_template` is the label that is assigned to the template data stored in the template file.

18. Run the `unpack` command on `OIGHOSTN` to unpack the template in the managed server domain directory, as follows:

```
cd $ORACLE_COMMON_HOME/common/bin
```

```
./unpack.sh -domain=$IGD_MSERVER_HOME \
-overwrite_domain=true \
-template=/full_path/scaleout_domain.jar \
-log_priority=DEBUG \
-log=/tmp/unpack.log \
-app_dir=$APPLICATION_HOME
```

In this example:

- Replace *$IGD_MSERVER_HOME* with the complete path to the domain home to be created on the local storage disk. This is the location where the copy of the domain is unpacked.
- Replace `/full_path/scaleout_domain.jar` with the complete path and file name of the domain template jar file that you created when you ran the `pack` command to pack up the domain on the shared storage device
- Replace *$APPLICATION_HOME* with the complete path to the Application directory for the domain on shared storage. See File System and Directory Variables Used in This Guide.

19. When scaling out the OAM_Cluster:

Register the new Managed Server with Oracle Access Management by doing the following:

a. Log in to the Access Management console using the URL outlined in URLs Used in This Chapter.
b. Go to the **System Configuration** tab.
c. Click **Server Instances**.
d. Select **Create** from the Actions menu.
e. Enter the following information:
   - **Server Name**: `oam_server3`
   - **Host**: Host that the server runs on
   - **Port**: Listen port that was assigned when the Managed Server was created

Copyright © 2018, 2025, Oracle and/or its affiliates.

- **OAM Proxy Port**: Port you want the Access Manager proxy to run on. This is unique for the host
- **Proxy Server ID**: `AccessServerConfigProxy`
- **Mode**: Set to same mode as existing Access Manager servers.

f. Go to the **Coherence** tab, and set **Local Port** to a unique value on the host.

g. Click **Apply**.

h. Restart the WebLogic Administration Server.

Add the newly created Access Manager server to all of the WebGate Profiles that might be using it, such as `Webgate_IDM`, and `IAMSuiteAgent`

For example, to add the Access Manager server to `Webgate_IDM`, access the Access Management console using the URL outlined in URLs Used in This Chapter and do the following:

a. Log in as the Access Manager Administrative User.

b. Go to the **System Configuration** tab.

c. Expand **Access Manager Settings** - **SSO Agents** - **OAM Agents**.

d. Click the open folder icon, then click **Search**. You should see the WebGate agent **Webgate_IDM**.

e. Click the agent **Webgate_IDM**.

f. Select **Edit** from the **Actions** menu.

g. Click **+** in the **Primary Server** list (or the **Secondary Server** list if this is a secondary server).

h. Select the newly created managed server from the **Server** list.

i. Set **Maximum Number of Connections** to `10`.

j. Click **Apply**.

k. Repeat the steps for all of the WebGate that are in use, and restart the new Managed Server.

20. Start Node Manager on the new host.

```
cd $NM_HOME
nohup ./startNodeManager.sh > ./nodemanager.out 2>&1 &
```

21. Start the new managed server.

22. Update the web tier configuration to include the new server. If you are using OHS, there is no need to add the new server to OHS. By default, the Dynamic Server List is used, which means that the list of servers in the cluster is automatically updated when a new node becomes part of the cluster. So, adding it to the list is not mandatory. The *WebLogicCluster* directive needs only a sufficient number of redundant `server:port` combinations to guarantee the initial contact in case of a partial outage.

    If there are expected scenarios where the Oracle HTTP Server is restarted and only the new server is up, update the `WebLogicCluster` directive to include the new server.

    For example:

```
<Location /oam>
 WLSRequest ON
 WLCookieName OAMJSESSIONID
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
 WebLogicCluster WebLogicCluster
oamhost1.example.com:14101,oamhost2.example.com:14101,oamhost2.example.com:
14101
</Location>
```

Verifying the Scale Out

After scaling out and starting the server, proceed with the following verifications:

1. Verify the correct routing to web applications.

   For example:

   a. Access the application on the load balancer:

      For SSL Terminated:

      ```
      http://igdinternal.example.com:7777/soa-infra
      ```

      For End to End SSL:

      ```
      https://igdinternal.example.com/soa-infra
      ```

   b. Check that there is activity in the new server also:

      In the **Remote Console**, go to **Monitoring Tree** and navigate to **Deployments** > **Application Runtime Data** > **soa-infra**.

   c. You can also verify that the web sessions are created in the new server:

      - In **Remote Console**, go to **Monitoring Tree** and navigate to **Deployments** > **Application Runtime Data** > **soa-infra**.
      - Go to **Component Runtimes** and click **<soa_server3_/soa-infra**.
      - Verify if there are sessions.

      You can use the sample URLs and the corresponding web applications that are identified in the following table, to check if the sessions are created in the new server for the cluster that you are scaling out:

Cluster to Verify	Sample URL to Test	Web Application Module
SOA_Cluster	`http://igdinternal.example.com:7777/soa-infra`	soa-infra > soa-infra
OAM_Cluster	`http://login.example.com/oam`	
AMA_Cluster	`http://iadadmin.example.com/access`	
OIM_Cluster	`https://oig.example.com/identity`	

2. Verify that JMS messages are being produced and consumed to the destinations, and produced and consumed from the destinations, in the three servers.

   a. In the **Remote Console**, go to the **Monitoring Tree**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

b. Navigate to **Dashboards** > **JMS Destinations.**

3. Verify the service migration, as described in Validating Automatic Service Migration.

Scaling in the Topology

This section describes how to scale in the topology for a cluster.
Perform the following steps to scale in the topology for a cluster:

1. Stop the managed server that you want to delete.
2. If you are using automatic service migration, verify that the resources corresponding to that server are not present in the remaining servers before you shrink and scale in the cluster. In case of using *exactly-once migration* policies, stop all the servers
3. Use the OAM Console to remove OAM Servers from the webgate configuration (If removing OAM Servers).
   a. Log in to the OAM console using the URL in URLs Used in This Chapter.
   b. Go to the **System Configuration** tab.
   c. Click **Server Instances**.
   d. Select the server instances you wish to remove and select **delete** from the actions menu.
   e. Click Apply.
4. Use the Oracle WebLogic Remote Console to delete the new server:
   a. Click **Edit Tree**.
   b. Go to **Environment** > **Servers**.
   c. Select the server that you want to delete.
   d. Click **Delete**.
   e. Click **Save** and **Commit changes** in the **Shopping Cart**.

> **Note**
>
> If migratable target was not deleted in the previous step, you get the following error message:
>
> ```
> The following failures occurred: --MigratableTargetMBean oam_server3_soa-
> failure-recovery (migratable) does not have a preferred server set.
> Errors must be corrected before proceeding.
> ```

5. Use the Oracle WebLogic Remote Console to update the subdeployment of each JMS Module that is used by the cluster that you are shrinking.

   Use the following table to identify the module for each cluster and perform this action for each module:

Copyright © 2018, 2025, Oracle and/or its affiliates.

Cluster to Scale in	Persistent Store	JMS Server to Delete from the Subdeployment
SOA_Cluster	UMSJMSSystemResource	UMSJMSServer_soa_scaled_3
	SOAJMSModule	SOAJMSServer_soa_scaled_3
	BPMJMSModule	BPMJMSServer_soa_scaled_3
OIM_Cluster	OIMJMSModule	OIMJMSServer_scaled_3

a. Click **Edit Tree**.
b. Go to **Services** > **JMS System Resources**.
c. Click the JMS module.
d. Click **Sub Deployment**.
e. Select the **Sub Deployment** Module
f. Unselect the JMS server that was created for the deleted server.
g. Click **Save** and **Commit changes** in the **Shopping Cart**.

6. Use the Oracle WebLogic Remote Console to delete the JMS servers:
   a. Click **Edit Tree**.
   b. Go to **Services** > **JMS Servers**.
   c. Select the JMS Servers that you created for the new server.
   d. Click **Delete**.
   e. Click **Save** and **Commit changes** in the **Shopping Cart**.
7. Use the Oracle WebLogic Remote Console to delete the JMS persistent stores:
   a. Click **Edit Tree**.
   b. Go to **Services** > **JDBC Stores**.
   c. Select the JDBC Store that you created for the new server.
   d. Click **Delete**.
   e. Click **Save** and **Commit changes** in the **Shopping Cart**.
8. If the machine that was hosting the deleted server is not used by any other servers you must delete it performing the following steps:
   a. Click **Edit Tree**.
   b. Go to **Environment** > **Machines**.
   c. Select the machine that you created for the new server.
   d. Click **Delete**.
   e. Click **Save** and **Commit changes** in the **Shopping Cart**.
9. Update the Web tier configuration to remove references to the deleted server.

Copyright © 2018, 2025, Oracle and/or its affiliates.

19

Configuring Single Sign-On for an Enterprise Deployment

You need to configure the Oracle HTTP Server WebGate in order to enable single sign-on with Oracle Access Manager.

- About Oracle Webgate
  Oracle WebGate is a web server plug-in that intercepts HTTP requests and forwards them to an existing Oracle Access Manager instance for authentication and authorization.
- General Prerequisites for Configuring Oracle HTTP Server WebGate
  Before you can configure Oracle HTTP Server WebGate, you must have installed and configured a certified version of Oracle Access Manager.
- Configuring Oracle HTTP Server WebGate for an Enterprise Deployment
  You need to configure Oracle HTTP Server WebGate for Oracle Access Manager on both WEBHOST1 and WEBHOST2.
- Adding a Load Balancer Certificate to WebGate
  Oracle WebGate 14c uses REST calls to interact with Oracle Access Manager 14c. To ensure that the communication works properly, you have to copy the load balancer certificates to WebGate Config and ensure that the REST endpoints are set correctly.
- Copying WebGates Artifacts to Web Tier
  When you created your Oracle Access Management installation, a WebGate called `Webgate_IDM` was created. In order for WebGate to communicate with the Access servers, you must copy the artifacts associated with this WebGate to the web tier.
- Restarting the Oracle HTTP Server Instance
  Restart the Oracle HTTP Server so the configuration changes take effect.
- Setting Up the WebLogic Server Authentication Providers
  To set up the WebLogic Server authentication providers, back up the configuration files, set up the Oracle Access Manager Identity Assertion Provider and set the order of providers.
- Adding the Administration Role to the New Administration Group
  The adding of administration role to a new administration group enables all users that belong to the group to be administrators for the domain.
- Configuring Oracle ADF and OPSS Security with Oracle Access Manager
  Some Oracle Fusion Middleware management consoles use Oracle Application Development Framework (Oracle ADF) security, which can integrate with Oracle Access Manager Single Sign-on (SSO). These applications can take advantage of Oracle Platform Security Services (OPSS) SSO for user authentication, but you must first configure the domain-level `jps-config.xml` file to enable these capabilities.

About Oracle Webgate

Oracle WebGate is a web server plug-in that intercepts HTTP requests and forwards them to an existing Oracle Access Manager instance for authentication and authorization.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Oracle WebGate software is installed as part of the Oracle HTTP Server software installation. See Registering and Managing OAM Agents in *Adminstrator's Guide for Oracle Access Management*. Oracle WebGate is available for Oracle HTTP Server.

General Prerequisites for Configuring Oracle HTTP Server WebGate

Before you can configure Oracle HTTP Server WebGate, you must have installed and configured a certified version of Oracle Access Manager.

For the most up-to-date information, see the certification document for your release on the *Oracle Fusion Middleware Supported System Configurations* page.

For WebGate certification matrix, click and open Certification Matrix for 14c Access Management WebGates.

> **Note**
>
> It is required that you use the WebGate version that is certified with your Oracle Access Manager deployment.

For more information about Oracle Access Manager, see the latest Oracle Identity and Access Management documentation, which you can find in the **Middleware** documentation on the Oracle Help Center.

Configuring Oracle HTTP Server WebGate for an Enterprise Deployment

You need to configure Oracle HTTP Server WebGate for Oracle Access Manager on both WEBHOST1 and WEBHOST2.

In the following procedure, replace the directory variables, such as *WEB_ORACLE_HOME* and *WEB_CONFIG_DIR*, with the values, as defined in File System and Directory Variables Used in This Guide.

1. Perform a complete backup of the web tier domain.
2. Change directory to the following location in the Oracle HTTP Server Oracle home:

   ```
   cd WEB_ORACLE_HOME/webgate/ohs/tools/deployWebGate/
   ```

   For Example:

   ```
   cd /u02/oracle/products/ohs/webgate/ohs/tools/deployWebGate
   ```

3. Run the following command to create the WebGate Instance directory and enable WebGate logging on OHS Instance:

   ```
   ./deployWebGateInstance.sh -w WEB_CONFIG_DIR -oh WEB_ORACLE_HOME
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

For example:

```
./deployWebGateInstance.sh -w /u02/oracle/config/domains/ohsDomain/config/
fmwconfig/components/OHS/ohs1 -oh /u02/oracle/products/ohs
```

4. Verify that a `webgate` directory and subdirectories was created by the `deployWebGateInstance` command:

   ```
   ls -lat WEB_ORACLE_HOME/ohs
   ```

   For Example:

   ```
   ls -lat /u02/oracle/products/ohs/webgate/ohs
   ```

   Output

   ```
   total 4
   drwxr-x---. 2 oracle oinstall 47 Dec 5 16:29 lib
   drwxr-x---. 2 oracle oinstall 103 Dec 5 16:29 config
   drwxr-x---. 8 oracle oinstall 88 Dec 5 16:29 .
   drwxr-x---. 4 oracle oinstall 40 Dec 5 16:29 tools
   drwxr-x---. 3 oracle oinstall 146 Dec 5 16:29 oamsso-bin
   drwxr-x---. 3 oracle oinstall 39 Dec 5 16:29 oamsso
   drwxr-x---. 29 oracle oinstall 4096 Dec 5 16:29 lang
   drwxr-x---. 3 oracle oinstall 17 Dec 5 16:29 ..
   ```

5. Run the following command to ensure that the `LD_LIBRARY_PATH` environment variable contains *`WEB_ORACLE_HOME`*`/lib` directory path:

   ```
   export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:WEB_ORACLE_HOME/lib
   ```

   For Example:

   ```
   export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/u02/oracle/products/ohs/lib
   ```

6. Change directory to the following directory

   `WEB_ORACLE_HOME``/webgate/ohs/tools/setup/InstallTools`

   For Example:

   ```
   cd /u02/oracle/products/ohs/webgate/ohs/tools/setup/InstallTools
   ```

7. Run the following command from the `InstallTools` directory.

   ```
   ./EditHttpConf -w $WEB_CONFIG_DIR -oh $WEB_ORACLE_HOME -o output_file_name
   ```

   For example:

   ```
   ./EditHttpConf -w /u02/oracle/config/domains/ohsDomain/config/fmwconfig/
   components/OHS/ohs1 -oh /u02/oracle/products/ohs
   ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

> **Note**
>
> The `-oh` *`WEB_ORACLE_HOME`* and `-o` *`output_file_name`* parameters are optional.

This command:

- Copies the `apache_webgate.template` file from the Oracle HTTP Server Oracle home to a new `webgate.conf` file in the Oracle HTTP Server configuration directory.
- Updates the `httpd.conf` file to add one line, so it includes the `webgate.conf`.
- Generates a WebGate configuration file. The default name of the file is `webgate.conf`, but you can use a custom name by using the `-o` *`output_file_name`* argument to the command.

Adding a Load Balancer Certificate to WebGate

Oracle WebGate 14c uses REST calls to interact with Oracle Access Manager 14c. To ensure that the communication works properly, you have to copy the load balancer certificates to WebGate Config and ensure that the REST endpoints are set correctly.

- Copying the LoadBalancer Certificates to WebGate Config
- Ensuring that the REST Endpoints are Set Correctly

Copying the LoadBalancer Certificates to WebGate Config

WebGate needs to trust your load balancer certificate. To ensure this trust, you should add the load balancer's certificate to the `cacert.pem` file, which is located in *`WEB_CONFIG_DIR`*`/webgate/config`.

You can obtain the certificate from the load balancer using a browser, such as Firefox. However, the easiest way to obtain the certificate is to use the `openssl` command. The syntax of the command is as follows:

```
openssl s_client -connect LOADBALANCER:PORT -showcerts </dev/null 2>/dev/null|
openssl x509 -outform PEM > LOADBALANCER.pem
```

For example:

```
openssl s_client -connect login.example.com:443 -showcerts </dev/null 2>/dev/
null|openssl x509 -outform PEM > login.example.com.pem
```

This command saves the certificate to a file named `login.example.com.pem`.

If you do not have load balancer certificates in your WebGate truststore, copy the `login.example.com.pem` file to *`WEB_CONFIG_DIR`*`/webgate/config renaming it to cacert.pem`.

For example:

```
cp login.example.com.pem WEB_CONFIG_DIR/webgate/config/cacert.pem
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

If you already have trusted certificates in WebGate, append the certificate to the `cacert.pem` file.

For example:

```
cp login.example.com.pem >> WEB_CONFIG_DIR/webgate/config/cacert.pem
```

Ensuring that the REST Endpoints are Set Correctly

To ensure that the REST endpoints are set correctly:

1. Log in to the OAM Console using the URL in URLs Used in This Chapter. Login using the OAM Administration user (`oamadmin`).
2. Click **Agents**.
3. Click **Search**.
4. Locate and click the name of the WebgGate agent from the search results to bring up the edit screen.
5. Expand the User Properties screen and check that the following parameters are defined correctly:
   - **OAMRestEndPointHostName** = `login.example.com`
   - **OAMRestEndPointPort** = `443`
   - **OAMServerCommunicationMode** = `HTTPS`

   These values should reflect the login point of your Oracle Access Manager installation. If unsure about these values, contact your OAM administrator.

Copying WebGates Artifacts to Web Tier

When you created your Oracle Access Management installation, a WebGate called `Webgate_IDM` was created. In order for WebGate to communicate with the Access servers, you must copy the artifacts associated with this WebGate to the web tier.

- Copying Generated Artifacts to the Oracle HTTP Server WebGate Instance Location

Copying Generated Artifacts to the Oracle HTTP Server WebGate Instance Location

After you run the idmTool configOAM, it creates a WebGate agent called Webgate_IDM. The process creates a number of artifacts relating to that agent in *`IAD_ASERVER_HOME`*`/output/Webgate_IDM`. These artifacts have to be copied to the Oracle HTTP Server configuration directory on the Web Tier hosts.

The location of the files in the Oracle HTTP Server configuration directory depends on the Oracle Access Manager security mode setting (OPEN or CERT).

The following table lists the required location of each generated artifact in the Oracle HTTP Server configuration directory.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 19-1 Web Tier Host Location to Copy the Generated Artifacts

File	Location in WEB_CONFIG_DIR
wallet/cwallet.sso[1]	*WEB_CONFIG_DIR*/webgate/config/wallet/
ObAccessClient.xml	*WEB_CONFIG_DIR*/webgate/config/
password.xml	*WEB_CONFIG_DIR*/webgate/config/
aaa_key.pem	*WEB_CONFIG_DIR*/webgate/config/
aaa_cert.pem	*WEB_CONFIG_DIR*/webgate/config/

[1] Copy `cwallet.sso` from the wallet folder and not from the output folder. Even though there are 2 files with the same name they are different. The one in the wallet sub directory is the correct one.

> **Note**
>
> If you need to redeploy the `ObAccessClient.xml` to `WEBHOST1` and `WEBHOST2`, delete the cached copy of `ObAccessClient.xml` and its lock file, `ObAccessClient.xml.lck` from the servers. The cache location on `WEBHOST1` is:
>
> *WEB_DOMAIN_HOME*/servers/ohs1/cache/
>
> And you must perform the similar step for the second Oracle HTTP Server instance on `WEBHOST2`:
>
> *WEB_DOMAIN_HOME*/servers/ohs2/cache/

Obtaining WebGate Artifacts

The easiest way to obtain the WebGate artifacts is to download them from the OAM console. To download, complete the following steps:

1. Log in to the OAM console using the URL in URLs Used in This Chapter. Login using the OAM Administration user (`oamadmin`).
2. Click **Agents**.
3. On the Search screen, click **Search**.
4. From the list of agents, select **Webgate_IDM** by clicking on its name.
5. Download the artifacts using the download button. A zip file gets downloaded on the host machine you are using.

Copy the downloaded zip file to the Oracle HTTP Server machine and unzip it to the *WEB_CONFIG_DIR*/webgate/config location. The files get extracted to the correct locations.

Restarting the Oracle HTTP Server Instance

Restart the Oracle HTTP Server so the configuration changes take effect.

For information about restarting the Oracle HTTP Server instance, see Restarting Oracle HTTP Server Instances by Using WLST in *Administering Oracle HTTP Server*.

Copyright © 2018, 2025, Oracle and/or its affiliates.

If you have configured Oracle HTTP Server in a WebLogic Server domain, you can also use Oracle Fusion Middleware Control to restart the Oracle HTTP Server instances. See Restarting Oracle HTTP Server Instances by Using Fusion Middleware Control in *Administering Oracle HTTP Server*.

Setting Up the WebLogic Server Authentication Providers

To set up the WebLogic Server authentication providers, back up the configuration files, set up the Oracle Access Manager Identity Assertion Provider and set the order of providers.

The following topics assumes that you have already configured the LDAP authenticator by following the steps in Creating a New LDAP Authenticator and Provisioning Enterprise Deployment Users and Group. If you have not already created the LDAP authenticator, then do so before you continue with this section.

> **Note**
>
> You only need to perform these steps in the **IAMGovernanceDomain**, they will already have been performed in the **IAMAccessDomain** as part of running configOAM.

- Backing Up the Configuration Files
  You should first back up the relevant configuration files to ensure their safety.
- Configuring a New Authenticator for Oracle WebLogic Server
- Setting Up the Oracle Access Manager Identity Assertion Provider
- Updating the Default Authenticator and Setting the Order of Providers

Backing Up the Configuration Files

You should first back up the relevant configuration files to ensure their safety.

Back up the following configuration files:

```
IAD_ASERVER_HOME/config/config.xml
IAD_ASERVER_HOME/config/fmwconfig/jps-config.xml
IAD_ASERVER_HOME/config/fmwconfig/system-jazn-data.xml
```

Also back up the `boot.properties` file for the Administration Server:

```
IAD_ASERVER_HOME/servers/AdminServer/security/boot.properties
```

Configuring a New Authenticator for Oracle WebLogic Server

After installing Oracle Identity Management, the Oracle WebLogic Server embedded LDAP server is the default authentication source (identity store). To use a new identity store (for example, OID or OUD), as the main authentication source, you must configure the Oracle WebLogic Server domain to use a new authentication provider to use the new store. For more information, see Security Providers in *Oracle WebLogic Remote Console Online Help*.

To configure a new authentication provider in Oracle WebLogic Server:

1. Use WebLogic Remote Console to connect to and manage the WebLogic Server domain through its Administration Server. See Connect to an Administration Server.

Copyright © 2018, 2025, Oracle and/or its affiliates.

2. In the **Edit Tree**, go to **Security**, then **Realms**, then **myrealm**, then **Authentication Providers**.
3. Click **New**.
4. In the **Name** field, enter `OID Provider`, or a name of your choosing.
5. Depending on your back-end LDAP directory, from the **Type** drop-down list, select `OracleInternetDirectoryAuthenticator` or `OracleUnifiedDirectoryAuthenticator`.
6. From the **Control Flag** drop-down list, select `SUFFICIENT`.
7. Click **Create**.
8. On the Configuration page for the new authentication provider, set the values on the **Provider-Specific Parameters** tab. Specify the settings from the tables below using appropriate values for the identity store you will be using in your environment.

Table 19-2 OID Authentication Provider Specific Values

Section Name	Field Name	Description
Connection	`Host`	The LDAP host name. For example, `idstore.example.com`.
Connection	`Port`	The LDAP host listening port number. For example, 1389 or 1636.
Connection	SSL Enabled	Set to true if you are communicating with your directory over SSL
Connection	`Principal`	The LDAP user DN used to connect to the LDAP server. For example: `cn=oimLDAP,cn=systemids,dc=example,dc=com`
Connection	`Credential`	The password for the LDAP administrative user entered as the Principal.
Users	`User Base DN`	Specifies the DN under which your users start. For example: `cn=users,dc=example,dc=com`
Users	`All Users Filter`	The LDAP search filter. For example, `(&(uid=*)(objectclass=person))`. The asterix (*) filters for all users. Click **More Info**... for details.
Users	`User From Name Filter`	The LDAP search filter. Click **More Info**... for details.
Users	`User Name Attribute`	The attribute that you want to use to authenticate (for example, `cn`, `uid`, or `mail`). Set as the default attribute for user name in the directory server. For example, `uid` .
Groups	`Group Base DN`	Specifies the DN that points to your Groups node. For example: `cn=groups,dc=example,dc=com`
General	`GUID attribute`	The attribute used to define object GUIDs in LDAP. `orclguid`

> **Note**
>
> You should not change this default value; in most cases the default value here is sufficient.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 19-3 OUD Authentication Provider Specific Values

Parameter	Sample Value	Value Description
Host	For example: `idstore.example.com`	The LDAP server's server ID.
Port	For example: 1389 or 1636	The LDAP server's port number.
Principal	For example: `cn=oimLDAP,cn=systemids,dc=example,dc=com`	Any LDAP user that has ReadOnly access to the users and groups stored in LDAP.
Credential	Enter LDAP password.	The password used to connect to the LDAP server.
SSL Enabled	Unchecked (clear)	Specifies whether SSL protocol is used when connecting to the LDAP server. If you select SSL, then you need to ensure that the certificate authority (CA) that issued the certificates for OUD are included in the WebLogic trust store.
User Base DN	For example: `cn=users,dc=example,dc=com`	Specifies the DN under which your users start.
All Users Filter	`(&(uid=*)(objectclass=person))`	Instead of a default search criteria for **All Users Filter**, search all users based on the `uid` value. If the **User Name Attribute** for the user object class in the LDAP directory structure is a type other than `uid`, then change that type in the **User From Name Filter** field. For example, if the **User Name Attribute** type is `cn`, then this field should be set to: `(&(cn=*)(objectclass=person)))`
User From Name Filter	For example: `(&(uid=%u)(objectclass=person))`	If the **User Name Attribute** for the user object class in the LDAP directory structure is a type other than `uid`, then change that type in the settings for the **User From Name** Filter. For example, if the **User Name Attribute** type is `cn`, then this field should be set to: `(&(cn=%u)(objectclass=person)))`.
User Name Attribute	For example: `uid`	The attribute of an LDAP user object that specifies the name of the user.
Use Retrieved User Name as Principal	Checked	Must be turned on.
Group Base DN	For example: `cn=groups,dc=example,dc=com`	Specify the DN that points to your Groups node.
All Groups Filter	`(&(cn=*)` `(objectclass=groupOfUniqueNames))`	Specify the group filter.
GUID Attribute	`entryuuid`	This value is prepopulated with `entryuuid` when `OracleUnifiedDirectoryAuthenticator` is used for OUD. Check this value if you are using Oracle Unified Directory as your authentication provider.
Static Member DN Attribute	`uniquemember`	

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 19-3 (Cont.) OUD Authentication Provider Specific Values

Parameter	Sample Value	Value Description
Static Group Name Attribute	`cn`	
Static Group Object Class	`groupofuniquenames`	
Static Group DNs from Member DN Filter	`(&(uniquemember=%M) (objectclass=groupofuniquenames))`	
Dynamic Group Name Attribute	`cn`	
Dynamic Group Object Class	`groupofURLs`	
Dynamic Member URL Attribute	`memberURL`	

9. Click **Save**.

For more information about configuring authentication providers in Oracle WebLogic Server, see Configuring Authentication Providers in *Administering Security for Oracle WebLogic Server*.

Perform the following steps to set up the default authenticator for use with the Identity Asserter:

1. In the **Edit Tree**, go to **Security**, then **Realms**, then **myrealm**, then **Authentication Providers**.
2. Select the default authentication provider (**DefaultAuthenticator**) to display its configuration page.
3. On the **Common** tab, from the **Control Flag** drop-down list, select `SUFFICIENT`.
4. Click **Save**.

Perform the following steps to reorder Providers:

1. In the **Edit Tree**, go to **Security**, then **Realms**, then **myrealm**, then **Authentication Providers**.
2. Select a provider name and use the arrow buttons to order the list of providers as follows:
   a. `OID Authenticator (SUFFICIENT)`
   b. `Default Authenticator (SUFFICIENT)`
3. Restart Oracle WebLogic Server.

Setting Up the Oracle Access Manager Identity Assertion Provider

Set up an Oracle Access Manager identity assertion provider in the Oracle WebLogic Server Remote Console.

To set up the Oracle Access Manager identity assertion provider:

1. Log in to the WebLogic Remote Console, if not already logged in.
2. In the Edit Tree, navigate to **Security** > **Realms** > **myrealm** > **Authentication Providers**.
3. Click **New**.
4. In the **Name** field, enter **OAM Provider** or a name of your choice.
5. From the **Type** drop-down list, select **OAMIdentityAsserter**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

6. Set the control flag to *REQUIRED*.
7. Click **Create**.

Updating the Default Authenticator and Setting the Order of Providers

Set the order of identity assertion and authentication providers in the WebLogic Remote Console.

To update the default authenticator and set the order of the providers:

1. Use WebLogic Remote Console to connect to and manage the WebLogic Server domain through its Administration Server.
2. In the **Edit** tree > navigate to **Security** > **Realms** > **myrealm** > **Authentication Providers**.
3. Select an authentication provider by selecting the check box.
4. From the table of providers, click the **DefaultAuthenticator**.
5. Set the Control Flag to `SUFFICIENT`.
6. Click **Save** to save the settings.
7. From the navigation breadcrumbs, click **Providers** to return to the list of providers.
8. Click **Reorder**.
9. Sort the providers to ensure that the OAM Identity Assertion provider is first and the DefaultAuthenticator provider is last.

Table 19-4 Sort order

Sort Order	Provider	Control Flag
1	OAMIdentityAsserter	`REQUIRED`
2	LDAP Authentication Provider	`SUFFICIENT`
3	OIMAuthenticationProvider	`SUFFICIENT`
4	DefaultIdentityAsserter	`N/A`
5	Trust Service Identity Asserter	`N/A`
6	DefaultAuthenticator	`SUFFICIENT`

10. Click **OK**.
11. Click **Activate Changes** to propagate the changes.
12. Shut down the Administration Server, Managed Servers, and any system components, as applicable.
13. Restart the Administration Server.
14. If you are going to configure ADF consoles with SSO, you can keep the managed servers down and restart them later. If not, you need to restart managed servers now.

- Setting the Default Authenticator
- Ordering the Authentication Providers

Copyright © 2018, 2025, Oracle and/or its affiliates.

Setting the Default Authenticator

In an Enterprise deployment, your users are stored in an LDAP directory. In order to be able to use the users in the LDAP directory, you must first change the Default Authenticator from REQUIRED to SUFFICIENT.

Use WebLogic Remote Console to connect to and manage the WebLogic Server domain through its Administration Server.

1. In the **Edit Tree**, navigate to **Security** > **Realms** > **myrealm** > **Authentication Providers**.
2. In the navigation tree, select the **Default Authenticator**.
3. Set the **Control Flag** to **SUFFICIENT** and click **Save**.

Ordering the Authentication Providers

The Authentication providers in the domain need to be in the required order to allow SSO integration.

Perform the following steps to set the order of the integration providers:

1. In the **Edit Tree**, navigate to **Security** > **Realms** > **myrealm** > **Authentication Providers**.
2. In the navigation tree, select the **Default Authenticator**.
3. Select an authentication provider and select the check box next to the Authentication provider name.
4. Click **Move Up** or **Move Down** to position the Authenticator in the required position.
5. Repeat until all the authenticators are in the order shown in Table 19-4.

> **Note**
>
> Ensure that the control flag of each authenticator is as appears in Table 19-4.

Adding the Administration Role to the New Administration Group

The adding of administration role to a new administration group enables all users that belong to the group to be administrators for the domain.

Perform the following steps to assign the administration role using the WebLogic Remote Console:

1. Log into the WebLogic Remote Console.
2. In the **Security Data** tree, navigate to **Realms** > **myrealm** > **Role Mappers** > **XACMLRoleMapper** > **Global** > **Roles**.
3. Select the **Admin role**.
4. Click **Add Condition**.
5. From the **Predicate List** drop-down menu, select **Group**.
6. In the **Group Argument Name** field, enter **WLSAdministrators** and click **OK**.

   The **WLSAdministrators** group is now listed.
7. Click **Save**.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Configuring Oracle ADF and OPSS Security with Oracle Access Manager

Some Oracle Fusion Middleware management consoles use Oracle Application Development Framework (Oracle ADF) security, which can integrate with Oracle Access Manager Single Sign-on (SSO). These applications can take advantage of Oracle Platform Security Services (OPSS) SSO for user authentication, but you must first configure the domain-level `jps-config.xml` file to enable these capabilities.

The domain-level `jps-config.xml` file is located in the following location after you create an Oracle Fusion Middleware domain:

```
$IAD_ASERVER_HOME/config/fmwconfig/jps-config.xml
```

> **Note**
>
> The domain-level `jps-config.xml` should not be confused with the `jps-config.xml` that is deployed with custom applications.

1. Change to the following directory:

   ```
   cd $ORACLE_COMMON_HOME/common/bin
   ```

2. If your WebLogic domain is running in secure mode set the environment variable `WLST_PROPERTIES` to include the location of your trust store. For example:

   ```
   export WLST_PROPERTIES="-Dweblogic.security.TrustKeyStore=CustomTrust -
   Dweblogic.security.CustomTrustKeyStoreFileName=/u01/oracle/config/
   keystores/idmTrustStore.p12 -
   Dweblogic.security.CustomTrustKeyStorePassPhrase=password -
   Dweblogic.security.SSL.ignoreHostnameVerification=true"
   ```

3. Start the WebLogic Server Scripting Tool (WLST):

   ```
   ./wlst.sh
   ```

   > **Note**
   >
   > If you are using an SSL enabled Admin Server, you must set the following environment variable before invoking WLST:
   >
   > ```
   > export WLST_PROPERTIES="-
   > Dweblogic.security.TrustKeyStore=CustomTrust -
   > Dweblogic.security.CustomTrustKeyStoreFileName=/u01/oracle/config/
   > keystores/idmTrustStore.p12 -
   > Dweblogic.security.CustomTrustKeyStorePassPhrase=password -
   > Dweblogic.security.SSL.ignoreHostnameVerification=true"
   > ```


Copyright © 2018, 2025, Oracle and/or its affiliates.

4. Connect to the Administration Server, by using the following WLST command:

   ```
   connect('admin_user','admin_password','admin_url')
   ```

   For example SSL Terminated

   ```
   connect('weblogic','Manager1','t3://iadadminvhn.example.com:7001')
   ```

   For example End to End SSL

   ```
   connect('weblogic','password','t3s://iadadminvhn.example.com:9002')
   ```

5. Run the `addOAMSSOProvider` command, as shown:

   ```
   addOAMSSOProvider(loginuri="/${app.context}/adfAuthentication",
   logouturi="/oam/logout.html")
   ```

   The following table defines the expected value for each argument in the `addOAMProvider` command.

   > **Note**
   >
   > Perform this action for each domain in your configuration.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Table 19-5 Expected Values for the Argument in the `addOAMProvider` command

Argument	Definition
loginuri	Specifies the URI of the login page **Note** For ADF security enabled applications, "/*context-root*/adfAuthentication" should be provided for the 'loginuri' parameter. For example: `/${app.context}/adfAuthentication` **Note** `${app.context}` must be entered as shown. At runtime, the application replaces the variable appropriately. Here is the flow: a. User accesses a resource that has been protected by authorization policies in OPSS, fox example. b. If the user is not yet authenticated, ADF redirects the user to the URI configured in *loginuri*. c. Access Manager, should have a policy to protect the value in *loginuri*: for example, "/*context-root*/adfAuthentication". d. When ADF redirects to this URI, Access Manager displays a Login Page (depending on the authentication scheme configured in Access Manager for this URI).
logouturi	Specifies the URI of the logout page. The value of the *loginurl* is usually `/oam/logout.html`.
autologinuri	Specifies the URI of the autologin page. This is an optional parameter.

6. Disconnect from the Administration Server by entering the following command:

   ```
   disconnect()
   ```

7. Restart the Administration Server and the managed servers.

Copyright © 2018, 2025, Oracle and/or its affiliates.

20

Troubleshooting

You can troubleshoot the common issues that may arise with the Identity and Access Management enterprise deployment. The solutions provided for the common problems help you resolve them quickly.

This chapter includes the following topics:

- Troubleshooting Oracle Access Management Access Manager
  Learn about some of the common problems that you may encounter with Oracle Access Manager and the actions you can take to resolve them.
- Troubleshooting Oracle Identity Governance
  Learn about some of the common problems that may arise with Oracle Identity Governance and the actions you can take to resolve the problem.
- Troubleshooting Oracle SOA Suite
  Learn about the transaction timeout error that may arise with Oracle SOA Suite and the action you can take to resolve the problem.
- Troubleshooting Integration OIGOAMIntegration.sh-configureLDAPConnector
  Learn about the error you may encounter during the integration process and the solution to fix this error.
- General Troubleshooting
  Learn about the error you may encounter when starting the Managed Server from the WebLogic Console and the resolution to fix the error.

Troubleshooting Oracle Access Management Access Manager

Learn about some of the common problems that you may encounter with Oracle Access Manager and the actions you can take to resolve them.

- Access Manager Runs out of Memory
- User Reaches the Maximum Allowed Number of Sessions
- Policies Do Not Get Created When Oracle Access Management Access Manager is First Installed
- You Are Not Prompted for Credentials After Accessing a Protected Resource
- Cannot Log In to Access Management Console
- Oracle Coherence Cluster Startup Errors in oam_policy_mgr Server Logs
- Errors in log File when Starting OAM Servers
- Too Many Redirects Error in Browser

Access Manager Runs out of Memory

Problem

After Access Manager has been running for a while, you see the following error message in the output:

Copyright © 2018, 2025, Oracle and/or its affiliates.

```
Attempting to allocate 1G bytes
There is insufficient native memory for the Java Runtime Environment to continue.
```

Possible reasons

- The system is out of physical RAM or swap space.
- In 32 bit mode, the process size limit was reached.

Solutions

- Reduce memory load on the system.
- Increase physical memory or swap space.
- Check if swap backing store is full.
- Use 64 bit Java on a 64 bit OS.
- Decrease Java heap size (-Xmx/-Xms).
- Decrease number of Java threads.
- Decrease Java thread stack sizes (-Xss).
- Disable compressed references (-XXcompressedRefs=false).
- Ensure that command line tool `adrci` can be executed from the command line.
  - at oracle.dfw.impl.incident.ADRHelper.invoke(ADRHelper.java:1309)
  - at oracle.dfw.impl.incident.ADRHelper.createIncident(ADRHelper.java:929
  - at oracle.dfw.impl.incident.DiagnosticsDataExtractorImpl.createADRIncident(Diagnostics DataExtractorImpl.java:1116)
- On both OAMHOST1 and OAMHOST2, edit the file `setSOADomainEnv.sh`, which is located in *`IAD_MSERVER_HOME`*`/bin` and locate the line which begins:

  ```
  PORT_MEM_ARGS=
  ```

  Change this line so that it reads:

  ```
  PORT_MEM_ARGS="-Xms768m -Xmx2560m"
  ```

User Reaches the Maximum Allowed Number of Sessions

Problem

The Access Manager server displays an error message similar to this:

```
The user has already reached the maximum allowed number of sessions. Please
close one of the existing sessions before trying to login again.
```

Solution

If users log in multiple times without logging out, they might overshoot the maximum number of configured sessions. You can modify the maximum number of configured sessions by using the Access Management Administration Console.

To modify the configuration by using the Access Management Administration Console, proceed as follows:

1. Go to **System Configuration** -> **Common Settings** -> **Session**

Copyright © 2018, 2025, Oracle and/or its affiliates.

2. Increase the value in the **Maximum Number of Sessions per User** field to cover all concurrent login sessions expected for any user. The range of values for this field is from 1 to any number.

Policies Do Not Get Created When Oracle Access Management Access Manager is First Installed

Problem

The Administration Server takes a long time to start after configuring Access Manager.

Solution

Tune the Access Manager database. When the Administration Server first starts after configuring Access Manager, it creates a number of default policies in the database. If the database is distant or in need of tuning, this can take a significant amount of time.

```
Resources
Authentication Policies
   Protected Higher Level Policy
   Protected Lower Level Policy
   Publicl Policy
Authorization Policies
   Authorization Policies
```

If you do not see these items, the initial population has failed. Check the Administration Server log file for details.

You Are Not Prompted for Credentials After Accessing a Protected Resource

Problem

When you access a protected resource, Access Manager should prompt you for your user name and password. For example, after creating a simple HTML page and adding it as a resource, you should see credential entry screen.

Solution

If you do not see the Credential Entry screen, perform the following steps:

1. Verify that host aliases for IAMAccessDomain have been set. You should have aliases for `IAMAccessDomain`:80, `IAMAccessDomain`:Null, `IADADMIN.example.com:80`, and `login.example.com:443`, where Port `80` is *`HTTP_PORT`* and Port `443` is *`HTTP_SSL_PORT`*.
2. Verify that WebGate is installed.
3. Verify that `ObAccessClient.xml` was copied from *`IAD_ASERVER_HOME`*`/output` to the WebGate Lib directory and that OHS was restarted.
4. When you first created the `ObAccessClient.xml` file, it was not formatted. When you restart OHS, re-examine the file to ensure that it is formatted. OHS gets a new version of the file from Access Manager when it first starts.
5. Shut down the Access Manager servers and access the protected resource. If you do not see an error saying Access Manager servers are not available, re-install WebGate.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Cannot Log In to Access Management Console

Problem

You cannot log in to the Access Management Console. The Administration Server diagnostic log might contain an error message similar to this:

```
Caused by: oracle.security.idm.OperationFailureException:
oracle.security.am.common.jndi.ldap.PoolingException [Root exception is
oracle.ucp.UniversalConnectionPoolException:
Invalid life cycle state.
 Check the status of the Universal Connection Pool]
         at
oracle.security.idm.providers.stdldap.UCPool.acquireConnection(UCPool.java:112)
```

Solution

Remove the `/tmp/UCP*` files and restart the Administration Server.

Oracle Coherence Cluster Startup Errors in oam_policy_mgr Server Logs

Problem

The oam_policy_mgr2 server has oam application deployment in failed state. The oam_policy_mgr2 server logs report request timeout exceptions while starting the cluster service, similar to following logs:

```
Oracle Coherence GE 3.7.1.13 <Warning> (thread=Cluster, member=n/a): Delaying
formation of a new cluster; IpMonitor failed to verify the reachability of senior
Member(Id=1, Timestamp=, Address=, MachineId=,
Location=site:,machine:IADADMINVHN,process:8499, Role=WeblogicServer); if this
persists it is likely the result of a local or remote firewall rule blocking
either ICMP pings, or connections to TCP port 7>

Error while starting cluster: com.tangosol.net.RequestTimeoutException: Timeout
during service start: ServiceInfo(Id=0, Name=Cluster, Type=Cluster
MemberSet=MasterMemberSet(
ThisMember=null
OldestMember=null
ActualMemberSet=MemberSet(Size=0
)
MemberId|ServiceVersion|ServiceJoined|MemberState
RecycleMillis=1200000
RecycleSet=MemberSet(Size=0
)
)
)
at
com.tangosol.coherence.component.util.daemon.queueProcessor.service.Grid.onStartupTimeou
t(Grid.CDB:3)

at
com.tangosol.coherence.component.util.daemon.queueProcessor.Service.start(Service.CDB:28)

at
com.tangosol.coherence.component.util.daemon.queueProcessor.service.Grid.start(Grid.CDB:6
)
```

Solution

Copyright © 2018, 2025, Oracle and/or its affiliates.

This is a known issue. In some of the environments, the Access Policy Manager Server that is not running on the same host as the WebLogic Administration Server is unable to start the coherence cluster service, which results in the oam application deployment to be in failed state. To solve this issue, you must create a server instance for the effected Access Policy Manager Server by completing the following steps:

1. Log in to the OAM console using the URL in URLs Used in This Chapter. Login using the OAM Administration user (`oamadmin`).
2. Click **Configuration**.
3. Click **Server Instances** from the configuration launch pad.
4. Click a new server instance for the Access Policy Manager WebLogic Managed Server, that is not running on the same machine as the IAMAccessDomain Admin Server. For example:
   - Name: oam_policy_mgr2
   - Port: 14150
   - Host: OAMHOST2 (For consolidated topology, the host will be IAMHOST2)

   > **Note**
   >
   > Provide the OAM Proxy details similar to the server instance for oam_server.

5. Click **Apply**.

Errors in log File when Starting OAM Servers

Problem

When you start the OAM Servers, errors similar to the following are seen in the log files which causes LCM heath check module to fail:

```
[oam_server1] [TRACE:16] [] [oracle.oam.config] [tid: DistributedCacheWorker:4] [userId:
<anonymous>] [ecid:
0000LGmRJqxB9DE5N7P5ie1N5mOd000004,1:16514] [APP: oam_server#11.1.2.0.0] [SRC_CLASS:
oracle.security.am.admin.config.util.MapUtil] [SRC_METHOD:
getDefaultedStringValue] property not found at path:[Ljava.lang.String;@43537067
Defaulting to value:,
[2016-04-20T06:55:39.982+00:00] [oam_server1] [TRACE:16] [] [oracle.oam.config] [tid:
DistributedCacheWorker:4] [userId: <anonymous>] [ecid:
0000LGmRJqxB9DE5N7P5ie1N5mOd000004,1:16514] [APP: oam_server#11.1.2.0.0] [SRC_CLASS:
oracle.security.am.admin.config.util.MapUtil] [SRC_METHOD: getStringValue] THROW[[
oracle.security.am.admin.config.ConfigurationException: Cannot get java.lang.String
value from configuration for key ResponseEscapeChar. Object null found.
at
oracle.security.am.admin.config.util.MapUtil.handleFailedAttributeAccess(MapUtil.java:447
)
at oracle.security.am.admin.config.util.MapUtil.getStringValue(MapUtil.java:130)
at oracle.security.am.admin.config.util.MapUtil.getDefaultedStringValue(MapUtil.java:147)
at
oracle.security.am.engines.common.identity.provider.util.IdStoreConfig.initializeConfig(I
dStoreConfig.java:76)
at
oracle.security.am.engines.common.identity.provider.util.IdStoreConfig.<init>(IdStoreConf
ig.java:69)
at
oracle.security.am.engines.common.identity.provider.util.IdStoreConfig.getConfig(IdStoreC
```


Copyright © 2018, 2025, Oracle and/or its affiliates.

```
onfig.java:128)
at
oracle.security.am.engines.common.identity.util.OAMUserAttribute.getStringValue(OAMUserAt
tribute.java:76)
at
oracle.security.am.engines.common.identity.util.OAMUserAttribute.toString(OAMUserAttribut
e.java:114)
at java.lang.String.valueOf(String.java:2849)
at java.lang.StringBuilder.append(StringBuilder.java:128)
at java.util.AbstractMap.toString(AbstractMap.java:523)
at java.lang.String.valueOf(String.java:2849)
at java.lang.StringBuilder.append(StringBuilder.java:128)
at
oracle.security.am.engines.common.identity.util.OAMIdentity.toString(OAMIdentity.java:678
)
at java.lang.String.valueOf(String.java:2849)
at java.lang.StringBuilder.append(StringBuilder.java:128)
at oracle.security.am.engines.sso.SSOSubject.toString(SSOSubject.java:238)
at java.lang.String.valueOf(String.java:2849)
at java.lang.StringBuilder.append(StringBuilder.java:128)
at oracle.security.am.engines.sme.impl.SessionImpl.toString(SessionImpl.java:629)
at java.lang.String.valueOf(String.java:2849)
at java.lang.StringBuilder.append(StringBuilder.java:128)
at
oracle.security.am.engines.sme.mapimpl.db.DbOraSmeStore.loadSession(DbOraSmeStore.java:17
05)
at
oracle.security.am.engines.sme.mapimpl.db.DbOraSmeStore.loadSession(DbOraSmeStore.java:16
91)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:606)
at
oracle.security.am.foundation.mapimpl.coherence.store.DataConnectionUtility.invokeSqlOper
ationWithRetries(DataConnectionUtility.java:275)
at oracle.security.am.engines.sme.mapimpl.db.DbOraSmeStore.load(DbOraSmeStore.java:1284)
at
com.tangosol.net.cache.ReadWriteBackingMap$CacheStoreWrapper.loadInternal(ReadWriteBackin
gMap.java:5676)
at
com.tangosol.net.cache.ReadWriteBackingMap$StoreWrapper.load(ReadWriteBackingMap.java:475
4)
at com.tangosol.net.cache.ReadWriteBackingMap.get(ReadWriteBackingMap.java:717)
at
com.tangosol.coherence.component.util.daemon.queueProcessor.service.grid.partitionedServi
ce.PartitionedCache$Storage.get(PartitionedCache.CDB:10)
at
com.tangosol.coherence.component.util.daemon.queueProcessor.service.grid.partitionedServi
ce.PartitionedCache.onGetRequest(PartitionedCache.CDB:23)
at
com.tangosol.coherence.component.util.daemon.queueProcessor.service.grid.partitionedServi
ce.PartitionedCache$GetRequest.run(PartitionedCache.CDB:1)
at com.tangosol.coherence.component.util.DaemonPool$WrapperTask.run(DaemonPool.CDB:1)
at com.tangosol.coherence.component.util.DaemonPool$WrapperTask.run(DaemonPool.CDB:32)
at com.tangosol.coherence.component.util.DaemonPool$Daemon.onNotify(DaemonPool.CDB:66)
at com.tangosol.coherence.component.util.Daemon.run(Daemon.CDB:42)
at java.lang.Thread.run(Thread.java:745)
]]
```

Solution

Copyright © 2018, 2025, Oracle and/or its affiliates.

This occurs when OAM servers cannot communicate with each other using the coherence port. This is often caused by iptables. The workaround for this issue is as follows:

1. Edit the file `/etc/sysconfig/iptables` on both OAMHOST1 and OAMHOST2 and add the following line:

```
# Generated by iptables-save v1.4.7 on Tue Apr 19 10:02:45 2016
*filter
:INPUT ACCEPT [593:243587]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [614:423013]
-A INPUT -p tcp -m state --state NEW -m tcp --dport 9095 -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m tcp --dport 9097 -j ACCEPT
COMMIT
```

   In the above set of lines, `9095` and `9097` are the coherence ports being used.

2. Save the file and restart the servers.

Too Many Redirects Error in Browser

Problem

When navigating from one application to another that uses the same OAM for SSO, you get a redirection error in the web browser. There are two different configurations to validate.

Solution 1:

1. Log in to the OAM Console at `iadadmin.example.com/oamconsole`.
2. From the Launch Pad, click the **Agents** icon.
3. In the resulting window > **Webgates** tab, click **search**. No search parameters need to be input.
4. In the search results, click the IAMSuiteAgent link.
5. Ensure that the Primary Cookie Domain is set to the domain that is used for the `login.example.com` domain. For example: example.com.
6. Restart all WebGate OHS instances.

Solution 2:

Ensure that the date and time on all OHS and OAM servers are within 60 seconds of each other. If they are not:

1. Ensure that the NTP setting are the same and valid on all OHS and OAM hosts.
2. Start or restart the *ntpd* service on all hosts.
3. Restart all WebGate OHS instances, the OAM domain AdminServer, and all Managed Servers.

Troubleshooting Oracle Identity Governance

Learn about some of the common problems that may arise with Oracle Identity Governance and the actions you can take to resolve the problem.

- OIM Bootstrap Process Fails
- java.io.FileNotFoundException When Running Oracle Identity Governance Configuration
- ResourceConnectionValidationxception When Creating User in Oracle Identity Governance

Copyright © 2018, 2025, Oracle and/or its affiliates.

- Oracle Identity Manager Reconciliation Jobs Fail
- OIM Reconciliation Jobs Fail When Running Against Oracle Unified Directory
- Cannot Open Reports from OIM Self Service Console
- Pending Violations Not Displaying the Correct List

OIM Bootstrap Process Fails

Problem

The OIM Bootstrap process fails after deploying composites. The error appears as follows:

```
Deployment of SOA Composites :-/<INSTALL_LOCATION>/Oracle_Home/idm/server/workflows/
composites/scajars/sca_DefaultRequestApproval_rev6.0.jar is successful>
<Jun 12, 2018 4:20:26,136 PM CEST> <Info> <oracle.iam.OIMPostConfigManager> <BEA-000000>
<updating feature:DEPLOYSOACOMPOSITESwith state :COMPLETEwith executionTime190108>
java.sql.SQLException: Connection closed
```

This is caused by a performance issue.

Solution

To resolve the issue temporarily, increase the inactivity timeouts on the following data sources:

- `oimJMSStoreDS`
- `oimOperationsDB`

The settings can be restored to their original values after the upgrade is complete.

1. Log in to the WebLogic Remote Console.
2. Click **Edit Tree**.
3. Click **Services**, then **Data Sources**, and then select the <*Data source name*>.
4. Click the **Connection Pool** tab.
5. Click the **Advanced** tab, increase the value of **Inactive Connection Timeout**.
6. Click **Save**.
7. Click the **Shopping Cart** and **Commit Changes**.
8. Restart the OIM Managed Server.

java.io.FileNotFoundException When Running Oracle Identity Governance Configuration

Problem

The following content was added to address bug 12390838

When you run Oracle Identity Manager configuration, the error `java.io.FileNotFoundException: soaconfigplan.xml (Permission denied)` may appear and Oracle Identity Manager configuration might fail.

Solution

To workaround this issue:

1. Delete the file `/tmp/soaconfigplan.xml`.

Copyright © 2018, 2025, Oracle and/or its affiliates.

2. Start the configuration again (*IGD_ORACLE_HOME*/bin/config.sh).

ResourceConnectionValidationxception When Creating User in Oracle Identity Governance

Problem

The following content was added to address bug 9816870

If you are creating a user in Oracle Identity Manager (by logging into Oracle Identity Manager System Administration Console, clicking the Administration tab, clicking the **Create User** link, entering the required information in the fields, and clicking **Save**) in an active-active Oracle Identity Manager configuration, and the Oracle Identity Manager server that is handling the request fails, you may see a "ResourceConnectionValidationxception" in the Oracle Identity Manager log file, similar to:

```
[2010-06-14T15:14:48.738-07:00] [oim_server2] [ERROR] [] [XELLERATE.SERVER]
[tid: [ACTIVE].ExecuteThread: '0' for queue: 'weblogic.kernel.Default
(self-tuning)'] [userId: xelsysadm] [ecid:
004YGJGmYrtEkJV6u3M6UH00073A0005EI,0:1] [APP: oim#11.1.1.3.0] [dcid:
12eb0f9c6e8796f4:-785b18b3:12938857792:-7ffd-0000000000000037] [URI:
/admin/faces/pages/Admin.jspx] Class/Method:
PooledResourceConnection/heartbeat encounter some problems: Operation timed
out[[
com.oracle.oim.gcp.exceptions.ResourceConnectionValidationxception: Operation
timed out
        at
oracle.iam.ldapsync.impl.repository.LDAPConnection.heartbeat(LDAPConnection.ja
va:162)
        at
com.oracle.oim.gcp.ucp.PooledResourceConnection.heartbeat(PooledResourceConnec
tion.java:52)
         .
         .
         .
```

Solution

Despite this exception, the user is created correctly.

Oracle Identity Manager Reconciliation Jobs Fail

Problem

Oracle Identity Manager reconciliation jobs fail, or one of the following messages is seen in the log files:

- Error-1

  ```
  LDAP Error 53 : [LDAP: error code 53 - Full resync required. Reason: The provided
  cookie is older than the start of historical in the server for the replicated
  domain : dc=example,dc=com]
  ```

- Error-2

  ```
  LDAP: error code 53 - Invalid syntax of the provided cookie
  ```

This error is caused by the data in the Oracle Unified Directory change log cookie expiring because Oracle Unified Directory has not been written to for a certain amount of time.

Solution

Copyright © 2018, 2025, Oracle and/or its affiliates.

1. Open a browser and access the OIG System Administration console using the URL in URLs Used in This Chapter.
2. Log in a as `xelsysadm` using the *`COMMON_IDM_PASSWORD`*.
3. Under **System Management**, click **Scheduler**.
4. Under **Search Scheduled Jobs**, enter `LDAP *` (there is a space before *) and hit **Enter**.
5. For each job in the search results, click on the job name on the left, then click **Disable** on the right.

   Do this for all jobs. If the job is already disabled do nothing.
6. Run the following commands on LDAPHOST1:

   ```
   cd OUD_ORACLE_INSTANCE/OUD/bin
   ./ldapsearch -h LDAPHOST1 -p 1389 -D "cn=oudadmin" -b "" -s base "objectclass=*"
   lastExternalChangelogCookie

   Password for user 'cn=oudadmin': <OudAdminPwd>
   dn: lastExternalChangelogCookie: dc=example,dc=com:00000140c682473c263600000862;
   ```

   Copy the output string that follows `lastExternalChangelogCookie:`. This value is required in the next step. For example,

   ```
   dc=example,dc=com:00000140c682473c263600000862;
   ```

   The Hex portion must be 28 characters long. If this value has more than one Hex portion then separate the 28char portions with spaces. For example:

   ```
   dc=example,dc=com:00000140c4ceb0c07a8d00000043 00000140c52bd0b9104200000042
   00000140c52bd0ba17b9000002ac 00000140c3b290b076040000012c;
   ```
7. Run each of the following LDAP reconciliation jobs once to reset the last change number.:
   - LDAP Role Delete Reconciliation
   - LDAP User Delete Reconciliation
   - LDAP Role Create and Update Reconciliation
   - LDAP User Create and Update Reconciliation
   - LDAP Role Hierarchy Reconciliation
   - LDAP Role Membership Reconciliation

   To run the jobs:

   a. Login to the OIM System Administration Console as the user `xelsysadm`.
   b. Under **System Configuration**, click **Scheduler**.
   c. Under **Search Scheduled Jobs**, enter `LDAP *` (there is a space before *) and hit **Enter**.
   d. Click on the job to be run.
   e. Set the parameter **Last Change Number** to the value obtained in step 6.

      For example:

      ```
      dc=example,dc=com:00000140c4ceb0c07a8d00000043 00000140c52bd0b9104200000042
      00000140c52bd0ba17b9000002ac 00000140c3b290b076040000012c;
      ```
   f. Click **Run Now**.
   g. Repeat for each of the jobs in the list at the beginning of this step.

Copyright © 2018, 2025, Oracle and/or its affiliates.

8. For each incremental recon job whose last changelog number has been reset, execute the job and check that the job now completes successfully.
9. After the job runs successfully, re-enable periodic running of the jobs according to your requirements.

If the error appears again after the incremental jobs have been re-enabled and run successfully ("Full resync required. Reason: The provided cookie is older..."), then increase the OUD cookie retention time. Although there is no hard and fast rule as to what this value should be, it should be long enough to avoid the issue, but small enough to avoid unnecessary resource consumption on OUD. One or two weeks should suffice. Run the following command on each OUD instance to increase the retention time to two weeks:

```
cd OUD_ORACLE_INSTANCE/bin

./dsconfig set-replication-server-prop --provider-name "Multimaster Synchronization" --
set replication-purge-delay:2w -D cn=oudadmin --trustAll -p 4444 -h LDAPHOSTn

Password for user 'cn=oudadmin':  <OudAdminPswd>
Enter choice [f]: f
```

OIM Reconciliation Jobs Fail When Running Against Oracle Unified Directory

Problem

Reconciliation jobs fail when running against Oracle Unified Directory (OUD). The following error is seen in the OIM WebLogic Server logs:

```
LDAP: error code 53 - Invalid syntax of the provided cookie
```

Solution

Perform the workaround described in Oracle Identity Manager Reconciliation Jobs Fail. If this workaround does not resolve the issue, try the following solution:

On each OIGHOST, update the *IGD_MSERVER_HOME*/config/fmwconfig/ovd/oim/adapters.os_xml file with the following parameter:

```
<param name="eclCookie" value="false"/>
```

Restart the OIM and SOA Managed Servers.

Cannot Open Reports from OIM Self Service Console

Problem

The reports cannot be opened from OIM Self Service Console.

Solution

When you enable the Identity Auditor feature in OIM, do the following configuration changes for the OIM-BI Publisher integration to work fine:

1. Log in to the IAMGovernanceDomain Enterprise Management Console.
2. Open the system MBean browser and update the MBean "oracle.iam:Location=oim_server1,name=Discovery,type=XMLConfig.DiscoveryConfig,XMLConfig=Config,Application=oim,ApplicationVersion=11.1.2.0.0" with Value as http://*igdadmin.example.com*/.

Copyright © 2018, 2025, Oracle and/or its affiliates.

Here, *igdadmin.example.com* is the Governance Domain admin Load balancer URL.

Pending Violations Not Displaying the Correct List

Problem

When viewing the pending violations list, you may see entries that are missing or entries that do not belong to the list.

Solution

If you encounter this issue, a restart of the OIG domain usually resolves it. If the issue is not resolved, raise a Service Request (SR) with Oracle Support.

Troubleshooting Oracle SOA Suite

Learn about the transaction timeout error that may arise with Oracle SOA Suite and the action you can take to resolve the problem.

- Transaction Timeout Error

Transaction Timeout Error

Problem

The following transaction timeout error appears in the log:

```
Internal Exception: java.sql.SQLException: Unexpected exception while enlisting
 XAConnection java.sql.SQLException: XA error: XAResource.XAER_NOTA start()
failed on resource 'SOADataSource_soaedg_domain': XAER_NOTA : The XID
is not valid
```

Solution

Check your transaction timeout settings, and be sure that the JTA transaction time out is less than the DataSource XA Transaction Timeout, which is less than the `distributed_lock_timeout` (at the database).

With the out of the box configuration, the SOA data sources do not set XA timeout to any value. The `Set XA Transaction Timeout` configuration parameter is unchecked in the WebLogic Remote Console. In this case, the data sources use the domain level JTA timeout which is set to `30`. Also, the default `distributed_lock_timeout` value for the database is `60`. As a result, the SOA configuration works correctly for any system where transactions are expected to have lower life expectancy than such values. Adjust these values according to the transaction times your specific operations are expected to take.

Troubleshooting Integration OIGOAMIntegration.sh-configureLDAPConnector

Learn about the error you may encounter during the integration process and the solution to fix this error.

Problem

The following content was added to address bug 27567130

Copyright © 2018, 2025, Oracle and/or its affiliates.

Whilst running configureLDAPConnector, you see the following error message:

```
2018-02-19 06:54:05] LDAPConnectorConfigTool.configureLDAPConnector:
exception: java.lang.reflect.UndeclaredThrowableException  [2018-02-19
06:54:05] javax.management.InstanceNotFoundException: Unable to  contact
MBeanServer for
oracle.iam:Location=oim_server1,name=SSOIntegrationMXBean,type=IAMAppRuntimeMB
  ean,Application=oim  at weblogic.utils.StackTraceDisabled.unknownMethod()
```

Solution

This is caused by the OIM Managed Server being called something other than `oim_server1`. This can be recovered by executing the following workaround.

Ensure that your OIM Managed Server is running.

1. Log in to Oracle Fusion Middleware control using the URL in URLs Used in This Chapter.
2. Start the System Mbean Browser by selecting Weblogic Domain and then clicking on System MBean browser.
3. Click on find and enter the Mbean name **SSOIntegrationMXBean** .
4. Click **Search**.
5. When the MBean is found, click **Operations > addContainerRules** .
6. Enter the following information:

   ```
   Oracle_Home set to the value of IGD_ORACLE_HOME dirType. set to OUD
   userContainer set to
   cn=users,
   dc=example,
   dc=com
   roleContatiner set to cn=groups,
   dc=example,dc=com
   ```

7. Click **Invoke** button.

General Troubleshooting

Learn about the error you may encounter when starting the Managed Server from the WebLogic Console and the resolution to fix the error.

- Cannot Start Managed Server from WebLogic Console

Cannot Start Managed Server from WebLogic Console

Problem

When you start a Managed Server from the WebLogic Console, the following error is shown:

```
. For server soa_server1, the Node Manager associated with machine OIGHOST1 is not
reachable.
. All of the servers selected are currently in a state which is incompatible with this
operation or are not associated with a running Node Manager or you are not authorized to
perform the action requested. No action will be performed.
```

Solution 1

Copyright © 2018, 2025, Oracle and/or its affiliates.

Check if the Node Manager is started on the target host. If not, start it.

Solution 2

Verify that the domain is listed in the file `nodemanager.domains`, which is located in the directory *`LOCAL_CONFIG_DIR`*`/nodemanger/hostname`. If not, do the following:

1. Start the WebLogic Scripting Tool (WLST) by running the following command from the location `$ORACLE_HOME/oracle_common/common/bin/`:

   ```
   ./wlst.sh
   ```

2. Connect to the domain you wish to add by running the following command:

   ```
   connect('weblogic_user','password','t3://ADMINVHN:AdminPort')
   ```

   > **Note**
   >
   > If using SSL use `'t3s://ADMINVHN:AdminPort`

   In this command:

   `weblogic_user` is the WebLogic Administration user. For example, `weblogic` or `weblogic_idm`.

   `password` is the password of the WebLogic Administration user.

   `ADMINVHN` is the Virtual host name of the Administration Server. For example, `IGDADMINVHN` or `IADADMINVHN`.

   `adminPort` is the port on which the Administration Server is running. For example, `7101`.

   Sample Command:

   `connect('weblogic_idm','<password>','t3://igdadminvhn.example.com:7101')` or is using SSL:`connect('weblogic_idm','<password>','t3s://igdadminvhn.example.com:9102')`

3. Enrol the domain using the following command:

   `nmEnroll(domainDir=`*`absolute_path_to_the_domain`*`,nm_Home=`*`absolute_path_to_the_nodemanager_home`*`)`

   For example:

   ```
   nmEnroll(domainDir='/u01/oracle/config/domains/
   IAMGovernanceDomain/',nmHome='/u02/oracle/config/nodemanger/hostname)')
   ```

   > **Note**
   >
   > For Managed Servers, the domain home should always be specified as the local Managed Server directory.

Copyright © 2018, 2025, Oracle and/or its affiliates.