Oracle® Fusion Middleware

Enterprise Deployment Guide for Oracle Identity Management

11*g* Release 2 (11.1.2.1)

E40782-06

February 2014

Documentation for system administrators that describes how to install and configure Oracle Identity Management components in an enterprise deployment for Oracle Fusion Middleware.

ORACLE

Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management 11*g* Release 2 (11.1.2.1)

E40782-06

Primary Authors: Ellen Desmond (Writer), Janga Aliminati (Architect), Michael Rhys (Contributing Engineer)

Contributors: Christelle Balon, Pradeep Bhat, Bruce Jiang, Louise Luo, Xiao Lin

Contents

3 Preparing the Network for an Enterprise Deployment

4 Preparing Storage for an Enterprise Deployment

5 Preparing the Servers for an Enterprise Deployment

6 Preparing the Database for an Enterprise Deployment

7 Installing and Configuring Oracle Unified Directory

8 Creating a Domain for an Enterprise Deployment

9 Preparing Identity Stores

10 Installing and Configuring Oracle Web Tier for an Enterprise Deployment

11 Extending the Domain to Include Oracle Access Management

12 Extending the Domain to Include Oracle Identity Manager

13 Setting Up Node Manager for an Enterprise Deployment

14 Configuring Server Migration for an Enterprise Deployment

17 Managing the Topology for an Enterprise Deployment

Index

List of Figures

List of Tables

Preface

This preface describes the audience, contents and conventions used in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*.

Audience

This guide is intended for system administrators who are responsible for installing and configuring Oracle Identity Management enterprise deployments.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at
http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit
http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit
http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

The following manuals in the Oracle Fusion Middleware documentation library provide additional information on the process of installing and configuring the Enterprise Deployment architecture:

- *Oracle Fusion Middleware Installation Guide for Oracle Identity Management*
- *Oracle Fusion Middleware Administrator's Guide*
- *Oracle Fusion Middleware High Availability Guide*
- *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*
- *Oracle Fusion Middleware Application Security Guide*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Unified Directory*
- *Oracle Fusion Middleware Repository Creation Utility User's Guide*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Virtual Directory*

- *Oracle Fusion Middleware Administrator's Guide for Oracle Directory Integration Platform*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Manager*
- *Oracle Fusion Middleware User's Guide for Oracle Identity Manager*
- *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide for Linux*
- *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide for Solaris Operating System*
- *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide for HP-UX*
- *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide for hp Tru64 UNIX*
- *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide for AIX Based Systems*
- *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide for Microsoft Windows*
- *Oracle Database Backup and Recovery User's Guide*

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.

What's New in This Guide

The following topics introduce the new and changed features of Oracle Identity and Access management and other significant changes that are described in this guide, and provides pointers to additional information.

New and Changed Features for 11*g* Release 2 (11.1.2.1)

Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management 11*g* Release 2 (11.1.2.1) is similar to the 11*g* Release 2 (11.1.2.0) version, but has been tested using 11*g* Release 2 (11.1.2.1) components.

1 Enterprise Deployment Overview

This chapter provides an overview of the enterprise topology for Oracle Identity Management.

This chapter contains the following sections:

Oracle Identity Management presents a comprehensive suite of products for all aspects of identity management.This guide describes reference enterprise topologies for the Oracle Identity Management Infrastructure components of Oracle Fusion Middleware. It also provides detailed instructions and recommendations to create the topologies by following the enterprise deployment guidelines.

1.1 About the Enterprise Deployment Guide

An enterprise deployment is an Oracle best practices blueprint based on proven Oracle high-availability technologies and recommendations for Oracle Fusion Middleware. The high-availability best practices described in this book make up one of several components of high-availability best practices for all Oracle products across the entire technology stack—Oracle Database, Oracle Fusion Middleware, Oracle Applications, Oracle Collaboration Suite, and Oracle Grid Control.

An Oracle Fusion Middleware enterprise deployment:

- Considers various business service level agreements (SLA) to make high-availability best practices as widely applicable as possible
- Leverages database grid servers and storage grid with low-cost storage to provide highly resilient, lower cost infrastructure
- Uses results from extensive performance impact studies for different configurations to ensure that the high-availability architecture is optimally configured to perform and scale to business needs
- Enables control over the length of time to recover from an outage and the amount of acceptable data loss from a natural disaster
- Evolves with each Oracle version and is completely independent of hardware and operating system

For more information on high availability practices, see the Oracle Database High Availability page on Oracle Technology Network at:

http://www.oracle.com/technetwork/database/features/availability/index-087701.html

Note: The Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management focuses on enterprise deployments in Linux environments. However, you can also implement enterprise deployments using UNIX environments.

1.2 Enterprise Deployment Guide Conventions

All UNIX and Linux command examples shown in this Guide are run using the `bash` shell.

1.3 Enterprise Deployment Terminology

This section identifies enterprise deployment terminology used in the guide.

- **Oracle home**: An Oracle home contains installed files necessary to host a specific product. For example, the SOA Oracle home contains a directory that contains binary and library files for Oracle SOA Suite. An Oracle home resides within the directory structure of the Middleware home. Each Oracle home can be associated with multiple Oracle instances or Oracle WebLogic Server domains.
- **Oracle Common home**: This environment variable and related directory path refers to the Oracle home that contains the binary and library files required for the Oracle Enterprise Manager Fusion Middleware Control and Java Required Files (JRF).
- **WebLogic Server home**: A WebLogic Server home contains installed files necessary to host a WebLogic Server. The WebLogic Server home directory is a peer of Oracle home directories and resides within the directory structure of the Middleware home.
- **Middleware home:** A Middleware home consists of the Oracle WebLogic Server home, and, optionally, one or more Oracle homes. A Middleware home can reside on a local file system or on a remote shared disk that is accessible through NFS.
- **Oracle instance**: An Oracle instance contains one or more active middleware system components, for example Oracle Web Cache, Oracle HTTP Server, or Oracle Internet Directory. You determine which components are part of an instance, either at install time or by creating and configuring an instance at a later time. An Oracle instance contains files that can be updated, such as configuration files, log files, temporary files.
- **failover**: When a member of a high availability system fails unexpectedly (unplanned downtime), in order to continue offering services to its consumers, the system undergoes a failover operation. If the system is an active-passive system, the passive member is activated during the failover operation and consumers are directed to it instead of the failed member. The failover process can be performed manually, or it can be automated by setting up hardware cluster services to detect failures and move cluster resources from the failed node to the standby node. If the system is an active-active system, the failover is performed by the load balancer entity serving requests to the active members. If an active member fails, the load balancer detects the failure and automatically redirects requests for the failed member to the surviving active members. See *Oracle Fusion Middleware High Availability Guide* for information on active-active and active-passive systems.

- **failback**: After a system undergoes a successful failover operation, the original failed member can be repaired over time and be re-introduced into the system as a standby member. If desired, a failback process can be initiated to activate this member and deactivate the other. This process reverts the system back to its pre-failure configuration.
- **hardware cluster**: A hardware cluster is a collection of computers that provides a single view of network services (for example: an IP address) or application services (for example: databases, Web servers) to clients of these services. Each node in a hardware cluster is a standalone server that runs its own processes. These processes can communicate with one another to form what looks like a single system that cooperatively provides applications, system resources, and data to users.

  A hardware cluster achieves high availability and scalability through the use of specialized hardware (cluster interconnect, shared storage) and software (health monitors, resource monitors). (The cluster interconnect is a private link used by the hardware cluster for heartbeat information to detect node death.) Due to the need for specialized hardware and software, hardware clusters are commonly provided by hardware vendors such as Sun, HP, IBM, and Dell. While the number of nodes that can be configured in a hardware cluster is vendor dependent, for the purpose of Oracle Fusion Middleware high availability, only two nodes are required. Hence, this document assumes a two-node hardware cluster for high availability solutions employing a hardware cluster.
- **cluster agent**: The software that runs on a node member of a hardware cluster that coordinates availability and performance operations with other nodes. Clusterware provides resource grouping, monitoring, and the ability to move services. A cluster agent can automate the service failover.
- **clusterware**: A software that manages the operations of the members of a cluster as a system. It allows one to define a set of resources and services to monitor via a heartbeat mechanism between cluster members and to move these resources and services to a different member in the cluster as efficiently and transparently as possible.
- **shared storage**: Shared storage is the storage subsystem that is accessible by all the machines in the enterprise deployment domain. Among other things, the following are located on the shared disk:
  - Middleware Home software
  - AdminServer Domain Home
  - JMS
  - Tlogs (where applicable)

  Managed Server homes can also be optionally located in the shared disk. The shared storage can be a Network Attached Storage (NAS), a Storage Area Network (SAN) or any other storage system that multiple nodes can access simultaneously and can read-write.
- **primary node**: The node that is actively running an Oracle Fusion Middleware instance at any given time and has been configured to have a backup/secondary node. If the primary node fails, Oracle Fusion Middleware instance is failed over to the secondary node. This failover can be manual or automated using the Clusterware for Administration Server. For a server migration based scenario, WebLogic Whole Server Migration is used for automated failover.

- **secondary node**: The node that is the backup node for an Oracle Fusion Middleware instance. This is where the active instance fails over when the primary node is no longer available. See the definition for primary node in this section.
- **network host name**: Network host name is a name assigned to an IP address either through the `/etc/hosts` file or through DNS resolution. This name is visible in the network that the machine to which it refers to is connected. Often, the network host name and physical host name are identical. However, each machine has only one physical host name but may have multiple network host names. Thus, a machine's network host name may not always be its physical host name.
- **physical host name**: This guide differentiates between the terms physical host name and network host name. This guide uses physical host name to refer to the "internal name" of the current machine. On Linux, this is the name returned by the `hostname` command.

  Physical host name is used by Oracle Fusion Middleware to reference the local host. During installation, the installer automatically retrieves the physical host name from the current machine and stores it in the Oracle Fusion Middleware configuration metadata on disk.
- **physical IP**: Physical IP refers to the IP of a machine on the network. In almost all cases, it is normally associated with the physical host name of the machine (see the definition of the physical host name). In contrast to a virtual IP, it is always associated with the same machine when on a network.
- **switchover**: During normal operation, active members of a system may require maintenance or upgrading. A switchover process can be initiated to allow a substitute member to take over the workload performed by the member that requires maintenance or upgrading, which undergoes planned downtime. The switchover operation ensures continued service to consumers of the system.
- **switchback**: When a switchover operation is performed, a member of the system is deactivated for maintenance or upgrading. When the maintenance or upgrading is completed, the system can undergo a switchback operation to activate the upgraded member and bring the system back to the pre-switchover configuration.
- **virtual host name**: Virtual host name is a network addressable host name that maps to one or more physical machines via a load balancer or a hardware cluster. For load balancers, the name "virtual server name" is used interchangeably with virtual host name in this book. A load balancer can hold a virtual host name on behalf of a set of servers, and clients communicate indirectly with the machines using the virtual host name. A virtual host name in a hardware cluster is a network host name assigned to a cluster virtual IP. Because the cluster virtual IP is not permanently attached to any particular node of a cluster, the virtual host name is not permanently attached to any particular node either.

  Note: Whenever the term "virtual host name" is used in this document, it is assumed to be associated with a virtual IP address. In cases where just the IP address is needed or used, it will be explicitly stated.

- **virtual IP**: Also, cluster virtual IP and load balancer virtual IP. Generally, a virtual IP can be assigned to a hardware cluster or load balancer. To present a single system view of a cluster to network clients, a virtual IP serves as an entry point IP

address to the group of servers which are members of the cluster. A virtual IP can be assigned to a server load balancer or a hardware cluster.

A hardware cluster uses a cluster virtual IP to present to the outside world the entry point into the cluster (it can also be set up on a standalone machine). The hardware cluster's software manages the movement of this IP address between the two physical nodes of the cluster while clients connect to this IP address without the need to know which physical node this IP address is currently active on. In a typical two-node hardware cluster configuration, each machine has its own physical IP address and physical host name, while there could be several cluster IP addresses. These cluster IP addresses float or migrate between the two nodes. The node with current ownership of a cluster IP address is active for that address.

A load balancer also uses a virtual IP as the entry point to a set of servers. These servers tend to be active at the same time. This virtual IP address is not assigned to any individual server but to the load balancer which acts as a proxy between servers and their clients.

These will be described in more detail in the following chapters.

1.4 Benefits of Oracle Recommendations

The Oracle Fusion Middleware configurations discussed in this guide are designed to ensure security of all transactions, maximize hardware resources, and provide a reliable, standards-compliant system for enterprise computing with a variety of applications. The security and high availability benefits of the Oracle Fusion Middleware configurations are realized through isolation in firewall zones and replication of software components.

This section contains the following topics:

- Section 1.4.1, "Built-in Security"
- Section 1.4.2, "High Availability"

1.4.1 Built-in Security

The Enterprise Deployment architectures are secure because every functional group of software components is isolated in its own zone, and all traffic is restricted by protocol and port. The following characteristics ensure security at all needed levels, as well as a high level of standards compliance:

- All external communication received on port 80 (*HTTP_PORT*) is redirected to port 443 (*HTTP_SSL_PORT*).
- External communication uses the Secure Socket Layer (SSL) secure Web Protocol. This is terminated at the site's load balancer.
- Communication from external clients does not go beyond the Load Balancing Router level.
- No direct communication from the Load Balancing Router to the data tier zone is allowed.
- Components are separated between zones on thethe web tier, application tier, and database tier.
- Direct communication across two firewalls at any one time is prohibited.
- If a communication begins in one firewall zone, it must end in the next firewall zone.

- LDAP directories can be isolated in a directory tier zone. (Oracle Unified Directory is in the application tier zone).
- Identity Management components are in the application tier zone.
- All communication between components across zones is restricted by port and protocol, according to firewall rules.

1.4.2 High Availability

The Enterprise Deployment architectures are highly available because each component or functional group of software components is replicated on a different computer and configured for component-level high availability.

2

Introduction and Planning

This chapter describes and illustrates the enterprise deployment reference topologies employed in this guide.

The key to a successful Enterprise Deployment is planning and preparation. The road map for installation and configuration at the end of this chapter directs you to the appropriate chapters for the tasks you need to perform. Use this chapter to help you map the examples used in this guide to your own deployment.

You can use Appendix B, "Worksheets for Identity Management Topology" to help you keep track of information.

This chapter contains the following topics:

2.1 Planning Your Deployment

An enterprise deployment for Identity Management consists of the following parts:

- A highly available database for storing policy information and information specific to the identity management components being deployed.
- Identity Management components installed in a highly available manner to support the creation and management of identity information, as well as to restrict access to resources based on the policies and identities stored within the database and identity store. Identity Management components can be divided into three categories:
  - Directory Services: one or more highly available directories for storing identity information
  - Identity Management Provisioning: Oracle Identity Manager
  - Access Control: Oracle Access Management
- A highly available web tier which is used to access Identity management components, to restrict access to those components and to ascertain the identity of people and processes trying to gain access to corporate resources.
- A highly available load balancer, which is used to distribute load between the web servers. The load balancer can also be used to off load SSL encryption to ensure

that communication between user sessions and Oracle Identity Management are encrypted, but without the overhead of having to enable SSL between the individual identity management components.

There are many ways that these component parts can be put together. Three topologies are shown in the next section. This guide explains in detail how to deploy them. These topologies are not the only ones supported by Oracle, but they are deemed to be the most common.

Note: This guide does not show how to create Oracle Internet Directory instances or the Oracle Internet Directory domain.

This section contains the following topics:

- Section 2.1.1, "Deployment Topologies"
- Section 2.1.2, "Which Topology Should I Use?"

2.1.1 Deployment Topologies

A topology is a deployment map of components. There are several different ways that Oracle Identity Management components can be installed to provide a working Identity and Access management solution. A topology can also be described as an architectural blueprint. This guide shows the most common deployment topologies for Oracle Identity and Access Management.

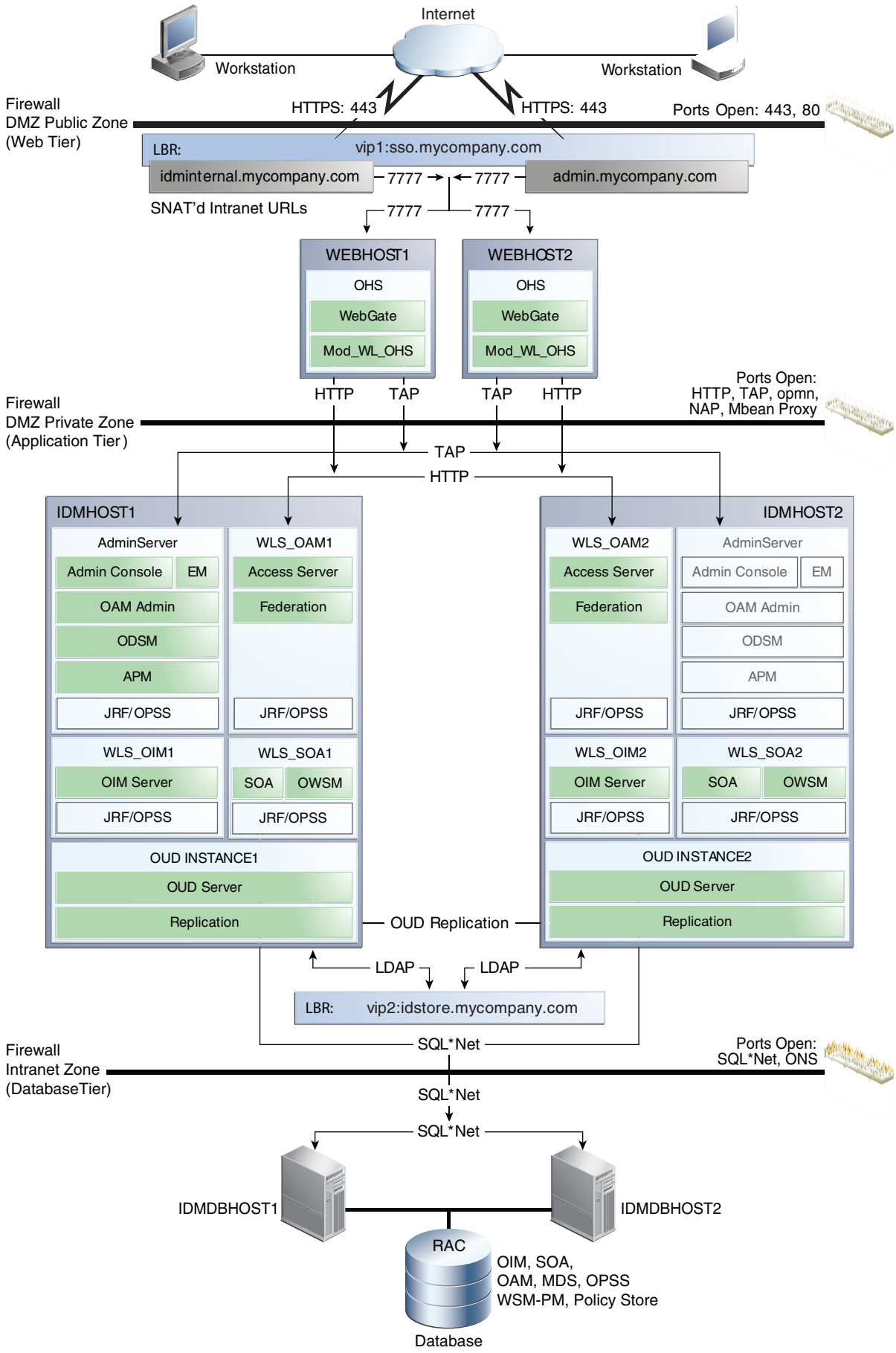
There are many different ways that Oracle Identity and Access Management can be deployed in an Enterprise Deployment. The two most common deployment models involve placing everything into a single domain and separating operational and managerial components into different domains. The three figures below show these two different deployment models.

This section contains the following topics:

- Section 2.1.1.1, "Single Domain Topology"
- Section 2.1.1.2, "Split Domain Topology"
- Section 2.1.1.3, "Three Domain Topology"

2.1.1.1 Single Domain Topology

Figure 2–1 Single Domain Topology


Internet
Workstation
Workstation
Firewall
DMZ Public Zone
(Web Tier)
HTTPS: 443
HTTPS: 443
Ports Open: 443, 80
LBR:
vip1:sso.mycompany.com
idminternal.mycompany.com
7777
7777
admin.mycompany.com
SNAT'd Intranet URLs
7777
7777
WEBHOST1
OHS
WebGate
Mod_WL_OHS
WEBHOST2
OHS
WebGate
Mod_WL_OHS
HTTP
TAP
TAP
HTTP
Ports Open:
HTTP, TAP, opmn,
NAP, Mbean Proxy
Firewall
DMZ Private Zone
(Application Tier)
TAP
HTTP
IDMHOST1
AdminServer
Admin Console
EM
OAM Admin
ODSM
APM
JRF/OPSS
WLS_OAM1
Access Server
Federation
JRF/OPSS
WLS_OIM1
OIM Server
JRF/OPSS
WLS_SOA1
SOA
OWSM
JRF/OPSS
OUD INSTANCE1
OUD Server
Replication
IDMHOST2
WLS_OAM2
Access Server
Federation
JRF/OPSS
AdminServer
Admin Console
EM
OAM Admin
ODSM
APM
JRF/OPSS
WLS_OIM2
OIM Server
JRF/OPSS
WLS_SOA2
SOA
OWSM
JRF/OPSS
OUD INSTANCE2
OUD Server
Replication
OUD Replication
LDAP
LDAP
LBR:
vip2:idstore.mycompany.com
SQL*Net
Ports Open:
SQL*Net, ONS
Firewall
Intranet Zone
(DatabaseTier)
SQL*Net
SQL*Net
IDMDBHOST1
IDMDBHOST2
RAC
OIM, SOA,
OAM, MDS, OPSS
WSM-PM, Policy Store
Database

This figure is a graphical representation of the enterprise topology. It includes icons and symbols that represent the hardware load balancer, host computers, firewalls, and other elements of the topology. At a high level, it shows the main components of the topology, including the following:

- The Web Tier: There are two servers, each of which hosts an Oracle HTTP Server and Oracle WebGate.
- The Application Tier: There are two servers, IDMHOST1 and IDMHOST2. Each contains managed servers for the following products:
  - Oracle Access Management, which hosts Access Server, Federation and corresponding JRF/OPSS processes
  - Oracle Identity Manager, which hosts an OIM Server and corresponding JRF/OPSS processes
  - SOA, which hosts a SOA Server and corresponding JRF/OPSS processes
  - Oracle Unified Directory. Each host has an instance of Oracle Unified Directory which is used as the LDAP directory for identity information. Each Oracle Unified Directory instance is kept up to date through Oracle Unified Directory replication.

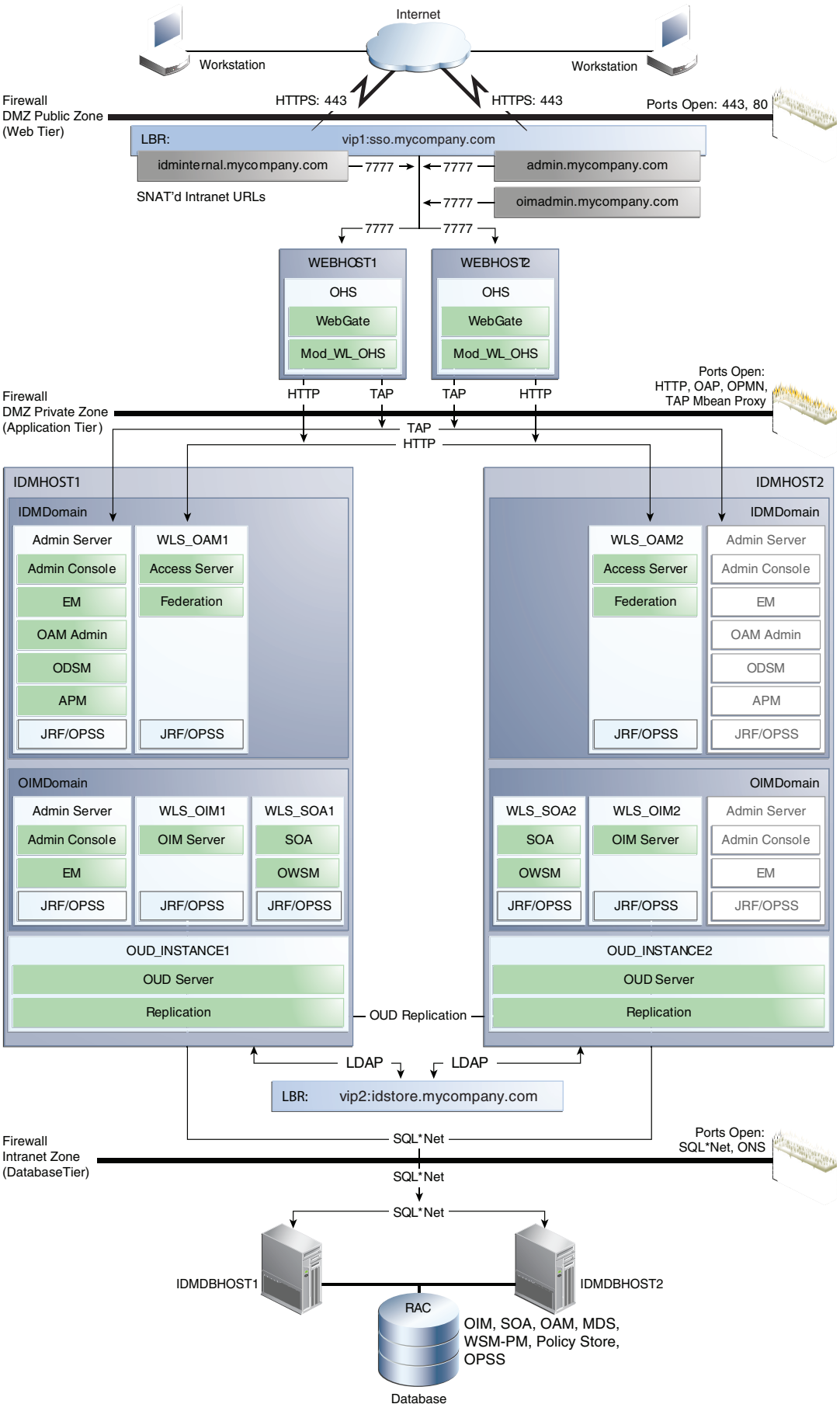
  IDMHOST1 also contains the WebLogic Administration Server, which hosts the WebLogic Console, Enterprise Manager Fusion Middleware Control, OAM Console, APM Console and ODSM (for Oracle Unified Directory). In the event of the failure of IDMHOST1, the WebLogic Administration Server can be started on IDMHOST2.
- The Data Tier: This is where the databases reside. The databases contain customer data and the schemas required by the application tier products.
- The Load Balancer: Inside the demilitarized zone (DMZ) is a load balancer which directs requests received on SSO.mycompany.com, ADMIN.mycompany.com and IDMINTERNAL.mycompany.com and directs requests to the Oracle HTTP servers. In the case of SSO.mycompany.com, requests are SSL encrypted. This is terminated at the load balancer. ADMIN.mycompany.com and IDMINTERNAL.mycompany.com handle requests using the HTTP protocol.

  In addition, the load balancer distributes LDAP requests among the Oracle Unified Directory instances on IDMHOST1 and IDMHOST2, using the load balancer virtual host IDSTORE.mycompany.com
- Firewalls: These are used to separate the Web, Application, and Directory tiers into different zones. WEBHOST1 and WEBHOST2 reside in the DMZ.

For more information, refer to the descriptions of the topology tiers in the sections that follow the diagrams. The instructions in this guide describe how to install and configure the software for this topology.

2.1.1.2 Split Domain Topology

Figure 2–2 Split Domain Topology


Internet
Workstation
Workstation
Firewall
DMZ Public Zone
(Web Tier)
HTTPS: 443
HTTPS: 443
Ports Open: 443, 80
LBR:
vip1:sso.mycompany.com
idminternal.mycompany.com
7777
7777
admin.mycompany.com
SNAT'd Intranet URLs
7777
oimadmin.mycompany.com
7777
7777
WEBHOST1
OHS
WebGate
Mod_WL_OHS
WEBHOST2
OHS
WebGate
Mod_WL_OHS
HTTP
TAP
TAP
HTTP
Ports Open:
HTTP, OAP, OPMN,
TAP Mbean Proxy
Firewall
DMZ Private Zone
(Application Tier)
TAP
HTTP
IDMHOST1
IDMDomain
Admin Server
Admin Console
EM
OAM Admin
ODSM
APM
JRF/OPSS
WLS_OAM1
Access Server
Federation
JRF/OPSS
IDMHOST2
IDMDomain
WLS_OAM2
Access Server
Federation
JRF/OPSS
Admin Server
Admin Console
EM
OAM Admin
ODSM
APM
JRF/OPSS
OIMDomain
Admin Server
Admin Console
EM
JRF/OPSS
WLS_OIM1
OIM Server
JRF/OPSS
WLS_SOA1
SOA
OWSM
JRF/OPSS
OIMDomain
WLS_SOA2
SOA
OWSM
JRF/OPSS
WLS_OIM2
OIM Server
JRF/OPSS
Admin Server
Admin Console
EM
JRF/OPSS
OUD_INSTANCE1
OUD Server
Replication
OUD Replication
OUD_INSTANCE2
OUD Server
Replication
LDAP
LDAP
LBR:
vip2:idstore.mycompany.com
SQL*Net
Firewall
Intranet Zone
(DatabaseTier)
Ports Open:
SQL*Net, ONS
SQL*Net
SQL*Net
IDMDBHOST1
IDMDBHOST2
RAC
OIM, SOA, OAM, MDS,
WSM-PM, Policy Store,
OPSS
Database

This figure is similar to Figure 2–1. It differs in that the WebLogic managed servers for Oracle Access Management are placed into a domain called IDMDomain, and the managed servers for Oracle Identity Manager and SOA components are placed into a domain called OIMDomain

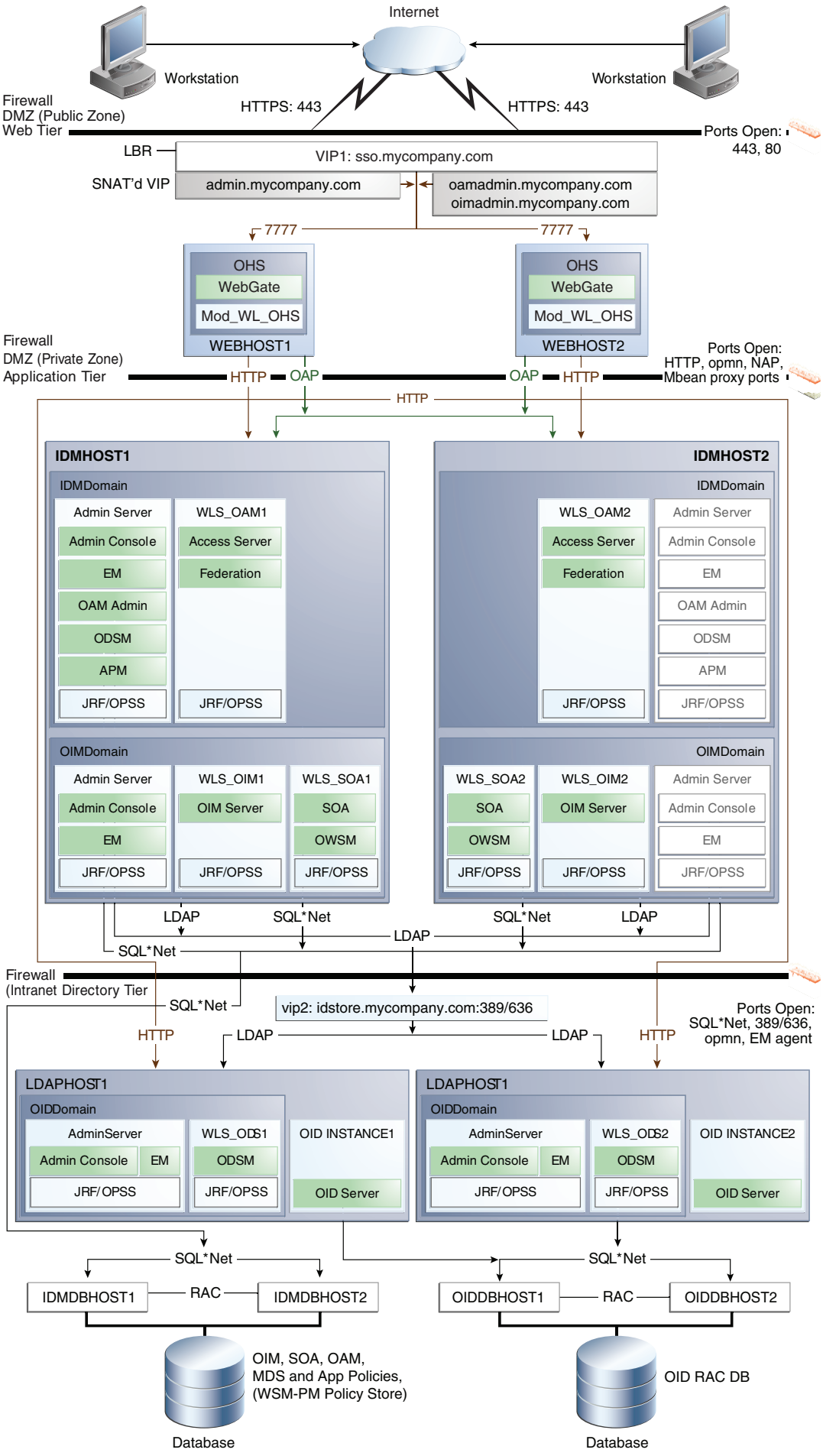
In addition, a second administration server is configured on IDMHOST1 to support the second domain, OIMDomain.

Chapter 16, "Creating a Split Domain Topology," describes how to install and configure the software for this topology.

For more information, refer to the descriptions of the topology tiers in the sections that follow the diagrams.

2.1.1.3 Three Domain Topology

Figure 2–3 Three Domain Topology


Internet
Workstation
Workstation
Firewall
DMZ (Public Zone)
Web Tier
HTTPS: 443
HTTPS: 443
Ports Open:
443, 80
LBR
VIP1: sso.mycompany.com
SNAT'd VIP
admin.mycompany.com
oamadmin.mycompany.com
oimadmin.mycompany.com
7777
7777
OHS
WebGate
Mod_WL_OHS
WEBHOST1
OHS
WebGate
Mod_WL_OHS
WEBHOST2
Firewall
DMZ (Private Zone)
Application Tier
HTTP
OAP
OAP
HTTP
Ports Open:
HTTP, opmn, NAP,
Mbean proxy ports
HTTP
IDMHOST1
IDMDomain
Admin Server
Admin Console
EM
OAM Admin
ODSM
APM
JRF/OPSS
WLS_OAM1
Access Server
Federation
JRF/OPSS
OIMDomain
Admin Server
Admin Console
EM
JRF/OPSS
WLS_OIM1
OIM Server
JRF/OPSS
WLS_SOA1
SOA
OWSM
JRF/OPSS
IDMHOST2
IDMDomain
WLS_OAM2
Access Server
Federation
JRF/OPSS
Admin Server
Admin Console
EM
OAM Admin
ODSM
APM
JRF/OPSS
OIMDomain
WLS_SOA2
SOA
OWSM
JRF/OPSS
WLS_OIM2
OIM Server
JRF/OPSS
Admin Server
Admin Console
EM
JRF/OPSS
LDAP
SQL*Net
LDAP
SQL*Net
LDAP
SQL*Net
Firewall
(Intranet Directory Tier
SQL*Net
vip2: idstore.mycompany.com:389/636
Ports Open:
SQL*Net, 389/636,
opmn, EM agent
HTTP
LDAP
LDAP
HTTP
LDAPHOST1
OIDDomain
AdminServer
Admin Console
EM
JRF/OPSS
WLS_ODS1
ODSM
JRF/OPSS
OID INSTANCE1
OID Server
LDAPHOST1
OIDDomain
AdminServer
Admin Console
EM
JRF/OPSS
WLS_ODS2
ODSM
JRF/OPSS
OID INSTANCE2
OID Server
SQL*Net
SQL*Net
IDMDBHOST1
RAC
IDMDBHOST2
OIDDBHOST1
RAC
OIDDBHOST2
OIM, SOA, OAM,
MDS and App Policies,
(WSM-PM Policy Store)
Database
OID RAC DB
Database

This figure is similar to Figure 2–2. It differs in that Oracle Unified Directory is replaced by Oracle Internet Directory. The Oracle Internet Directory components are moved into the Database Tier, which has been renamed Directory Tier.

In the Directory Tier, there are two new hosts, OIDHOST1 and OIDHOST2. These hosts contain the Oracle Internet Directory instances. There is also a separate domain called OIDDomain which contains a WebLogic Administration Server and a pair of WebLogic managed servers, each of which hosts ODSM.

The Directory Tier also contains an additional database, which contains the Oracle Internet Directory schemas.

For more information, refer to the descriptions of the topology tiers in the sections that follow the diagrams.

2.1.2 Which Topology Should I Use?

The single domain topology and the split domain topology each have advantages and disadvantages.

This section contains the following topics:

- Section 2.1.2.1, "Single Domain Topology"
- Section 2.1.2.2, "Split Domain Topology"
- Section 2.1.2.3, "Three Domain Topology"
- Section 2.1.2.4, "Summary"

2.1.2.1 Single Domain Topology

The advantages of the single domain topology are:

- It is suitable for the majority of implementations
- Every component is contained within a single WebLogic domain, making setup, management and patching simpler.
- It is simpler to set up than the Split Domain Model.
- The same topology is suitable for production and non-production systems
- This is the same deployment model used by Identity Management for Fusion Applications
- There are no complications introduced by having different IAM components in different domains.
- Enterprise Manager Fusion Middleware Control shows a single consolidated view of all of the identity management components.

The disadvantages of the single deployment model are:

- Identity Provisioning and Access Control are provided by the same deployment.
- If a patch must be applied, and application of that patch requires the entire domain to be stopped, then service is interrupted to both Provisioning and Access Control.
- A patch set that contains domain level updates can introduce compatibility issues. This is less likely to occur in a domain which has only Identity and Access Management installed, that is, a domain with no Oracle Internet Directory, Oracle Virtual Directory, or Oracle Identity Federation.

2.1.2.2 Split Domain Topology

The split domain topology is suitable for large organizations requiring finite control over each component in the deployment.

The main advantages of the Split Domain topology are related to patching flexibility. Specifically:

- As each component (Oracle Identity Manager and Access Manager) reside in different domains, you can apply patches (even domain level ones) so that they update only the component they are targeted at.
- You can patch Administrative Components such as Oracle Identity Manager without the need for a controlled outage, which you would require when updating an Operational component such as Access Manager).

The disadvantage of the split deployment topology are:

- It is more complex to setup.
- Components must be configured to work cross-domain.
- Multiple software deployments are required, so there are more Middleware Homes.
- Generic patches must be deployed to each domain.
- Management of the domain is more complex. Specifically:
  - Two Administration Consoles
  - Two instances of Fusion Middleware Control
  - No single view of all the identity management components

2.1.2.3 Three Domain Topology

This topology is essentially the same as the split domain topology, but it uses Oracle Internet Directory instead of Oracle Unified Directory. In order to facilitate patching of Oracle Internet Directory independently of other components, Oracle Internet Directory is configured using a separate domain.

The guide assumes that you already have a highly available Oracle Internet Directory deployed, and does not cover its setup. For information on setting up highly available Oracle Internet Directory, see the "Oracle Internet Directory High Availability" section in *Oracle Fusion Middleware High Availability Guide*.

2.1.2.4 Summary

Unless you specifically require complete control of patching, that is, the ability to patch each identity management component separately, Oracle recommends that you use the single domain topology.

2.2 Understanding the Topologies

Each topology is divided into tiers for increased security and protection. The tiers are separated by firewalls that control access from one tier to the next. The goal is to prevent unauthorized traffic. In an Internet-facing topology, for instance, it should not be possible to directly access a database from the Internet zone. Only applications deployed in the application zone should have access to the database.

Each of the diagrams above shows three tiers:

- Web Tier

- Application Tier
- Database Tier

Although it is not shown on the figures, there can also be a directory tier (which is often included in the database tier). If a dedicated directory tier is introduced, LDAP directories can be placed within that tier. This is most commonly done in deployments using Oracle Internet Directory, which is closely tied to a database.

This section contains the following topics:

2.2.1 About the Web Tier

The web tier is in the DMZ Public Zone. The HTTP Servers are deployed in the web tier.

Most of the Identity Management components can function without the web tier, but for most enterprise deployments, the web tier is desirable. To support enterprise level single sign-on using products such as Oracle Single Sign-On and Access Manager, the web tier is required.

While components such as Oracle Enterprise Manager Fusion Middleware Control and Oracle Directory Services Manager can function without a web tier, they can be configured to use a web tier, if desired.

In the web tier:

- WEBHOST1 and WEBHOST2 have Oracle HTTP Server, WebGate (an Access Manager component), and the `mod_wl_ohs` plug-in module installed. The `mod_wl_ohs` plug-in module enables requests to be proxied from Oracle HTTP Server to a WebLogic Server running in the application tier.
- WebGate (an Oracle Access Management component) in Oracle HTTP Server uses Oracle Access Protocol (OAP) to communicate with Access Manager running on IDMHOST1 and IDMHOST2, in the Identity Management DMZ. WebGate and Access Manager are used to perform operations such as user authentication.

On the firewall protecting the web tier, the HTTP ports are 443 (*HTTP_SSL_PORT*) for HTTPS and 80 (*HTTP_PORT*) for HTTP. Port 443 is open.

2.2.1.1 Architecture Notes

Oracle HTTP Servers on WEBHOST1 and WEBHOST2 are configured with `mod_wl_ohs`, and proxy requests for the Oracle Enterprise Manager, Oracle Directory Integration Platform, and Oracle Directory Services Manager Java EE applications deployed in WebLogic Server on IDMHOST1 and IDMHOST2.

2.2.1.2 High Availability Provisions

If the Oracle HTTP server fails on the WEBHOST, Oracle Process Management and Notification (OPMN) server attempts to restart it.

2.2.1.3 Security Provisions

The Oracle HTTP Servers process requests received using the URLs `SSO.mycompany.com` and `ADMIN.mycompany.com`. The name `ADMIN.mycompany.com` is only resolvable inside the firewall. This prevents access to sensitive resources such as the Oracle WebLogic Server console and Oracle Enterprise Manager Fusion Middleware Control console from the public domain.

2.2.2 About the Application Tier

The application tier is the tier where Java EE applications are deployed. Products such as Oracle Identity Manager, Oracle Directory Integration Platform, Oracle Directory Services Manager and Oracle Enterprise Manager Fusion Middleware Control are the key Java EE components that can be deployed in this tier. Applications in this tier benefit from the High Availability support of Oracle WebLogic Server.

The Identity Management applications in the application tier interact with the directory tier as follows:

- They leverage the directory tier for enterprise identity information.
- In some cases, they leverage the directory tier (and sometimes the database in the data tier) for application metadata.
- Oracle Enterprise Manager Fusion Middleware Control and Oracle Directory Services Manager are administration tools that provide administrative functionalities to the components in the application tier as well as the directory tier.
- WebLogic Server has built-in web server support. If enabled, the HTTP listener exists in the application tier as well. However, for the enterprise deployment shown in Figure 1-1, customers have a separate web tier relying on web servers such as Apache or Oracle HTTP Server.

In the application tier, IDMHOST1 and IDMHOST2 include the following components:

- The operational component of the infrastructure. This component is Oracle Access Management Access Manager (OAM). This is an J2EE applications which is run within Oracle WebLogic Server.
- Oracle Identity Manager and Oracle Entitlements Server Policy Manager, which are used for user provisioning and policy management.
- The administrative components of Identity management, including Oracle Identity Manager, which is used for user provisioning. Note: These servers also run Oracle SOA which is used exclusively by Oracle Identity Manager.

  Oracle Identity Manager (OIM) communicates with the directory tier.

In addition:

- IDMHOST1 hosts an Oracle WebLogic Administration Server. Inside the administration server are managerial and navigational components for the domain including: Oracle WebLogic Console, Oracle Enterprise Manager Fusion Middleware Control, Oracle Access Management Console, and Oracle Directory Services Manager (ODSM) for Oracle Unified Directory. The WebLogic Administration server is a singleton process. That is, it can only be started on one server at a time. In the event that the host running the administration server fails, the Administration server can be manually started on a different host.

2.2.2.1 About WebLogic Domains

A domain is the basic administration unit for WebLogic Server instances. A domain consists of one or more WebLogic Server instances (and their associated resources) that you manage with a single Administration Server. You can define multiple domains based on different system administrators' responsibilities, application boundaries, or geographical locations of servers. Conversely, you can use a single domain to centralize all WebLogic Server administration activities.

In the context of Identity Management, it is recommended that you deploy the Identity Management components, plus SOA, in a separate WebLogic Server domain from the one where SOA, Web Center Portal and other customer applications might be deployed. In a typical enterprise deployment, the administration of identity management components such as LDAP directory, single sign-on solutions, and provisioning solutions is done by a different set of administrators from those who administer the middleware infrastructure and applications. Oracle Identity Manager can be deployed into a separate dedicated domain so that it can be patched independently of other products.

It is technically possible to deploy everything in a single domain in a development or test environment. However, in a production environment, the recommendation to use separate domains creates a logical administrative boundary between the identity management stack and the rest of the middleware and application deployment.

2.2.2.2 About LDAP Directories

Identity information is stored with an LDAP compliant directory. Oracle supports the following directories natively:

- Oracle Unified Directory
- Oracle Internet Directory

If you want to use a different directory, such as Microsoft Active Directory, you can either use Oracle Virtual Directory to present that information or use Oracle Directory Integration Platform to synchronize the users and groups from the other directory.

The standard LDAP port is 389 *(LDAP_LBR_PORT)* for the non-SSL port and 636 (*LDAP_LBR_SSL_PORT*) for the SSL port. LDAP services are often used for white pages lookup by clients such as email clients in the intranet. The ports 389 and 636 on the load balancer are typically redirected to the non-privileged ports used by the individual directory instances. LDAP requests are distributed among the Oracle Unified Directory, Oracle Internet Directory, or Oracle Virtual Directory hosts using a hardware load balancer.

If you have a complex directory implementation where identity information is split among multiple back end directories, you can use this implementation with Oracle Identity Management by using Oracle Virtual Directory.

Many organizations have an existing directory deployment. If you do not have an existing deployment, then you can use this guide to learn how to configure Oracle Unified Directory to take on this role. If you have an existing directory, you can use this guide to learn how to configure that directory for use with Oracle Identity Management. If you do not have a directory and you want to use a directory other than Oracle Unified Directory, refer to your directory documentation for information about configuring these directories in a highly available manner.

2.2.2.2.1 About Oracle Unified Directory If you store your identity information in Oracle Unified Directory, this information is stored locally in a Berkeley database. To ensure

high availability, this information is replicated to other Oracle Unified Directory instances using Oracle Unified Directory 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*.

Oracle Unified Directory normally keeps track of changes by using an internal change number. The change number is specific to each Oracle Unified Directory instance, which can cause issues if one Oracle Unified Directory instance fails without the entry being replicated. Such a failure can impact Oracle Identity Manager reconciliation.

Patch 16943171 allows Oracle Identity Manager to use a newer, more reliable Oracle Unified Directory mechanism for tracking changes, which relies on the use of cookies. This mechanism eliminates the issues that can arise on failover of Oracle Unified Directory. After Patch 16943171 is installed, Oracle Identity Manager uses cookie mode when automatically creating reconciliation jobs. If you decide to create custom reconciliation jobs, then you must force the use of Oracle Unified Directory cookie mode.

2.2.2.2.2 About Oracle Internet Directory and Oracle Virtual Directory If you are using Oracle Internet Directory as your Identity Store, you can configure it to use multimaster replication as described in the *Oracle Fusion Middleware High Availability Guide* chapter Configuring Identity Management for Maximum High Availability. This enables you

to maintain the same naming contexts on multiple directory servers. It can improve performance by providing more servers to handle queries and by bringing the data closer to the client. It improves reliability by eliminating risks associated with a single point of failure.

Oracle Identity Management, which includes Oracle Internet Directory, is on a different release cycle from Oracle Identity and Access Management. If your identity information is in Oracle Internet Directory, Oracle recommends that, for ease of patching, you place the Oracle Identity Management components, such as ODSM, in a separate MW_HOME and domain. It is, however, supported to have the components reside in the same domain. If you do not intend to manage your directories using Oracle Directory Services Manager, then you do not need to have a separate WebLogic domain as described in the directory topologies.

If you are using Oracle Unified Directory or Oracle Internet Directory exclusively, you do not need to use Oracle Virtual Directory.

Oracle Internet Directory and Oracle Virtual Directory are closely tied to the database tier for the following reasons:

- Oracle Internet Directory relies on Oracle Database as its back end.
- Oracle Virtual Directory provides virtualization support for other LDAP services or databases or both.

2.2.2.2.3 High Availability Provisions In addition to the directory deployments, which are typically active/active, the following features provide high availability for the directories:

- If the Oracle Internet Directory fails, Oracle Process Management and Notification (OPMN) server attempts to restart it.
- If the Oracle Virtual Directory fails, Oracle Process Management and Notification (OPMN) server attempts to restart it.
- There is no automatic restart if Oracle Unified Directory fails. Oracle Unified Directory relies on requests being redirected to a surviving instance by the load balancer.

2.2.2.3 Architecture Notes

- An embedded version of Oracle Entitlement Server is used to control access to Oracle Fusion Middleware components.
- Oracle Entitlements Server uses a centralized policy store that is stored within a database.
- Access Manager uses the OPSS Policy Store to store policy information.
- In a split domain configuration, OIM and SOA are installed into a separate domain from other components.
- The Oracle WebLogic Server console, Oracle Enterprise Manager Fusion Middleware Control, and Oracle Access Management console are always bound to the listen address of the Administration Server.
- The WebLogic administration server is a singleton service. It runs on only one node at a time. In the event of failure, it is restarted on a surviving node.
- The managed servers WLS_OAM1 and WLS_OAM2 are deployed in a cluster and Access Manager applications deployed to the cluster.

- The managed servers WLS_OIM1 and WLS_OIM2 are deployed in a cluster and Access Manager applications deployed to the cluster.
- The managed servers WLS_SOA1 and WLS_SOA2 are deployed in a cluster and Access Manager applications deployed to the cluster.

2.2.2.4 High Availability Provisions

- OAM Server, Oracle Identity Manager, and SOA are active-active deployments; these servers communicate with the data tier at run time.
- The WebLogic Administration Server and Oracle Enterprise Manager deployment is active-passive (where other components are active-active). There is one Administration Server per domain.
- The WebLogic Administration Server is a singleton component deployed in an active-passive configuration. If the primary fails or the Administration Server on IDMHOST1 does not start, the Administration Server on the secondary host can be started. If a WebLogic managed server fails, the node manager running on that host attempts to restart it.

2.2.2.5 Security Provisions

Oracle WebLogic Server Console, Oracle Enterprise Manager Fusion Middleware Control console, and Oracle Access Management Console are only accessible through a virtual host configured on the load balancer, which is only available inside the firewall.

2.2.3 About the Optional Directory Tier

No directory tier is shown in Section 2.1.1, "Deployment Topologies," as this Guide features the use of Oracle Unified Directory, which typically resides in the application tier. If you are using Oracle Internet Directory or Oracle Virtual Directory, however, you might need a directory tier. You can also put Oracle Unified Directory in a directory tier for added security, as the directory tier is typically protected by firewalls. Applications above the directory tier access LDAP services through a designated LDAP host port.

The directory tier is typically managed by directory administrators providing enterprise LDAP service support.

2.2.4 About the Database Tier

Starting with 11g Release 2 (11.1.2), policy information is stored in the database. The database is also used for storing information specific to the identity management components being deployed.

In some cases, the directory tier and data tier might be managed by the same group of administrators. In many enterprises, however, database administrators own the data tier while directory administrators own the directory tier.

Although not shown in the diagram, you might be required to place directories into the Database tier for added security. This is most likely to be required in deployments which use Oracle Internet Directory, which requires access to a database for storing its information.

2.3 Hardware Requirements for an Enterprise Deployment

The deployments shown in the topology diagrams use a small number of powerful servers, which makes the deployment simpler. It is, however, not mandatory to use such powerful servers. You can, alternatively, distribute managed servers over a larger number of smaller servers.

If you are planning to deploy on the same number of servers shown in the diagrams, these servers should have the minimum specification shown in Table 2–1.

For detailed requirements, or for requirements for other platforms, see *Oracle Fusion Middleware System Requirements and Specifications*.

Table 2–1 Typical Hardware Requirements

Server	Processor	Disk	Memory	TMP Directory	Swap
Database Host IDMDBHOST*n*	4 or more X Pentium 1.5 GHz or greater	nXm n=Number of disks, at least 4 (striped as one disk). m=Size of the disk (minimum of 30 GB)	6-16 GB	Default	Default
WEBHOST*n*	2 or more X Pentium 1.5 GHz or greater	10 GB	4 GB	Default	Default
IDMHOST*n*	4 or more X Pentium 1.5 GHz or greater	20 GB	16 GB	Default	Default

These are the typical hardware requirements. For each tier, carefully consider the load, throughput, response time and other requirements to plan the actual capacity required. The number of nodes, CPUs, and memory required can vary for each tier based on the deployment profile.Production requirements may vary depending on applications and the number of users.

The Enterprise Deployment configurations described in this guide use two servers for each tier to provide failover capability. This, however, does not presume adequate computing resources for any application or user population. If the system workload increases such that performance is degraded, you can add WebLogic servers or directory/OHS instances as described in Section 17.4, "Scaling Enterprise Deployments".

Note: Oracle recommends configuring all nodes in the topology identically with respect to operating system levels, patch levels, user accounts, and user groups.

2.4 Software Components for an Enterprise Deployment

This section describes the software required for an Oracle Identity Management enterprise deployment.

This section contains the following topics:

- Section 2.4.1, "Software Versions"
- Section 2.4.2, "About Obtaining Software"
- Section 2.4.3, "Summary of Oracle Homes"
- Section 2.4.4, "About Installing Software"

- Section 2.4.5, "Applying Patches and Workarounds"

2.4.1 Software Versions

Table 2–2, " Software Versions Used" lists the Oracle software you need to obtain before starting the procedures in this guide.

Table 2–2 Software Versions Used

Short Name	Product	Version
OHS11G	Oracle HTTP Server	11.1.1.6.0
JRockit	Oracle JRockit	jrockit-jdk1.6.0_29-R28.2.0-4.0.1 or newer
WLS	Oracle WebLogic Server	10.3.6.0
IAM	Oracle Identity and Access Management	11.1.2.1.0
SOA	Oracle SOA Suite	11.1.1.6.0
WebGate	WebGate 11*g*	11.1.2.1.0
RCU	Repository Creation Assistant	11.1.2.1.0
OUD	Oracle Unified Directory	11.1.2.1.0
IDM (optional)	Oracle Identity Management	11.1.2.1.0

2.4.2 About Obtaining Software

For complete information about downloading Oracle Fusion Middleware software, see the *Oracle Fusion Middleware 11g Release 1 Download, Installation, and Configuration Readme* for this release, at: `http://docs.oracle.com/cd/E23104_01/download_readme.htm`

2.4.3 Summary of Oracle Homes

Oracle binaries are installed into an Oracle Fusion Middleware home. Individual products are installed into Oracle homes within the Middleware home. Table 2–3 is a summary of the Middleware homes and Oracle homes used in this document.

The installation and configuration of Oracle Identity Management is outside the scope of this Guide. See *Oracle Fusion Middleware High Availability Guide* for more information.

Table 2–3 Summary of Homes

Home Name	Home Description	Products Installed
MW_HOME	Consists of the Oracle WebLogic Server home and, optionally, one or more Oracle homes.	
WL_HOME	This is the root directory in which Oracle WebLogic Server is installed. The `WL_HOME` directory is a peer of Oracle home directory and resides within the *MW_HOME*.	Oracle WebLogic Server

Table 2–3 (Cont.) Summary of Homes

Home Name	Home Description	Products Installed
IDM_ORACLE_HOME	Contains the binary and library files for Oracle Identity Management and is located in: *IAM_MW_HOME*/idm.	Oracle Internet Directory Oracle Virtual Directory Oracle Directory Services Manager
IAM_ORACLE_HOME	Contains the binary and library files required for Oracle Identity and Access Management and is located in *IAM_MW_HOME*/iam.	Oracle Access Manager Oracle Identity Management
OUD_ORACLE_HOME	Contains the binary and library files required for Oracle Unified Directory and is located in *IAM_MW_HOME*/oud	Oracle Unified Directory
WEB_ORACLE_HOME	Contains the binary and library files required for Oracle HTTP Server. and is located in *WEB_MW_HOME*/web.	Oracle WebGate
SOA_ORACLE_HOME	Contains the binary and library files required for the Oracle SOA Suite.Required only when creating topologies with OIM and is located in *MW_HOME*/soa.	Oracle SOA Suite
ORACLE_COMMON_HOME	Contains the generic Oracle home files. This Oracle home is created automatically by any product installation and is located in *MW_HOME*/oracle_common.	Generic commands

2.4.4 About Installing Software

You perform software installation using the Oracle Installer, which resides on the installation media. In this guide, you deploy software into different directories, which are described in Chapter 4, "Preparing Storage for an Enterprise Deployment."

The installation tasks you must perform can be summarized as follows:

- Install OHS onto local storage on hosts in the web tier
- Install Oracle WebGate onto local storage on hosts in the web tier
- Install Oracle WebLogic Server onto shared storage available to hosts in the application tier.
- Install Oracle Identity and Access Management onto shared storage available to hosts in the application tier
- If you use Oracle Unified Directory, install it onto shared storage available to hosts in the application tier.
- If you are using Oracle Internet Directory or Oracle Virtual Directory, you install Oracle Identity Management onto hosts in the Data Tier. If you plan to use these products and do not have a highly available directory, you must create one as described in the *Oracle Fusion Middleware High Availability Guide*. In addition to Oracle Internet Directory or Oracle Virtual Directory, you must also install ODSM for Oracle Internet Directory and Oracle Virtual Directory.

If you are creating a split domain configuration, you must install Oracle Identity Management into two separate locations on the application tier to facilitate independent patching.

Detailed instructions for the installation of the Oracle software are found in the component chapters in this book.

Some products, such as Oracle Internet Directory and Oracle Virtual Directory, require you to run a script that sets the permissions of some files to `root`.

Oracle strongly recommends that you read the release notes for any additional installation and deployment considerations prior to starting the setup process.

2.4.5 Applying Patches and Workarounds

See the Oracle Fusion Middleware Release Notes for your platform and operating system for a list of patches to apply. You **must** apply the patches to ensure that your software operates as expected.

Caution: In particular, Patch 16943171 is critical if you are using Oracle Unified Directory in active-active mode, as shown in the topology diagrams. Failure to apply this patch might result in data inconsistency in the event of a failover.

Patches are available for download from `http://support.oracle.com`. You can find instructions for deploying each patch in the enclosed `README.html` file.

2.5 Road Map for the Reference Topology Installation and Configuration

Before beginning your Oracle Identity Management enterprise deployment, review the flow chart in Figure 2–4, "Flow Chart of the Oracle Identity Management Enterprise Deployment Process". This flow chart illustrates the high-level process for completing the enterprise deployment documented in this guide. Table 2–4 describes the steps in the flow chart and directs you to the appropriate section or chapter for each step.

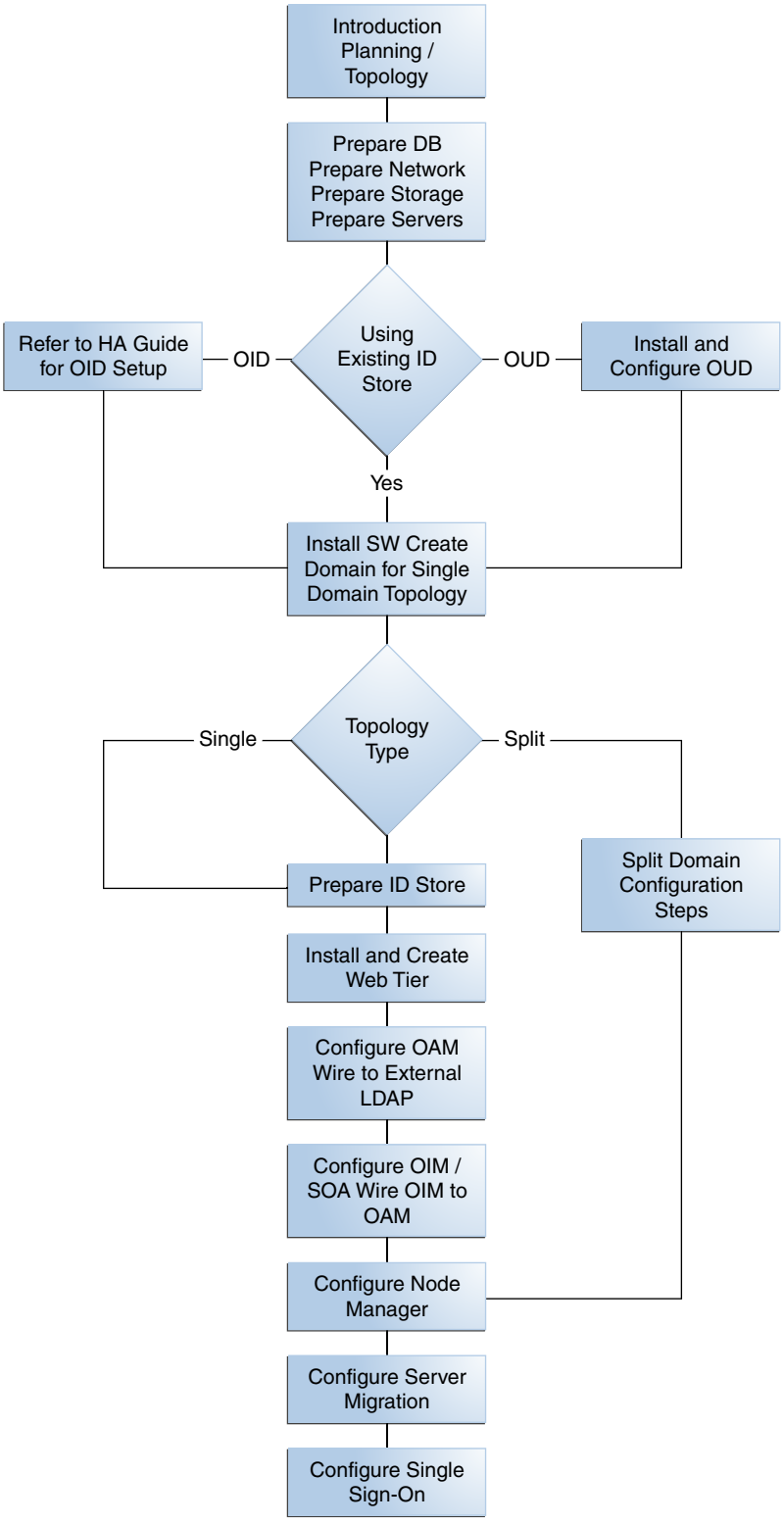
This section covers the following topics:

- Section 2.5.1, "Flow Chart of the Oracle Identity Management Enterprise Deployment Process"
- Section 2.5.2, "Steps in the Oracle Identity Management Enterprise Deployment Process"

2.5.1 Flow Chart of the Oracle Identity Management Enterprise Deployment Process

Figure 2–4, "Flow Chart of the Oracle Identity Management Enterprise Deployment Process" provides a flow chart of the Oracle Identity Management enterprise deployment process. Review this chart to become familiar with the steps that you must follow, based on the existing environment.

Figure 2–4 Flow Chart of the Oracle Identity Management Enterprise Deployment Process



2.5.2 Steps in the Oracle Identity Management Enterprise Deployment Process

Table 2–4 describes each of the steps in the enterprise deployment process flow chart for Oracle Identity Management, shown in Figure 2–4. The table also provides information on where to obtain more information about each step in the process.

Table 2–4 Steps in the Oracle Identity Management Enterprise Deployment Process

Step	Description	More Information
Prepare your Network for Enterprise Deployment	To prepare your network for an enterprise deployment, understand concepts, such as virtual server names and IPs and virtual IPs, and configure your load balancer by defining virtual host names.	Chapter 3, "Preparing the Network for an Enterprise Deployment"
Prepare your Storage for Enterprise Deployment	To prepare your file system for an enterprise deployment, review the terminology for directories and directory environment variables, and configure shared storage.	Chapter 4, "Preparing Storage for an Enterprise Deployment"
Prepare your Servers for an Enterprise Deployment	To prepare your servers for an enterprise deployment, ensure that your servers meet hardware and software requirements, enable Unicode support and Virtual IP Addresses, mount shared storage, configure users and groups, and, if necessary, install software onto multihomed systems.	Chapter 5, "Preparing the Servers for an Enterprise Deployment"
Prepare your Database for Enterprise Deployment	To prepare your database for an enterprise deployment, review database requirements, create database services, load the metadata repository, in the Oracle RAC database, configure Identity Management schemas for transactional recovery privileges, and back up the database.	Chapter 6, "Preparing the Database for an Enterprise Deployment"
Extend the Domain for Oracle Unified Directory?	Extend the existing WebLogic domain by running the Configuration Wizard to configure Oracle Unified Directory.	Chapter 7, "Installing and Configuring Oracle Unified Directory."
Create a Domain	Run the Configuration Wizard to create the domains (OIMDomain) and include OAM and OIM.	Chapter 8, "Creating a Domain for an Enterprise Deployment"
Prepare Identity and Policy Stores	Prepare the Identity and Policy Stores in an Oracle Identity Management enterprise deployment.	Chapter 9, "Preparing Identity Stores"
Configure the Web Tier	Configure the Oracle Web Tier by associating the Oracle Web tier with the Oracle WebLogic Domain, Configuring Oracle HTTP Server with the load balancer, and configuring virtual host names.	Chapter 10, "Installing and Configuring Oracle Web Tier for an Enterprise Deployment"
Extend the Domain for Oracle Access Management?	Run the Configuration Wizard again and extend the domain to include Oracle Access Management.	Chapter 11, "Extending the Domain to Include Oracle Access Management"

Table 2–4 (Cont.) Steps in the Oracle Identity Management Enterprise Deployment Process

Step	Description	More Information
Extend the Domain for Oracle Identity Manager?	Run the Configuration Wizard again and extend the domain to include Oracle Identity Manager.	Chapter 12, "Extending the Domain to Include Oracle Identity Manager"
Set up Node Manager	Set up Node manager by enabling host name verification, starting Node Manager, and configuring WebLogic Servers to use custom keystores.	Chapter 13, "Setting Up Node Manager for an Enterprise Deployment"
Configure Server Migration	Configure server migration for the WLS_OIM1, WLS_SOA1, WLS_OIM2, and WLS_SOA2 Managed Servers. The WLS_OIM1 and WLS_SOA1 Managed Server are configured to restart on IDMHOST2 should a failure occur. The WLS_OIM2 and WLS_SOA2 Managed Servers are configured to restart on IDMHOST1 should a failure occur.	Chapter 14, "Configuring Server Migration for an Enterprise Deployment"
Configure Single Sign-on for Administration Consoles in an Enterprise Deployment	Configure single sign-on (SSO) for administration consoles in an Identity Management Enterprise deployment.	Chapter 15, "Configuring Single Sign-on for Administration Consoles in an Enterprise Deployment"

3

Preparing the Network for an Enterprise Deployment

This chapter describes the prerequisites for the Oracle Identity Management Infrastructure enterprise deployment topologies.

This chapter includes the following topics:

3.1 Overview of Preparing the Network for an Enterprise Deployment

You must configure several virtual servers and associated ports on the load balancer for different types of network traffic and monitoring. These virtual servers should be configured to the appropriate real hosts and ports for the services running. Also, the load balancer should be configured to monitor the real host and ports for availability so that the traffic to these 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.

3.2 Planning Your Network

As shown in the deployment topology figures in Section 2.2, "Understanding the Topologies," each deployment is spread across multiple zones. A zone is a means of restricting access to components of your infrastructure to those that actually need it. In the examples in this guide, two zones are shown.

- The public zone–This is where the outside world gains access to your systems. You place into this zone only those components that the outside world must access, such as the Load Balancers and Web Tiers. If users from the outside world attempts to access any servers or services below this zone, they are prevented from doing so by firewalls.

  The public zone is configured so that the servers in this zone can interact with the application servers in the private zone.

- The intranet zone–This is where you place servers that contain core services, such as databases. These services are very tightly controlled by the organization as they contain the most sensitive data.

By using this approach, you restrict access to information to only those components that require it. This approach is useful where you have users coming in from outside of your organization. If, instead of an extranet, you are setting up an intranet, where all communication is from trusted sources, then you might reasonably decide to do away with one or more of the zones.

3.3 About Virtual Server Names Used by the Topologies

Virtual server names are configured on your load balancer. A load balancer can be configured so that traffic received on a given IP Address is distributed to a pool of dedicated servers associated with that IP address.

This load balancer IP address is associated with a name, known as a virtual server name, which is defined in DNS. The load balancer includes this name in HTTP headers so that it can be distinguished by Oracle HTTP Server.

This section contains the following topics:

- Section 3.3.1, "Virtual Host Names"
- Section 3.3.2, "Virtual Server names"

3.3.1 Virtual Host Names

The load balancer is configured with a number of virtual host names, depending on the access required DNS is set up in such a way that these virtual host names are accessible in the areas where they are used. For example:

Public communication is configured using a virtual host which is resolvable both inside and outside of the organization.

Interprocess communication is configured using a virtual host which is only resolvable in the private zone.

Administration access is resolvable only inside the organization.

3.3.2 Virtual Server names

The Identity Management enterprise topologies use the following virtual server names:

- IDSTORE.mycompany.com
- ADMIN.mycompany.com
- IDMINTERNAL.mycompany.com
- SSO.mycompany.com

Ensure that the virtual server names are associated with IP addresses and are part of your DNS. The computers on which Oracle Fusion Middleware is running must be able to resolve these virtual server names.

You define the virtual server names on the load balancer using the procedure in Section 3.4, "Configuring the Load Balancers"

The rest of this guide assumes that the deployment is one of those shown in Chapter 2, "Introduction and Planning."

3.3.2.1 IDSTORE.mycompany.com

- This virtual server is enabled on LBR2. It acts as the access point for all Identity Store LDAP traffic. Traffic to both the SSL and non-SSL is configured. The clients access this service using the address `IDSTORE.mycompany.com:636` for SSL and `IDSTORE.mycompany.com:389` for non-SSL.
- If your Identity Store is accessed through Oracle Virtual Directory, monitor the heartbeat of the Oracle Virtual Directory processes on OVDHOST1 and OVDHOST2. If an Oracle Virtual Directory process stops, the load balancer must continue to route the LDAP traffic to a surviving Oracle Virtual Directory instance.
- If your Identity Store is in Oracle Internet Directory and is accessed directly, monitor the heartbeat of the Oracle Internet Directory processes on the Oracle Internet Directory Hosts. If an Oracle Internet Directory process stops, the load balancer must continue to route the LDAP traffic to a surviving Oracle Internet Directory instance.
- If your Identity Store is in Oracle Unified Directory and is accessed directly, monitor the heartbeat of the Oracle Unified Directory processes. If an Oracle Unified Directory process stops, the load balancer must continue to route the LDAP traffic to a surviving Oracle Unified Directory instance.
- If your identity store is in Oracle Unified Directory, this virtual server directs traffic received on port 389 *(LDAP_LBR_PORT)* to each of the Oracle Unified Directory instances on port 1389 (*LDAP_DIR_PORT*).
- If your identity store is in Oracle Unified Directory, this virtual server directs traffic received on port 636 (*LDAP_LBR_SSL_PORT*) to each of the Oracle Unified Directory instances on port 1636 *(LDAP_DIR_SSL_PORT*).
- If your identity store is in Oracle Internet Directory, this virtual server directs traffic received on port 389 *(LDAP_LBR_PORT)* to each of the Oracle Internet Directory instances on port 3060.
- If your identity store is in Oracle Internet Directory, this virtual server directs traffic received on port 636 (*LDAP_LBR_SSL_PORT*) to each of the Oracle Internet Directory instances on port 3131.
- If your identity store is in Oracle Virtual Directory, this virtual server directs traffic received on port 389 *(LDAP_LBR_PORT)* to each of the Oracle Virtual Directory instances on port 6501.
- If your identity store is in Oracle Virtual Directory, this virtual server directs traffic received on port 636 (*LDAP_LBR_SSL_PORT*) to each of the Oracle Virtual Directory instances on port 7501.

3.3.2.2 ADMIN.mycompany.com

- This virtual server is enabled on LBR1. It acts as the access point for all internal HTTP traffic that gets directed to the administration services. The incoming traffic from clients is non-SSL enabled. Thus, the clients access this service using the address `ADMIN.mycompany.com:80` and in turn forward these to port 7777 (*OHS_PORT*) on WEBHOST1 and WEBHOST2. The services accessed on this virtual host include the WebLogic Administration Server Console, Oracle Enterprise Manager Fusion Middleware Control, Oracle Authorization Policy Manager, and Oracle Directory Services Manager.
- 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 `ADMIN.mycompany.com` virtual host.

3.3.2.3 IDMINTERNAL.mycompany.com

- This virtual server is enabled on LBR1. The incoming traffic from clients is non-SSL enabled. Thus, the clients access this service using the address `IDMINTERNAL.mycompany.com:80` and in turn forward these to port `7777` (*OHS_PORT*) on WEBHOST1 and WEBHOST2. The SOA Managed servers access this virtual host to callback Oracle Identity Manager web services
- Create rules in the firewall to block outside traffic from accessing this virtual host. Only traffic inside the DMZ should be able to access these URLs on the `IDMINTERNAL.mycompany.com` virtual host.

3.3.2.4 SSO.mycompany.com

- This is the virtual name which fronts all Identity Management components, including Oracle Access Management and Oracle Identity Manager.
- This virtual server is enabled on LBR1. It acts as the access point for all HTTP traffic that gets directed to the single sign on services. The incoming traffic from clients is SSL enabled. Thus, the clients access this service using the address `SSO.mycompany.com:443` and in turn forward these to port 7777 (*OHS_PORT*) on WEBHOST1 and WEBHOST2. All the single sign on enabled protected resources are accessed on this virtual host.
- Configure this virtual server in the load balancer with both port 80 (*HTTP_PORT*) and port 443 (*HTTP_SSL_PORT)*.
- This virtual host must be configured to preserve the client IP address for a request. In some load balancers, you configure this by enabling the load balancer to insert the original client IP address of a request in an X-Forwarded-For HTTP header.

3.4 Configuring the Load Balancers

Several virtual servers and associated ports must be configured on the load balancer for different types of network traffic and monitoring. These should be configured to the appropriate real hosts and ports for the services running. Also, the load balancer should be configured to monitor the real host and ports for availability so that the traffic to these 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.

There are two load balancer devices in the recommended topologies. One load balancer is set up for external HTTP traffic and the other load balancer is set up for internal LDAP traffic. A deployment may choose to have a single load balancer device due to a variety of reasons. While this is supported, the deployment should consider the security implications of doing this and if found appropriate, open up the relevant firewall ports to allow traffic across the various zones. It is worth noting that in either case, it is highly recommended to deploy a given load balancer device in fault tolerant mode.

This section contains the following topics:

- Section 3.4.1, "Load Balancer Requirements"
- Section 3.4.2, "Load Balancer Configuration Procedures"
- Section 3.4.3, "Load Balancer Configuration"

3.4.1 Load Balancer Requirements

The enterprise topologies use an external load balancer. This external load balancer must have the following features:

- Ability to load-balance traffic to a pool of real servers through a virtual host name: Clients access services 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.
- Monitoring of ports (HTTP and HTTPS).
- Virtual servers and port configuration: Ability to configure virtual server names and ports on your external load balancer, and 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 WebLogic Clusters, the load balancer must 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.
- Resource monitoring / port monitoring / process failure detection: The load balancer must be able to detect service and node failures (through notification or some other means) and to stop directing non-Oracle Net traffic to the failed node. If your external load balancer has the ability to automatically detect failures, you should use it.
- Fault tolerant mode: It is highly recommended that you configure the load balancer to be in fault-tolerant mode.
- Other: It is highly recommended that you configure the load balancer virtual server to return immediately to the calling client when the back-end 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.
- SSL acceleration, which refers to off loading the public-key encryption algorithms involved in SSL transactions to a hardware accelerator. This feature is recommended, but not required.
- Ability to terminate SSL requests at the load balancer and forward traffic to the backend real servers using the equivalent non-SSL protocol. For example, the load balancer must be able to forward HTTPS requests as HTTP. This feature is sometimes called "SSL termination." It is required for this Enterprise Deployment.
- Ability to Preserve the Client IP Addresses: The Load Balancer must have the capability to insert the original client IP address of a request in an X-Forwarded-For HTTP header to preserve the Client IP Address.
- Ability to add `WL-Proxy-SSL: true` to the HTTP Request Header. Some load balancers do this automatically.

3.4.2 Load Balancer Configuration Procedures

The procedures for configuring a load balancer differ, depending on the specific type of load balancer. Refer to the vendor supplied documentation for actual steps. The following steps outline the general configuration flow:

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 example, for load balancing between the web hosts you create a pool of servers which would direct requests to hosts WEBHOST1 and WEBHOST2 on port *7777* (*OHS_PORT*).
2. Create rules to determine whether or not a given host and service is available and assign it to the pool of servers described in Step 1.
3. Create a Virtual Server on the load balancer. This is the address and port that receives requests used by the application. For example, to load balance Web Tier requests you would create a virtual host for `SSO.mycompany.com:80`.
4. If your load balancer supports it, specify whether or not the virtual server is available internally, externally or both. Ensure that internal addresses are only resolvable from inside the network.
5. Configure SSL Termination, if applicable, for the virtual server.
6. Assign the Pool of servers created in Step 1 to the virtual server.
7. Tune the time out settings as listed in Table 3–3, " Ports Used in the Oracle Identity Management Enterprise Deployment Topologies". This includes time to detect whether a service is down.

3.4.3 Load Balancer Configuration

For an Identity Management deployment, configure your load balancer as shown in Table 3–1.

Table 3–1 Load Balancer Configuration

Virtual Host	Server Pool	Protocol	SSL Termination	External	Other Required Configuration/Comments
`SSO.mycompany.com:80`	`WEBHOST1.mycompany.com:7777` `WEBHOST2.mycompany.com:7777`	HTTP	No	Yes	Identity Management requires that the following be added to the HTTP header: `Header Name: IS_SSL`[1] `Header Value: ssl`

Table 3–1 (Cont.) Load Balancer Configuration

Virtual Host	Server Pool	Protocol	SSL Termination	External	Other Required Configuration/Comments
SSO.mycompany.com:443	WEBHOST1.mycompany.com:7777 WEBHOST2.mycompany.com:7777	HTTP	Yes	Yes	Identity Management requires that the following be added to the HTTP header: Header Name: IS_SSL Header Value: ssl
ADMIN.mycompany.com:80	WEBHOST1.mycompany.com:7777 WEBHOST2.mycompany.com:7777	HTTP	No	No	
IDMINTERNAL.mycompany.com:80	WEBHOST1.mycompany.com:7777 WEBHOST2.mycompany.com:7777	HTTP	No	No	
IDSTORE.mycompany.com:389	IDMHOST1.mycompany.com:1389 IDMHOST2.mycompany.com:1389	LDAP	No	No	Only required if Identity Management users are stored in an Oracle Unified Directory.
IDSTORE.mycompany.com:636	IDMHOST1.mycompany.com:1636 IDMHOST2.mycompany.com:1636	LDAP	No	No	Only required if Identity Management users are stored in an Oracle Unified Directory.
IDSTORE.mycompany.com:389	OIDHOST1.mycompany.com:3060 OIDHOST2.mycompany.com:3060	LDAP	No	No	This will generally have been setup as part of your OID installation. It is used to load balance LDAP calls to OID.

Table 3–1 (Cont.) Load Balancer Configuration

Virtual Host	Server Pool	Protocol	SSL Termination	External	Other Required Configuration/Comments
IDSTORE.mycompany.com:636	OIDHOST1.mycompany.com:3131 OIDHOST2.mycompany.com:3131	LDAP	No	No	This will generally have been setup as part of your OID installation. It is used to load balance LDAP calls to OID.
IDSTORE.mycompany.com:389	OVDHOST1.mycompany.com:6501 OVDHOST2.mycompany.com:6501	LDAP	No	No	This will generally have been setup as part of your OVD installation. It is used to load balance LDAP calls to OVD.
IDSTORE.mycompany.com:636	OVDHOST1.mycompany.com:7501 OVDHOST2.mycompany.com:7501	LDAP	No	No	This will generally have been setup as part of your OVD installation. It is used to load balance LDAP calls to OVD.

[1] For information about configuring IS_SSL, see "About User Defined WebGate Parameters" in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*.

3.5 About IP Addresses and Virtual IP Addresses

A virtual IP address is an unused IP Address which belongs to the same subnet as the host's primary IP address. It is assigned to a host manually and Oracle WebLogic Managed servers are configured to listen on this IP Address. In the event of the failure of the node where the IP address is assigned, the IP address is assigned to another node in the same subnet, so that the new node can take responsibility for running the managed servers assigned to it.

The following is a list of the Virtual IP addresses required by Oracle Identity Management:

- ADMINVHN.mycompany.com

  In Enterprise deployments the WebLogic Administration Server must be able to continue processing requests after the host it is residing on fails. A virtual IP address should be provisioned in the application tier so that it can be bound to a network interface on any host in the application tier. The WebLogic Administration Server is configured later to listen on this virtual IP address, as discussed later in this manual. The virtual IP address fails over along with the Administration Server from IDMHOST1 to IDMHOST2, or vice versa.

- SOAHOSTXVHN.mycompany.com

One virtual IP address is required for each SOA managed server. This enables the servers to participate in Server migration.

Provision a virtual IP address in the application tier so that it can be bound to a network interface on any host in the application tier.

- OIMHOSTXVHN.mycompany.com

  One virtual IP Address is required for each Oracle Identity Manager managed server. This enables the servers to participate in Server migration.

  Provision a virtual IP address in the application tier so that it can be bound to a network interface on any host in the application tier.

Configure the Administration Server and the managed servers to listen on different virtual IPs and physical IPs as illustrated in Figure 3–1.

Figure 3–1 IP Addresses and VIP Addresses

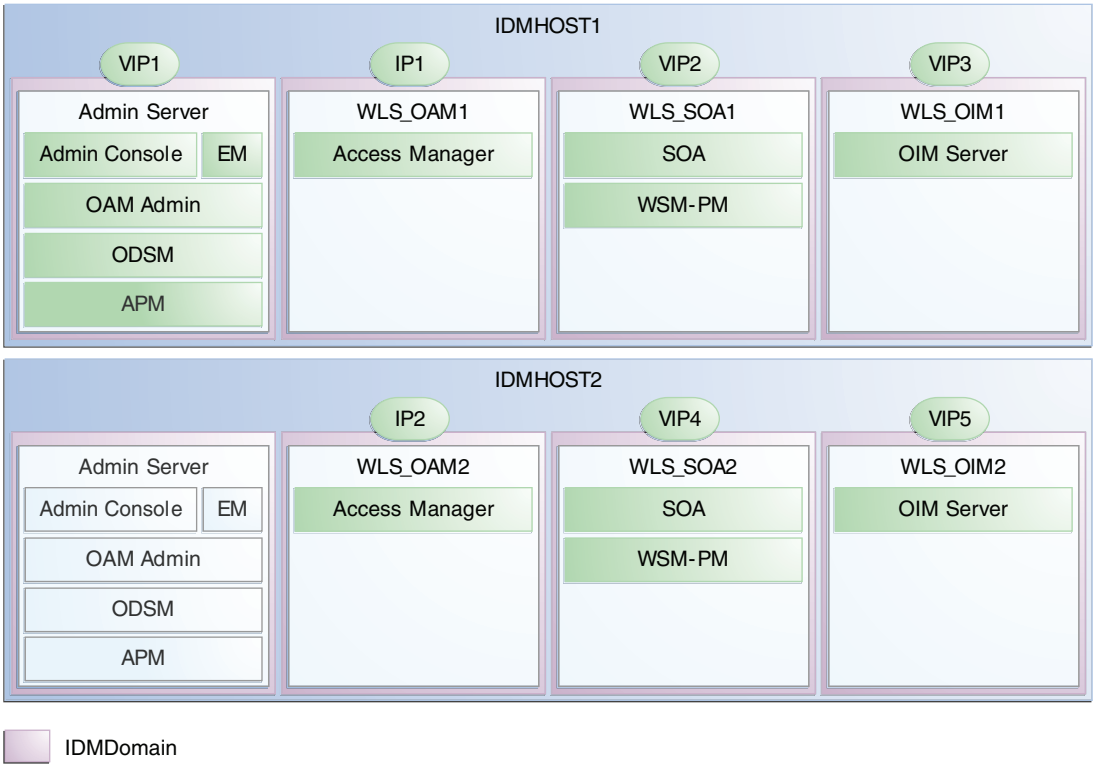


Table 3–2 provides descriptions of the various virtual hosts.

Table 3–2 VIP Addresses and Virtual Hosts

Virtual IP	VIP Maps to...	Description
VIP1	ADMINVHN	ADMINVHN is the virtual host name that is the listen address for the Administration Server and fails over with manual failover of the Administration Server. It is enabled on the node where the Administration Server process is running (IDMHOST1 by default).

Table 3–2 (Cont.) VIP Addresses and Virtual Hosts

Virtual IP	VIP Maps to...	Description
VIP2	SOAHOST1VHN	SOAHOST1VHN is the virtual host name that maps to the listen address for WLS_SOA1 and fails over with server migration of this managed server. It is enabled on the node where WLS_SOA1 process is running (IDMHOST1 by default).
VIP3	OIMHOST1VHN	OIMHOST1VHN is the virtual host name that maps to the listen address for the WLS_OIM1 server and fails over with server migration of this server. It is enabled in the node where the WLS_OIM1 process us running (IDMHOST1 by default)
VIP4	SOAHOST2VHN	SOAHOST2VHN is the virtual host name that maps to the listen address for WLS_SOA2 and fails over with server migration of this managed server. It is enabled on the node where WLS_SOA2 process is running (IDMHOST2 by default).
VIP5	OIMHOST2VHN	OIMHOST2VHN is the virtual host name that maps to the listen address for the WLS_OIM2 server and fails over with server migration of this server. It is enabled in the node where the WLS_OIM2 process us running (IDMHOST2 by default)

3.6 About Firewalls and Ports

Many Oracle Fusion Middleware components and services use ports. As an administrator, you must know the port numbers used by these services, and to ensure that the same port number is not used by two services on a host.

Most port numbers are assigned after installation. You can use different port numbers if you want to. The port numbers shown in Table 3–3 are examples that are used throughout this guide for consistency. If you use different port numbers, you must substitute those values for the values in the table wherever they are used.

Table 3–3 lists the ports used in the Oracle Identity Management topologies, including the ports that you must open on the firewalls in the topologies.

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 database tier.

You can use the Port Mapping worksheet in Appendix B to help you keep track of your port usage.

Table 3–3 Ports Used in the Oracle Identity Management Enterprise Deployment Topologies

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Timeout
Browser request	FW0	80	HTTP / Load balancer	Both	Timeout depends on all HTML content and the type of process model used for Oracle Identity Management.
Browser request	FW0	443	HTTPS / Load balancer	Both	Timeout depends on all HTML content and the type of process model used for Oracle Identity Management.
Browser request	FW1	80	HTTPS / Load Balancer	Outbound (for intranet clients)	Timeout depends on all HTML content and the type of process model used for IDM.
Browser request	FW1	443	HTTPS / Load Balancer	Outbound (for intranet clients)	Timeout depends on all HTML content and the type of process model used for IDM.
Load balancer to Oracle HTTP Server	n/a	7777	HTTP	n/a	See Section 3.4, "Configuring the Load Balancers."
OHS registration with Administration Server	FW1	7001	HTTP/t3	Inbound	Set the timeout to a short period (5-10 seconds).
Oracle WebLogic Administration Server access from web tier	FW1	7001	HTTP / Oracle HTTP Server and Administration Server	Inbound	N/A
Enterprise Manager Agent - web tier to Enterprise Manager	FW1	5160	HTTP / Enterprise Manager Agent and Enterprise Manager	Both	N/A
Oracle HTTP Server to WLS_OAM	FW1	14100	HTTP / Oracle HTTP Server to WebLogic Server	Inbound	Timeout depends on the `mod_weblogic` parameters used.
Oracle HTTP Server WLS_OIM	FW1	14000	HTTP / Oracle HTTP Server to WebLogic Server	Inbound	Timeout depends on the mod_ weblogic parameters used
Oracle HTTP Server WLS_SOA	FW1	8001	HTTP / Oracle HTTP Server to WebLogic Server	Both	Timeout depends on the mod_ weblogic parameters used
Oracle HTTP Server management by Administration Server	FW1	OPMN remote port (6701) and OHS Admin Port (7779)	TCP and HTTP, respectively	Outbound	Set the timeout to a short period, such as 5-10 seconds.
OAM Server	FW1	5575	OAP	Both	N/A
Access Manager Coherence port	FW1	9095	TCMP	Both	N/A
Oracle Coherence Port	FW1	8000 - 8088	TCMP	Both	N/A
IDMDomain Oracle WebLogic Administration Server access from directory tier	FW2	7001	HTTP / Oracle Internet Directory, Oracle Virtual Directory, and Administration Server	Outbound	N/A

Table 3–3 (Cont.) Ports Used in the Oracle Identity Management Enterprise Deployment Topologies

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Timeout
Enterprise Manager Agent - directory tier to Enterprise Manage, if directory tier is implemented.	FW2	5160	HTTP / Enterprise Manager Agent and Enterprise Manager	Both	N/A
Oracle HTTP Server to Administration Server	FW2	OPMN remote port	HTTP / Administration Server to OPMN	Inbound	N/A
Application Tier to Database Listener	FW2	1521	SQL*Net	Both	Timeout depends on all database content and on the type of process model used for Oracle Identity Management.
Oracle Notification Server (ONS)	FW2	6200	ONS	Both	Required for Gridlink. An ONS server runs on each database server.
LDAP Port	FW2	Depends on directory	LDAP	Inbound	Ideally, these connections should be configured not to time out.
LDAP SSL Port	FW2	Depends on directory	LDAP SSL	Inbound	Ideally, these connections should be configured not to time out.
Node Manager	N/A	5556	TCP/IP	N/A	N/A

Note: Additional ports might need to be opened across the firewalls to enable applications in external domains, such as SOA or WebCenter Portal domains, to authenticate against this Identity Management domain.

3.7 Managing Access Manager Communication Protocol

This section discusses Oracle Access Protocol (OAP) and provides an overview of a user request.

This section contains the following topics:

- Section 3.7.1, "Access Manager Protocols"
- Section 3.7.2, "Overview of Integration Requests"
- Section 3.7.3, "Overview of User Request"
- Section 3.7.4, "About the Unicast Requirement for Communication"

3.7.1 Access Manager Protocols

Oracle Access Protocol (OAP) enables communication between Access System components (for example, OAM Server, WebGate) during user authentication and authorization. This protocol was formerly known as NetPoint Access Protocol (NAP) or COREid Access Protocol.

3.7.2 Overview of Integration Requests

Oracle Access Management Access Manager is responsible for creating sessions for users. When Access Manager is integrated with another Identity Management

component, such as Oracle Identity Manager, authentication is delegated to that component.

A typical request flow is as follows:

1. The user tries to access a resource for the first time.
2. WebGate intercepts the request and detects that the user is not authenticated.
3. Access Manager credential collector is invoked and the user enters a user name and password in response to a prompt. Access Manager knows that password policy requires the password to be changed at first login, so the user's browser is redirected to Oracle Identity Manager.
4. The user is prompted to change password and set up challenge questions.
5. At this point, Oracle Identity Manager has authenticated the user using the newly entered password. Oracle Identity Manager creates a TAP request to say that Access Manager can create a session for the user. That is, the user will not be expected to log in again. This is achieved by adding a token to the user's browser that Access Manager can read.

   The TAP request to Access Manager will include such things as:

   - Where the Access Manager servers are located.
   - What web gate profile to use.
   - WebGate profile password.
   - Certificates, if Access Manager is working in simple or cert mode.

3.7.3 Overview of User Request

The request flow when a user requests access is as follows:

1. The user requests access to a protected resource over HTTP or HTTPS.
2. The WebGate intercepts the request.
3. The WebGate forwards the request to the OAM Server over Oracle Access Protocol to determine if the resource is protected, how, and whether the user is authenticated (if not, there is a challenge).
4. The OAM Server checks the directory server for credentials such as a user ID and password, sends the information back to WebGate over Oracle Access Protocol, and generates an encrypted cookie to authenticate the user.
5. Following authentication, the WebGate prompts the OAM Server over Oracle Access Protocol and the OAM Server looks up the appropriate security policies, compares them to the user's identity, and determines the user's level of authorization.

   - If the access policy is valid, the user is allowed to access the desired content and/or applications.
   - If the policy is false, the user is denied access and redirected to another URL determined by the organization's administrator.

3.7.4 About the Unicast Requirement for Communication

Oracle recommends that the nodes in the topology communicate using unicast communication. Unlike multicast communication, unicast does not require

cross-network configuration. Using unicast avoids network errors due to multicast address conflicts.

In unicast messaging mode, the default listening port of the server is used if no channel is configured. Cluster members communicate to the group leader when they need to send a broadcast message which is usually the heartbeat message. When the cluster members detect the failure of a group leader, the next oldest member becomes the group leader. The frequency of communication in unicast mode is similar to the frequency of sending messages on multicast port.

The following considerations apply when using unicast to handle cluster communications:

- All members of a WebLogic cluster must use the same message type. Mixing multicast and unicast messaging is not allowed.
- Individual cluster members cannot override the cluster messaging type.
- The entire cluster must be shut down and restarted to change the message modes from unicast to multicast or from multicast to unicast.
- JMS topics configured for multicasting can access WebLogic clusters configured for unicast because a JMS topic publishes messages on its own multicast address that is independent of the cluster address. However, the following considerations apply:
  - The router hardware configurations that allow unicast clusters may not allow JMS multicast subscribers to work.
  - JMS multicast subscribers need to be in a network hardware configuration that allows multicast accessibility. (That is, JMS subscribers must be in a multicast-enabled network to access multicast topics.)

4

Preparing Storage for an Enterprise Deployment

This chapter describes how to prepare storage for an Oracle Identity Management enterprise deployment.

The storage model described in this guide was chosen for maximum availability, best isolation of components, symmetry in the configuration, and facilitation of backup and disaster recovery. The rest of the guide uses a directory structure and directory terminology based on this storage model. Other directory layouts are possible and supported.

This chapter contains the following topics:

4.1 Overview of Preparing Storage for Enterprise Deployment

It is important to set up your storage in a way that makes the enterprise deployment easier to understand, configure, and manage. Oracle recommends setting up your storage according to information in this chapter. The terminology defined in this chapter is used in diagrams and procedures throughout the guide.

Use this chapter as a reference to help understand the directory variables used in the installation and configuration procedures. Other directory layouts are possible and supported, but the model adopted in this guide is chosen 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.

4.2 Terminology for Directories and Directory Variables

This section describes the directory variables used throughout this guide for configuring the Oracle Identity Management enterprise deployment. You are not required to set these as environment variables. The following directory variables are used to describe the directories installed and configured in the guide:

- **ORACLE_BASE**: This environment variable and related directory path refers to the base directory under which Oracle products are installed. For example: `/u01/oracle`

- **MW_HOME**: This variable and related directory path refers to the location where Oracle Fusion Middleware resides. A *MW_HOME* has a *WL_HOME*, an *ORACLE_COMMON_HOME* and one or more *ORACLE_HOMEs*. An example of a typical *MW_HOME* is:

  `/u01/oracle/products/access`

  There is a different *MW_HOME* for each domain.

  In this guide, this value might be preceded by a product suite abbreviation, for example: *IAM_MW_HOME*, *OIM_MW_HOME*, *WEB_MW_HOME*.

- **WL_HOME**: This variable and related directory path contains installed files necessary to host a WebLogic Server, for example *MW_HOME*`/wlserver_10.3`. The `WL_HOME` directory is a peer of Oracle home directory and resides within the *MW_HOME*.

- **ORACLE_HOME**: This variable points to the location where an Oracle Fusion Middleware product, such as Oracle HTTP Server, Oracle SOA Suite, or Oracle Internet Directory is installed and the binaries of that product are being used in a current procedure. In this guide, this value might be preceded by a product suite abbreviation, for example: `IAM_ORACLE_HOME`, `OIM_ORACLE_HOME`, `WEB_ORACLE_HOME`. For more information about homes, see Table 2–3, " Summary of Homes".

- **ORACLE_COMMON_HOME**: This variable and related directory path refer to the location where the Oracle Fusion Middleware Common Java Required Files (JRF) Libraries and Oracle Fusion Middleware Enterprise Manager Libraries are installed. An example is: *MW_HOME*`/oracle_common`

- **Domain directory**: This path refers to the file system location where the Oracle WebLogic domain information (configuration artifacts) is stored. Different WebLogic Servers can use different domain directories even when in the same node as described Section 4.4, "About Recommended Locations for the Different Directories."

- **ORACLE_INSTANCE**: An Oracle instance contains one or more system components, such as Oracle Web Cache, Oracle HTTP Server, or Oracle Internet Directory. An Oracle instance directory contains updatable files, such as configuration files, log files, and temporary files. An example is: `/u02/private/oracle/config/instances/web1`

  In this guide, this value might be preceded by a product suite abbreviation, such as `WEB_ORACLE_INSTANCE`.

- **JAVA_HOME**: This is the location where Oracle JRockit is installed.

- **ASERVER_HOME**: This is the primary location of the domain configuration. A typical example is: `/u01/oracle/config/domains/`*domain_name*

- **MSERVER_HOME**: This is a copy of the domain configuration used to start and stop managed servers. A typical example is: `/u02/private/oracle/config/domains/`*domain_name*

4.3 About File Systems

After you create the partitions on your storage, you must place file systems on the partitions so that you can store the Oracle files. For local or direct attached shared storage, the file system type is most likely the default type for your operating system, for example: EXT3 for Linux.

If your shared storage is on network attached storage (NAS), which is accessed by two or more hosts either exclusively or concurrently, then you must use a supported

clustered file system such as NFS version 3 or 4. Such file systems provide conflict resolution and locking capabilities.

4.4 About Recommended Locations for the Different Directories

This section contains the following topics:

- Section 4.4.1, "Recommendations for Binary (Middleware Home) Directories"
- Section 4.4.2, "Recommendations for Domain Configuration Files"
- Section 4.4.3, "Shared Storage Recommendations for JMS File Stores and Transaction Logs"
- Section 4.4.4, "Recommended Directory Locations"

4.4.1 Recommendations for Binary (Middleware Home) Directories

The following sections describe guidelines for using shared storage for your Oracle Fusion Middleware middleware home directories:

- Section 4.4.1.1, "About the Binary (Middleware Home) Directories"
- Section 4.4.1.2, "About Sharing a Single Middleware Home for Multiple Domains"
- Section 4.4.1.3, "About Using Redundant Binary (Middleware Home) Directories"

4.4.1.1 About the Binary (Middleware Home) Directories

When you install any Oracle Fusion Middleware product, you install the product binaries into a Middleware home. The binary files installed in the Middleware home are read-only and remain unchanged unless the Middleware home is patched or upgraded to a newer version.

In a typical production environment, the Middleware home files are saved in a separate location from the domain configuration files, which you create using the Oracle Fusion Middleware Configuration Wizard.

The Middleware home for an Oracle Fusion Middleware installation contains the binaries for Oracle WebLogic Server, the Oracle Fusion Middleware infrastructure files, and any Oracle Fusion Middleware product-specific directories.

For more information about the structure and content of an Oracle Fusion Middleware home, see *Oracle Fusion Middleware Installation Guide for Oracle Enterprise Content Management Suite*.

4.4.1.2 About Sharing a Single Middleware Home for Multiple Domains

Oracle Fusion Middleware enables you to configure multiple Oracle WebLogic Server domains from a single Middleware home. This allows you to install the Middleware home in a single location on a shared volume and reuse the Middleware home for multiple host installations.

When a Middleware home is shared by multiple servers on different hosts, there are some best practices to keep in mind. In particular, be sure that the Oracle Inventory on each host is updated for consistency and for the application of patches.

To update the oraInventory for a host and attach a Middleware home on shared storage, use the following command:

```
ORACLE_HOME/oui/bin/attachHome.sh
```

For more information about the Oracle inventory, see "Oracle Universal Installer Inventory" in the *Oracle Universal Installer Concepts Guide.*

4.4.1.3 About Using Redundant Binary (Middleware Home) Directories

For maximum availability, Oracle recommends using redundant binary installations on shared storage.

In this model, you install two identical Middleware homes for your Oracle Fusion Middleware software on two different shared volumes. You then mount one of the Middleware homes to one set of servers, and the other Middleware home to the remaining servers. Each Middleware home has the same mount point, so the Middleware home always has the same path, regardless of which Middleware home the server is using.

Should one Middleware home become corrupted or unavailable, only half your servers are affected. For additional protection, Oracle recommends that you disk mirror these volumes.

If separate volumes are not available on shared storage, Oracle recommends simulating separate volumes using different directories within the same volume and mounting these to the same mount location on the host side. Although this does not guarantee the protection that multiple volumes provide, it does allow protection from user deletions and individual file corruption.

4.4.2 Recommendations for Domain Configuration Files

The following sections describe guidelines for using shared storage for the Oracle WebLogic Server domain configuration files you create when you configure your Oracle Fusion Middleware products in an enterprise deployment:

- Section 4.4.2.1, "About Oracle WebLogic Server Administration and Managed Server Domain Configuration Files"
- Section 4.4.2.2, "Shared Storage Requirements for Administration Server Domain Configuration Files"
- Section 4.4.2.3, "Local Storage Requirements for Managed Server Domain Configuration Files"

4.4.2.1 About Oracle WebLogic Server Administration and Managed Server Domain Configuration Files

When you configure an Oracle Fusion Middleware product, you create or extend an Oracle WebLogic Server domain. Each Oracle WebLogic Server domain consists of a single Administration Server and one or more managed servers.

For more information about Oracle WebLogic Server domains, see *Oracle Fusion Middleware Understanding Domain Configuration for Oracle WebLogic Server*.

In an enterprise deployment, it is important to understand that the managed servers in a domain can be configured for active-active high availability. However, the Administration server cannot. The Administration Server is a singleton service. That is, it can be active on only one host at any given time.

ASERVER_HOME is the primary location of the domain configuration. *MSERVER_HOME* is a copy of the domain configuration that is used to start and stop managed servers. The WebLogic Administration Server automatically copies configuration changes applied to the *ASERVER_HOME* domain configuration to all those *MSERVER_HOME* configuration directories that have been registered to be part of the domain. However, the *MSERVER_HOME* directories also contain deployments and data specific to the managed servers.

For that reason, when performing backups, you must include both *ASERVER_HOME* and *MSERVER_HOME*.

4.4.2.2 Shared Storage Requirements for Administration Server Domain Configuration Files

Administration Server configuration files must reside on Shared Storage. This allows the administration server to be started on a different host should the primary host become unavailable. The directory where the administration server files is located is known as the *ASERVER_HOME* directory. This directory is located on shared storage and mounted to the Administration Server host and to each host running Oracle Identity Manager.

Managed Server configuration Files should reside on local storage to prevent performance issues associated with contention. The directory where the managed server configuration files are located is known as the *MSERVER_HOME* directory. It is highly recommended that managed server domain configuration files be placed onto local storage.

4.4.2.3 Local Storage Requirements for Managed Server Domain Configuration Files

If you must use shared storage, it is recommended that you create a storage partition for each node and mount that storage exclusively to that node

The configuration steps provided for this enterprise deployment topology assume that a local domain directory for each node is used for each managed server.

Note: Oracle Unified Directory is not supported on NFS.

4.4.3 Shared Storage Recommendations for JMS File Stores and Transaction Logs

JMS file stores and JTA transaction logs must be placed on shared storage in order to ensure that they are available from multiple hosts for recovery in the case of a server failure or migration.

For more information about saving JMS and JTA information in a file store, see "Using the WebLogic Persistent Store" in *Oracle Fusion Middleware Configuring Server Environments for Oracle WebLogic Server*.

4.4.4 Recommended Directory Locations

This section describes the recommended use of shared and local storage.

This section includes the following topics:

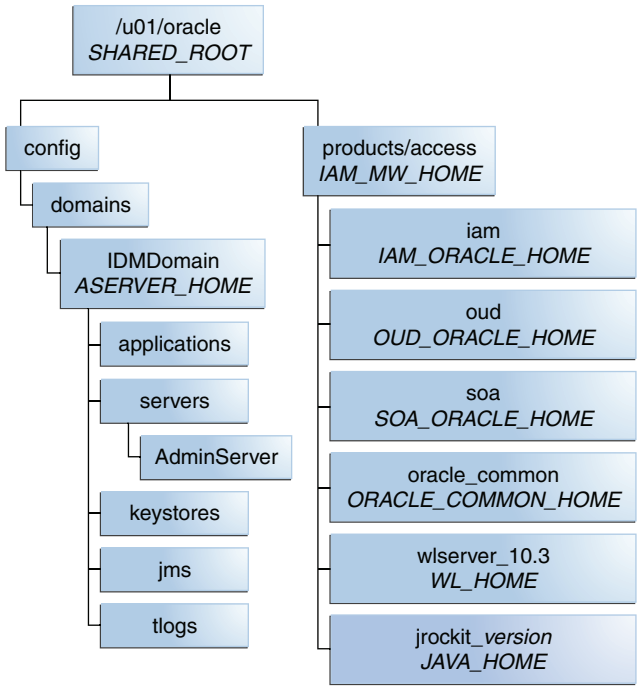
- Section 4.4.4.1, "Shared Storage"
- Section 4.4.4.2, "Local Storage"

4.4.4.1 Shared Storage

In an Enterprise Deployment, it is recommended that the volume `VOL1/OracleIDM` be created on shared storage on hosts IDMHOST1 and IDMHOST2. The mount point must be `/u01/oracle`.

You can mount shared storage either exclusively or shared.

Figure 4–1 Shared Storage



The figure shows the shared storage directory hierarchy. Under the mount point, `/u01/oracle` (*SHARED_ROOT*) are the directories `config` and `products`.

The directory `config` contains `domains`, which contains `IDMDomain` (*ASERVER_HOME*). `IDMDomain` has five subdirectories: `applications`, `servers`, `keystores`, `jms`, and `logs`. The `servers` directory has a subdirectory, `AdminServer`.

The directory `products` contains the directory `access` (*IAM_MW_HOME*), which has six subdirectories: `iam` (*IAM_ORACLE_HOME*), `oud` (*OUD_ORACLE_HOME*), `soa` (*SOA_ORACLE_HOME*), `oracle_common` (*ORACLE_COMMON_HOME*), `wlserver_10.3` (*WL_HOME*), and `jrockit_version` (*JAVA_HOME*).

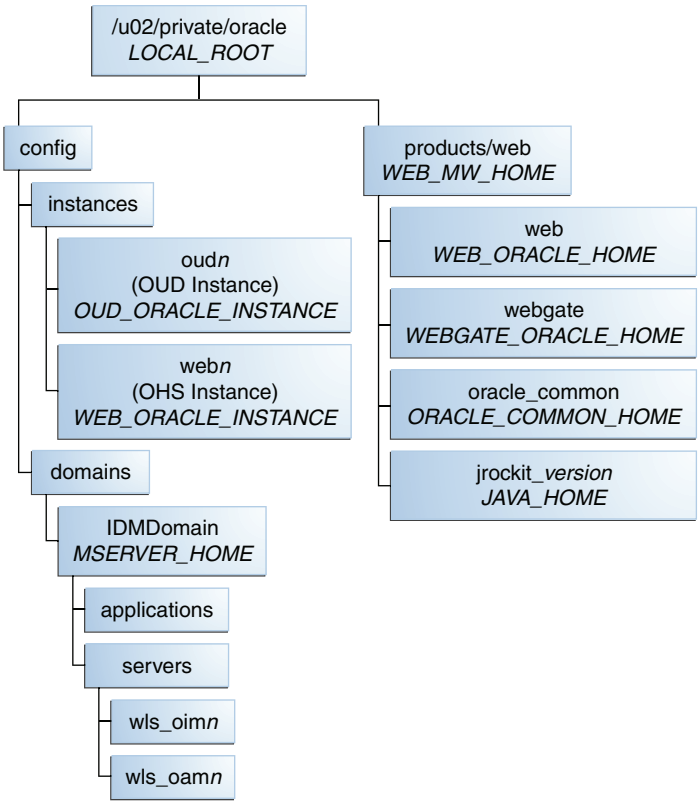
4.4.4.2 Local Storage

In an Enterprise Deployment it is recommended that the following directories be created on local storage:

Table 4–1 Local Storage Directories

Tier	Environment Variable	Directory	Hosts
Web Tier	*WEB_MW_HOME*	`/u02/private/oracle/products/web`	WEBHOST1 WEBHOST2
Web Tier	*WEB_ORACLE_INSTANCE*	`/u02/private/oracle/config/instances/web`*n*	WEBHOST1 WEBHOST2
Application Tier	*OUD_ORACLE_INSTANCE*	`/u02/private/oracle/config/instances/oud`*n*	IDMHOST1 IDMHOST2
	MSERVER_HOME	`/u02/private/oracle/config/domains/IDMDomain`	IDMHOST1 IDMHOST2

Figure 4–2 Local Storage



The figure shows the local storage directory hierarchy. The top level directory, `/u02/private/oracle` (*LOCAL_ROOT*), has two subdirectories, `config` and `products`.

The directory `config` has two subdirectories, `instances` and `domains`. The `domains` directory contains `oudn` (*OUD_ORACLE_INSTANCE*), where *n* is the OUD instance, and `webn` (*WEB_ORACLE_INSTANCE*), where *n* is the OHS Instance. The `domains` directory contains `IDMDomain` (*MSERVER_HOME*), which contains `applications` and `servers`. The `servers` directory contains `wls_oimn` and `wls_oamn`, where *n* is the OIM and OAM instance, respectively.

The `products` directory contains the `web` directory (*WEB_MW_HOME*), which has four subdirectories: `web` (*WEB_ORACLE_HOME*), `webgate` (*WEBGATE_ORACLE_HOME*), `oracle_common` (*ORACLE_COMMON_HOME*), and `jrockit_version` (*JAVA_HOME*).

Note: While it is recommended that you put *WEB_ORACLE_INSTANCE* directories onto local storage, you can use shared storage. If you use shared storage, you must ensure that the HTTP lock file is placed on discrete locations.

5

Preparing the Servers for an Enterprise Deployment

This chapter describes how to prepare the servers for an enterprise deployment.

It contains the following sections:

5.1 Overview of Preparing the Servers

Before you deploy Oracle Fusion Middleware on new hardware, you must set up the servers you plan to use so that the Oracle Software can work in an optimum fashion. Specifically, you must ensure that:

- The servers are running a certified operating system with the required software patches installed.
- You have configured the UNIX Kernel correctly.
- You have created Users and Groups to own the Oracle software.

The settings described in this chapter are only a guide. After using your Oracle software, you should use operating system utilities to tune the configuration to ensure that you are maximizing the potential of your servers.

5.2 Verifying Your Server and Operating System

Ensure that the server and operating system that you plan to use is a certified combination for the products you plan to use. Refer to Oracle Certification Matrix for details.

5.3 Meeting the Minimum Hardware Requirements

In order to use a server in an Oracle Enterprise Deployment you must verify that it meets the minimum specification described in Section 2.3, "Hardware Requirements for an Enterprise Deployment.". If you plan to use a different deployment architecture, for example, one with more or fewer components deployed on a different number of boxes, you must check *Oracle Fusion Middleware System Requirements and Specifications* to ensure that you have the minimum specification to support the products you plan to deploy on these servers.

If you are deploying to a virtual server environment, such as Oracle Exalogic, ensure that each of the virtual servers meets the minimum requirements.

Ensure that you have sufficient local disk and shared storage is configured as described in Chapter 4, "Preparing Storage for an Enterprise Deployment."

Allow sufficient swap and temporary space. Specifically:

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

5.4 Meeting Operating System Requirements

Before starting your operating provisioning you must perform the following tasks:

1. Install a certified operating system.
2. Install all necessary patches and packages as listed in the Release Notes.

This section includes the following topics:

- Section 5.4.1, "Meeting UNIX and Linux Requirements."

5.4.1 Meeting UNIX and Linux Requirements

This section includes the following topics:

- Section 5.4.1.1, "Configure Kernel Parameters."
- Section 5.4.1.2, "Setting the Open File Limit."
- Section 5.4.1.3, "Setting Shell Limits."
- Section 5.4.1.4, "Configuring Local Hosts File."

5.4.1.1 Configure Kernel Parameters

The kernel parameter and shell limit values shown below are recommended values only. For production database systems, 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 below 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 are deploying a database onto the host, you might need to modify additional kernel parameters. Refer to the 11*g* Release 2 *Oracle Grid Infrastructure Installation Guide* for your platform.

Table 5–1 UNIX Kernel Parameters

Parameter	Value
kernel.sem	256 32000 100 142
kernel.shmmax	4294967295

To set these parameters:

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

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

5.4.1.2 Setting the Open File Limit

On all UNIX operating systems, the minimum Open File Limit should be 4096.

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

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 commands below.

C shell:

```
limit descriptors
```

Bash:

```
ulimit -n
```

5.4.1.3 Setting Shell Limits

To change the shell limits, login as `root` and edit the `/etc/security/limits.conf` file.

Add the following lines:

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

For the most recent suggested values, see *Oracle Fusion Middleware System Requirements and Specifications*.

After editing the file, reboot the machine.

5.4.1.4 Configuring Local Hosts File

Before you begin the installation of the Oracle software, ensure that your local `hosts` file is formatted like this:

```
IP_Address Fully_Qualified_Name Short_Name
```

5.5 Enabling Unicode Support

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 SOA Suite technologies, for example, are based on Unicode.

If the operating system is configured to use a non-UTF-8 encoding, Oracle SOA Suite 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.

5.6 Enabling Virtual IP Addresses

The enterprise deployment requires that certain hosts, such as those running the WebLogic Administration Server or SOA managed servers, use virtual IP addresses. You must enable the appropriate IP address on each server.

Section 3.5, "About IP Addresses and Virtual IP Addresses," describes the mapping of IP Addresses to servers.

This section includes the following topics:

- Section 5.6.1, "Virtual IP Addresses to Enable."
- Section 5.6.2, "Enabling Virtual Addresses by Using the Command Line."

5.6.1 Virtual IP Addresses to Enable

Virtual IP Addresses are required for failover of the WebLogic Administration Server, regardless of whether other Oracle Fusion Middleware components are installed later or not.

You associate the Administration Server with a virtual IP address. This allows the Administration Server to be started on a different host if the primary host fails.

Check that the virtual host is enabled as follows:

Table 5–2 Virtual Hosts for Domain

VIP	Enabled on Host
ADMINVHN.mycompany.com	IDMHOST1
OIMHOST1VHN.mycompany.com	IDMHOST1
OIMHOST2VHN.mycompany.com	IDMHOST2
SOAHOST1VHN.mycompany.com	IDMHOST1
SOAHOST2VHN.mycompany.com	IDMHOST2

Note: This is the DNS name associated with the floating IP address. It is not the DNS name of the virtual host configured on the load balancer.

5.6.2 Enabling Virtual Addresses by Using the Command Line

To enable a virtual IP address, you must associate the IP address with a network interface and set the netmask of that interface. On a Linux host, perform the steps listed in this section. For other operating systems, refer to your manufacturer documentation.

To enable the virtual IP address, you use the `ifconfig` command to set the IP address and netmask of the interface. Then you use the `arping` command to enable your network to register the new location of the virtual IP address. Run the following commands as `root`:

```
/sbin/ifconfig interface:index IPAddress netmask netmask
/sbin/arping -q -U -c 3 -I interface IPAddress
```

where *`interface`* is `eth0`, `eth1`, and so forth, and *`index`* is `0`, `1`, `2`, and so forth.

For example:

```
/sbin/ifconfig eth0:1 100.200.140.206 netmask 255.255.255.0
/sbin/arping -q -U -c 3 -I eth0 100.200.140.206
```

Validate that the address is available by `ping`ing it from another node, for example:

```
/bin/ping 100.200.140.206
```

5.7 Mounting Shared Storage onto the Host

The shared storage configured in Chapter 4, "Preparing Storage for an Enterprise Deployment" must be available on the hosts that use it. Mount the shared storage to all servers that require access to it.

Each host must have appropriate privileges set within the NAS or 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 UNIX or Linux using NFS storage.

You must create and mount shared storage locations so that IDMHOST1 and IDMHOST2 can see the same location for 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 ID 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 execute the files, but they do not have write privileges. For more information about installation and configuration privileges, see the "Understanding Installation and Configuration Privileges and Users" section in the *Oracle Fusion Middleware Installation Planning Guide*.

`nasfiler` is the shared storage filer.

From IDMHOST1:

```
mount -t nfs nasfiler:VOL1/OracleIDM /u01/oracle
```

From IDMHOST2:

```
mount -t nfs nasfiler:VOL1/OracleIDM /u01/oracle
```

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 you just configured.

For example:

```
$ cd newly mounted directory
$ touch testfile
```

Verify that the owner and permissions are correct:

```
$ ls -l testfile
```

Then remove the file:

```
$ rm testfile
```

Note: The shared storage can be a NAS or SAN device. The following illustrates an example of creating storage for a NAS device from IDMHOST1. 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:VOL1/OracleIDM /u01/oracle
```

Contact your storage vendor and machine administrator for the correct options for your environment.

5.8 Configuring Users and Groups

Groups

You must create the following groups on each node.

- `oinstall`
- `dba`

Users

You must create the following user on each node.

- `nobody`–An unprivileged user.
- `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.

Notes:

- 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.

5.9 Installing Oracle Software onto a Server with Multiple Network Addresses

You can install Oracle Identity Management components on a multihomed system. A multihomed system is has with multiple IP addresses. Typically, each IP address is associated with a different network card on the system. Each IP address is associated with a host name. You can create aliases for each host name.

The Installer retrieves the fully qualified domain name from the first entry in `/etc/hosts` file. For example, if your file looks like the following sample file, the Installer retrieves `MYHOST1.mycompany.com` for configuration:

```
127.0.0.1 localhost.localdomain localhost
10.222.333.444 MYHOST1.mycompany.com myhost1
20.222.333.444 DEVHOST2.mycompany.com devhost2
```

6

Preparing the Database for an Enterprise Deployment

This chapter describes how to install and configure the Identity Management database repositories.

This chapter contains the following topics:

6.1 Overview of Preparing the Databases for an Identity Management Enterprise Deployment

The Identity Management components in the enterprise deployment use database repositories. This chapter describes how to perform the following steps:

- Verify the database requirements as described in Section 6.2, "Verifying the Database Requirements for an Enterprise Deployment."
- Install and configure the Oracle database repositories. See the installation guides listed in the "Related Documents" section of the Preface and Section 6.3, "Installing the Database for an Enterprise Deployment."
- Create database services, as described in Section 6.4, "Creating Database Services."
- Create the required Oracle schemas in the database using the Repository Creation Utility (RCU). See Section 6.6, "Loading the Identity Management Schemas in the Oracle RAC Database by Using RCU."

6.2 Verifying the Database Requirements for an Enterprise Deployment

Before loading the metadata repository into your databases, check that they meet the requirements described in these subsections:

6.2.1 Databases Required

For Oracle Identity management, a number of separate databases are recommended. Table 6–1 provides a summary of these databases. Which database or databases you use depends on the topology that you are implementing.

The Oracle Metadata Services (MDS) Repository is a particular type of repository that contains metadata for some Oracle Fusion Middleware components. It can also include custom Java EE applications developed by your organization.

Table 6–1 Mapping between Databases and Schemas

Database Names	Database Hosts	Service Names	Schemas in Database
IDMDB	IDMDBHOST1 IDMDBHOST2	`OAMEDG.mycompany.com`	OAM, IAU, ORASDPM, MDS
		`OIMEDG.mycompany.com`	OIM, SOA_INFRA, MDS
		`OESEDG.mycompany.com`	OPSS, MDS

The following sections apply to all the databases listed in Table 6–1.

6.2.2 Database Host Requirements

The database used to store the metadata repository should be highly available in its own right, for maximum availability Oracle recommends the use of an Oracle Real Application Clusters (RAC) database.

Ideally the database should use Oracle Automatic Storage Management (ASM) for the storage of data, however this is not necessary.

If using ASM, then ASM should be installed into its own Oracle home and have two disk groups:

- One for the Database Files
- One for the Flash Recovery Area

If you are using Oracle ASM, best practice is to also use Oracle Managed Files.

6.2.3 Database Versions Supported

To check if your database is certified or to see all certified databases, refer to the "Certified Databases" section in the Certification Document:

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

To determine the version of your installed Oracle Database, execute the following query at the SQL prompt:

```
select version from sys.product_component_version where product like 'Oracle%';
```

6.2.4 Patching the Oracle Database

Patches are required for some versions of Oracle Database.

6.2.4.1 Patch Requirements for Oracle Database 11g (11.1.0.7)

Table 6–2 lists patches required for Oracle Identity Manager configurations that use Oracle Database 11*g* (11.1.0.7). Before you configure Oracle Identity Manager 11*g*, be sure to apply the patches to your Oracle Database 11*g* (11.1.0.7) database.

Table 6–2 Required Patches for Oracle Database 11g (11.1.0.7)

Platform	Patch Number and Description on My Oracle Support
Linux	7614692: BULK FEATURE WITH 'SAVE EXCEPTIONS' DOES NOT WORK IN ORACLE 11G
	7000281: DIFFERENCE IN FORALL STATEMENT BEHAVIOR IN 11G
	8327137: WRONG RESULTS WITH INLINE VIEW AND AGGREGATION FUNCTION
	8617824: MERGE LABEL REQUEST ON TOP OF 11.1.0.7 FOR BUGS 7628358 7598314

6.2.4.2 Patch Requirements for Oracle Database 11g (11.2.0.2.0)

If you are using Oracle Database 11*g* (11.2.0.2.0), make sure that you download and install the appropriate version (based on the platform) for the RDBMS Patch Number 10259620. This is a prerequisite for installing the Oracle Identity Manager schemas.

Table 6–3 lists the patches required for Oracle Identity Manager configurations that use Oracle Database 11*g* Release 2 (11.2.0.2.0). Make sure that you download and install the following patches before creating Oracle Identity Manager schemas.

Table 6–3 Required Patches for Oracle Database 11g (11.2.0.2.0)

Platform	Patch Number and Description on My Oracle Support
Linux x86 (32-bit) Linux x86 (64-bit)	RDBMS Interim Patch#10259620.

If this patch is not applied, then problems might occur in user and role search and manager lookup. In addition, search results might return empty result.

Note:

- Apply this patch in ONLINE mode. Refer to the readme.txt file bundled with the patch for the steps to be followed.
- In some environments, the RDBMS Interim Patch has been unable to resolve the issue, but the published workaround works. Refer to the note "Wrong Results on 11.2.0.2 with Function-Based Index and OR Expansion due to fix for Bug:8352378 [Metalink Note ID 1264550.1]" at http://support.oracle.com for the workaround. This note can be followed to set the parameters accordingly with the only exception that they need to be altered at the Database Instance level by using ALTER SYSTEM SET <param>=<value> scope=<memory> or <both>.

6.2.5 About Initialization Parameters

The databases must have the following minimum initialization parameters defined:

Table 6–4 Minimum Initialization Parameters for Oracle RAC Databases

Parameter	Value
aq_tm_processes	1
dml_locks	200
job_queue_processes	10
open_cursors	800[1]
session_max_open_files	50
sessions	500
processes	500
sga_target	512M
pga_aggregate_target	100M
sga_max_size	4G
session_cached_cursors	500

[1] OAM requires a minimum of 800 open cursors in the database. When OIM and OAM are available, the number of open cursors should be 1500.

Note: For guidelines on setting up optimum parameters for the Database, see *Oracle Fusion Middleware Performance and Tuning Guide*.

6.3 Installing the Database for an Enterprise Deployment

Install and configure the database repository as follows.

Oracle Clusterware

- For 10*g* Release 2 (10.2), see the *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide* for your platform, listed in "Related Documents".

- For 11*g* Release 1 (11.1), see *Oracle Clusterware Installation Guide*.

Automatic Storage Management

- For 10*g* Release 2 (10.2), see *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide* for your platform, listed in "Related Documents".
- For 11*g* Release 1 (11.1), see *Oracle Clusterware Installation Guide*.
- When you run the installer, select the **Configure Automatic Storage Management** option in the **Select Configuration** screen to create a separate Automatic Storage Management home.

Oracle Real Application Clusters

- For 10*g* Release 2 (10.2), see *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide* for your platform, listed in "Related Documents".
- For 11*g* Release 1 (11.1), see *Oracle Real Application Clusters Installation Guide*.

Oracle Real Application Clusters Database

Create a Real Applications Clusters Database with the following characteristics:

- Database must be in archive log mode to facilitate backup and recovery.
- Optionally, enable the Flashback database.
- Create UNDO tablespace of sufficient size to handle any rollback requirements during the Oracle Identity Manager reconciliation process.
- Database is created with ALT32UTF8 character set.

6.4 Creating Database Services

This section describes how to configure the database for Oracle Fusion Middleware 11g metadata. It contains the following topics:

- Section 6.4.1, "Creating Database Services for 10.x and 11.1.x Databases"
- Section 6.4.2, "Creating Database Services for 11.2.x Databases"
- Section 6.4.3, "Database Tuning"

6.4.1 Creating Database Services for 10.x and 11.1.x Databases

For complete instructions on creating database services, see the chapter on Workload Management in the *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide*. Oracle recommends that a specific database service be used for a product suite, even when product suites share the same database. It is also recommended that the database service used is different than the default database service.

Use the `CREATE_SERVICE` subprogram to create the database services for the components in your topology. The lists of services to be created are listed in Table 6–1, " Mapping between Databases and Schemas".

1. Log on to SQL*Plus as the `sysdba` user by typing:

```
sqlplus "sys/password as sysdba"
```

Then run the following command to create a service called `OAMEDG.mycompany.com` for Access Manager:

```
EXECUTE DBMS_SERVICE.CREATE_SERVICE
(SERVICE_NAME => 'OAMEDG.mycompany.com',
NETWORK_NAME => 'OAMEDG.mycompany.com');
```

2. Add the service to the database and assign it to the instances using srvctl:

```
srvctl add service -d idmdb -s OAMEDG.mycompany.com -r idmdb1,idmdb2
```

3. Start the service using `srvctl`:

```
srvctl start service -d idmdb -s OAMEDG.mycompany.com
```

6.4.2 Creating Database Services for 11.2.x Databases

Use `srvctl` to create the database services for the components in your topology. The lists of services to be created are listed in Table 6–1, " Mapping between Databases and Schemas".

1. Create service using the command `srvctl add service`, as follows.

```
srvctl add service -d idmdb -s OAMEDG.mycompany.com -r idmdb1,idmdb2 -q FALSE
-m NONE -e SELECT -w 0 -z 0
```

The meanings of the command-line arguments are as follows:

Option	Argument
-d	Unique name for the database
-s	Service name
-r	Comma separated list of preferred instances
-q	AQ HA notifications (TRUE or FALSE)
-e	Failover type (NONE, SESSION, or SELECT)
-m	Failover method (NONE or BASIC)
-w	Failover delay (integer)
-z	Failover retries (integer)

2. Start the Service using `srvctl start service`

```
srvctl start service -d idmdb -s OAMEDG.mycompany.com
```

3. Validate the service started by using `srvctl status service`, as follows:

```
srvctl status service -d idmdb -s OAMEDG.mycompany.com
Service OAMEDG.mycompany.com is running on instance(s) idmdb1,idmdb2
```

4. Validate that the service was created correctly by using `srvctl config service`:

```
srvctl config service -d idmdb -s OAMEDG.mycompany.com
Service name: OAMEDG.mycompany.com
Service is enabled
Server pool: IDMDB_OAMEDG.mycompany.com
Cardinality: 2
Disconnect: false
```

```
Service role: PRIMARY
Management policy: AUTOMATIC
DTP transaction: false
AQ HA notifications: false
Failover type: SELECT
Failover method: NONE
TAF failover retries: 0
TAF failover delay: 0
Connection Load Balancing Goal: LONG
Runtime Load Balancing Goal: NONE
TAF policy specification: NONE
Edition:
Preferred instances: idmdb1,idmdb2
Available instances:
```

Note: For more information about the SRVCTL command, see the *Oracle Real Application Clusters Administration and Deployment Guide*.

6.4.3 Database Tuning

The database parameters defined in Section 6.3, "Installing the Database for an Enterprise Deployment" are only a guide. You might need to perform additional tuning after the system is in use. For more information, see *Database Performance Tuning Guide*.

Refresh the database statistics after you initially load the database, and on an ongoing basis. To do that, issue the following SQL*Plus command:

```
exec DBMS_STATS.GATHER_SCHEMA_STATS(OWNNAME=> '<OIM_SCHEMA>', ESTIMATE_
PERCENT=>DBMS_STATS.AUTO_SAMPLE_SIZE, DEGREE=>8, OPTIONS=>'GATHER AUTO', NO_
INVALIDATE=>FALSE);
```

6.5 Preparing the Database for Repository Creation Utility (RCU)

To prepare the Oracle Database, follow the instructions in the section "RCU Requirements for Oracle Databases" in the *Oracle Fusion Middleware System Requirements and Specifications*.

Execute the following commands to create XATRANS Views:

```
cd DB_ORACLE_HOME/rdbms/admin
sqlplus / as sysdba
@xaview.sql
```

6.6 Loading the Identity Management Schemas in the Oracle RAC Database by Using RCU

You run RCU to create the collection of schemas used by Identity Management and Management Services.

1. Start RCU by issuing this command:

   ```
   RCU_HOME/bin/rcu &
   ```

2. On the Welcome screen, click **Next**.

3. On the Create Repository screen, select the **Create** operation to load component schemas into a database. Then click **Next**.
4. On the Database Connection Details screen, provide the information required to connect to an existing database. For example:

   Database Type: `Oracle Database`

   - **Host Name:** Enter the VIP address of one of the RAC database nodes or the database SCAN address, for example: `DB-SCAN.mycompany.com`
   - **Port:** The port number for the database listener *(DB_LSNR_PORT)*. For example: `1521`
   - **Service Name:** The service name of the database. For example `OAMEDG.mycompany.com`.

     Use the service names for the components you will select from the table in Step 6.
   - **Username:** `sys`
   - **Password:** The sys user password
   - **Role:** `SYSDBA`

   Click **Next**.
5. On the Check Prerequisites screen, click `OK` after the prerequisites have been validated.
6. On the Select Components screen, provide the following values:

   Create a New Prefix: Enter a prefix to be added to the database schemas. Note that all schemas are required to have a prefix. For example, enter `EDG`.

   Components: Select the appropriate components from the following table for the topology you are using.

Product	RCU Option	Service Name	Comments
Oracle Platform Security Services	AS Common Schemas–Oracle Platform Security Service		Required to hold policy store information. Mandatory for all topologies.
Oracle Access Management Access Manager	Identity Management–Access Manager	`OAMEDG.mycompany.com`	Audit Services will also be selected.
Oracle Identity Manager	Identity Management–Oracle Identity Manager	`OAMEDG.mycompany.com`	Metadata Services, SOA infrastructure, and User Messaging will also be selected.

   Click **Next**.

Notes: If your topology requires more than one database, the following important considerations apply:

- Be sure to install the correct schemas in the correct database.
- You might have to run the RCU more than once to create all the schemas for a given topology.
- Table 6–1 in this chapter provides the recommended mapping between the schemas and their corresponding databases. Refer to this table to ensure that the correct details are entered in this screen.

7. On the Check Prerequisites screen, click **OK** after the prerequisites have been validated.
8. On the Schema Passwords screen, enter the passwords for the schemas. You can choose to use either the same password for all the schemas or different passwords for each of the schemas. Oracle recommends choosing different passwords for different schema's to enhance security

   Click **Next**.
9. On the Map Tablespaces screen, accept the defaults and click **Next**.
10. On the confirmation screen, click **OK** to allow the creation of the tablespaces.
11. On the Creating tablespaces screen, click OK to acknowledge creation of the tablespaces.
12. On the Summary screen, the summary and verify that the details provided are accurate. Click **Create** to start the schema creation process.
13. On the Completion summary screen, verify that the schemas were created.

    Click **Close** to exit.

6.7 Backing up the Database

After you have prepared your database, back it up as described in Section 17.6.3.3, "Backing Up the Database."

7

Installing and Configuring Oracle Unified Directory

This chapter describes how to install and configure Oracle Unified Directory (OUD) in the enterprise deployment.

This chapter includes the following topics:

7.1 Overview of Installing and Configuring Oracle Unified Directory

Oracle Unified Directory is a required component in the Identity Management enterprise topologies. You use it as the Identity Store, that is, for storing information about users and groups.

In this chapter, you configure two instances of Oracle Unified Directory by using Oracle Unified Directory configuration assistant.

7.2 Prerequisites for Configuring Oracle Unified Directory Instances

Before configuring the Oracle Unified Directory Instances on IDMHOST1 and IDMHOST2 ensure that the following tasks have been performed:

- Synchronize the time on the individual IDMHOSTs nodes so that there is a discrepancy of no more than 250 seconds between them.
- Ensure that the load balancer is configured.

7.3 Installing Oracle Unified Directory

Perform these steps to install Oracle Unified Directory on shared storage, perform the following steps from either IDMHOST1 or IDMHOST2.

Ensure that the system, patch, kernel and other requirements are met. These are listed in *Oracle Fusion Middleware Installation Guide for Oracle Identity Management* in the

Oracle Fusion Middleware documentation library for the platform and version you are using.

Install JDK as described in Section 8.2.1.1, "Installing Oracle JRockit." To start the Oracle Fusion Middleware 11g Oracle Identity Management Installer, change directory to Disk1 of the installation media and enter the command:

```
./runInstaller
```

Then proceed as follows:

On the Specify Inventory Directory screen, do the following:/u02/private/oracle/oraInventory

- Enter *HOME*`/oraInventory` (`/u02/private/oracle/oraInventory`), where *HOME* is the home directory of the user performing the installation (this is the recommended location).
- Enter the OS group for the user performing the installation.
- Click **Next**.

Follow the instructions on screen to execute `createCentralInventory.sh` as `root`.

1. On the Welcome screen, click **Next**.
2. On the Install Software Updates screen, choose whether to skip updates, check with Oracle Support for updates, or search for updates locally.

   Click **Next**.
3. On the Prerequisite Checks screen, verify that the checks complete successfully, then click **Next**.
4. On the specify Installation Screen Enter:
   - **OUD Base Location Home**: *IAM_MW_HOME*
   - **Oracle Home Directory**: `oud`

   Click **Next**.
5. On the installation Summary Screen click **Install**.
6. On the Installation Progress Screen click **Next**.
7. On the installation complete Screen click **Finish**.

7.4 Configuring the Oracle Unified Directory Instances

Follow these steps to configure Oracle Unified Directory components in the application tier on IDMHOST1 and IDMHOST2. During the configuration you will also configure Oracle Unified Directory replication servers.

This section contains the following topics:

7.4.1 Configuring Oracle Unified Directory on IDMHOST1

Ensure that ports 1389 (*LDAP_DIR_PORT*), 1636 *(LDAP_DIR_SSL_PORT*), 4444 (*LDAP_DIR_ADMIN_PORT*), and 8989 *(LDAP_DIR_REPL_PORT*) 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.

On Linux:

```
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.

On Linux:

Remove the entries for ports 1389, 1636, 4444, and 8989 in the `/etc/services` file and restart the services or restart the computer.

Set the environment variable JAVA_HOME

Set the environment variable INSTANCE_NAME to:

```
../../../../u02/private/oracle/config/instances/oud1
```

Note that the tool creates the instance home relative to the *OUD_ORACLE_HOME*, so you must include previous directories to get the instance created in *OUD_ORACLE_INSTANCE*.

Change Directory to *OUD_ORACLE_HOME*

Start the Oracle Unified Directory configuration assistant by executing the command:

```
./oud-setup
```

1. On the Welcome screen, click **Next**.
2. On the Server Settings screen, enter:
   - **Host Name**: The name of the host where Oracle Unified Directory is running, for example: IDMHOST1.mycompany.com
   - **LDAP Listener Port**: 1389 (*LDAP_DIR_PORT*)
   - **Administration Connector Port**: 4444 (*LDAP_DIR_ADMIN_PORT*)
   - **LDAP Secure Access**: Click **Configure**
   - In the Security Options page, enter:
     - **SSL Access**: Selected.
     - **Enable SSL on Port**: 1636 *(LDAP_DIR_SSL_PORT*)
     - **Certificate**: Generate Self Signed Certificate OR provide details of your own certificate.
     - Click **OK**
   - **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**.
3. On the Topology Options screen:
   - Select: **This server will be part of a replication topology**

- Enter: **Replication Port:** 8989
- Select: **Configure As Secure,** if you wish replication traffic to be encrypted.
- There is already a server in the topology. Leave it deselected.

Click **Next**.

4. On the Directory Data screen, enter:
   - **Directory Base DN**: `dc=mycompany,dc=com`
   - **Directory Data**: Only create base entry

   Click **Next**.
5. On the Oracle Components Integration screen, click **Next**.
6. On the Runtime Options screen, click **Next**.
7. On the Review screen, verify that the information displayed is correct and click **Finish**.
8. On the Finished screen, click **Close**.

7.4.2 Validating Oracle Unified Directory on IDMHOST1

After configuration, you can validate that Oracle Unified Directory is working by performing a simple search. To do this issue the following command:

```
OUD_ORACLE_INSTANCE/OUD/bin/ldapsearch -h IDMHOST1.mycompany.com -p 1389 -D
cn=oudadmin -b "" -s base "(objectclass=*)" supportedControl
```

If Oracle Unified Directory is working correctly, you will see a list `supportedControl` entries returned.

7.4.3 Configuring an Additional Oracle Unified Directory Instance on IDMHOST2

Ensure that ports 1389 (*LDAP_DIR_PORT*), 1636 *(LDAP_DIR_SSL_PORT*), 4444 (*LDAP_DIR_ADMIN_PORT*), 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.

On Linux:

```
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.

On Linux:

Remove the entries for ports 1389, 1636, 4444, and 8989 in the `/etc/services` file and restart the services or restart the computer.

Set the environment variable `JAVA_HOME`

Set the environment variable `INSTANCE_NAME` to `../../../../u02/private/oracle/config/instances/oud2`.

Note the tool creates the instance home relative to the *OUD_ORACLE_HOME*, so you must include previous directories to get the instance created in *OUD_ORACLE_INSTANCE*.

Change Directory to *OUD_ORACLE_HOME*

Start the Oracle Unified Directory configuration assistant by executing the command:

```
./oud-setup
```

1. On the Welcome screen, click **Next**.
2. On the Server Settings screen, enter:
   - **Host Name**: The name of the host where Oracle Unified Directory is running, for example: IDMHOST2
   - **LDAP Listener Port**: 1389 (*LDAP_DIR_PORT*)
   - **Administration Connector Port**: 4444 (*LDAP_DIR_ADMIN_PORT*)
   - LDAP Secure Access
     - Click **Configure**
     - Select **SSL Access**
     - **Enable SSL on Port**: 1636 *(LDAP_DIR_SSL_PORT*)
     - **Certificate**: Generate Self Signed Certificate OR provide details of your own certificate.
     - Click **OK**
   - **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**.
3. On the Topology Options screen, enter
   - **This server will be part of a replication topology**
   - **Replication Port:** 8989
   - Select **Configure As Secure,** if you wish replication traffic to be encrypted.
   - **There is already a server in the topology**: Selected.

     Enter the following:
     - **Host Name**: The name of an existing Oracle Unified Directory server host, for example: IDMHOST1.mycompany.com
     - **Administrator Connector Port**: 4444 (*LDAP_DIR_ADMIN_PORT*)
     - **Admin User**: Name of the Oracle Unified Directory admin user on IDMHOST1, 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.**

   Click **Next**.
4. On The Create Global Administrator Screen Enter:
   - **Global Administrator ID:** 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**.

5. On the Data Replication Screen. select `dc=mycompany,dc=com` and click **Next**.
6. On the Oracle Components Integration screen, click **Next**.
7. On the Runtime Options Screen Click Next.
8. On the Review Screen, check that the information displayed is correct and click **Finish**.
9. On the Finished screen, click **Close**.

7.4.4 Validating Oracle Unified Directory on IDMHOST2

After configuration you can validate that Oracle Unified Directory is working by performing a simple search. To do this issue the following command:

```
OUD_ORACLE_INSTANCE/OUD/bin/ldapsearch -h IDMHOST2.mycompany.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.

7.4.5 Enable Oracle Unified Directory Assured Replication

As discussed in Section 2.2.2.2.1, "About Oracle Unified Directory," you must ensure that data read from every Oracle Unified Directory instance is current. You do this by enabling Oracle Unified Directory Assured Replication in Safe Read Mode, as follows:

1. On IDMHOST1, issue the following command:

```
OUD_ORACLE_INSTANCE/OUD/bin/dsconfig -h IDMHOST1 -p 4444 -D "cn=oudadmin" -j
./password_file -n \
set-replication-domain-prop \
--provider-name "Multimaster Synchronization" \
--domain-name "dc=mycompany,dc=com" \
--advanced \
--set assured-type:safe-read \
--trustAll
```

2. Confirm that the operation has been successful by issuing the command:

```
OUD_ORACLE_INSTANCE/OUD/bin/dsconfig -h IDMHOST1 -p 4444 -D "cn=oudadmin" -j
./password_file -n \
get-replication-domain-prop \
--provider-name "Multimaster Synchronization" \
--domain-name "dc=mycompany,dc=com" \
--advanced \
--property assured-type --property assured-timeout --property group-id \
--trustAll
```

where *password_file* is a file that contains the OUD administrator password.

If Safe Mode is enabled, the output looks similar to this:

```
Property        : Value(s)
----------------:----------
assured-timeout : 2 s
assured-type    : safe-read
group-id        : 1
```

3. Repeat steps 1-2 for each Oracle Unified Directory instance, for example: IDMHOST2.

7.4.6 Relaxing Oracle Unified Directory Object Creation Restrictions

Oracle Identity Management requires that a number of object classes be created in Oracle Unified Directory. You must perform the following step so that Oracle Unified Directory allows creation of the needed object classes.

Execute the following command on each Oracle Unified Directory instance:

```
OUD_ORACLE_INSTANCE/OUD/dsconfig -h IDMHOST1 -p 4444 -D "cn=oudadmin" -j
./password_file -n \
        set-global-configuration-prop \
        --set single-structural-objectclass-behavior:warn \
        --trustAll
```

Repeat the command for each Oracle Unified Directory instance, for example: IDMHOST2.

7.4.7 Validating Oracle Unified Directory Through the Load Balancer

In addition, validate that you can access Oracle Unified Directory through the load balancer by issuing the command:

```
OUD_ORACLE_INSTANCE/OUD/bin/ldapsearch -h LDAP_LBR_HOST -p LDAP_LBR_PORT -D OUD_
Adminisitrator -b "" -s base "(objectclass=*)" supportedControl
```

For example:

```
OUD_ORACLE_INSTANCE/OUD/bin/ldapsearch -h IDSTORE.mycompany.com -p 389 -D
cn=oudadmin -b "" -s base "(objectclass=*)" supportedControl
```

To check that Oracle Unified Directory replication is enabled, issue the command:

```
OUD_ORACLE_INSTANCE/OUD/bin/status
```

If you are asked how you wish to trust the server certificate, valid options are:

- Automatically trust
- Use a truststore
- Manually validate

Select your choice.

You are then prompted for the Administrator bind DN (`cn=oudadmin`) and its password.

Next, you see output similar to the following example. Replication will be set to enable.

```
--- Server Status ---
Server Run Status: Started
Open Connections: 2

--- Server Details ---
Host Name: idmhost1
Administrative Users: cn=oudadmin
Installation Path: /u01/oracle/products/access/oud
Instance Path: /u02/private/oracle/config/instances/oud1/OUD
Version: Oracle Unified Directory 11.1.2.1.0
```

```
Java Version: 1.6.0_29
Administration Connector: Port 4444 (LDAPS)

--- Connection Handlers ---
Address:Port : Protocol : State
-------------:-------------:---------
-- : LDIF : Disabled
8989 : Replication : Enabled
0.0.0.0:161 : SNMP : Disabled
0.0.0.0:1389 : LDAP : Enabled
0.0.0.0:1636 : LDAPS : Enabled
0.0.0.0:1689 : JMX : Disabled

--- Data Sources ---
Base DN: dc=mycompany,dc=com
Backend ID: userRoot
Entries: 1
Replication: Enabled
Missing Changes: 0
Age Of Oldest Missing Change: <not available>
```

7.5 Post-Configuration Task

In an environment in which LDAP synchronization is enabled, certain operations against OUD fail with the following error in OUD logs:

```
The request control with Object Identifier (OID) "1.2.840.113556.1.4.319" cannot
be used due to insufficient access rights
```

To workaround this issue:

1. Change the ACIs on control 1.2.840.113556.1.4.319 from ldap://all to ldap://anyone in OUD config file *OUD_INSTANCE*/config/config.ldif file, as shown:

   Change:

   ```
   ds-cfg-global-aci: (targetcontrol="1.3.6.1.1.12 || 1.3.6.1.1.13.1 ||
   1.3.6.1.1.13.2 || 1.2.840.113556.1.4.319 || 1.2.826.0.1.3344810.2.3 ||
   2.16.840.1.113730.3.4.18 || 2.16.840.1.113730.3.4.9 || 1.2.840.113556.1.4.473
   || 1.3.6.1.4.1.42.2.27.9.5.9") (version 3.0; acl "Authenticated users control
   access"; allow(read) userdn="ldap:///all";)
   ```

   To:

   ```
   ds-cfg-global-aci: (targetcontrol="2.16.840.1.113730.3.4.2 ||
   2.16.840.1.113730.3.4.17 || 2.16.840.1.113730.3.4.19 || 1.3.6.1.4.1.4203.1.10.2
   || 1.3.6.1.4.1.42.2.27.8.5.1 || 2.16.840.1.113730.3.4.16 ||
   2.16.840.1.113894.1.8.31 || 1.2.840.113556.1.4.319") (version 3.0; acl
   "Anonymous control access"; allow(read) userdn="ldap:///anyone";)
   ```

2. Restart OUD and Oracle Identity Manager servers.

7.6 Backing Up the Oracle Unified Directory installation

Perform a backup of the Middleware home and of Oracle Unified Directory, as described in Section 17.6.3, "Performing Backups During Installation and Configuration."

8

Creating a Domain for an Enterprise Deployment

This chapter describes how to create a domain using the Configuration Wizard, Oracle WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control. The topology you are creating dictates the number of domains you need to create. Once the initial domain has been created, it can be extended with other products as described later on in this book.

> **Note:** Oracle strongly recommends that you read the release notes for any additional installation and deployment considerations prior to starting the setup process.

This chapter contains the following sections.

8.1 Overview of Creating a Domain

Table 8–1 lists the steps for creating a WebLogic domain, including post-configuration tasks.

Table 8–1 Steps for Creating a WebLogic Domain

Step	Description	More Information
Create a WebLogic Domain	Run the Configuration Wizard to create WebLogic domain.	Section 8.4, "Running the Configuration Wizard to Create a Domain"
Post-Configuration and Verification Tasks	Follow the instructions for post-configuration and validation tasks.	Section 8.5, "Post-Configuration and Verification Tasks"
Back Up the Domain	Back up the newly configured WebLogic domain.	Section 8.7, "Backing Up the WebLogic Domain"

Once this domain is created and configured you can extend the domain to include other Identity Management components, as described in the next chapters.

8.2 Installing Oracle Fusion Middleware Home

As described in Section 4.4, "About Recommended Locations for the Different Directories," you install Oracle Fusion Middleware software in at least two storage locations for redundancy.

You must install the following components of Oracle Fusion Middleware to create a Middleware home (*MW_HOME*):

1. Oracle WebLogic Server: Section 8.2.1, "Installing Oracle WebLogic Server and Creating the Fusion Middleware Home"
2. One or more of the Oracle Fusion Middleware components
   a. Section 8.2.2, "Installing Oracle Identity and Access Management"
   b. Section 8.2.3, "Installing the Oracle SOA Suite"
3. Oracle Fusion Middleware for Identity Management

8.2.1 Installing Oracle WebLogic Server and Creating the Fusion Middleware Home

This section describes how to obtain and install Oracle WebLogic Server.

8.2.1.1 Installing Oracle JRockit

1. Download the version of Oracle JRockit for your platform from:

   ```
   http://www.oracle.com/technetwork/middleware/jrockit/downloads/index.html
   ```

2. Add execute permissions to Oracle JRockit. For example:

   ```
   chmod +x jrockit-1.6.0_29-R28.2.0-4.0.1-linux-x64.bin
   ```

3. Start the Oracle JRockit installer by issuing the command:

   ```
   ./jrockit-version.bin
   ```

   For example:

   ```
   ./jrockit-1.6.0_29-R28.2.0-4.0.1-linux-x64.bin
   ```

4. On the Welcome Screen, click **Next**.
5. On the Choose Product Installation Directories screen, enter the Product Installation Directory, which is inside your Middleware Home.
6. On the Optional Components Screen, click **Next**.
7. On the Installation Complete screen, click **Done**.

8.2.1.2 Installing WebLogic Server Using the Generic Installer

1. Download the Oracle WebLogic Server Generic Installer from:
   `http://edelivery.oracle.com`
2. Add Oracle JRockit to your path. For example, on Linux, issue the command:

   ```
   export PATH=IAM_MW_HOME/jrockit-jdk1.6.0_29-R28.2.0-4.0.1/bin:$PATH
   ```

3. Check the version of java by issuing the command:

```
java -version
```

Ensure that the 64-bit version is displayed if you are using a 64-bit operating system.

4. Start the WebLogic installer using the appropriate command:

   64-Bit Operating System

   ```
   java -d64 -jar wls1036_generic.jar
   ```

   32-Bit Operating System

   ```
   java -jar wls1036_generic.jar
   ```

5. On the Welcome screen, click **Next**.
6. On the Choose Middleware Home screen, select: **Create a New Middleware Home**

   For the Middleware Home directory enter the path to `IAM_MW_HOME`, for example:

   ```
   /u01/oracle/products/access
   ```

   Click **Next**.
7. A warning is displayed, informing you that the directory is not empty and asking if you want to proceed.

   Click **Yes**.
8. On the Register for Security Updates screen, enter your My Oracle Support username and password so that you can be notified of security updates.

   Click **Next**.
9. On the Choose Install Type screen, select **Typical**.

   Note: Oracle WebLogic Server and Oracle Coherence are installed.

10. On the JDK Selection screen, select the Oracle JRockit JDK that you installed earlier. It should be listed by default.

    Note: The examples documented in this guide use Oracle JRockit. Any certified version of Java can be used for this procedure and is fully supported unless otherwise noted.

11. On the Choose Product Installation Directories screen, accept the following:
    - **Middleware Home Directory**: *`IAM_MW_HOME`*
    - **Product Installation Directories for WebLogic Server**: *`IAM_MW_HOME`*`/wlserver_10.3`
    - **Oracle Coherence**: *`IAM_MW_HOME`*`/coherence_3.6`

    Click **Next**.
12. On the Installation Summary screen, click **Next** to start the install process
13. On the Installation complete screen, deselect **Run Quickstart**.
14. Click **Done** to exit the WebLogic Server Installer.

8.2.2 Installing Oracle Identity and Access Management

Oracle Identity and Access Management includes the following products:

- Oracle Access Management Access Manager
- Oracle Identity Manager

Perform the steps in this section to install Oracle Identity and Access Management on the hosts identified in Table 2–2, " Software Versions Used".

Ensure that the system, patch, kernel and other requirements are met. These are listed in the *Oracle Fusion Middleware Installation Guide for Oracle Identity Management* in the Oracle Fusion Middleware documentation library for the platform and version you are using.

To start the Oracle Fusion Middleware 11*g* Installer for Oracle Identity and Access Management, change directory to Disk1 of the installation media and enter the command:

```
./runInstaller
```

When the installer prompts you for a JRE/JDK location, enter the Oracle SDK location created in the Oracle WebLogic Server installation, for example:

```
IAM_MW_HOME/jrockit_version
```

Then perform these installation steps:

1. On the Specify Inventory Directory screen, enter values for the Oracle Inventory Directory and the Operating System Group Name. For example:
   - **Specify the Inventory Directory**: `/u02/private/oracle/oraInventory`
   - **Operating System Group Name**: `oinstall`

   A dialog box appears with the following message:

   ```
   Certain actions need to be performed with root privileges before the install
   can continue. Please execute the script
   /u02/private/oracle/oraInventory/createCentralInventory.sh now from another
   window and then press "Ok" to continue the install. If you do not have the root
   privileges and wish to continue the install select the "Continue installation
   with local inventory" option.
   ```

   Log in as `root` and run:

   ```
   /u02/private/oracle/oraInventory/createCentralInventory.sh
   ```

   This sets the required permissions for the Oracle Inventory Directory and then brings up the Welcome screen.

   Note: The Oracle Inventory screen is not shown if an Oracle product was previously installed on the host. If the Oracle Inventory screen is not displayed for this installation, check the following:

   1. The `/etc/oraInst.loc` file exists.
   2. The Inventory directory listed is valid.
   3. The user performing the installation has write permissions for the Inventory directory.

2. On the Install Software Updates screen, choose whether to skip updates, check with Oracle Support for updates or search for updates locally.

   Click **Next**.

3. On the Welcome screen click **Next**.

4. On the Prerequisite Checks screen, verify that the checks complete successfully, then click **Next**.

5. On the Specify Installation Location screen, enter the following values:

   - **Oracle MiddleWare Home**: Select a previously installed Middleware Home from the drop-down list. For example: *IAM_MW_HOME*
   - **Oracle Home Directory**: Enter `iam` as the Oracle home directory name.

   Click **Next**.

6. On the Application Server Screen select **WebLogic Server** and click **Next**.

7. On the Installation Summary screen, click **Install**.

8. On the Installation Progress screen, click **Next**.

9. On the Installation Complete screen, click **Finish**.

8.2.3 Installing the Oracle SOA Suite

Perform these steps to install the Oracle SOA Suite.

Ensure that the system, patch, kernel and other requirements are met. These are listed in the *Oracle Fusion Middleware Installation Guide for Oracle SOA Suite and Oracle Business Process Management Suite* in the Oracle Fusion Middleware documentation library for the platform and version you are using.

To start the Oracle Fusion Middleware 11*g* SOA Suite Installer, change directory to Disk1 of the installation media and enter the following command.

```
./runInstaller
```

When the installer prompts you for a JRE/JDK location, enter the Oracle SDK location created in the Oracle WebLogic Server installation, for example:

IAM_MW_HOME`/jrockit_`*version*

Then perform these installation steps:

1. On the Specify Inventory Directory screen, enter values for the Oracle Inventory Directory and the Operating System Group Name. For example:

   - **Specify the Inventory Directory**: `/u02/private/oracle/oraInventory`
   - **Operating System Group Name**: `oinstall`

   A dialog box appears with the following message:

   ```
   Certain actions need to be performed with root privileges before the install
   can continue. Please execute the script
   /u02/private/oracle/oraInventory/createCentralInventory.sh now from another
   window and then press "Ok" to continue the install. If you do not have the root
   privileges and wish to continue the install select the "Continue installation
   with local inventory" option.
   ```

   Log in as `root` and run:

   ```
   /u02/private/oracle/oraInventory/createCentralInventory.sh
   ```

This sets the required permissions for the Oracle Inventory Directory and then brings up the Welcome screen.

Note: The Oracle Inventory screen is not shown if an Oracle product was previously installed on the host. If the Oracle Inventory screen is not displayed for this installation, check the following:

1. The `/etc/oraInst.loc` file exists.
2. The Inventory directory listed is valid.
3. The user performing the installation has write permissions for the Inventory directory.

---

2. On the Welcome screen, click **Next**.
3. On the **Install Software Updates** screen, choose whether to register with Oracle Support for updates or search for updates locally.

   Click **Next**.
4. On the Prerequisite Checks screen, verify that the checks complete successfully, and then click **Next**.
5. On the Specify Installation Location screen, enter the following values:
   - **Oracle Middleware Home**: Select a previously installed Middleware Home from the drop-down list. For example: *`IAM_MW_HOME`*
   - **Oracle Home Directory**: Enter `soa` as the Oracle home directory name.

   Note: You must use the same Oracle home directory name for Oracle SOA Suite on all hosts.

   ---
6. Click **Next**.
7. On the Application Server screen, choose your Application Server, for example: Web Logic Server.

   Click **Next**.
8. On the Installation Summary screen, click **Install**.
9. On the Installation Process screen, click **Next**.
10. On the Installation Complete screen, click **Finish**.

8.3 About Console URLs and Domains

The component URLs related to the domains, and the user names used to access them, are listed in the following table.

Table 8–2 URLs Available After Web Tier Integration

Component	URL	User
WebLogic Console	http://ADMIN.mycompany.com/console	weblogic
Fusion Middleware Control	http://ADMIN.mycompany.com/em	weblogic

8.4 Running the Configuration Wizard to Create a Domain

Run the WebLogic Configuration Wizard on IDMHOST1 once for each domain to be created. In later chapters you will extend these domains to include the components of your topology.

To create a domain:

1. Ensure that the database where you installed the repository is running. For Oracle RAC databases, all instances should be running, so that the validation check later in the procedure is more reliable.
2. Change directory to the location of the Configuration Wizard. This is within ORACLE_COMMON_HOME.

   ```
   cd ORACLE_COMMON_HOME/common/bin
   ```

3. Start the Oracle Fusion Middleware Configuration Wizard by typing:

   ```
   ./config.sh
   ```

4. On the Welcome screen, select **Create a New WebLogic Domain**, and click **Next**.
5. On the Select Domain Source screen, select the following products:
   - **Oracle Entitlements Server for Admin Server [iam]**
   - **Oracle Enterprise Manager [oracle_common]**
   - **Oracle Platform Security Service [iam]**
   - **Oracle Directory Services Manager [oud]** (if using Oracle Unified Directory)
   - **Oracle JRF [oracle_common]**

   Click **Next**.
6. On the Specify Domain Name and Location screen, enter
   - **Domain name**: `IDMDomain`
   - **Domain location**:

     ```
     /u01/oracle/config/domains
     ```

   - **Application location**:

     ```
     ASERVER_HOME/applications
     ```

   Ensure that the domain directory matches the directory and shared storage mount point recommended in Section 4.4, "About Recommended Locations for the Different Directories."

   Click **Next**.
7. On the Configure Administrator Username and Password screen, enter the username (default is `weblogic`) and password to be used for the domain's administrator. For example:
   - **Name**: `weblogic`
   - **User Password**: *`password for weblogic user`*
   - **Confirm User Password**: *`password for weblogic user`*
   - **Description:** `This user is the default administrator.`

   Click **Next**.

8. On the Configure Server Start Mode and JDK screen, do the following:
   - For WebLogic Domain Startup Mode, select **Production Mode**.
   - For JDK Selection, select **JRockit SDK**

   Click **Next**.

   Note: The next step and all steps through Step 12, "On the Test Component Schema," are only relevant if the domain being created is IDMDomain or OIMDomain.

9. On the Configure JDBC Component Schema screen, select the following:
   - **OPSS Schema**

   For the Oracle RAC configuration for component schemas, select **Convert to GridLink.**

   Click **Next**.

10. The Gridlink RAC Component Schema screen appears. In this screen, enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU.
    - **Driver**: Select **Oracle's driver (Thin) for GridLink Connections,Versions:10 and later**.
    - Select **Enable FAN**.
    - Do one of the following:
      - If **SSL** is not selected for ONS notifications to be encrypted, deselect **SSL**.
      - Select **SSL** and provide the appropriate wallet and wallet password.
    - **Service Listener**: Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the parameter `remote_listener` in the database:

```
SQL>show parameter remote_listener;
NAME            TYPE   VALUE
-----------------------------------------------------------
remote_listener string DB-SCAN.mycompany.com:1521
```

    Note:
    - For Oracle Database 11g Release 1 (11.1), use the virtual IP and port of each database instance listener, for example: `DBHOST1-VIP.mycompany.com` (port `1521`) and `DBHOST2-VIP.mycompany.com` (port `1521`), where `1521` is *`DB_LSNR_PORT`*
    - For Oracle Database 10g, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources see Appendix A, "Using Multi Data Sources with Oracle RAC."

- **ONS Host**: Enter the SCAN address for the Oracle RAC database and the ONS remote port, as reported by the database when you invoke the following command:

```
srvctl config nodeapps -s
ONS exists: Local port 6100, remote port 6200, EM port 2016
```

Note: For Oracle Database 11g Release 1 (11.1), use the hostname and port of each database's ONS service, for example: DBHOST1.mycompany.com (port 6200) and DBHOST2.mycompany.com (port 6200)

Enter the following RAC component schema information:

Schema Name	Service Name	Schema Owner	Password
OPSS Schema	OESEDG.mycompany.com	EDG_OPSS	*password*

If you prefer to use RAC Multi Data Sources, see Appendix A, "Using Multi Data Sources with Oracle RAC."

Click **Next**.

11. In the Test JDBC Data Sources screen, confirm that all connections are successful. The connections are tested automatically. The Status column displays the results. If all connections are not successful, click **Previous** to return to the previous screen and correct your entries.

    Click **Next** when all the connections are successful.

12. On the Test Component Schema screen, the Wizard attempts to validate the data sources. If the data source validation succeeds, click **Next**. If it fails, click **Previous**, correct the problem, and try again.

13. On the Select Optional Configuration screen, select the following:

    - **Administration Server**
    - **Managed Servers, Clusters and Machines**

    Click **Next**.

14. On the Configure the Administration Server screen, enter the following values:

    - **Name**: `AdminServer`
    - **Listen Address**: `ADMINVHN.mycompany.com`
    - **Listen Port**: `7001` (*`WLS_ADMIN_PORT`*)
    - **SSL Listen Port**: `7002` (*`WLS_ADMIN_SSL_PORT`*)
    - **SSL Enabled**: Selected

    Click **Next**.

15. On the Configure Managed Servers screen, click **Next**.

16. On the Configure Clusters screen, click **Next**.

17. On the Configure Machines screen, click the **Unix Machine** tab and then click **Add** to add the following machine. The machine name does not need to be a valid host name or listen address, it is just a unique identifier of a node manager location:

- **Name**: `ADMINHOST`
- **Node manager listen address**: `LOCALHOST`

Note: The virtual host machine must point to LOCALHOST because LOCALHOST is the relative internal address for whatever machine is active. The node manager associated with the Administration Server changes when the Administration Server fails over because the Administration Server uses the localhost attribute in conjunction with the first host and then again, after failover, in conjunction with the second host.

---

18. Click **Next**.
19. On the Assign Servers to Machines screen, assign servers to machines as follows:
    - ***ADMINHOST***: **AdminServer**

    where ***ADMINHOST*** is the name value entered in Step 17, for example:

    ADVINVHN.mycompany.com

    Click **Next**.
20. On the Configuration Summary screen, validate that your choices are correct, then click **Create**.
21. On the Create Domain screen, click **Done**.

8.5 Post-Configuration and Verification Tasks

After configuring the domain with the configuration Wizard, follow these instructions for post-configuration and verification.

This section includes the following topics:

8.5.1 Copying OIM Adapter Template

This section is required only if you are using Oracle Unified Directory in active-active mode, as shown in the topology diagrams.

After installing Oracle Identity and Access Management, apply Patch 16943171.

Then manually copy the file `adapter_template_oim.xml` from *ORACLE_COMMON_HOME*`/modules/oracle.ovd_11.1.1/templates/` to *IAM_ORACLE_HOME*`/libovd/`. For example:

```
cp ORACLE_COMMON_HOME/modules/oracle.ovd_11.1.1/templates/adapter_template_oim.xml
IAM_ORACLE_HOME/libovd/
```

8.5.2 Creating boot.properties for the WebLogic Administration Servers

Create a `boot.properties` file for the Administration Server on the host IDMHOST1. If the file already exists, edit it. The `boot.properties` file enables the Administration Server to start without prompting you for the administrator username and password.

For each Administration Server:

1. Create the following directory structure.

   ```
   mkdir -p ASERVER_HOME/servers/AdminServer/security
   ```

2. In a text editor, create a file called boot.properties in the last directory created in the previous step, and enter the username and password in the file. For example:

   ```
   username=weblogic
   password=password for weblogic user
   ```

3. Save the file and close the editor.

> **Note:** The username and password entries in the file are not encrypted until you start the Administration Server, as described in Section 8.5.5, "Updating the Node Manager Credentials." For security reasons, minimize the time the entries in the file are left unencrypted. After you edit the file, start the server as soon as possible so that the entries are encrypted.

8.5.3 Reassociate the Domain with the Existing OPSS Policy Store

Before starting your domain for the first time, you must reassociate the domain with the OPSS policy store in the database. To do this perform the following steps.

To reassociate the first domain with the OPSS security store use the following command:

```
ORACLE_COMMON_HOME/common/bin/wlst.sh IAM_ORACLE_
HOME/common/tools/configureSecurityStore.py -d ASERVER_HOME -c IAM -m create -p
opss_schema_password
```

Validate that the above commands have been successful by issuing the command:

```
ORACLE_COMMON_HOME/common/bin/wlst.sh IAM_ORACLE_
HOME/common/tools/configureSecurityStore.py -d ASERVER_HOME -m validate
```

8.5.4 Starting Node Manager

Perform these steps to start Node Manager on IDMHOST1 to create the `nodemanager.properties` file.

1. Run the `startNodeManager.sh` script located under the *`WL_HOME`*`/server/bin` directory.
2. Run the `setNMProps.sh` script to set the `StartScriptEnabled` property to `true`:

```
cd IAM_MW_HOME/oracle_common/common/bin
./setNMProps.sh
```

> **Note:** You must use the `StartScriptEnabled` property to avoid class loading failures and other problems.

3. Stop the Node Manager by killing the Node Manager process.

Start Node Manager on IDMHOST1 and IDMHOST2 by running the `startNodeManager.sh` script located under the *`IAM_MW_HOME`*`/wlserver_10.3/server/bin` directory.

8.5.5 Updating the Node Manager Credentials

You start the Administration server by using WLST and connecting to Node Manager. The first start of the Administration Server with Node Manager, however, requires that you change the default username and password that the Configuration Wizard sets for Node Manager. Therefore you must use the start script for the Administration Server for the first start. Follow these steps to start the Administration Server using Node Manager.

Steps 1-4 are required for the first start operation, but subsequent starts require only Step 4.

1. Start the Administration Server using the start script in the domain directory.

```
cd ASERVER_HOME/bin
./startWebLogic.sh
```

2. Use the Administration Console to update the Node Manager credentials on IDMDomain.
   a. In a browser, go to the listen address for the domain. For example:

      `http://ADMINVHN.mycompany.com:7001/console` where `7001` is *`WLS_ADMIN_PORT`*, as described in Section B.3.

   b. Log in as the administrator.
   c. Click **Lock and Edit**.
   d. Click ***domain_name*** in the Domain Structure menu.
   e. Select **Security** tab then **General** tab.
   f. Expand **Advanced Options**.
   g. Enter a new username for Node Manager or make a note of the existing one and update the Node Manager password.
   h. Click **Save**.
   i. Click **Activate Changes**.
3. Stop the WebLogic Administration Server by issuing the command `stopWebLogic.sh` located under the *`ASERVER_HOME`*`/bin` directory.

4. Start WLST and connect to the Node Manager with `nmConnect` and the credentials you just updated. Then start the WebLogic Administration Server using `nmStart`.

   ```
   cd ORACLE_COMMON_HOME/common/bin
   ./wlst.sh
   ```

   Once in the WLST shell, execute the following commands:

   ```
   nmConnect('Admin_User','Admin_Password', 'ADMINHOST1','Port',
     'domain_name','ASERVER_HOME')
   nmStart('AdminServer')
   ```

   where *Port* is *NMGR_PORT* in Section B.3, *domain_name* is the name of the domain and *Admin_User* and *Admin_Password* are the Node Manager username and password you entered in Step 2. For example:

   ```
   nmConnect('admin','password', 'IDMHOST1','5556',
     'IDMDomain','ASERVER_HOME')
   nmStart('AdminServer')
   ```

8.5.6 Validating the WebLogic Administration Server

Perform these steps to ensure that the Administration Server is properly configured:

1. In a browser, go to the Oracle WebLogic Server Administration Console at the URL:

   `http://ADMINVHN.mycompany.com:7001/console`, where `7001` is *WLS_ADMIN_PORT*, as described in Section B.3.

2. Log in as the WebLogic administrator, for example: `weblogic`.
3. Check that you can access Oracle Enterprise Manager Fusion Middleware Control at `http://ADMINVHN.mycompany.com:7001/em`.
4. Log in to Oracle Enterprise Manager Fusion Middleware Control as the WebLogic administrator, for example: `weblogic`.

8.5.7 Enabling WebLogic Plug-in

In Enterprise deployments, Oracle WebLogic Server is fronted by Oracle HTTP servers. The HTTP servers are, in turn, fronted by a load balancer, which performs SSL translation. In order for internal loopback URLs to be generated with the `https` prefix, Oracle WebLogic Server must be informed that it receives requests through the Oracle HTTP Server WebLogic plug-in.

The plug-in can be set at either the domain, cluster, or Managed Server level. Because all requests to Oracle WebLogic Server are through the Oracle OHS plug-in, set it at the domain level.

To do this perform the following steps:

1. Log in to the Oracle WebLogic Server Administration Console at `http://ADMINVHN.mycompany.com/console`.
2. Click **Lock and Edit**.
3. Click ***domain_name***, for example: **IDMDomain** in the Domain Structure Menu.
4. Click the **Configuration** tab.
5. Click the **Web Applications** sub tab.

6. Select **WebLogic Plugin Enabled**.
7. Click **Save** and **Activate the Changes**.

8.5.8 Disabling Host Name Verification for the Oracle WebLogic Administration Server

This step is required if you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server. (See Chapter 13, "Setting Up Node Manager for an Enterprise Deployment.") If you have not configured the server certificates, you will receive errors when managing the different WebLogic Servers. To avoid these errors, disable host name verification while setting up and validating the topology, and enable it again once the EDG topology configuration is complete as described in Chapter 13, "Setting Up Node Manager for an Enterprise Deployment."

Perform these steps to disable host name verification:

1. Go to the Oracle WebLogic Server Administration Console at: `http://ADMINVHN.mycompany.com:7001/console`, where `7001` is *`WLS_ADMIN_PORT`*, as described in Section B.3.
2. Log in as the user `weblogic`, using the password you specified during the installation.
3. Click **Lock** and **Edit**.
4. Expand the Environment node in the Domain Structure window.
5. Click **Servers**. The Summary of Servers page appears.
6. Select **AdminServer(admin)** in the **Name** column of the table. The Settings page for AdminServer(admin) appears.
7. Click the **SSL** tab.
8. Click **Advanced**.
9. Set Hostname Verification to **None**, if it is not already set.
10. Click **Save**.
11. Click **Activate Changes**.

8.5.9 Stopping and Starting the WebLogic Administration Server

Stop the Administration Server as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components"

Note: *`Admin_User`* and *`Admin_Password`* are only used to authenticate connections between Node Manager and clients. They are independent from the server administration ID and password and are stored in the *`ASERVER_HOME`*`/config/nodemanager/nm_password.properties` file.

8.6 Testing Manual Failover the WebLogic Administration Server

Test failover of the Administration Server to IDMHOST2 and then back to IDMHOST1, as described in Section 17.9, "Manually Failing Over the WebLogic Administration Server"

8.7 Backing Up the WebLogic Domain

Back up the Middleware home, the database and the WebLogic domain as described in Section 17.6.3, "Performing Backups During Installation and Configuration."

9 Preparing Identity Stores

This chapter describes how to prepare the Identity Store in an Oracle Identity Management enterprise deployment.

It contains the following sections:

9.1 Overview of Preparing Identity Stores

Preparing the Identity Store involves extending the schema of the directory to support Oracle Access Management Access Manager and Oracle Identity Manager, then seeding the Identity Store with system users that will be used when building the Identity Management topology.

9.2 Backing up the LDAP Directories

The procedures described in this chapter change the configuration of the LDAP directories that host the Identity Store. Before performing any of these tasks, back up your LDAP directories, as described in Section 17.6.3, "Performing Backups During Installation and Configuration."

9.3 Prerequisites

Before proceeding, ensure that the following statements are true:

- A High Availability LDAP directory, such as Oracle Unified Directory, is available.
- Other directories, such as Active Directory, are installed and available (if required).

9.4 Preparing the Identity Store

This section describes how to prepare the Identity Store. It contains the following topics:

9.4.1 Overview of Preparing the Identity Store

Before you can use a directory to support Access Manager, you must extend the directory to include Object classes required by Access Manager in the LDAP directory you are using.

In addition to extending the directory schema, you must create a number of users. These users are used later on in the guide for such things as:

- Accessing the directory using a dedicated user.
- Accessing Access Manager, the directory, and WebLogic after these products have off loaded authentication to an external directory.

9.4.2 Creating the Configuration File

Create a property file, `idstore.props`, on IDMHOST1 to use when preparing the Identity Store. The file will have the following structure:

Oracle Unified Directory Example

```
# Common
IDSTORE_HOST: IDMHOST1.mycompany.com
IDSTORE_PORT: 1389
IDSTORE_ADMIN_PORT: 4444
IDSTORE_KEYSTORE_FILE: OUD_ORACLE_INSTANCE/OUD/config/admin-keystore
IDSTORE_KEYSTORE_PASSWORD: Password key
IDSTORE_BINDDN: cn=oudadmin
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=mycompany,dc=com
IDSTORE_SEARCHBASE: dc=mycompany,dc=com
IDSTORE_USERNAMEATTRIBUTE: cn
IDSTORE_LOGINATTRIBUTE: uid
IDSTORE_USERSEARCHBASE: cn=Users,dc=mycompany,dc=com
IDSTORE_NEW_SETUP: true
POLICYSTORE_SHARES_IDSTORE: true
# OAM
IDSTORE_OAMADMINUSER:oamadmin
IDSTORE_OAMSOFTWAREUSER:oamLDAP
OAM11G_IDSTORE_ROLE_SECURITY_ADMIN:OAMAdministrators
# OAM and OIM
IDSTORE_SYSTEMIDBASE: cn=systemids,dc=mycompany,dc=com
# OIM
IDSTORE_OIMADMINGROUP: OIMAdministrators
```

```
IDSTORE_OIMADMINUSER: oimLDAP
# WebLogic
IDSTORE_WLSADMINUSER : weblogic_idm
IDSTORE_WLSADMINGROUP : WLSAdmins
```

Oracle Internet Directory Example

```
# Common
IDSTORE_HOST: OIDHOST1.mycompany.com
IDSTORE_PORT: 3060
IDSTORE_BINDDN: cn=orcladmin
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=mycompany,dc=com
IDSTORE_SEARCHBASE: dc=mycompany,dc=com
IDSTORE_USERNAMEATTRIBUTE: cn
IDSTORE_LOGINATTRIBUTE: uid
IDSTORE_USERSEARCHBASE: cn=Users,dc=mycompany,dc=com
POLICYSTORE_SHARES_IDSTORE: true
IDSTORE_NEW_SETUP: true
# OAM
IDSTORE_OAMADMINUSER:oamadmin
IDSTORE_OAMSOFTWAREUSER:oamLDAP
OAM11G_IDSTORE_ROLE_SECURITY_ADMIN:OAMAdministrators
# OAM and OIM
IDSTORE_SYSTEMIDBASE: cn=systemids,dc=mycompany,dc=com
# OIM
IDSTORE_OIMADMINGROUP: OIMAdministrators
IDSTORE_OIMADMINUSER: oimLDAP
# WebLogic
IDSTORE_WLSADMINUSER : weblogic_idm
IDSTORE_WLSADMINGROUP : WLSAdmins
```

Where:

- `IDSTORE_HOST` and `IDSTORE_PORT` are, respectively, the host and port of your Identity Store directory. Specify the back end directory here, rather than OVD. In the case of OID and OUD, specify, respectively, one of the Oracle Internet Directory or Oracle Unified Directory instances, for example:

  OID: `OIDHOST1` and `3060`

  OUD: `IDMHOST1` and `1389`

- `IDSTORE_ADMIN_PORT` (*`LDAP_DIR_ADMIN_PORT`*) is 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` is 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_ORACLE_INSTANCE/OUD/config`*. If you are not using Oracle Unified Directory, you can leave out this parameter. This file must be located on the same host that the `idmConfigTool` command is running on. The command uses this file to authenticate itself with OUD.

- `IDSTORE_KEYSTORE_PASSWORD` is the encrypted password of the Oracle Unified Directory keystore. This value can be found in the file *`OUD_ORACLE_INSTANCE`*`/OUD/config/admin-keystore.pin`. If you are not using Oracle Unified Directory, you can leave out this parameter.

- `IDSTORE_BINDDN` is an administrative user in the Identity Store Directory

- IDSTORE_GROUPSEARCHBASE is the location in the directory where Groups are Stored.
- IDSTORE_SEARCHBASE is the location in the directory where Users and Groups are stored.
- IDSTORE_USERNAMEATTRIBUTE is the name of the directory attribute containing the user's name. Note that this is different from the login name.
- IDSTORE_LOGINATTRIBUTE is the LDAP attribute which contains the users Login name.
- IDSTORE_USERSEARCHBASE is the location in the directory where Users are Stored.
- IDSTORE_NEW_SETUP is always set to true for Oracle Unified Directory. If you are not using OUD, you do not need to specify this attribute.
- POLICYSTORE_SHARES_IDSTORE is set to `true` for IDM 11*g*.
- IDSTORE_OAMADMINUSER is the name of the user you want to create as your Access Manager Administrator.
- IDSTORE_OAMSOFTWAREUSER is a user that gets created in LDAP that is used when Access Manager is running to connect to the LDAP server.
- OAM11G_IDSTORE_ROLE_SECURITY_ADMIN is the name of the group which is used to allow access to the OAM console.
- IDSTORE_SYSTEMIDBASE is the location of a container in the directory where users can be placed when you do not want them in the main user container. This happens rarely but one example is the Oracle Identity Manager reconciliation user which is also used for the bind DN user in Oracle Virtual Directory adapters.
- IDSTORE_OIMADMINGROUP Is the name of the group you want to create to hold your Oracle Identity Manager administrative users.
- IDSTORE_OIMADMINUSER is the user that Oracle Identity Manager uses to connect to the Identity store.
- IDSTORE_WLSADMINUSER: The username to be used for logging in to the web logic domain once it is enabled by SSO.
- IDSTORE_WLSADMINGROUP: is 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.

Use OIM entries only if your topology includes Oracle Identity Manager. Use OAM entries only if your topology includes Access Manager.

9.4.3 Preparing a Directory for Access Manager and Oracle Identity Manager

This section explains how to deploy Identity Management components to support Oracle Unified Directory, Oracle Internet Directory, or Active Directory as the identity store.

It contains the following topics:

- Section 9.4.3.1, "Configuring Oracle Unified Directory and Oracle Internet Directory for Use with Access Manager and Oracle Identity Manager"
- Section 9.4.3.2, "Configuring Active Directory for Use with Access Manager and Oracle Identity Manager"

9.4.3.1 Configuring Oracle Unified Directory and Oracle Internet Directory for Use with Access Manager and Oracle Identity Manager

Pre-configuring the Identity Store extends the schema in Oracle Unified Directory or Oracle Internet Directory.

> **Note:** You do not need to preconfigure the Identity Store unless you are using Access Manager or Oracle Identity Manager.

To do this, perform the following tasks on IDMHOST1:

1. Set `MW_HOME` to *`IAM_MW_HOME`*.

   Set `ORACLE_HOME` to *`IAM_ORACLE_HOME`*.

   Set `JAVA_HOME` to *`JAVA_HOME`*.

2. Configure the Identity Store by using the command `idmConfigTool`, which is located at:

   `IAM_ORACLE_HOME``/idmtools/bin`

   > **Note:** When you run the `idmConfigTool`, it creates or appends to the file `idmDomainConfig.param`. This file is generated in the same directory that the `idmConfigTool` is run from. To ensure that each time the tool is run, the same file is appended to, always run the `idmConfigTool` from the directory:
   >
   > *`IAM_ORACLE_HOME`*`/idmtools/bin`

   The syntax of the command is:

   ```
   idmConfigTool.sh -preConfigIDStore input_file=configfile
   ```

   For example:

   ```
   idmConfigTool.sh -preConfigIDStore input_file=idstore.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.

   Sample command output:

   ```
   Enter ID Store Bind DN password :
   Dec 4, 2012 11:39:19 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
   INFO: -> LOADING: /u01/oracle/products/access/iam/idmtools/templates/oud/oud_
   schema_extn.ldif
   Dec 4, 2012 11:39:20 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
   INFO: -> LOADING:
   /u01/oracle/products/access/iam/oam/server/oim-intg/ldif/ojd/schema/ojd_oam_
   pwd_schema_add.ldif
   Dec 4, 2012 11:39:20 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
   INFO: -> LOADING:
   /u01/oracle/products/access/iam/oam/server/oim-intg/ldif/ojd/schema/ojd_user_
   schema_add.ldif
   Dec 4, 2012 11:39:20 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
   INFO: -> LOADING:
   /u01/oracle/products/access/iam/oam/server/oim-intg/ldif/ojd/schema/ojd_user_
   index_generic.ldif
   ```

```
Dec 4, 2012 11:39:21 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/oracle/products/access/iam/idmtools/templates/oud/add_
oraclecontext_container.ldif
Dec 4, 2012 11:39:21 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/oracle/products/access/iam/idmtools/templates/oud/oud_
indexes_extn.ldif
Dec 4, 2012 11:39:21 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/oracle/products/access/iam/idmtools/templates/oud/idm_
idstore_groups_template.ldif
Dec 4, 2012 11:39:21 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/oracle/products/access/iam/idmtools/templates/oud/idm_
idstore_groups_acl_template.ldif
Dec 4, 2012 11:39:21 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING:
/u01/oracle/products/access/iam/idmtools/templates/oud/systemid_pwdpolicy.ldif
Dec 4, 2012 11:39:21 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/oracle/products/access/iam/idmtools/templates/oud/fa_
pwdpolicy.ldif
The tool has completed its operation. Details have been logged to
automation.log
```

3. 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.

Note: In addition to creating users, `idmConfigTool` creates the following groups:

- `orclFAUserReadPrivilegeGroup`
- `orclFAUserWritePrivilegeGroup`
- `orclFAUserWritePrefsPrivilegeGroup`
- `orclFAGroupReadPrivilegeGroup`
- `orclFAGroupWritePrivilegeGroup`

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

9.4.3.2 Configuring Active Directory for Use with Access Manager and Oracle Identity Manager

This section describes how to configure Active Directory. Extend the schema in Active Directory as follows.

Note: The order in which you perform the steps is critical!

---

1. Locate the following files:

   IDM_ORACLE_HOME`/oam/server/oim-intg/ldif/ad/schema/ADUserSchema.ldif`

   IDM_ORACLE_HOME`/oam/server/oim-intg/ldif/ad/schema/AD_oam_pwd_schema_add.ldif`

2. In both these files, replace the `domain-dn` with the appropriate `domain-dn` value

3. Use `ldapadd` from the command line to load the two LDIF files, as follows.

   ```
   ldapadd -h activedirectoryhostname -p activedirectoryportnumber -D AD_
   administrator -q -c -f file
   ```

   where *AD_administrator* is a user which has schema extension privileges to the directory

   For example:

   ```
   ldapadd -h "ACTIVEDIRECTORYHOST.mycompany.com" -p 389 -D adminuser –q -c -f
   ADUserSchema.ldif
   ldapadd -h "ACTIVEDIRECTORYHOST.mycompany.com" -p 389 -D adminuser -q -c -f AD_
   oam_pwd_schema_add.ldif
   ```

   Note: After the `-D` you can specify either a DN or *user@domain.com*.

4. Then go to:

   IAM_MW_HOME`/oracle_common/modules/oracle.ovd_11.1.1/oimtemplates`

   Run the following command to extend Active Directory schema:

   ```
   sh extendadschema.sh -h AD_host -p AD_port -D 'administrator@mydomain.com' -AD
   "dc=mydomain,dc=com" -OAM true
   ```

9.4.4 Creating 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 IDMHOST1:

1. Set `MW_HOME` to *IAM_MW_HOME*.

   Set `ORACLE_HOME` to *IAM_ORACLE_HOME*.

   Set `JAVA_HOME` to *JAVA_HOME*.

2. Configure the Identity Store by using the command `idmConfigTool`, which is located at:

   IAM_ORACLE_HOME`/idmtools/bin`

   Note: When you run the `idmConfigTool`, it creates or appends to the file `idmDomainConfig.param`. This file is generated in the same directory that the `idmConfigTool` is run from. To ensure that each time the tool is run, the same file is appended to, always run the `idmConfigTool` from the directory:

   IAM_ORACLE_HOME`/idmtools/bin`

   The syntax of the command on Linux is:

   ```
   idmConfigTool.sh -prepareIDStore mode=MODE input_file=configfile
   ```

The value selected for *MODE* determines the type of users to be created. Possible values for *MODE* include: `OAM`, `OIM`, and `WLS`.

Run the command once for each of the components that is in your topology.

- 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 Administration Console and Oracle Enterprise Manager Fusion Middleware Control. Type:

  ```
  idmConfigTool.sh -prepareIDStore mode=WLS input_file=idstore.props
  ```

  Run this command first.

- If your topology includes Access Manager, you must seed the Identity Store with users that are required by Access Manager. Type:

  ```
  idmConfigTool.sh -prepareIDStore mode=OAM input_file=idstore.props
  ```

- If your topology includes Oracle Identity Manager, you must seed the Identity Store with the `xelsysadm` user and assign it to an Oracle Identity Manager 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:

  ```
  idmConfigTool.sh -prepareIDStore mode=OIM input_file=idstore.props
  ```

  Note: This command also creates a container in your Identity Store for reservations.

  The password assigned to the `xelsysadm` user must conform to the following rules:

  - Six characters or more
  - One or more numeric character
  - Two or more alphabetic characters
  - Start with alphabetic character
  - One or more lowercase character

When the command runs, you are prompted to enter the password of the account you are connecting to the Identity Store with.

3. 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.

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

9.4.5 Add Missing Oracle Internet Directory Object Class

Bug 14341069 is caused by a missing object class in Oracle Internet Directory. The workaround it is to add this object class manually.

1. Create a file called update_oid.ldif with the following contents:

```
dn: cn=subschemasubentry
changetype: modify
delete: objectclasses
objectclasses: ( 2.16.840.1.113894.200.2.1 NAME 'orclIDXPerson' SUP
inetorgperson AUXILIARY MAY ( middleName $ orclActiveStartDate $
orclActiveEndDate $ orclIsEnabled $ orclTimeZone $ c $ orclGenerationQualifier
$ orclHireDate $ orclAccessibilityMode $ orclColorContrast $ orclFontSize $
orclnumberFormat $ orclcurrency $ orcldateFormat $ orcltimeFormat $
orclembeddedHelp $ orclFALanguage $ orclFATerritory $
orclDisplayNameLanguagePreference $ orclImpersonationGranter $
orclImpersonationGrantee $ orclMTTenantGUID $ orclMTTenantUName $ orclMTUid $
orclFAUserID $ orclFAPersonID $ orclFAPartyID ))

dn: cn=subschemasubentry
changetype: modify
add: attributetypes
attributetypes: ( 2.16.840.1.113894.200.1.7 NAME 'orclPwdExpirationDate'
EQUALITY caseIgnoreMatch SYNTAX '1.3.6.1.4.1.1466.115.121.1.15' SINGLE-VALUE
USAGE userApplications )

dn: cn=subschemasubentry
changetype: modify
add: objectclasses
objectclasses: ( 2.16.840.1.113894.200.2.1 NAME 'orclIDXPerson' SUP
inetorgperson AUXILIARY MAY ( middleName $ orclActiveStartDate $
orclActiveEndDate $ orclIsEnabled $ orclTimeZone $ c $ orclGenerationQualifier
$ orclHireDate $ orclAccessibilityMode $ orclColorContrast $ orclFontSize $
orclnumberFormat orclcurrency $ orcldateFormat $ orcltimeFormat $
orclembeddedHelp $ orclFALanguage $ orclFATerritory $
orclDisplayNameLanguagePreference $ orclImpersonationGranter $
orclImpersonationGrantee $ orclMTTenantGUID $ orclMTTenantUName $ orclMTUid $
orclFAUserID $ orclFAPersonID $ orclFAPartyID $ orclPwdExpirationDate ) )
```

2. Update Oracle Internet Directory using the command:

```
ldapmodify -D cn=orcladmin -h OIDHOST1.mycompany.com -p 3060 -f update_oid.ldif
```

9.4.6 Add Missing Oracle Unified Directory Permission

This section describes a workaround for a missing permission in Oracle Unified Directory.

Create a file called add_aci.ldif with the following contents:

```
dn: cn=Reserve,dc=mycompany,dc=com
changetype: modify
delete: aci
aci: (version 3.0; acl "oim reserve group container acl"; allow (read,add,delete)
groupdn="ldap:///cn=OIMAdministrators,cn=Groups,dc=mycompany,dc=com"; deny (all)
userdn="ldap:///anyone";)

dn: cn=Reserve,dc=mycompany,dc=com
changetype: modify
add: aci
aci: (target = "ldap:///cn=Reserve,dc=mycompany,dc=com")(targetattr = "*")(version
3.0; acl "Allow OIMAdministrators Group add, read and write access to all
attributes"; allow (add, read, search, compare,write, delete, import,export)
(groupdn = "ldap:///cn=OIMAdministrators,cn=Groups,dc=mycompany,dc=com");)
```

Update Oracle Unified Directory using the command:

```
ldapmodify -D cn=oudadmin -h IDMHOST1.mycompany.com -p 1389 -f add_aci.ldif
```

9.4.7 Granting Oracle Unified Directory Change Log Access

If you are using Oracle Unified Directory and Oracle Identity Manager, you must now grant access to the changelog. You do this by performing the following steps on all OUD hosts, that is, on IDMHOST1 and IDMHOST2:

1. On the host where OUD is running (for example, IDMHOST), create a file called `mypasswordfile` that contains the password you use to connect to OUD.
2. Remove the existing change log permission by issuing the command on one of the replicated OUD hosts:

```
OUD_ORACLE_INSTANCE/bin/dsconfig set-access-control-handler-prop \
--remove
global-aci:"(target=\"ldap:///cn=changelog\")(targetattr=\"*\")(version 3.0;
acl \"External changelog access\"; deny (all) userdn=\"ldap:///anyone\";)" \
        --hostname OUD_HOST \
        --port OUD_ADMIN_PORT \
        --trustAll \
        --bindDN cn=oudadmin \
        --bindPasswordFile passwordfile \
        --no-prompt
```

   For example:

```
OUD_ORACLE_INSTANCE/bin/dsconfig set-access-control-handler-prop \
--remove
global-aci:"(target=\"ldap:///cn=changelog\")(targetattr=\"*\")(version 3.0;
acl \"External changelog access\"; deny (all) userdn=\"ldap:///anyone\";)" \
        --hostname IDMHOST1.mycompany.com \
        --port 4444 \
        --trustAll  \
        --bindDN cn=oudadmin \
        --bindPasswordFile mypasswordfile \
        --no-prompt
```

3. Then add the following new ACI:

```
OUD_ORACLE_INSTANCE/bin/dsconfig set-access-control-handler-prop \
--add global-aci:"(target=\"ldap:///cn=changelog\")(targetattr=\"*\")(version
3.0; acl \"External changelog access\"; allow
(read,search,compare,add,write,delete,export)
groupdn=\"ldap:///cn=OIMAdministrators,cn=groups,dc=mycompany,dc=com\";)" \
        --hostname OUD_HOST \
        --port OUD_ADMIN_PORT \
        --trustAll \
        --bindDN cn=oudadmin \
        --bindPasswordFile passwordfile \
        --no-prompt
```

   For example:

```
OUD_ORACLE_INSTANCE/bin/dsconfig set-access-control-handler-prop \
--add global-aci:"(target=\"ldap:///cn=changelog\")(targetattr=\"*\")(version
3.0; acl \"External changelog access\"; allow
(read,search,compare,add,write,delete,export)
groupdn=\"ldap:///cn=OIMAdministrators,cn=groups,dc=mycompany,dc=com\";)" \
        --hostname IDMHOST1.mycompany.com \
        --port 4444 \
        --trustAll \
```

```
        --bindDN cn=oudadmin \
        --bindPasswordFile mypasswordfile \
        --no-prompt
```

4. Then add the following new ACI:

```
OUD_ORACLE_INSTANCE/bin/dsconfig set-access-control-handler-prop \
--add global-aci:"(targetcontrol=\"1.3.6.1.4.1.26027.1.5.4\")(version 3.0; acl
\"OIMAdministrators control access\"; allow(read)
groupdn=\"ldap:///cn=OIMAdministrators,cn=groups,dc=mycompany,dc=com\";)" \
        --hostname OUD_HOST \
        --port OUD_ADMIN_PORT \
        --trustAll \
        --bindDN cn=oudadmin \
        --bindPasswordFile passwordfile \
        --no-prompt
```

For example:

```
OUD_ORACLE_INSTANCE/bin/dsconfig set-access-control-handler-prop \
--add global-aci:"(targetcontrol=\"1.3.6.1.4.1.26027.1.5.4\")(version 3.0; acl
\"OIMAdministrators control access\"; allow(read)
groupdn=\"ldap:///cn=OIMAdministrators,cn=groups,dc=mycompany,dc=com\";)" \
        --hostname IDMHOST1.mycompany.com \
        --port 4444 \
        --trustAll \
        --bindDN cn=oudadmin \
        --bindPasswordFile mypasswordfile \
        --no-prompt
```

5. Then add the following ACI:

```
OUD_ORACLE_INSTANCE/bin/dsconfig set-access-control-handler-prop \
--add
global-aci:"(target=\"ldap:///\")(targetscope=\"base\")(targetattr=\"lastExtern
alChangelogCookie\")(version 3.0; acl \"User-Visible lastExternalChangelog\";
allow (read,search,compare)
groupdn=\"ldap:///cn=OIMAdministrators,cn=groups,dc=mycompany,dc=com\";)" \
        --hostname OUD_HOST \
        --port OUD_ADMIN_PORT \
        --trustAll \
        --bindDN cn=oudadmin \
        --bindPasswordFile passwordfile \
        --no-prompt
```

For example:

```
OUD_ORACLE_INSTANCE/bin/dsconfig set-access-control-handler-prop \
--add
global-aci:"(target=\"ldap:///\")(targetscope=\"base\")(targetattr=\"lastExtern
alChangelogCookie\")(version 3.0; acl \"User-Visible lastExternalChangelog\";
allow (read,search,compare)
groupdn=\"ldap:///cn=OIMAdministrators,cn=groups,dc=mycompany,dc=com\";)" \
        --hostname IDMHOST1.mycompany.com \
        --port 4444 \
        --trustAll \
        --bindDN cn=oudadmin \
        --bindPasswordFile mypasswordfile \
        --no-prompt
```

9.4.8 Creating Oracle Unified Directory Indexes

When you run the `idmConfigTool` to prepare an Oracle Unified Directory identity store, it creates indexes for the data on the instance against which it is run. You must manually create these indexes on each of the remaining Oracle Unified Directory instances in the configuration.

To do this, on IDMHOST2, issue the following commands:

```
OUD_ORACLE_INSTANCE/OUD/bin/ldapmodify -h IDMHOST2.mycompany.com -Z -X -p 4444 -a
-D "cn=oudadmin" -j mypasswordfile -c  -f IAM_ORACLE_
HOME/oam/server/oim-intg/ldif/ojd/schema/ojd_user_index_generic.ldif

OUD_ORACLE_INSTANCE/OUD/bin/ldapmodify -h IDMHOST2.mycompany.com -Z -X -p 4444 -a
-D "cn=oudadmin" -j mypasswordfile -c  -f IAM_ORACLE_
HOME/idmtools/templates/oud/oud_indexes_extn.ldif
```

Once the indexes have been created on every IDMHOST, rebuild the indexes as follows:

1. Shut down Oracle Unified Directory by issuing the command:

   ```
   OUD_ORACLE_INSTANCE/OUD/bin/stop-ds
   ```

2. Execute the command:

   ```
   OUD_ORACLE_INSTANCE/OUD/bin/rebuild-index --rebuildAll -b "dc=mycompany,dc=com"
   ```

3. Restart Oracle Unified Directory by issuing the command:

   ```
   OUD_ORACLE_INSTANCE/OUD/bin/start-ds
   ```

Repeat Steps 1-3 to rebuild the indexes for every IDMHOST, including the host which the `idmConfigTool` was run against, to maintain availability only stop the directory for which you are rebuilding the indexes.

9.4.9 Creating Access Control Lists in Directories Other than Oracle Internet Directory and Oracle Unified Directory

In the preceding sections, you seeded the Identity Store with users and artifacts for the Oracle components. If your Identity Store is hosted in a directory other than Oracle Internet Directory or Oracle Unified Directory, such as Microsoft Active Directory, you must set up the access control lists (ACLs) to provide appropriate privileges to the entities you created. 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 Manager user `oimLDAP` 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 Manager administration group. The Oracle Identity Manager user is added as its member. The Oracle Identity Manager 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 Manager admin group to perform read/write operations.

9.5 Creating Adapters in Oracle Virtual Directory

If you access your LDAP directory through Oracle Virtual Directory, you must link Oracle Virtual Directory to the back end LDAP directory by creating adapters. This section describes how.

The procedure is slightly different, depending on the directory you are connecting to. The following sections show how to create and validate adapters for supported directories:

- Section 9.5.1, "Ensuring the Change Log Generation is Enabled in Oracle Internet Directory"
- Section 9.5.2, "Creating Oracle Virtual Directory Adapters for Oracle Internet Directory and Active Directory"
- Section 9.5.3, "Validating the Oracle Virtual Directory Adapters"

9.5.1 Ensuring the Change Log Generation is Enabled in Oracle Internet Directory

Before you create a change log adapter in Oracle Virtual Directory, you must ensure that the back end Oracle Internet Directory servers have changelog generation enabled.

To test whether a directory server has changelog generation enabled, type:

```
ldapsearch -h directory_host -p ldap_port -D bind_dn -q -b '' -s base
'objectclass=*' lastchangenumber
```

For example:

```
ldapsearch -h OIDHOST1 -p 3060 -D "cn=orcladmin" -q -b '' -s base 'objectclass=*'
lastchangenumber
```

If the command output includes `lastchangenumber` with a value, changelog generation is enabled. If changelog generation is not enabled, enable it as described in the "Enabling and Disabling Changelog Generation by Using the Command Line" section of *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*.

9.5.2 Creating Oracle Virtual Directory Adapters for Oracle Internet Directory and Active Directory

You can use `idmConfgTool` to create the Oracle Virtual Directory User and Changelog adapters for Oracle Internet Directory and Active Directory. Oracle Identity Manager requires adapters. It is highly recommended, though not mandatory, that you use Oracle Virtual Directory to connect to Oracle Internet Directory.

To do this, perform the following tasks on IDMHOST1:

1. Set `MW_HOME` to *`IAM_MW_HOME`*.

   Set `ORACLE_HOME` to *`IAM_ORACLE_HOME`*.

   Set `JAVA_HOME` to *`JAVA_HOME`*.

2. Create a properties file for the adapter you are configuring called `ovd1.props`. The contents of this file depends on whether you are configuring the Oracle Internet Directory adapter or the Active Directory Adapter.

   - **Oracle Internet Directory** adapter properties file:

     ```
     ovd.host:OVDHOST1.mycompany.com
     ovd.port:8899
     ovd.binddn:cn=orcladmin
     ovd.password:ovdpassword
     ovd.oamenabled:true
     ovd.ssl:true
     ldap1.type:OID
     ldap1.host:OIDIDSTORE.mycompany.com
     ldap1.port:3060
     ldap1.binddn:cn=oimLDAP,cn=systemids,dc=mycompany,dc=com
     ldap1.password:oidpassword
     ldap1.ssl:false
     ldap1.base:dc=mycompany,dc=com
     ldap1.ovd.base:dc=mycompany,dc=com
     usecase.type: single
     ```

   - **Active Directory** adapter properties file:

     ```
     ovd.host:OVDHOST1.mycompany.com
     ovd.port:8899
     ovd.binddn:cn=orcladmin
     ovd.password:ovdpassword
     ovd.oamenabled:true
     ovd.ssl:true
     ldap1.type:AD
     ldap1.host:ADIDSTORE.mycompany.com
     ldap1.port:636
     ldap1.binddn:cn=adminuser
     ldap1.password:adpassword
     ldap1.ssl:true
     ldap1.base:dc=mycompany,dc=com
     ldap1.ovd.base:dc=mycompany,dc=com
     usecase.type: single
     ```

   The following list describes the parameters used in the properties file.

   - `ovd.host` is the host name of a server running Oracle Virtual Directory.
   - `ovd.port` is the https port used to access Oracle Virtual Directory.
   - `ovd.binddn` is the user DN you use to connect to Oracle Virtual Directory.
   - `ovd.password` is the password for the DN you use to connect to Oracle Virtual Directory.
   - `ovd.oamenabled` is always `true` in Fusion Applications deployments.
   - `ovd.ssl` is set to `true`, as you are using an https port.
   - `ldap1.type` is set to OID for the Oracle Internet Directory back end directory or set to AD for the Active Directory back end directory.

- `ldap1.host` is the host on which back end directory is located. Use the load balancer name.
- `ldap1.port` is the port used to communicate with the back end directory.
- `ldap1.binddn` is the bind DN of the `oimLDAP` user.
- `ldap1.password` is the password of the `oimLDAP` user
- `ldap1.ssl` is set to `true` if you are using the back end's SSL connection, and otherwise set to `false`. This should always be set to `true` when an adapter is being created for AD.
- `ldap1.base` is the base location in the directory tree.
- `ldap1.ovd.base` is the mapped location in Oracle Virtual Directory.
- `usecase.type` is set to `Single` when using a single directory type.

3. Configure the adapter by using the `idmConfigTool` command, which is located at:

   IAM_ORACLE_HOME`/idmtools/bin`

   Note: When you run the `idmConfigTool`, it creates or appends to the file `idmDomainConfig.param`. This file is generated in the same directory that the `idmConfigTool` is run from. To ensure that each time the tool is run, the same file is appended to, always run the `idmConfigTool` from the directory:

   IAM_ORACLE_HOME`/idmtools/bin`

   The syntax of the command on Linux is:

   ```
   idmConfigTool.sh -configOVD input_file=configfile [log_file=logfile]
   ```

   For example:

   ```
   idmConfigTool.sh -configOVD input_file=ovd1.props
   ```

   The command requires no input. The output looks like this:

   ```
   The tool has completed its operation. Details have been logged to logfile
   ```

Run this command for each Oracle Virtual Directory instance in your topology, with the appropriate value for `ovd.host` in the property file.

9.5.3 Validating the Oracle Virtual Directory Adapters

Perform the following tasks by using ODSM:

1. Access ODSM at:

   `http://`*HOSTNAME*`.mycompany.com:`*port*`/odsm`

2. Connect to Oracle Virtual Directory.
3. Go the **Data Browser** tab.
4. Expand **Client View** so that you can see each of your user adapter root DN's listed.
5. Expand the user adapter root DN, if there are objects already in the back end LDAP server, you should see those objects here.

6. ODSM doesn't support changelog query, so you cannot expand the `cn=changelog` subtree.

   Perform the following tasks by using the command-line:

   - Validate the user adapters by typing:

     ```
     ldapsearch -h directory_host -p ldap_port -D "cn=orcladmin" -q  -b <user_
     search_base> -s sub "objectclass=inetorgperson" dn
     ```

     For example:

     ```
     ldapsearch -h OVDHOST1.mycompany.com -p 6501 -D "cn=orcladmin" -q -b
     "cn=Users,dc=mycompany,dc=com" -s sub "objectclass=inetorgperson" dn
     ```

     Supply the password when prompted.

     You should see the user entries that already exist in the back end LDAP server.

   - Validate changelog adapters by typing:

     ```
     ldapsearch -h directory_host -p ldap_port -D "cn=orcladmin" -q  -b
     "cn=changelog" -s one "changenumber>=0"
     ```

     For example:

     ```
     ldapsearch -h OVDHOST1 -p 6501 -D "cn=orcladmin" -q -b "cn=changelog" -s
     one "changenumber>=0"
     ```

     The command returns logs of data, such as creation of all the users. It returns without error if the changelog adapters are valid.

   - Validate lastchangenumber query by typing:

     ```
     ldapsearch -h directory_host -p ldap_port -D "cn=orcladmin" -q -b
     "cn=changelog" -s base 'objectclass=*' lastchangenumber
     ```

     For example:

     ```
     ldapsearch -h OVDHOST1 -p 6501 -D "cn=orcladmin" -q -b "cn=changelog" -s
     base 'objectclass=*' lastchangenumber
     ```

     The command returns the latest change number generated in the back end LDAP server.

9.6 Backing Up the Identity Stores

Back up your LDAP directories, as described in Section 17.6.3, "Performing Backups During Installation and Configuration."

10

Installing and Configuring Oracle Web Tier for an Enterprise Deployment

This chapter describes how to configure the Oracle Web Tier for an Oracle Identity Management enterprise deployment.

This chapter includes the following topics:

10.1 Overview of Installing and Configuring the Web Tier

This chapter describes how to install Oracle HTTP server and associate the Oracle Web Tier with the WebLogic Server domain. Once the Web tier is associated with the WebLogic Server, you can monitor it using the Oracle Fusion Middleware Console.

You then configure the load balancer to route all HTTP requests to WEBHOST1 and WEBHOST2.

The last section describes how to define the Oracle HTTP Server directives to route requests to the load balancer virtual hosts you defined in Chapter 3, "Preparing the Network for an Enterprise Deployment."

10.2 Install and Configure the Web Tier

This section contains the following topics:

10.2.1 Prerequisites

- Before configuring the Oracle Web Tier software, you must install it on WEBHOST1 and WEBHOST2, as described in Section 10.2.3, "Installing Oracle HTTP Server." Run the Configuration Wizard to define the instance home, the instance name, and the Oracle HTTP Server component name.
- Ensure that port *7777* (*OHS_PORT*) is not in use. Because Oracle HTTP Server is installed by default on port *7777*, you must ensure that port *7777* is not used by any other service on the nodes. To check if this port is in use, run the following command before installing Oracle HTTP Server. You must free the port if it is in use.

  ```
  netstat -an | grep 7777
  ```

- Create a file containing the ports used by Oracle HTTP Server. On Disk1 of the installation media, locate the file `stage/Response/staticports.ini`. Copy it to a file called `ohs_ports.ini`. Delete all entries in `ohs_ports.ini` except for `OHS PORT` and `OPMN Local Port`. Change the values of those ports to `7777` and `6700`, respectively.

  Note: If the port names in the file are slightly different from `OHS PORT` and `OPMN Local Port`, use the names in the file.

10.2.2 Installing Oracle JRockit

Install Oracle JRockit on WEBHOST1 and WEBHOST2 as described in Section 8.2.1.1, "Installing Oracle JRockit."

10.2.3 Installing Oracle HTTP Server

This section explains how to install Oracle HTTP Server on WEBHOST1 and WEBHOST2.

This section contains the following topics:

- Section 10.2.3.1, "Verifying Prerequisites"
- Section 10.2.3.2, "Running the Installer"

10.2.3.1 Verifying Prerequisites

Prior to installing the Oracle HTTP server, check that your machines meet the following requirements:

1. Ensure that the system, patch, kernel, and other requirements are met as specified in *Oracle Fusion Middleware Installation Guide for Oracle Web Tier*.
2. On Linux platforms, if the `/etc/oraInst.loc` file exists, verify that its contents are similar to this:

   ```
   inventory_loc=/u02/private/oracle/oraInventory
   inst_group=oinstall
   ```

   Ensure that the inventory directory is correct and that you have write permissions for that directory.

   If the `/etc/oraInst.loc` file does not exist, you can skip this step.

10.2.3.2 Running the Installer

As described in Section 4.4, "About Recommended Locations for the Different Directories," you install the Oracle HTTP Server onto a local disk. You can install it on shared storage, but if you do that, you must allow access from the Web Tier DMZ to your shared disk array, which is undesirable. If you decide to install onto shared disk then please see the Release Notes for further configuration information.

Before Starting the install, ensure that the following environment variables are not set on Linux platforms.

- `LD_ASSUME_KERNEL`
- `ORACLE_INSTANCE`

To start Oracle Universal Installer on Linux, change directory to Disk 1 of the installation media and issue the command

```
./runInstaller
```

Proceed as follows:

1. On the Specify Oracle Inventory Directory screen, enter *`HOME`*`/oraInventory`, where *`HOME`* is the home directory of the user performing the installation. (This is the recommended location).

   Enter the OS group for the user performing the installation.

   Click **Next**.

2. On the Welcome screen, click **Next**.

3. On the Install Software Updates screen, choose whether to skip updates, check with Oracle Support for updates or search for updates locally.

   Click **Next**.

4. On the Select Installation Type screen, select **Install Software** –> **Do Not Configure**

   Click **Next**.

5. On the Prerequisite Checks screen, click **Next**.

6. On the Specify Installation Location screen, specify the following values:

   - **Oracle Middleware Home Location (Installation Location)**: *`WEB_MW_HOME`*. For example: `/u02/private/oracle/products/web`
   - **Oracle Home Directory**: `web`

7. On the Specify Security Updates screen, choose whether to receive security updates from Oracle support.

   Click **Next**.

8. On the Installation Summary screen, review the selections to ensure that they are correct (if they are not, click **Back** to modify selections on previous screens), and click **Install**.

10.2.4 Running the Configuration Wizard to Configure the HTTP Server

The steps for configuring the Oracle Web Tier are the same for WEBHOST1 and WEBHOST2.

Perform these steps to configure the Oracle web tier:

1. Change the directory to the location of the Oracle Fusion Middleware Configuration Wizard:

   ```
   cd WEB_ORACLE_HOME/bin
   ```

2. Start the Configuration Wizard:

   ```
   ./config.sh
   ```

Enter the following information into the configuration wizard:

1. On the Welcome screen, click **Next**.
2. On the Configure Component screen, select: **Oracle HTTP Server**.

   Ensure that Associate Selected Components with WebLogic Domain is selected.

   Ensure Oracle Web Cache is **NOT** selected.

   Click **Next**.
3. On the Specify WebLogic Domain Screen, enter
   - **Domain Host Name**: `ADMINVHN.mycompany.com`
   - **Domain Port No**: `7001`, where `7001` is *WLS_ADMIN_PORT* in Section B.3.
   - **User Name**: Weblogic Administrator User (For example: `weblogic`)
   - **Password**: Password for the Weblogic Administrator User account

   Click **Next**.
4. On the Specify Component Details screen, specify the following values:

   Enter the following values for WEBHOST*n*, where *n* is 1 or 2:

   - **Instance Home Location**: *WEB_ORACLE_INSTANCE* (`/u02/private/oracle/config/instances/webn`)
   - **Instance Name**: `webn`
   - **OHS Component Name**: `ohsn`

   Click **Next**.
5. On the Configure Ports screen, you use the `ohs_ports.ini` file you created in Section 10.2.1, "Prerequisites" to specify the ports to be used. This enables you to bypass automatic port configuration.
   a. Select **Specify Ports using a Configuration File**.
   b. In the file name field specify `ohs_ports.ini`.
   c. Click **Browse**, then click **Next**.
6. In the Specify Security Updates screen, choose whether you want to receive security updates from Oracle support and if you do, enter your e-mail address.
7. On the Installation Summary screen, review the selections to ensure that they are correct. If they are not, click **Back** to modify selections on previous screens.

   Click **Configure**.

   On the Configuration screen, the wizard launches multiple configuration assistants. This process can be lengthy. When it completes, click **Next**.

   On the Installation Complete screen, click **Finish** to confirm your choice to exit.

10.3 Post Configuration Tasks

This section describes tasks for configuring Oracle HTTP Server for the WebLogic Domain, and for verifying the configuration. Perform these steps on each web host.

This section includes the following topics:

- Section 10.3.1, "Configuring Oracle HTTP Server to Run as Software Owner"
- Section 10.3.2, "Update Oracle HTTP Server Runtime Parameters"
- Section 10.3.3, "Creating Virtual Hosts to Support Identity Management"

10.3.1 Configuring Oracle HTTP Server to Run as Software Owner

By default, the Oracle HTTP server runs as the user `nobody`. In the Identity Management installation, the Oracle HTTP server should run as the Software owner and group.

To cause it to run as the appropriate user and group, edit the file `httpd.conf`, which is located in:

WEB_ORACLE_INSTANCE`/config/OHS/`*component_name*

Find the section in `httpd.conf` where `User` is defined.

Change this section to read:

```
User User_who_installed_the_software
Group Group_under_which_the_HTTP_server_runs
```

`Group` is typically the default user group, for example: `oinstall`.

For example:

```
<IfModule !mpm_winnt_module>
#
# If you wish httpd to run as a different user or group, you must run
# httpd as root initially and it will switch.
#
# User/Group: The name (or #number) of the user/group to run httpd as.
# . On SCO (ODT 3) use "User nouser" and "Group nogroup".
# . On HPUX you may not be able to use shared memory as nobody, and the
# suggested workaround is to create a user www and use that user.
# NOTE that some kernels refuse to setgid(Group) or semctl(IPC_SET)
# when the value of (unsigned)Group is above 60000;
# don't use Group #-1 on these systems!
#
User oracle
Group oinstall
</IfModule>
```

10.3.2 Update Oracle HTTP Server Runtime Parameters

By default, the Oracle HTTP Server contains parameter values that are suitable for most applications. These values, however, must be adjusted in IDM Deployments.

Proceed as follows:

Edit the file `httpd.conf`, which is located in:

WEB_ORACLE_INSTANCE`/config/OHS/`*component_name*`/`

Find the entry that looks like this:

```
<IfModule mpm_worker_module>
```

Update the values in this section as follows:

```
<IfModule mpm_worker_module>
  ServerLimit 20
  StartServers 2
  MaxClients 1000
  MinSpareThreads 200
  MaxSpareThreads 800
  ThreadsPerChild 50
  MaxRequestsPerChild 10000
  AcceptMutex fcntl
  LockFile "${ORACLE_INSTANCE}/diagnostics/logs/${COMPONENT_TYPE}/${COMPONENT_
NAME}/http_lock"
</IfModule>
```

Save the file.

10.3.3 Creating Virtual Hosts to Support Identity Management

In order for Oracle HTTP server to service Oracle Identity Management, you must create a number of files to add support for virtual hosts. Each of the files in the following sections creates a virtual host definition and declares a number of URLs which can be accessed from within it. By enclosing the location directives inside the virtual host these locations will only be available when invoked using the virtual host name. For example, you will be able to access the WebLogic console by using the URL `http://ADMIN.mycompany.com/console` but not by using the URL: `https://SSO.mycompany.com/console`

The following sections show sample configuration files for a complete Identity Management deployment. If you are only doing a partial deployment only include those entries applicable to components you are deploying. If you extend your domain at a later date with extra components then you must update the files below with the entries required to support the components you are using.

10.3.3.1 Enable Virtual Host Support

Before creating virtual host directives, you must enable the Oracle HTTP Server to listen for virtual hosts on the default OHS listen port.

To do this, on each web host, edit the file `httpd.conf`, which is located in the directory: *`WEB_ORACLE_INSTANCE`*`/config/OHS/`*`component_name`*

Locate the line that looks like this:

```
#NameVirtualHost *:80
```

Add the following entry to the file, using `7777` or whatever your *OHS_PORT* value is, and save the file.

```
NameVirtualHost *:7777
```

10.3.3.2 Create Virtual Host Definitions

Create the following files on each web host in the directory: *`WEB_ORACLE_INSTANCE`*`/config/OHS/`*`component_name`*`/moduleconf`

10.3.3.2.1 Create Virtual Host for ADMIN.mycompany.com Create a file called `admin_vh.conf`. This will contain a list of locations which are supported by clients accessing the domain using `ADMIN.mycompany.com`.

```
<VirtualHost *:7777>

   ServerName ADMIN.mycompany.com:80
   RewriteEngine On
   RewriteOptions inherit
   ServerAdmin you@your.address

###################################
## General Domain Configuration
###################################

   # Admin Server and EM
   <Location /console>
      SetHandler weblogic-handler
      WebLogicHost ADMINVHN.mycompany.com
      WebLogicPort 7001
   </Location>

   <Location /consolehelp>
      SetHandler weblogic-handler
      WebLogicHost ADMINVHN.mycompany.com
      WebLogicPort 7001
   </Location>

   <Location /em>
      SetHandler weblogic-handler
      WebLogicHost ADMINVHN.mycompany.com
      WebLogicPort 7001
   </Location>

################################################
## Entries Required by Oracle Entitlements Server
################################################

   # APM
   <Location /apm>
      SetHandler weblogic-handler
      WebLogicHost ADMINVHN.mycompany.com
      WebLogicPort 7001
   </Location>

################################################
## Entries Required by Oracle Unified Directory
################################################

   # OUD ODSM
   <Location /odsm>
      SetHandler weblogic-handler
      WebLogicHost ADMINVHN.mycompany.com
      WebLogicPort 7001
   </Location>

############################################
## Entries Required by Oracle Access Manager
############################################
```

```
   # OAM Console
   <Location /oamconsole>
      SetHandler weblogic-handler
      WebLogicHost ADMINVHN.mycompany.com
      WebLogicPort 7001
   </Location>

##############################################
## Entries Required by Oracle Identity Manager
##############################################

   # OIM self and advanced admin webapp consoles(canonic webapp)
   <Location /oim>
     SetHandler weblogic-handler
     WLCookieName    oimjsessionid
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
    </Location>

   # OIM, xlWebApp - Legacy 9.x webapp (struts based)
   <Location /xlWebApp>
     SetHandler weblogic-handler
     WLCookieName    oimjsessionid
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

   # OIM self service console
   <Location /identity>
     SetHandler weblogic-handler
     WLCookieName oimjsessionid
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

   # OIM Nexaweb WebApp - used for workflow designer and DM
   <Location /Nexaweb>
     SetHandler weblogic-handler
     WLCookieName    oimjsessionid
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

   <Location /sysadmin>
     SetHandler weblogic-handler
     WLCookieName oimjsessionid
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

</VirtualHost>
```

10.3.3.2.2 Create Virtual Host for SSO.mycompany.com Create a file called `sso_vh.conf`. This will contain a list of locations which are supported by clients accessing the domain using `SSO.mycompany.com`. These are the main entry points for external users.

```
<VirtualHost *:7777>
   ServerName https://SSO.mycompany.com:443
```

```
   RewriteEngine On
   RewriteOptions inherit
   ServerAdmin you@your.address

##############################################
## Entries Required by Oracle Access Manager
##############################################

   # OAM Configuration
   <Location /oam>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WLCookieName OAM_JSESSIONID
      WebLogicCluster IDMHOST1.mycompany.com:14100,IDMHOST2.mycompany.com:14100
   </Location>

   # Required if using Oracle Identity Federation
   <Location /oamfed>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WLCookieName OAM_JSESSIONID
      WebLogicCluster IDMHOST1.mycompany.com:14100,IDMHOST2.mycompany.com:14100
  </Location>

   # Required if using Oracle Identity Federation
   <Location /sts>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WLCookieName OAM_JSESSIONID
      WebLogicCluster IDMHOST1.mycompany.com:14100,IDMHOST2.mycompany.com:14100
   </Location>
################################################
## Entries Required by Oracle Identity Manager
################################################

   # OIM, xlWebApp - Legacy 9.x webapp (struts based)
   <Location /xlWebApp>
     SetHandler weblogic-handler
     WLProxySSL ON
     WLProxySSLPassThrough ON
     WLCookieName    oimjsessionid
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

   # OIM self service console
   <Location /identity>
     SetHandler weblogic-handler
     WLProxySSL ON
     WLProxySSLPassThrough ON
     WLCookieName oimjsessionid
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

</VirtualHost>
```

10.3.3.2.3 Create Virtual Host for IDMINTERNAL.mycompany.com Create a file called `idminternal_vh.conf`. This will contain a list of locations which are supported by clients accessing the domain using `IDMINTERNAL.mycompany.com`. These entries are used by internal callbacks.

```
<VirtualHost *:7777>
   ServerName http://IDMINTERNAL.mycompany.com:80
   RewriteEngine On
   RewriteOptions inherit
   ServerAdmin you@your.address

################################################
## Entries Required by Oracle Identity Manager
################################################

   # Provide the OIM Managed Server Port
   <Location /workflowservice>
     SetHandler weblogic-handler
     WLCookieName    oimjsessionid
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

   # OIM, SOA Infra
   <Location /soa-infra>
     SetHandler weblogic-handler
     WLCookieName oimjsessionid
     WebLogicCluster SOAHOST1VHN:8001,SOAHOST2VHN:8001
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

   # OIM, Used for provisioning-callback.
   <Location /provisioning-callback>
     SetHandler weblogic-handler
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

   # OIM, SOA Callback webservice for SOD - Provide the SOA Managed Server Ports
   <Location /sodcheck>
     SetHandler weblogic-handler
     WLCookieName oimjsessionid
     WebLogicCluster SOAHOST1VHN:8001,SOAHOST2VHN:8001
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

   # OIM, SOA Callback
   <Location /integration>
     SetHandler weblogic-handler
     WLCookieName oimjsessionid
     WebLogicCluster SOAHOST1VHN:8001,SOAHOST2VHN:8001
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

   # OIM, spml xsd profile
   <Location /spml-xsd>
     SetHandler weblogic-handler
     WLCookieName    oimjsessionid
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
```

```
   </Location>

   # OIM, spml dsml profile
   <Location /spmlws>
     SetHandler weblogic-handler
     PathTrim /weblogic
     WLCookieName oimjsessionid
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

   # OIM, role-sod profile
   <Location /role-sod>
     SetHandler weblogic-handler
     WLCookieName oimjsessionid
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

   # OIM, used for Callback service.
   <Location /callbackResponseService>
     SetHandler weblogic-handler
     WLCookieName    oimjsessionid
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

   # OIM, UMS Email Support
   <Location /ucs>
     SetHandler weblogic-handler
     WLCookieName oimjsessionid
     WebLogicCluster SOAHOST1VHN:8001,SOAHOST2VHN:8001
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

   <Location /reqsvc>
     SetHandler weblogic-handler
     WLCookieName oimjsessionid
     WebLogicCluster
OIMHOSTVHN1.mycompany.com:14000,OIMHOSTVHN2.mycompany.com:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

</VirtualHost>
```

10.4 Restart Oracle HTTP Server

Restart OHS on WEBHOST1 as follows:

```
WEB_ORACLE_INSTANCE/bin/opmnctl restartproc ias-component=ohs1
```

Restart OHS on WEBHOST2:

```
WEB_ORACLE_INSTANCE/bin/opmnctl restartproc ias-component=ohs2
```

10.5 Setting the Front End URL for the Administration Console

Oracle WebLogic Server Administration Console tracks changes that are made to ports, channels and security using the console. When changes made through the console are activated, the console validates its current listen address, port and protocol. If the listen address, port and protocol are still valid, the console redirects the HTTP request, replacing the host and port information with the Administration Server's listen address and port. When the Administration Console is accessed using a load balancer, you must change the Administration Server's front end URL so that the user's browser is redirected to the appropriate load balancer address.

Setting the front end host and port as described in this section forces all access to the applications deployed in the WebLogic administration server to go through the Oracle HTTP server. This means that, once single sign-on is enabled in the domain, access to administration consoles such as Weblogic Console, and OAM Administration console is under the control of Access Manager.

Note: Once Single Sign-On is enabled, you can only access applications deployed in the administration server through the Oracle HTTP server. At least one Access Manager managed server must be running.

You cannot start managed servers by using the WebLogic Console until at least one Access Manager managed server has been started. See Section 17.1.3.1, "Starting an Access Manager Managed Server When None is Running" for instructions on starting an Access Manager managed server without using the console

If you do not want to protect applications deployed in the Administration Server with Oracle Single Sign-on, that is, if you want to allow direct access to those applications, bypassing corporate security, you can do so by not setting the front end host and port as described in this section. Oracle does not recommend this.

To make this change, perform the following steps:

1. Log in to Oracle WebLogic Server Administration Console at the URL:

   `http://ADMINVHN.mycompany.com:7001/console`, where `7001` is *`WLS_ADMIN_PORT`*, as described in Section B.3.

2. Click **Lock and Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers** to open the Summary of Servers page.
5. Select **AdminServer(admin)** in the Names column of the table. The Settings page for AdminServer(admin) appears.
6. Click the **Protocols** tab.
7. Click the **HTTP** tab.
8. Set the **Front End Host** and **Front End HTTP PORT** fields to your load balancer address, as follows.
   - **Front End Host**: `ADMIN.mycompany.com`
   - **Front End HTTP PORT**: 80 (*HTTP_PORT*)
9. Save and activate the changes.
10. Restart Administration server.

To eliminate redirections, best practice is to disable the Administration console's `Follow changes` feature. To do this, log in to the administration console and click **Preferences**->**Shared Preferences**. Deselect **Follow Configuration Changes** and click **Save**.

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. See Section 17.10, "Troubleshooting" for possible causes.

Note: After restarting the domain and the Oracle HTTP Server, the Oracle HTTP Server should appear as a manageable target in Oracle Enterprise Manager Fusion Middleware Control. To verify this, log in to Fusion Middleware Control. The WebTier item in the navigation tree should show that Oracle HTTP Server has been registered.

10.6 Validating the Configuration

After the installation is completed, perform the following validations.

- Check that you can access the Oracle HTTP Server by using following URLs:

  `http://WEBHOST1.mycompany.com:7777/`

  (where `7777` is the `OHS_PORT`, as described in Section B.3)

  `http://WEBHOST2.mycompany.com:7777/`

  `https://SSO.mycompany.com/`

  `http://IDMINTERNAL.mycompany.com`

- Validate Access to Oracle Directory Services Manager for Oracle Unified Directory using the URL:

  `http://ADMIN.mycompany.com/odsm`

  and create a connection to one of the local Oracle Unified Directory servers.

- Validate Access to Oracle Entitlements Server Policy Manager using the URL:

  `http://ADMIN.mycompany.com/apm`

  Log in using the WebLogic Administrator account for APM, for example: `weblogic`

- Validate Access to WebLogic Console using the URL

  `http://ADMIN.mycompany.com/console`

  Log in using the WebLogic Administrator account, for example: `weblogic`.

- Validate Access to Oracle Enterprise Manager Fusion Middleware Control using the URL

  `http://ADMIN.mycompany.com/em`

  Log in using the WebLogic Administrator account, for example: `weblogic`.

10.7 Summary of Web Tier URLs

Table 10–1 Web Tier URLs

Server or Console	URL
Oracle HTTP Server SSO	`https://SSO.mycompany.com/`
Oracle HTTP Server Internal	`http://IDMINTERNAL.mycompany.com/`
Oracle Directory Services Manager for Oracle Unified Directory	`http://ADMIN.mycompany.com/odsm`
Oracle Entitlements Server Policy Manager	`http://ADMIN.mycompany.com/apm`
WebLogic Console	`http://ADMIN.mycompany.com/console`
Oracle Enterprise Manager Fusion Middleware Control	`http://ADMIN.mycompany.com/em`

10.8 Backing up the Web Tier Configuration

Back up the Web Tier binaries and Domain Home, as described in Section 17.6.3, "Performing Backups During Installation and Configuration."

11

Extending the Domain to Include Oracle Access Management

This chapter describes how to extend the domain to include Oracle Access Management Access Manager in the Oracle Identity Management enterprise deployment.

This chapter includes the following topics:

11.1 Overview of Extending the Domain to Include Oracle Access Management Access Manager

Access Manager enables your users to seamlessly gain access to web applications and other IT resources across your enterprise. It provides a centralized and automated single sign-on (SSO) solution, which includes an extensible set of authentication methods and the ability to define workflows around them. It also contains an authorization engine, which grants or denies access to particular resources based on properties of the user requesting access as well as based on the environment from which the request is made. Comprehensive policy management, auditing, and integration with other components of your IT infrastructure enrich this core functionality.

Access Manager consists of several components, including OAM Server, Oracle Access Management Console, and WebGates. The OAM Server includes all the components necessary to restrict access to enterprise resources. The Oracle Access Management Console is the administrative console to Access Manager. WebGates are web server agents that act as the actual enforcement points for Access Manager. Follow the instructions in this chapter and Chapter 15, "Configuring Single Sign-on for Administration Consoles in an Enterprise Deployment" to install and configure the Access Manager components necessary for your enterprise deployment.

11.2 About Domain URLs

After you complete this chapter, the following URL will be available:

Table 11–1 OAM URLs After Web Tier Configuration

Component	URLs	User	SSO User
OAM Console	`http://ADMIN.mycompany.com/oamconsole`	`weblogic`	`oamadmin`
Oracle Enterprise Manager Fusion Middleware Control	`http://ADMIN.mycompany.com/em`	`weblogic`	`weblogic_idm`
Oracle Directory Services Manager	`http://ADMIN.mycompany.com/odsm`	`weblogic`	`weblogic_idm`
Oracle Entitlements Server Policy Manager	`http://ADMIN.mycompany.com/apm`	`weblogic`	`oamadmin`

11.3 Using Different Directory Configurations

Access Manager normally uses a single LDAP store to hold Identity Information. You can, however, configure Oracle Access Management Access Manager to use multiple directory stores of different types, such as Oracle Virtual Directory and a third party directory.

When you use multiple directories, you can present the directories to Access Manager as a single consolidated directory, using Oracle Virtual Directory. Alternatively, you can configure Access Manager to access each directory individually.

For more information, see the following chapters in *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*:

- "Configuring an Identity Store with Multiple Directories"
- "Integrating Oracle Internet Directory with Oracle Access Manager"

To learn more about the different types of directory configuration for Access Manager, consult the Access Manager documentation listed under "Related Documents" in the Preface. Customers considering these variations should adjust their directory and Access Manager deployment accordingly.

11.4 Prerequisites

Before you configure Access Manager, ensure that the following tasks have been performed on IDMHOST1 and IDMHOST2:

1. Prepare the Identity Store as described in Chapter 9, "Preparing Identity Stores."
2. Configure Oracle Web Tier on WEBHOST1 and WEBHOST2 as described in Chapter 10, "Installing and Configuring Oracle Web Tier for an Enterprise Deployment."
3. Configure the load balancer as described in Section 3.3, "About Virtual Server Names Used by the Topologies."

11.5 Extending Domain with Access Manager

Start the configuration wizard on IDMHOST1 by executing the command:

```
IAM_MW_HOME/oracle_common/common/bin/config.sh
```

Then proceed as follows:

1. On the Welcome screen, select **Extend an Existing WebLogic Domain**. Click **Next**.
2. On the Select a WebLogic Domain screen, using the navigator, select the domain home of the WebLogic Administration Server, for example: *ASERVER_HOME*

   Click **Next**
3. On the Select Extension Source screen, select **Oracle Access Management [iam]**.

   Click **Next**
4. On the Configure JDBC Component Schema screen, do the following:

   Select **OAM Infrastructure**.

   For the Oracle RAC configuration for component schemas, select **Convert to GridLink**.

   Click **Next**.
5. The Gridlink RAC Component Schema screen appears. In this screen, enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU.
   - **Driver**: Select Oracle's driver (Thin) for GridLink Connections,Versions:10 and later.
   - Select **Enable FAN**.
   - Do one of the following:
     - If SSL is not configured for ONS notifications to be encrypted, deselect **SSL**.
     - Select **SSL** and provide the appropriate wallet and wallet password.
   - **Service Listener**: Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the parameter `remote_listener` in the database:

     ```
     SQL>show parameter remote_listener;
     NAME            TYPE   VALUE
     ------------------------------------------------------------
     remote_listener string DB-SCAN.MYCOMPANY.COM:1521
     ```

Notes:

- For Oracle Database 11*g* Release 1 (11.1), use the virtual IP and port of each database instance listener, for example: `DBHOST1-VIP.mycompany.com` (port `1521`) and `DBHOST2-VIP.mycompany.com` (port `1521`), where `1521` is *`DB_LSNR_PORT`*
- For Oracle Database 10*g*, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources see Appendix A, "Using Multi Data Sources with Oracle RAC."

- **ONS Host**: Enter the SCAN address for the Oracle RAC database and the ONS remote port as reported by the database:

```
srvctl config nodeapps -s
ONS exists: Local port 6100, remote port 6200, EM port 2016
```

Note: For Oracle Database 11g Release 1 (11.1), use the hostname and port of each database's ONS service, for example:

`DBHOST1.mycompany.com (port 6200)`

and

`DBHOST2.mycompany.com (port 6200)`

Enter the following RAC component schema information:

Schema Name	Service Name	Schema Owner	Password
Access Management	`OAMEDG.mycompany.com`	`EDG_OAM`	*password*

6. In the Test JDBC Data Sources screen, confirm that all connections were successful.

   The connections are tested automatically. The **Status** column displays the results. If all connections are not successful, click **Previous** to return to the previous screen and correct your entries.

   Click **Next** when all the connections are successful.

7. On the Test Component Schema screen, the Wizard attempts to validate the data sources. If the data source validation succeeds, click **Next**. If it fails, click **Previous**, correct the problem, and try again.

8. On the Select Optional Configuration screen, select **Managed Servers, Clusters and Machines**.

   Click **Next**

9. When you first enter the Configure Managed Servers screen, a managed server called oam_server1 is created automatically. Rename oam_server1 to WLS_OAM1 and update its attributes as shown in the following table. Then, add a new managed server called WLS_OAM2 with the following attributes.

Name	Listen Address	Listen Port	Port Variable	SSL Listen Port	SSL Enabled
WLS_OAM1	IDMHOST1.mycompany.com	14100	*OAM_PORT*	N/A	No
WLS_OAM2	IDMHOST2.mycompany.com	14100	*OAM_PORT*	N/A	No

Notes:

- Do not change the configuration of the managed servers that were configured as a part of previous deployments.
- Do not delete the default managed servers that are created. Rename them as described.

Click **Next**.

10. On the Configure Clusters screen, create a cluster by clicking **Add**. Supply the following information:

Name	Cluster Messaging Mode
oam_cluster	Unicast

Leave all other fields at the default settings and click **Next**.

11. On the Assign Servers to Clusters screen, associate the Managed Servers with the cluster. Click the cluster name in the right pane. Click the Managed Server under Servers, then click the arrow to assign it to the cluster.

Assign servers to the cluster as follows:

Cluster	Server
oam_cluster	WLS_OAM1
	WLS_OAM2

Note: Do not change the configuration of any clusters which have already been configured as part of previous application deployments.

Click **Next**.

12. On the Configure Machines screen, create a machine for each host in the topology. Click the **Unix Machine** tab and then click **Add** to add the following machines:

Note: "Name" can be any unique string. "Node Manager Listen Address" must be a resolvable host name.

Name	Node Manager Listen Address	Node Manager Listen Port	Port Variable
IDMHOST1.mycompany.com	IDMHOST1.mycompany.com	5556	*NMGR_PORT*
IDMHOST2.mycompany.com	IDMHOST2.mycompany.com	5556	*NMGR_PORT*

Leave all other fields to their default values.

Note: The machine name does not need to be a valid host name or listen address; it is just a unique identifier of a Node Manager location

Click **Next**.

13. On the Assign Servers to Machines screen, assign servers to machines as follows:

    IDMHOST1: **WLS_OAM1**

    IDMHOST2: **WLS_OAM2**

    Click **Next** to continue.

14. On the Configuration Summary screen, click **Extend** to extend the domain.

    Note: If you receive a warning that says:

    ```
    CFGFWK: Server listen ports in your domain configuration conflict
    with ports in use by active processes on this host
    ```

    Click **OK**.

    This warning appears if Managed Servers have been defined as part of previous installs and can safely be ignored.

15. On the Installation Complete screen, click **Done**.

16. Restart WebLogic Administration Server as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

11.6 Configuring Access Manager

This section contains the following topics:

- Section 11.6.1, "Removing IDM Domain Agent"
- Section 11.6.2, "Setting a Global Passphrase"
- Section 11.6.3, "Configuring Access Manager by Using the IDM Configuration Tool"
- Section 11.6.4, "Validating the Configuration"
- Section 11.6.5, "Updating Newly-Created Agent"
- Section 11.6.6, "Modifying Access Manager Resources"
- Section 11.6.7, "Updating Existing WebGate Agents"

- Section 11.6.8, "Perform Bug 13824816 Workaround"

11.6.1 Removing IDM Domain Agent

By default, the IDMDomainAgent provides single sign-on capability for administration consoles. In enterprise deployments, WebGate handles single sign-on, so you must remove the IDMDomainAgent. Remove the IDMDomainAgent as follows:

Log in to the WebLogic console at the URL listed in Section 17.2, "About Identity Management Console URLs."

Then:

1. Select **Security Realms** from the **Domain Structure** Menu
2. Click **myrealm**.
3. Click the **Providers** tab.
4. Click **Lock and Edit** from the Change Center.
5. In the list of authentication providers, select **IAMSuiteAgent**.
6. Click **Delete**.
7. Click **Yes** to confirm the deletion.
8. Click **Activate Changes** from the Change Center.
9. Restart WebLogic Adminisration Server and ALL running Managed Servers, as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

11.6.2 Setting a Global Passphrase

By default, Access Manager is configured to use the Open security model. If you plan to change this mode using `idmConfigTool`, you must set a global passphrase. Although you need not set the global passphrase and the web gate access password to be the same, it is recommended that you do.You do this by performing the following steps.

1. Log in to the OAM console at the URL listed in Section 17.2, "About Identity Management Console URLs."

   as the WebLogic administration user.
2. Click the **System Configuration** tab.
3. Click **Access Manager Settings** located in the Access Manager section.
4. Select **Open** from the **Actions** menu. The access manager settings are displayed.
5. If you plan to use Simple security mode for OAM servers, supply a global passphrase.
6. Click **Apply**.

11.6.3 Configuring Access Manager by Using the IDM Configuration Tool

Now that the initial installation is done, you must perform the following tasks:

- Configure Access Manager to use an external LDAP Directory, (`IDSTORE.mycompany.com`).

- Create Access Manager WebGate Agent.

You perform these tasks by using `idmConfigTool`.

> **Note:** Two parameter settings determine whether you are configuring Access Manager with Oracle Identity Manager integration or Access Manager alone.
>
> - To configure Access Manager with Oracle Identity Manager integration, set `OAM11G_OIM_INTEGRATION_REQ` to `true` and specify a value for `OAM11G_OIM_OHS_URL`.
> - To configure Access Manager without Oracle Identity Manager, set `OAM11G_OIM_INTEGRATION_REQ` to `false`.
>
> These parameters are used to add extra links, such as Forgotten Password, to the Access Manager credential collection page
>
> If you configure Access Manager without Oracle Identity Manager, then decide to add Oracle Identity Manager at a later date, you must run this command again to configure Access Manager with Oracle Identity Manager integration.

Perform the following tasks on IDMHOST1:

1. Set `MW_HOME` to *`IAM_MW_HOME`*.

   Set `ORACLE_HOME` to *`IAM_ORACLE_HOME`*.

   Set `JAVA_HOME` to *`JAVA_HOME`*.

2. Create a properties file called `config_oam1.props` with the following contents:

```
WLSHOST: ADMINVHN.mycompany.com
WLSPORT: 7001
WLSADMIN: weblogic
WLSPASSWD: Admin Password
IDSTORE_DIRECTORYTYPE: OUD
IDSTORE_HOST: IDSTORE.mycompany.com
IDSTORE_PORT: 389
IDSTORE_BINDDN: cn=oudadmin
IDSTORE_USERNAMEATTRIBUTE: cn
IDSTORE_LOGINATTRIBUTE: uid
OAM11G_SERVER_LOGIN_ATTRIBUTE: uid
IDSTORE_USERSEARCHBASE: cn=Users,dc=mycompany,dc=com
IDSTORE_SEARCHBASE: dc=mycompany,dc=com
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=mycompany,dc=com
IDSTORE_SYSTEMIDBASE: cn=systemids,dc=mycompany,dc=com
IDSTORE_OAMSOFTWAREUSER: oamLDAP
IDSTORE_OAMADMINUSER: oamadmin
PRIMARY_OAM_SERVERS: IDMHOST1.mycompany.com:5575,IDMHOST2.mycompany.com:5575
WEBGATE_TYPE: ohsWebgate11g
ACCESS_GATE_ID: Webgate_IDM
OAM11G_OIM_WEBGATE_PASSWD: password to be assigned to WebGate
COOKIE_DOMAIN: .mycompany.com
OAM11G_WG_DENY_ON_NOT_PROTECTED: true
OAM11G_IDM_DOMAIN_OHS_HOST: SSO.mycompany.com
OAM11G_IDM_DOMAIN_OHS_PORT: 443
OAM11G_IDM_DOMAIN_OHS_PROTOCOL: https
OAM11G_SERVER_LBR_HOST: SSO.mycompany.com
OAM11G_SERVER_LBR_PORT: 443
```

```
OAM11G_SERVER_LBR_PROTOCOL: https
OAM11G_OAM_SERVER_TRANSFER_MODE: simple
OAM_TRANSFER_MODE: simple
OAM11G_IDM_DOMAIN_LOGOUT_URLS:
/console/jsp/common/logout.jsp,/em/targetauth/emaslogout.jsp
OAM11G_IDSTORE_ROLE_SECURITY_ADMIN: OAMAdministrators
OAM11G_SSO_ONLY_FLAG: false
COOKIE_EXPIRY_INTERVAL: 120
OAM11G_IMPERSONATION_FLAG: false
OAM11G_OIM_INTEGRATION_REQ: false
OAM11G_OIM_OHS_URL:https://SSO.mycompany.com:443
SPLIT_DOMAIN:true
```

Where:

- `WLSHOST` (*`ADMINVHN`*) is the host of your administration server. This is the virtual name.
- `WLSPORT` is the port of your administration server, *`WLS_ADMIN_PORT`* in Section B.3.
- `WLSADMIN` is the WebLogic administrative user you use to log in to the WebLogic console.
- `WLSPASSWD` is the WebLogic administrator password.
- `IDSTORE_DIRECTORYTYPE` is `OUD`, `OID` or `OVD`.
- `IDSTORE_HOST` and `IDSTORE_PORT` are the host and port of the Identity Store directory when accessed through the load balancer. These are *`LDAP_LBR_HOST`* and *`LDAP_LBR_PORT`* in the Section B.3 worksheet.
- `IDSTORE_BINDDN` is an administrative user in the Identity Store directory.
- `IDSTORE_USERSEARCHBASE` is the location in the directory where Users are stored.
- `IDSTORE_GROUPSEARCHBASE` is the location in the directory where Groups are stored.
- `IDSTORE_SEARCHBASE` is the location in the directory where Users and Groups are stored.
- `IDSTORE_SYSTEMIDBASE` is the location of a container in the directory where the user oamLDAP is stored.
- `IDSTORE_OAMSOFTWAREUSER` is the name of the user you created in Section 9.4, "Preparing the Identity Store" to be used to interact with LDAP.
- `IDSTORE_OAMADMINUSER` is the name of the user you created in Section 9.4, "Preparing the Identity Store" to access your OAM Console.
- `PRIMARY_OAM_SERVERS` is a comma separated list of your OAM Servers and the proxy ports they use, for example: IDMHOST1:*OAM_PROXY_PORT*

Note: To determine the proxy ports your OAM Servers use:

1. Log in to the OAM console at the URL listed in Section 17.2, "About Identity Management Console URLs."
2. Click the **System Configuration** tab.
3. Expand **Server Instances** under the Common Configuration section
4. Click an OAM Server, such as **WLS_OAM1**, and select **Open** from the **Actions** menu.
5. Proxy port is the one shown as **Port**.

- `ACCESS_GATE_ID` is the name you want to assign to the WebGate.
- `OAM11G_OIM_WEBGATE_PASSWD` is the password to be assign to the WebGate.
- `OAM11G_IDM_DOMAIN_OHS_HOST` is the name of the load balancer which is in front of the OHS's.
- `OAM11G_IDM_DOMAIN_OHS_PORT` is the port that the load balancer listens on (*HTTP_SSL_PORT*).
- `OAM11G_IDM_DOMAIN_OHS_PROTOCOL` is the protocol to use when directing requests at the load balancer.
- `OAM11G_WG_DENY_ON_NOT_PROTECTED`, when set to `false`, allows login pages to be displayed. It should be set to `true` when using webgate11g.
- `OAM_TRANSFER_MODE` is the security model that the Oracle Access Manager Servers function in. Valid values are `simple` and `open`. If you use the `simple` mode, you must define a global passphrase, as defined in Section 11.6.2, "Setting a Global Passphrase."
- `OAM11G_OAM_SERVER_TRANSFER_MODE` is the security model that the OAM Servers function in, as defined in Section 11.6.2, "Setting a Global Passphrase."
- `OAM11G_IDM_DOMAIN_LOGOUT_URLS` is set to the various logout URLs.
- `OAM11G_SSO_ONLY_FLAG` confgures Access Manager as authentication only mode or normal mode, which supports authentication and authorization.

  If `OAM11G_SSO_ONLY_FLAG` is `true`, the OAM Server operates in authentication only mode, where all authorizations return true by default without any policy validations. In this mode, the server does not have the overhead of authorization handling. This is recommended for applications which do not depend on authorization policies and need only the authentication feature of the OAM Server.

  If the value is `false`, the server runs in default mode, where each authentication is followed by one or more authorization requests to the OAM Server. WebGate allows the access to the requested resources or not, based on the responses from the OAM Server.
- `OAM11G_IMPERSONATION_FLAG` is set to `true` if you are configuring OAM Impersonation.
- `OAM11G_SERVER_LBR_HOST` is the name of the load balancer fronting your site. This and the following two parameters are used to construct your login URL.
- `OAM11G_SERVER_LBR_PORT` is the port that the load balancer is listening on (*HTTP_SSL_PORT*).

- OAM11G_SERVER_LBR_PROTOCOL is the URL prefix to use.
- OAM11G_OIM_INTEGRATION_REQ should be set to true if you are building a topology which contains both OAM and OIM. Otherwise set to false at this point. This value is only set to true when performing Access Manager/Oracle Identity Manager integration and is set during the integration phase.
- OAM11G_OIM_OHS_URL should be set to the URL of your load balancer. This parameter is only required if your topology contains OAM and OIM.
- COOKIE_DOMAIN is the domain in which the WebGate functions.
- WEBGATE_TYPE is the type of WebGate agent you want to create.
- OAM11G_IDSTORE_NAME is the Identity Store name. If you already have an Identity Store in place which you wish to reuse (rather than allowing the tool to create a new one for you), then set the value of this parameter to the name of the Identity Store you wish to reuse.
- OAM11G_SERVER_LOGIN_ATTRIBUTE when set to uid, ensures that when users log in, their username is validated against the uid attribute in LDAP.
- SPLIT_DOMAIN should be set to true If you are creating a domain with just OAM or OAM located in a different domain from OIM (Split Domain). Otherwise, it is not necessary to specify this parameter.

3. Configure Access Manager using the command idmConfigTool which is located at:

   IAM_ORACLE_HOME/idmtools/bin

   Note: When you run the idmConfigTool, it creates or appends to the file idmDomainConfig.param. This file is generated in the same directory that the idmConfigTool is run from. To ensure that each time the tool is run, the same file is appended to, always run the idmConfigTool from the directory:

   IAM_ORACLE_HOME/idmtools/bin

   The syntax of the command on Linux is:

   ```
   idmConfigTool.sh -configOAM input_file=configfile
   ```

   For example:

   ```
   idmConfigTool.sh -configOAM input_file=config_oam1.props
   ```

   When the command runs you are prompted to enter the password of the account you are connecting to the Identity Store with. You are also asked to specify the passwords you want to assign to these accounts:

   - IDSTORE_PWD_OAMSOFTWAREUSER
   - IDSTORE_PWD_OAMADMINUSER

4. 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.
5. Restart WebLogic Administration Server as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

Note: After you run `idmConfigTool`, several files are created that you need for subsequent tasks. Keep these in a safe location.

Two 11*g* WebGate profiles are created: `Webgate_IDM`, which is used for intercomponent communication and `Webgate_IDM_11g`, which is used by 11*g* Webgates.

The following files exist in the directory *`ASERVER_HOME`*`/output/Webgate_IDM_11g`. You need these when you install the WebGate software.

- `cwallet.sso`
- `ObAccessClient.xml`
- `password.xml`

Additionally, you need the files `aaa_cert.pem` and `aaa_key.pem`, which are located in the directory *`ASERVER_HOME`*`/output/Webgate_IDM`.

11.6.4 Validating the Configuration

To Validate that this has completed correctly.

1. Access the OAM console at: `http://ADMIN.mycompany.com/oamconsole`
2. Log in as the Access Manager administration user you created in Section 9.4, "Preparing the Identity Store," for example, `oamadmin`.
3. Click the **System Configuration** tab
4. Expand **Access Manager** - **SSO Agents** - **OAM Agents**.
5. Click the open folder icon, then click **Search**.
6. You should see the WebGate agents `Webgate_IDM` and `Webgate_IDM_11g`, which you created in Section 11.6.3, "Configuring Access Manager by Using the IDM Configuration Tool."

11.6.5 Updating Newly-Created Agent

After generating the initial configuration, you must edit the configuration and add advanced configuration entries.

1. Select **System Configuration** Tab
2. Select **Access Manager - SSO Agents - OAM Agent** from the directory tree. Double-click or select the open folder icon.
3. On the displayed search page click **Search** to perform an empty search.
4. Click the Agent `Webgate_IDM`.
5. Select **Open** from the Actions menu.
6. Set **Maximum Number of Connections** to `10` for all of the OAM Servers listed in the primary servers list.
7. If the following **Logout URLs** are not listed, add them:
   - `/oamsso/logout.html`
   - `/console/jsp/common/logout.jsp`

- `/em/targetauth/emaslogout.jsp`

8. Click **Apply**.
9. Repeat Steps 4 through 7 for the WebGate agent Webgate_IDM_11g.
10. Click **Policy Configuration** tab.
11. Click **Host Identifiers.**
12. Click **Open**.
13. Click **Search**.
14. Click **IAMSuiteAgent**.
15. Click **+** in the **Host Name Variations** box.
16. Enter the following information:
    - **Host Name**: `ADMIN.mycompany.com`
    - **Port**: `80` (*HTTP_PORT*)
17. Click **Apply**.

11.6.6 Modifying Access Manager Resources

During deployment, a number of resources are created in Access Manager with protection levels set. In order for Oracle Identity Manager to function correctly, one of these resources needs to be modified and one created.

To do this perform the following steps:

1. Log in to the OAM console at the URL listed in (About Identity and Access Management Console URLs.")
2. Navigate to **IAM Suite** application domain.
3. Navigate to **Resources** tab
4. Click New Resource and enter the following information:
    - **Type**: http
    - **Description**: `provisioning-callback`
    - **Host Identifier**: `IAMSuiteAgent`
    - **Resource URL**: `/provisioning-callback/**`
    - **Protection Level**: `Excluded`
    - **Authentication Policy**: `n/a`
    - **Authorization Policy**: `n/a`
5. Click **Apply**.
6. Locate the resource `/identity/**` and click on it in the **Search** results window.
7. Click **Edit**.
8. Change the **Protection Level** to **Excluded**.
9. Click **Apply**.

11.6.7 Updating Existing WebGate Agents

If you have changed the OAM security model using the idmConfigTool you must change the security model used by any existing Webgates to reflect this change.

To do this, perform the following steps:

1. Log in to the Oracle Access Management Console as the Access Manager administration user you created in Section 9.4, "Preparing the Identity Store," at the URL listed in Section 17.2, "About Identity Management Console URLs."
2. Click the **System Configuration** tab.
3. Expand **Access Manager - SSO Agents**.
4. Click **OAM Agents** and select **Open** from the **Actions** menu.
5. In the Search window, click **Search**.
6. Click each Agent that was not created by `idmconfigTool` in Section 11.6.3, "Configuring Access Manager by Using the IDM Configuration Tool", for example: **IAMSuiteAgent**.
7. Set the Security value to the new security model. Add any missing Access Manager servers to the displayed list.

   Click **Apply**.

11.6.8 Perform Bug 13824816 Workaround

To work around Bug 13824816, add a condition to the Admin role using the WebLogic Administration Server Console.

Note: Perform this step now only if you specified the parameter `SPLIT_DOMAIN` as `true` when you performed the steps in Section 11.6.3, "Configuring Access Manager by Using the IDM Configuration Tool.".

If you perfomed those steps with the parameter `SPLIT_DOMAIN` set to `false`, perform the steps in this section AFTER you have integrated Oracle Identity Management with Oracle Access Manager. Section 12.23.5, "Perform Bug 13824816 Workaround, if Necessary" will remind you when you reach that point.

To add conditions to the Admin role in the Security Realm:

1. Log in to the WebLogic Administration Server Console at the URL listed in Section 17.2, "About Identity Management Console URLs."
2. In the left pane of the console, click **Security Realms**.
3. On the Summary of Security Realms page, click **myrealm** under the Realms table.
4. On the Settings page for myrealm, click the **Roles & Policies** tab.
5. On the Realm Roles page, expand the **Global Roles** entry under the Roles table. This brings up the entry for Roles.
6. Click the **Roles** link to go to the Global Roles page.
7. On the Global Roles page, click the **Admin** role to go to the Edit Global Role page:
8. On the Edit Global Roles page, under Role Conditions, click **Add Conditions**.

9. On the Choose a Predicate page, select **Group** from the predicates list and click **Next**.
10. On the Edit Arguments Page, specify `OAMAdministrators` in the **Group Argument** field and click **Add**.
11. Click **Finish** to return to the Edit Global Rule page.

    The Role Conditions now show the `OAMAdministrators` Group as an entry.
12. Click **Save** to finish adding the Admin role to the `OAMAdministrators` Group.

11.7 Configuring Access from Web Tier

If you are adding Access Manager to an existing domain, don't forget to include OAM in the Web Tier configuration as described in Section 10.3.3, "Creating Virtual Hosts to Support Identity Management."

11.8 Deploying Managed Server Configuration to Local Storage

Once the configuration is complete, you must propagate the Oracle Identity Manager configuration to the managed server directory on IDMHOST1 and IDMHOST2.

You do this by packing and unpacking the domain, you pack the domain first on IDMDomain on IDMHOST1 then unpack it on IDMHOST1 and IDMHOST2.

Follow these steps to propagate the domain to the managed server domain directory.

1. Invoke the `pack` utility from *`ORACLE_COMMON_HOME`*`/common/bin/` on IDMHOST1.

   ```
   ./pack.sh -domain=ASERVER_HOME -template=iam_domain.jar  -template_name="IAM
   Domain" -managed=true
   ```

   This creates a file called `iam_domain.jar`. Copy this file to IDMHOST2.
2. On IDMHOST1 and IDMHOST2, invoke the utility `unpack`, which is also located in the directory: ORACLE_COMMON_HOME/common/bin/

   ```
   ./unpack.sh -domain=MSERVER_HOME -template=iam_domain.jar -overwrite_
   domain=true -app_dir=MSERVER_HOME/applications
   ```

11.9 Starting Managed Servers WLS_OAM1 and WLS_OAM2

Start the managed servers WLS_OAM1 and WLS_OAM2 as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

11.10 Validating Access Manager

You can validate Access Manager by using the `oamtest` tool. To do this, perform the following steps:

1. Ensure that *`JAVA_HOME`* is set in your environment.
2. Add *`JAVA_HOME`*`/bin` to your *`PATH`*, for example:

   ```
   export PATH=$JAVA_HOME/bin:$PATH
   ```
3. Change directory to:

   `IAM_ORACLE_HOME``/oam/server/tester`

4. Start the test tool in a terminal window using the command:

   ```
   java -jar oamtest.jar
   ```

5. When the OAM test tool starts, enter the following information in the **Server Connection** section of the page:

   - **Primary IP Address**: `IDMHOST1.mycompany.com`
   - **Port**: `5575` (*OAM_PROXY_PORT*)
   - **Agent ID**: `Webgate_IDM_11g`
   - **Agent Password**: *`webgate password`*

   Note: if you configured simple mode, you must select **Simple** and provide the global passphrase.

   Click **Connect**.

   In the status window you see:

   ```
   [reponse] Connected to primary access server
   ```

6. In the **Protected Resource URI** section enter:

   - **Scheme**: `http`
   - **Host**: `ADMIN.mycompany.com`
   - **Port**: `80` (*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**: *`oamadmin password`*

   Click **Authenticate**.

   In the status window, you see:

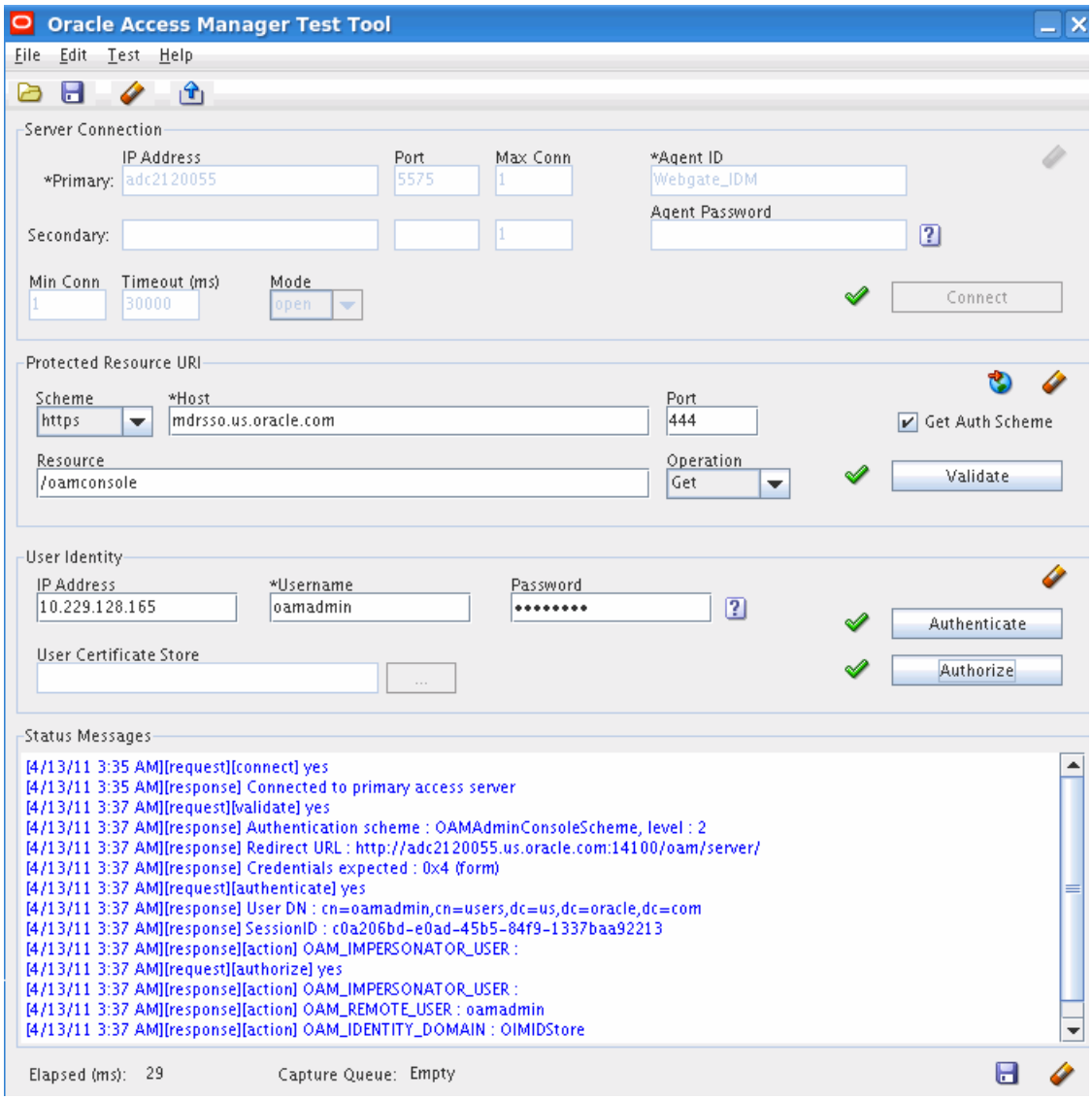
   ```
   [request] [authenticate] yes
   ```

   Click **Authorize**.

   In the status window you see.

   ```
   [request] [authorize] yes
   ```

The following is an example of a test:



Repeat this test for each access server in the topology, remembering to change the connection details for each server.

11.11 Creating a Single Keystore for Integrating Access Manager with Other Components

When you configure Access Manager to work using the simple transport protocol, all traffic to Access Manager is encrypted. When you integrate Access Manager with other components, such as Oracle Identity Manager, you must enable the product being integrated to understand this encryption. (This is not necessary when the transport model is open.) You do this by using a keystore.

When you change Access Manager to use the simple protocol, keystores are created automatically in the directory *ASERVER_HOME*`/output/webgate-ssl`. This directory contains the following files:

- `oamclient-keystore.jks`–contains the private key.
- `oamclient-truststore.jks`–contains the Access Manager simple mode CA certificate

These files are accessed using the Global Passphrase defined at the time of enabling Access Manager in simple mode.

Some products require configuring with both of the files above and some products, such as Oracle Identity Manager require a single consolidated keystore.

To create a keystore suitable for use by Oracle Identity Manager, perform the following steps.

1. Change directory to *ASERVER_HOME*/output/webgate-ssl, for example:

   ```
   cd ASERVER_HOME/output/webgate-ssl
   ```

2. Copy the file oamclient-keystore.jks to ssoKeystore.jks, for example

   ```
   cp oamclient-keystore.jks ssoKeystore.jks
   ```

3. Import the trust store into the new keystore ssoKeystore.jks using the command:

   ```
   keytool -importcert -file IAM_ORACLE_HOME/oam/server/config/cacert.der
   -trustcacerts -keystore PathName_to_keystore -storetype JKS
   ```

   Enter the keystore password when prompted.

   For example:

   ```
   keytool -importcert -file IAM_ORACLE_HOME/oam/server/config/cacert.der
   -trustcacerts -keystore ssoKeystore.jks -storetype JKS
   ```

   Note: The files ssoKeystore.jks and oamclient-truststore.jks are required when you integrate Access Manager running in Simple mode with Oracle Identity Manager. When you integrate these components, you are asked to copy these files to the *ASERVER_HOME*/config/fmwconfig directory. If you subsequently extend the domain on machines where these files have been placed using pack/unpack, you must recopy ssoKeystore.jks and oamclient-truststore.jks after unpacking.

11.12 Backing Up the Access Manager Configuration

Back up the database, the WebLogic domain, and the LDAP directories, as described in Section 17.6.3, "Performing Backups During Installation and Configuration."

12

Extending the Domain to Include Oracle Identity Manager

This chapter describes how to install and configure Oracle Identity Manager for use in the Oracle Identity Management Enterprise Deployment Topology.

This chapter contains the following topics:

12.1 Overview of Extending the Domain to Include Oracle Identity Manager

Oracle Identity Manager is a user provisioning and administration solution that automates the process of adding, updating, and deleting user accounts from applications and directories. It also improves regulatory compliance by providing granular reports that attest to who has access to what. Oracle Identity Manager is available as a standalone product or as part of Oracle Identity Management.

Automating user identity provisioning can reduce Information Technology (IT) administration costs and improve security. Provisioning also plays an important role in regulatory compliance. Key features of Oracle Identity Manager include password management, workflow and policy management, identity reconciliation, reporting and auditing, and extensibility through adapters.

Oracle Identity Manager provides the following key functionalities:

- User Administration
- Workflow and Policy
- Password Management
- Audit and Compliance Management
- Integration Solutions
- User Provisioning
- Organization and Role Management

For details about Oracle Identity Manager, see the *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Manager*.

12.2 About Domain URLs

After you complete this chapter, the following URL will be available:

Table 12–1 OIM URLs

Component	URLs	SSO User
Self-service Console	`https://SSO.mycompany.com/identity`	xelsysadm
OIM Administration Console	`http://ADMIN.mycompany.com/sysadmin`	xelsysadm

12.3 Prerequisites

Before extending the domain with Oracle Identity Manager, ensure that the following tasks have been performed:

1. Ensure that the virtual IP addresses for the Oracle Identity Manager and SOA managed servers have been provisioned and enabled. See Section 3.5, "About IP Addresses and Virtual IP Addresses" for details
2. Ensure that you have created the wlfullclient.jar file, as described in Section 12.5, "Creating the wlfullclient.jar File."
3. Ensure the Identity Store is installed and configured.
4. Provision the Oracle Identity Management users as described in Section 9.4, "Preparing the Identity Store."
5. Stop all the managed servers running in your domain, as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components," before extending the domain with Oracle Identity Manager.

Note: Oracle SOA deployed along with Oracle Identity Manager is used exclusively for Oracle Identity Manager work flow. It cannot be used for other purposes.

12.4 Provisioning the OIM Login Modules Under the WebLogic Server Library Directory

Due to issues with versions of the configuration wizard, some environmental variables are not added to the *ASERVER_HOME*/bin/setDomainenv.sh script. This causes certain install sequences to fail. This section is a temporary workaround for that problem. The steps in this section must be performed on all *MW_HOME*s that are associated with the domain hosting Oracle Identity Manager, that is, *IAM_MW_HOME*.

Apply the following steps across all the WebLogic Server homes in the domain.

1. Copy the OIMAuthenticator.jar, oimmbean.jar, oimsigmbean.jar and oimsignaturembean.jar files located under the *IAM_ORACLE_HOME*/server/loginmodule/wls directory to the *IAM_MW_HOME*/wlserver_10.3/server/lib/mbeantypes directory.

   ```
   cp $IAM_ORACLE_HOME/server/loginmodule/wls/* $IAM_MW_HOME/wlserver_
   10.3/server/lib/mbeantypes
   ```

2. Change directory to *MW_HOME*/wlserver_10.3/server/lib/mbeantypes/

   ```
   cd $IAM_MW_HOME/wlserver_10.3/server/lib/mbeantypes
   ```

3. Change the permissions on these files to 750 by using the chmod command.

   ```
   chmod 750 *
   ```

12.5 Creating the wlfullclient.jar File

Oracle Identity Manager uses the wlfullclient.jar library for certain operations. Oracle does not ship this library, so you must create this library manually. Oracle recommends creating this library under the *IAM_MW_HOME*/wlserver_10.3/server/lib directory on all the machines hosting Oracle Identity Manager in the application tier of your environment, such as *IAM_MW_HOME* and *OIM_MW_HOME*.

Follow these steps to create the wlfullclient.jar file:

1. Navigate to the *IAM_MW_HOME*/wlserver_10.3/server/lib directory
2. Set your *JAVA_HOME* environment variable and ensure that the *JAVA_HOME*/bin directory is in your path.
3. Create the wlfullclient.jar file by running:

```
java -jar wljarbuilder.jar
```

12.6 Synchronize System Clocks

Oracle SOA uses Quartz to maintain its jobs and schedules in the database. Synchronize the system clocks for the SOA WebLogic cluster to enable proper functioning of jobs, adapters, and Oracle B2B.

12.7 Extending the Domain to Configure Oracle Identity Manager and Oracle SOA Suite

You must extend your domain to include Oracle Identity Manager. When extending the domain, you must do so from the host that is running the domain's Administration Server. This is the domain IDMDomain on IDMHOST1.

To extend the domain with Oracle Identity Manager, start the configuration wizard on IDMHOST1 by executing the command:

ORACLE_COMMON_HOME/common/bin/config.sh

Proceed as follows

1. On the Welcome screen, select **Extend an existing WebLogic Domain**.

   Click **Next**.
2. On the Select WebLogic Domain Directory screen, select the location of the domain directory for IDMDomain, for example:
   /u01/oracle/config/domains/IDMDomain

   Click **Next**.
3. On the Select Extension Source screen, select **Extend my domain automatically to support the following added products**. From the list below, select: **Oracle Identity Manager**.

   Notes:

   - **Oracle SOA Suite** and **Oracle WSM Policy Manager** are selected automatically. If Oracle WSM Policy Manager has already been installed, the choice is not available.
   - When you select **Oracle Identity Manager**, **Oracle JRF WebServices Asynchronous services** is selected automatically.

   Select **Next**.
4. On the Configure JDBC Component Schemas screen, do the following.

   Select all the data sources listed on the page:

   - **SOA Infrastructure**

- **User Messaging Service**
- **OIM MDS Schema**
- **OWSM MDS Schema**
- **SOA MDS Schema**
- **OIM Schema**

Select **Convert to GridLink**.

Click **Next**.

5. The Gridlink RAC Component Schema screen appears. In this screen, enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU.

   Select all the schemas for your component. Do not select schemas listed for previously configured components.

   For each entry provide the following common information.

   - **Driver**: Select Oracle's driver (Thin) for GridLink Connections,Versions:10 and later.
   - Select **Enable FAN**.
   - Do one of the following:
     - If SSL is not configured for ONS notifications to be encrypted, deselect **SSL**.
     - Select **SSL** and provide the appropriate wallet and wallet password.
   - **Service Listener**: Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the parameter `remote_listener` in the database:

     ```
     SQL>show parameter remote_listener;

     NAME            TYPE   VALUE
     ------------------------------------------------------------
     remote_listener string DB-SCAN.mycompany.com:1521
     ```

     Note: For Oracle Database 11*g* Release 1 (11.1), use the virtual IP and port of each database instance listener, for example:

     ```
     DBHOST1-VIP.mycompany.com (port 1521)
     ```

     and

     ```
     DBHOST2-VIP.mycompany.com (port 1521) (DB_LSNR_PORT)
     ```

     For Oracle Database 10*g*, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources see "Verifying Adapters for Multiple Directory Identity Stores by Using ODSM" in *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.

   - **ONS Host**: Enter the SCAN address for the Oracle RAC database and the ONS remote port as reported by the database:

```
srvctl config nodeapps -s
ONS exists: Local port 6100, remote port 6200, EM port 2016
```

Note: For Oracle Database 11g Release 1 (11.1), use the hostname and port of each database's ONS service, for example:

```
DBHOST1.mycompany.com (port 6200)
```

and

```
DBHOST2.mycompany.com (port 6200)
```

Enter the following RAC component schema information:

Schema Name	Service Name	Schema Owner	Password
OIM Schema	OIMEDG.mycompany.com	EDG_OIM	*password*
SOA Infrastructure	OIMEDG.mycompany.com	EDG_SOAINFRA	*password*
User Messaging Service	OIMEDG.mycompany.com	EDG_ORASDPM	*password*
OIM MDS Schema	OIMEDG.mycompany.com	EDG_MDS	*password*
OWSM MDS Schema	OIMEDG.mycompany.com	EDG_MDS	*password*
SOA MDS Schema	OIMEDG.mycompany.com	EDG_MDS	*password*

If you prefer to use RAC multi datasources, see "Verifying Adapters for Multiple Directory Identity Stores by Using ODSM" in *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.

Click **Next**.

6. On the Test Component Schema screen, the Configuration Wizard attempts to validate the data sources. If the data source validation succeeds, click **Next**. If it fails, click **Previous**, correct the problem, and try again.

   Click **Next**.

7. On the Select Optional Configuration screen, Select:

   - **JMS Distributed Destination**
   - **Managed Servers, Clusters and Machines**
   - **JMS File Store**

   Click **Next**.

8. On the JMS Distributed Destination screen, ensure that all the JMS system resources listed on the screen are uniform distributed destinations. If they are not, select **UDD** form the drop down box. Ensure that the entries look like this:

JMS System Resource	Uniform/Weighted Distributed Destination
UMSJMSSystemResource	UDD

JMS System Resource	Uniform/Weighted Distributed Destination
SOAJMSModule	UDD
OIMJMSModule	UDD
BPMJMSModule	UDD
JRFWSAsyncjmsModule	UDD

Click **Next**.

An Override Warning box with the following message is displayed:

```
CFGFWK-40915: At least one JMS system resource has been selected for conversion
to a Uniform Distributed Destination (UDD). This  conversion will take place
only if the JMS System resource is assigned to a cluster
```

Click **OK** on the Override Warning box.

9. When you first enter the Configure Managed Servers screen, two managed servers called oim_server1 and soa_server1 are created automatically. Rename soa_server1 to WLS_SOA1 and oim_server1 to WLS_OIM1 and update their attributes as shown in the following table. Then, add two new managed servers called WLS_OIM2 and WLS_SOA2 with the following attributes.

Name	Listen Address	Listen Port	Port Variable	SSL Listen Port	SSL Enabled
WLS_SOA1	SOAHOST1VHN	8001	*SOA_PORT*	N/A	No
WLS_SOA2	SOAHOST2VHN	8001	*SOA_PORT*	N/A	No
WLS_OIM1	OIMHOST1VHN	14000	*OIM_PORT*	N/A	No
WLS_OIM2	OIMHOST2VHN	14000	*OIM_PORT*	N/A	No

Notes:

- Do not change the configuration of the managed servers that were configured as a part of previous deployments.
- Do not delete the default managed servers that are created. Rename them as described.

10. On the Configure Clusters screen, create each cluster by clicking **Add**. Supply the following information:

Name	Messaging Mode	Multicast Address	Multicast Port	Cluster Address
oim_cluster	unicast	n/a	n/a	OIMHOST1VHN:14000, OIMHOST2VHN:14000[1]
soa_cluster	unicast	n/a	n/a	SOAHOST1VHN:8001,SOAHOST2VHN:8001[2]

[1] Where 14000 is the *OIM_PORT* from Section B.3

[2] Where 8001 is the *SOA_PORT* from Section B.3

Leave all other fields at the default settings and click **Next**.

> **Note:** Do not change the configuration of the clusters that were configured as a part of previous deployments.

11. On the Assign Servers to Clusters screen, associate the managed servers with the cluster. Click the cluster name in the right pane. Click the managed server under **Servers**, then click the arrow to assign it to the cluster. Assign the following values:

Cluster	Server
oim_cluster	WLS_OIM1
	WLS_OIM2
soa_cluster	WLS_SOA1
	WLS_SOA2

> **Note:** Do not make any changes to clusters that already have entries defined.

Click **Next**.

12. On the Configure Machines screen, create a machine for each host in the topology, if they have not already been created.
    a. Click the **Unix Machine** tab.
    b. **Name**: Name of the host. Best practice is to use the DNS name.
    c. **Node Manager Listen Address**: DNS name of the machine.
    d. **Node Manager Port**: Port for Node Manager

    Provide the information shown in the following table.

Name	Node Manager Listen Address	Node Manager Listen Port	Port Variable
IDMHOST1	IDMHOST1	5556	*NMGR_PORT*
IDMHOST2	IDMHOST2	5556	*NMGR_PORT*

    Leave the default values for all other fields.

    Delete the default local machine entry under the **Machines** tab.

    Click **Next**.

13. On the Assign Servers to Machines screen, assign servers to machines as follows:
    - **IDMHOST1**: `WLS_OIM1` and `WLS_SOA1`
    - **IDMHOST2**: `WLS_OIM2` and `WLS_SOA2`

    Click **Next** to continue.

14. On the Configure JMS File Stores screen, update the directory locations for the JMS file stores. Provide the information shown in the following table.

Name	Directory
UMSJMSFileStore_auto_1	*ASERVER_HOME*/jms/UMSJMSFileStore_auto_1
UMSJMSFileStore_auto_2	*ASERVER_HOME*/jms/UMSJMSFileStore_auto_2
BPMJMSServer_auto_1	*ASERVER_HOME*/jms/BPMJMSServer_auto_1
BPMJMSServer_auto_2	*ASERVER_HOME*/jms/BPMJMSServer_auto_2
SOAJMSFileStore_auto_1	*ASERVER_HOME*/jms/SOAJMSFileStore_auto_1
SOAJMSFileStore_auto_2	*ASERVER_HOME*/jms/SOAJMSFileStore_auto_2
OIMJMSFileStore_auto_1	*ASERVER_HOME*/jms/OIMJMSFileStore_auto_1
OIMJMSFileStore_auto_2	*ASERVER_HOME*/jms/OIMJMSFileStore_auto_2
JRFWSAsyncFileStore_auto_1	*ASERVER_HOME*/jms/JRFWSAsyncFileStore_auto_1
JRFWSAsyncFileStore_auto_2	*ASERVER_HOME*/jms/JRFWSAsyncFileStore_auto_2

Click **Next**.

15. On the Configuration Summary screen, click **Extend** to extend the domain.
16. On the Installation Complete screen, click **Done**.
17. Restart WebLogic Administration Server, as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

12.8 Deploying Oracle Identity Manager and Oracle SOA to Managed Server Domain Directory on IDMHOST1 and IDMHOST2

Once the configuration is complete, you must propagate the Oracle Identity Manager configuration to the managed server directory on IDMHOST1 and IDMHOST2.

You do this by packing and unpacking the domain. You pack the domain first on IDMDomain on IDMHOST1, then unpack it on IDMHOST1 and IDMHOST2.

Follow these steps to propagate the domain to the managed server domain directory.

1. Invoke the pack utility from *ORACLE_COMMON_HOME*/common/bin/ on IDMHOST1.

   ```
   ./pack.sh -domain=ASERVER_HOME -template=oim_domain.jar -template_name="OIM
   Domain" -managed=true
   ```

2. This creates a file called oim_domain.jar. Copy this file to IDMHOST2.
3. On IDMHOST1 and IDMHOST2, invoke the utility unpack, which is also located in the directory: *ORACLE_COMMON_HOME*/common/bin/

   ```
   ./unpack.sh -domain=MSERVER_HOME -template=oim_domain.jar -overwrite_
   domain=true -app_dir=MSERVER_HOME/applications
   ```

12.9 Configuring Oracle Coherence for Deploying Composites

Although deploying composites uses multicast communication by default, Oracle recommends using unicast communication in SOA enterprise deployments. Use unicast if you disable multicast communication for security reasons.

Unicast communication does not enable nodes to discover other cluster members in this way. Consequently, you must specify the nodes that belong to the cluster. You do not need to specify all of the nodes of a cluster, however. You need only specify enough nodes so that a new node added to the cluster can discover one of the existing nodes. As a result, when a new node has joined the cluster, it is able to discover all of the other nodes in the cluster. Additionally, in configurations such as SOA enterprise deployments where multiple IPs are available in the same system, you must configure Oracle Coherence to use a specific host name to create the Oracle Coherence cluster.

Note: An incorrect configuration of the Oracle Coherence framework used for deployment may prevent the SOA system from starting. The deployment framework must be properly customized for the network environment on which the SOA system runs. Oracle recommends the configuration described in this section.

This section contains the following topics:

- Section 12.9.1, "Enabling Communication for Deployment Using Unicast Communication."
- Section 12.9.2, "Specifying the Host Name Used by Oracle Coherence."

12.9.1 Enabling Communication for Deployment Using Unicast Communication

Specify the nodes using the `tangosol.coherence.wka<n>` system property, where `<n>` is a number between 1 and 9. You can specify up to 9 nodes. Start the numbering at 1. This numbering must be sequential and must not contain gaps. In addition, specify the host name used by Oracle Coherence to create a cluster through the `tangosol.coherence.localhost` system property. This local host name should be the virtual host name used by the SOA server as the listener addresses (SOAHOST1VHN and SOAHOST2VHN). Set this property by adding the `-Dtangosol.coherence.localhost` parameters to the Arguments field of the Oracle WebLogic Server Administration Console's Server Start tab.

Tip: To guarantee high availability during deployments of SOA composites, specify enough nodes so that at least one of them is running at any given time.

Note: SOAHOST1VHN is the virtual host name that maps to the virtual IP where WLS_SOA1 listening (in SOAHOST1). SOAHOST2VHN is the virtual host name that maps to the virtual IP where WLS_SOA2 is listening (in SOAHOST2).

12.9.2 Specifying the Host Name Used by Oracle Coherence

Use the Administration Console to specify a host name used by Oracle Coherence.

To add the host name used by Oracle Coherence:

1. Log into the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Environment** node.
3. Click **Servers**. The Summary of Servers page appears.
4. Click the name of the server (**WLS_SOA1** or **WLS_SOA2**, which are represented as hyperlinks) in Name column of the table. The settings page for the selected server appears.
5. Click **Lock & Edit**.
6. Click the **Server Start** tab.
7. Enter the following for WLS_SOA1 and WLS_SOA2 into the Arguments field.

   For WLS_SOA1, enter the following:

   ```
   -Dtangosol.coherence.wka1=SOAHOST1VHN
   -Dtangosol.coherence.wka2=SOAHOST2VHN
   -Dtangosol.coherence.localhost=SOAHOST1VHN
   ```

   For WLS_SOA2, enter the following:

   ```
   -Dtangosol.coherence.wka1=SOAHOST1VHN
   -Dtangosol.coherence.wka2=SOAHOST2VHN
   -Dtangosol.coherence.localhost=SOAHOST2VHN
   ```

   Note: There should be no breaks in lines between the different `-D` parameters. Do not copy or paste the text to your Administration Console's arguments text field. It may result in HTML tags being inserted in the Java arguments. The text should not contain other text characters than those included the example above.

Note: The Coherence cluster used for deployment uses port 8088 by default. This port can be changed by specifying a different port (for example, 8089) with the -Dtangosol.coherence.wkan.port and -Dtangosol.coherence.localport startup parameters. For example:

WLS_SOA1 (enter the following into the Arguments field on a single line, without a carriage return):

```
-Dtangosol.coherence.wka1=SOAHOST1VHN
-Dtangosol.coherence.wka2=SOAHOST2VHN
-Dtangosol.coherence.localhost=SOAHOST1VHN
-Dtangosol.coherence.localport=8089
-Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
```

WLS_SOA2 (enter the following into the Arguments field on a single line, without a carriage return):

```
-Dtangosol.coherence.wka1=SOAHOST1VHN
-Dtangosol.coherence.wka2=SOAHOST2VHN
-Dtangosol.coherence.localhost=SOAHOST2VHN
-Dtangosol.coherence.localport=8089
-Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
```

For more information about Coherence Clusters see the *Oracle Coherence Developer's Guide*.

8. Click **Save** and **Activate Changes**.

Notes:

- You must ensure that these variables are passed to the managed server correctly. (They should be reflected in the server's output log.) Failure of the Oracle Coherence framework can prevent the soa-infra application from starting.
- The multicast and unicast addresses are different from the ones used by the WebLogic Server cluster for cluster communication. SOA guarantees that composites are deployed to members of a single WebLogic Server cluster even though the communication protocol for the two entities (the WebLogic Server cluster and the groups to which composites are deployed) are different.

9. Stop the WebLogic Administration Server on IDMHOST1. by using the WebLogic Administration Console as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."
10. Start the Administration Server on IDMHOST1 using the Node Manager, as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."
11. Start SOA server `WLS_SOA1`.
12. If desired, start other servers that you shut down in Section 12.3, "Prerequisites."

12.10 Configuring Oracle Identity Manager

You must configure the Oracle Identity Manager server instance before you can start the Oracle Identity Manager and SOA Managed Servers. This is performed on IDMHOST1. The Oracle Identity Management Configuration Wizard loads the Oracle Identity Manager metadata into the database and configures the instance.

Before proceeding, ensure that the following are true:

- The Administration Server is up and running.
- SOA Managed Server is up and running.
- The environment variables *DOMAIN_HOME* and *WL_HOME* are ***not*** set in the current shell.

The Oracle Identity Management Configuration Wizard is located under the Identity Management Oracle home. To start the Configuration Wizard, type:

IAM_ORACLE_HOME/bin/config.sh

Proceed as follows:

1. On the Welcome screen, click **Next**
2. On the Components to Configure screen, Select **OIM Server**.

   Click **Next**.
3. On the Database screen, provide the following values:
   - **Connect String**: The connect string for the Oracle Identity Manager database:

     IDMDB1-VIP.mycompany.com:1521:OIMEDG1^IDMDB2-VIP.mycompany.com:1521:OIMEDG2@OIMEDG.mycompany.com where 1521 is the *DB_LSNR_PORT* port from Section B.3.

     If you are using Oracle Database 11.2, replace the vip address and port with the 11.2 SCAN address and port.
   - **OIM Schema User Name**: EDG_OIM
   - **OIM Schema password**: *password*
   - **MDS Schema User Name**: EDG_MDS
   - **MDS Schema Password**: *password*

   Click **Next**.
4. On the WebLogic Administration Server screen, provide the following details for the WebLogic Administration Server:
   - **URL**: The URL to connect to the WebLogic Administration Server. For example:

     t3://ADMINVHN.mycompany.com:7001, where Port 7001 is *WLS_ADMIN_PORT*
   - **UserName**: weblogic.
   - **Password**: Password for the weblogic user

   Click **Next**.
5. On the OIM Server screen, provide the following values:
   - **OIM Administrator Password**: Password for the Oracle Identity Manager Administrator. This is the password for the xelsysadm user. The password

must contain an uppercase letter and a number. Best practice is to use the same password that you assigned to the user `xelsysadm` in Section 9.4, "Preparing the Identity Store."

- **Confirm Password**: Confirm the password·
- **OIM HTTP URL**: Proxy URL for the Oracle Identity Manager Server. This is the URL for the Hardware load balancer that is front ending the OHS servers for Oracle Identity Manager. For example: `http://IDMINTERNAL.mycompany.com:80`.
- **Enable LDAP Sync**: Selected.

Click **Next**.

6. On the LDAP Server Screen, the information you enter is dependent on your implementation. Provide the following details:
   - **Directory Server Type**:
     - `OUD`, if your Identity Store is Oracle Unified Directory.
     - `OID`, if your Identity Store is in Oracle Internet Directory.
     - `OVD` if you access your Identity Store through Oracle Virtual Directory.
   - **Directory Server ID**: A name for your directory server. For example: `IdStore`. This is only required if the directory type is `OID` or `OUD`.
   - **Server URL**: The LDAP server URL. For example: `ldap://IDSTORE.mycompany.com:389`
   - **Server User**: The user name for connecting to the LDAP Server. For example: `cn=oimLDAP,cn=systemids,dc=mycompany,dc=com`
   - **Server Password**: The password for connecting to the LDAP Server.
   - **Server Search DN**: The Search DN, if you are accessing your IDStore using Oracle Virtual Directory Server. For example: `dc=mycompany,dc=com`.

   Click **Next**.
7. On the LDAP Server Continued screen, provide the following LDAP server details:
   - **LDAP Role Container**: The DN for the Role Container. This is the container where the Oracle Identity Manager roles are stored. For example: `cn=Groups,dc=mycompany,dc=com` ·
   - **LDAP User Container**: The DN for the User Container. This is the container where the Oracle Identity Manager users are stored. For example: `cn=Users,dc=mycompany,dc=com`·
   - **User Reservation Container**: The DN for the User Reservation Container. For example: `cn=Reserve,dc=mycompany,dc=com`.

   Click **Next**.
8. On the Configuration Summary screen, verify the summary information.

   Click **Configure** to configure the Oracle Identity Manager instance
9. On the Configuration Progress screen, once the configuration completes successfully, click **Next**.
10. On the Configuration Complete screen, view the details of the Oracle Identity Manager Instance configured.

Click **Finish** to exit the Configuration Wizard.

11. Restart WebLogic Administration Server, as described inSection 17.1, "Starting and Stopping Oracle Identity Management Components."

12.11 Copy SOA Directory

Copy the `soa` directory located under *`ASERVER_HOME`* on IDMHOST1 to *`MSERVER_HOME`* directory on IDMHOST1 and IDMHOST2.

For example:

```
scp -rp ASERVER_HOME/soa user@IDMHOST2:MSERVER_HOME
```

12.12 Starting SOA and Oracle Identity Manager Managed Servers on IDMHOST1 and IDMHOST2

Follow this sequence of steps to start the WLS_OIM1 and WLS_SOA1 Managed Servers on IDMHOST1:

1. Validate that the Administration Server started up successfully by bringing up the Oracle WebLogic Administration Console.
2. If the WLS_SOA1 Managed Server is still running, stop and restart it as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."
3. Start the WLS_OIM1 Managed Server using the WebLogic Administration Console as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

Follow this sequence of steps to start the WLS_OIM2 and WLS_SOA2 Managed Servers on IDMHOST2:

1. Validate that the Administration Server started up successfully by bringing up the Oracle WebLogic Administration Console.
2. Start the WLS_SOA2 Managed Server, using the WebLogic Administration Console as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."
3. Start the WLS_OIM2 Managed Server using the WebLogic Administration Console as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

12.13 Validating Oracle Identity Manager Instance on IDMHOST1 and IDMHOST2

Validate the Oracle Identity Manager Server Instances by bringing up the Oracle Identity Manager Console in a web browser at:

```
http://OIMHOST1VHN.mycompany.com:14000/identity
```

```
http://OIMHOST1VHN.mycompany.com:14000/sysadmin
```

```
http://OIMHOST2VHN.mycompany.com:14000/identity/
```

```
http://OIMHOST2VHN.mycompany.com:14000/sysadmin/
```

Log in using the `xelsysadm` username and password.

Note: When you log in for the first time, you are prompted to setup Challenge Questions. Please do so before proceeding further.

Validate Oracle SOA Suite using the URLs:

`http://SOAHOST1VHN.mycompany.com:8001/soa-infra`

`http://SOAHOST2VHN.mycompany.com:8001/soa-infra`

where `8001` is *`SOA_PORT`* in Section B.3.

Log in as the `weblogic` user.

12.14 Configuring Oracle Identity Manager to Reconcile from ID Store

In the current release, the `LDAPConfigPostSetup` script enables all the LDAPSync-related incremental Reconciliation Scheduler jobs, which are disabled by default. The LDAP configuration post-setup script is located under the *`IAM_ORACLE_HOME`*`/server/ldap_config_util` directory. Run the Script on IDMHOST1, as follows:

1. Edit the `ldapconfig.props` file located under the *`IAM_ORACLE_HOME`*`/server/ldap_config_util` directory and provide the following values:

Parameter	Value	Description
`OIMProviderURL`	`t3://OIMHOST1VHN.mycompany.com:14000,OIMHOST2VHN.mycompany.com:14000`[1]	List of Oracle Identity Manager managed servers.
`LDAPURL`	Specify the URL for the Oracle Virtual Directory instance, for example: `ldap://IDSTORE.mycompany.com:389`	Identity Store URL. Only required if IDStore is accessed using Oracle Virtual Directory.
`LDAPAdminUserName`	`cn=oimLDAP,cn=systemids,dc=mycompany,dc=com`	Name of use used to connect to Identity Store. Only required if your Identity Store is in Oracle Virtual Directory. This user should **not** be located in `cn=Users,dc=mycompany,dc=com`.
`LIBOVD_PATH_PARAM`	*`MSERVER_HOME`*`/config/fmwconfig/ovd/oim`	Required unless you access your identity store using Oracle Virtual Directory.

[1] Where `14000` is the *`OIM_PORT`* from Section B.3.

Note: `usercontainerName`, `rolecontainername`, and `reservationcontainername` are not used in this step.

2. Save the file.
3. Set `MW_HOME` to *`IAM_MW_HOME`*.

Set ORACLE_HOME to *IAM_ORACLE_HOME*.

Set JAVA_HOME to *JAVA_HOME*.

Set WL_HOME to *MW_HOME*/wlserver_10.3.

Set APP_SERVER to weblogic.

Set OIM_ORACLE_HOME to *IAM_ORACLE_HOME*.

Set DOMAIN_HOME set *MSERVER_HOME*.

4. Run LDAPConfigPostSetup.sh. The script prompts for the LDAP admin password and the Oracle Identity Manager admin password. For example:

```
IAM_ORACLE_HOME/server/ldap_config_util/LDAPConfigPostSetup.sh path_to_
property_file
```

For example:

```
IAM_ORACLE_HOME/server/ldap_config_util/LDAPConfigPostSetup.sh IAM_ORACLE_
HOME/server/ldap_config_util
```

12.15 Configuring Oracle Identity Manager to Work with the Oracle Web Tier

This section describes how to configure Oracle Identity Manager to work with the Oracle Web Tier.

This section contains the following topics:

- Section 12.15.1, "Configuring Oracle HTTP Servers to Front End the Oracle Identity Manager and SOA Managed Servers"
- Section 12.15.2, "Changing Host Assertion in WebLogic"
- Section 12.15.3, "Updating SOA Endpoints"
- Section 12.15.4, "Validating Web Tier Integration"

12.15.1 Configuring Oracle HTTP Servers to Front End the Oracle Identity Manager and SOA Managed Servers

If you are adding OIM to an existing domain then don't forget to include OIM in the Web Tier configuration as described in Section 10.3, "Post Configuration Tasks."

12.15.2 Changing Host Assertion in WebLogic

Because the Oracle HTTP Server acts as a proxy for WebLogic, by default certain CGI environment variables are not passed through to WebLogic. These include the host and port. You must tell WebLogic that it is using a virtual site name and port so that it can generate internal URLs appropriately.

To do this, log in to the WebLogic administration console at the URL listed in Section 17.2, "About Identity Management Console URLs." Proceed as follows:

1. Select **Clusters** from the home page or, alternatively, select **Environment** -> **Clusters** from the **Domain** structure menu.
2. Click **Lock and Edit** in the Change Center Window to enable editing.
3. Click the **Cluster Name** (**soa_cluster**).
4. In the **Configuration** tab, select the **HTTP** subtab.

Enter:

- **Frontend Host**: `IDMINTERNAL.mycompany.com`
- **Frontend HTTP Port**: `80` (*HTTP_PORT*)

5. Click **Save**.
6. Click **Activate Changes** in the Change Center window to enable editing.

12.15.3 Updating SOA Endpoints

Update SOA endpoints, as follows:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control at the address listed in Section 17.2, "About Identity Management Console URLs."
2. Expand the **SOA** folder in the Navigation pane and right click **soa-infra**
3. Select **SOA Administration -> Common Properties**
4. Click on the link **More SOA Infra Advanced Configuration Properties**.
5. Edit the following properties and apply the changes:
   - **ServerURL**: `http://IDMINTERNAL.mycompany.com:80`
   - **CallbackServerURL**: `http://IDMINTERNAL.mycompany.com:80`
   - **HttpServerURL**: `http://IDMINTERNAL.mycompany.com:80`
6. Click **Apply**.
7. Restart WLS_SOA1 and WLS_SOA2 as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

12.15.4 Validating Web Tier Integration

Validate web tier integration as follows:

12.15.4.1 Validating Oracle Identity Manager Instance from the Web Tier

Validate the Oracle Identity Manager Server Instance by bringing up the Oracle Identity Manager Console in a web browser. at:

`https://SSO.mycompany.com:443/identity`

and

`http://ADMIN.mycompany.com/sysadmin`

Log in using the `xelsysadm` username and password.

12.15.4.2 Validating Accessing SOA from the Web Tier

Validate SOA by accessing the URL:

`http://IDMINTERNAL.mycompany.com:80/soa-infra`

and logging in as the WebLogic administration user.

Note: After WebGate is enabled, **soa-infra** is not available.

12.16 Configuring a Default Persistence Store for Transaction Recovery

The WLS_OIM and WLS_SOA Managed Servers have a transaction log that stores information about committed transactions that are coordinated by the server that might not have been completed. The WebLogic Server uses this transaction log for recovery from system crashes or network failures. To leverage the migration capability of the Transaction Recovery Service for the servers within a cluster, store the transaction log in a location accessible to a server and its backup servers.

> **Note:** Preferably, this location should be on a dual-ported SCSI disk or on a Storage Area Network (SAN).

Perform these steps to set the location for the default persistence stores for the Oracle Identity Manager and SOA Servers:

1. Create the following directory on the shared storage:

   ASERVER_HOME/tlogs

2. Log in to the Oracle WebLogic Server Administration Console.
3. Click **Lock and Edit**.
4. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node.

   The Summary of Servers page is displayed.
5. Click the name of either the Oracle Identity Manager or the SOA server (represented as a hyperlink) in the **Name** column of the table.
6. The Settings page for the selected server is displayed, and defaults to the **Configuration** tab.
7. Open the **Services** sub tab.
8. Under the **Default Store** section of the page, provide the path to the default persistent store on shared storage. The directory structure of the path is as follows:
   - For Oracle Identity Manager Servers: *ASERVER_HOME*/tlogs
   - For SOA Servers: *ASERVER_HOME*/tlogs

   > **Note:** To enable migration of the Transaction Recovery Service, specify a location on a persistent storage solution that is available to other servers in the cluster. All the servers that are a part of the cluster must be able to access this directory.

9. Click **Save and Activate**.
10. Repeat these steps, selecting the other SOA server on the Summary of Servers page.
11. Restart the Oracle Identity Manager and SOA Managed Servers, as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components," to make the changes take effect.

12.17 Configuring UMS Email Notification

This section describes how to configure UMS email notification. This is optional. The following steps assume that an email server has been set up and that Oracle Identity Management can use it to send the email notifications.

1. Log in to the Oracle Enterprise Manager Fusion Middleware Control instance that is associated with Oracle Identity Manager, at the URL listed in Section 17.2, "About Identity Management Console URLs.".
2. Expand **User Messaging Service**.
3. Right click **usermessagingdriver-email (WLS_SOA1)** and select **email driver properties**.
4. Enter the following information:
   - **OutgoingMailServer**: name of the SMTP server, for example: SMTP.mycompany.com
   - **OutgoingMailServerPort**: port of the SMTP server, for example: 465 for SSL outgoing mail server and 25 for non-SSL
   - **OutgoingMailServerSecurity**: The security setting used by the SMTP server Possible values can be None/TLS/SSL. If the mail server is configured to accept SSL requests, perform these additional steps to remove DemoTrust store references from the SOA environment:
     a. Modify the *MSERVER_HOME*`/bin/setDomainEnv.sh` file to remove the DemoTrust references `-Djavax.net.ssl.trustStore=`*WL_HOME*`/server/lib/DemoTrust.jks` from `EXTRA_JAVA_PROPERTIES`.
     b. Modify the `startManagedWeblogic.sh` file on IDMHOST1 and IDMHOST2. Remove the `weblogic.security.SSL.trustedCAKeyStore` property set in `JAVA_OPTIONS` from this file. That is, remove the line that looks like this:

        ```
        JAVA_OPTIONS="-Dweblogic.security.SSL.trustedCAKeyStore="{MW_
        HOME}/server/server/lib/cacerts" ${JAVA_OPTIONS}"
        ```

     c. Restart Oracle Identity Manager and the OIM and SOA managed servers.
   - **OutgoingUsername**: Any valid username
   - **OutgoingPassword:**
     a. Choose **Indirect Password**, **Create New User**
     b. Provide a unique string for **Indirect Username/Key**, for example: `OIMEmailConfig`. This will mask the password and not expose it in clear text in the configuration file.
     c. Provide valid password for this account.

   Click **Apply**.

   Repeat Steps 3 and 4 for each SOA server.
5. From the Navigator Select **WebLogic Domain** -> **DomainName**.
6. From the menu, select **System Mean Browser**.
7. Expand **Application Defined MBeans** -> **oracle.iam** -> **Server: WLS_OIM1** -> **Application: oim** -> **IAMAppRuntimeMBean**.
8. Click **UMSEmailNotificationProviderMBean**.

9. Enter:
   - **WSUrl**: `http://IDMINTERNAL.mycompany.com:80/ucs/messaging/webservice`
   - **Policies**: Leave blank.
   - **CSFKey**: `Notification.Provider.Key`
10. Click **Apply**.

12.18 Add Load Balancer Certificate to SOA Keystore

Using a browser, obtain the certificate for SSO.mycompany.com. (Refer to your browser documentation to determine how to do this.) Save the file to IDMHOST1 in the `.pem` format, for example: `/tmp/sso.pem`.

Then import the certificate into the SOA keystore using the `keytool` command, which is provided as part of the JDK (Java Development Kit). Proceed as follows:

1. Set the environment variables.
   - Set `JAVA_HOME` to *`JAVA_HOME`*.
   - Set `PATH` to *`JAVA_HOME`*`/bin:$PATH`.
2. Change directory to *`WL_HOME`*`/server/lib`.

   ```
   cd WL_HOME/server/lib
   ```

3. Add the certificate to the SOA keystore using the following command:

   ```
   keytool -import -file /tmp/sso.pem -alias SSOAlias -keystore DemoTrust.jks
   -storepass DemoTrustKeyStorePassPhrase
   ```

12.19 Excluding Users from Oracle Identity Manager Reconciliation

By default Oracle Identity Management reconciles all users that are located in the LDAP `container cn=Users`. Once reconciled, these users are subject to the usual password ageing policies defined in Oracle Identity Manager. This is not desirable for system accounts. It is recommended that you exclude the following accounts from this reconciliation:

- `xelsysadm`
- `oimLDAP`
- `oamLDAP`

To exclude these users from reconciliation and discard failed reconciliation events, perform the steps in the following sections, using ODSM and the OIM Console.

This section contains the following topics:

- Section 12.19.1, "Adding the orclAppIDUser Object Class to the User by Using ODSM."
- Section 12.19.2, "Closing Failed Reconciliation Events by Using the OIM Console."

12.19.1 Adding the orclAppIDUser Object Class to the User by Using ODSM

Users can be excluded from OIM reconciliation by attaching the object class `orclAppIDUser` to each of the users.

The example below is for Oracle Unified Directory using ODSM for Oracle Unified Directory. If you are using Oracle Internet Directory or Oracle Virtual Directory you can use ODSM for OID/OVD to accomplish this. For directories other than Oracle Unified Directory, Oracle Internet Directory or Oracle Virtual Directory, refer to your system documentation for information on how to do this.

1. Log in to ODSM at:

   `http://ADMIN.mycompany.com/odsm`

2. Connect to one of the LDAP instances that hosts the user to be excluded.

   - **Server**: One of the Oracle Unified Directory hosts, for example: `IDMHOST1.mycompany.com`
   - **Administration Port**: The Oracle Unified Directory administration port (*`LDAP_DIR_ADMIN_PORT`*), for example: `4444`
   - **User Name**: Directory Administrator, for example: `cn=oudadmin`

   If prompted, trust the server certificate.

3. Select **Data Browser**.
4. Click on the user you want to exclude, navigating to the user in the directory tree, for example: `Root -> dc=mycompany,dc=com -> cn=systemids -> cn=UserId`
5. Click the **Attributes** tab.
6. Click **+** in the **Object Class** list (in mandatory properties)
7. Enter the property name: `orclAppIDUser`
8. Click **Apply**.

Repeat Steps 1-8 for each user to be excluded.

12.19.2 Closing Failed Reconciliation Events by Using the OIM Console

This step is required to clear out failed reconciliation events. Failed reconciliation events are repeatedly retried, which puts an unnecessary load on the system.

1. Log in to the OIM Administration Console as the `xelsysadm` user, using the URL: `http://ADMIN.mycompany.com/sysadmin`
2. Click **Reconciliation** under **Event Management**.
3. Click **Advanced Search**.
4. In the **Current Status** field, select **Equals**. In the **Search** box, select **Creation Failed** from the list.
5. Click **Search**.
6. Select each of the events.
7. From the Actions menu, select **Close Event**.
8. In the Confirmation window enter a justification, such as `Close Failed Reconciliation Events`.
9. Click **Closed**.

10. Click **OK** to acknowledge the confirmation message.

12.20 Enabling Oracle Identity Manager to Connect to SOA Using the Administrative Users Provisioned in LDAP

Oracle Identity Manager connects to SOA as SOA administrator, with the username `weblogic` by default. As mentioned in the previous sections, a new administrator user is provisioned in the central LDAP store to manage Identity Management Weblogic Domain.

Perform the following postinstallation steps to enable Oracle Identity Manager to work with the Oracle WebLogic Server administrator user provisioned in the central LDAP store. This enables Oracle Identity Manager to connect to SOA without any problem:

1. Log in to Enterprise Manager at the URL listed in Section 17.2, "About Identity Management Console URLs."
2. Select **Farm_IDMDomain** –> **Identity and Access** –> **OIM** –> **oim(11.1.2.0.0).**
3. Select **System MBean Browser** from the menu or right click to select it.
4. Select **Application defined Mbeans** –> **oracle.iam** –> **Server: WLS_OIM1** –> **Application: oim** –> **XML Config** –> **Config** –> **XMLConfig.SOAConfig** –> **SOAConfig**
5. Change the **username** attribute to the Oracle WebLogic Server administrator username provisioned in Section 9.4, "Preparing the Identity Store," for example: `weblogic_idm`.

   Change **SOA Config RMI URL** to:

   `cluster:t3://soa_cluster`

   Change **SOA Config SOAP URL** to:

   `http://IDMINTERNAL.mycompany.com:80`
6. Click **Apply**.
7. Select **Weblogic Domain** –> **IDMDomain** from the Navigator.
8. Select **Security** –> **Credentials** from the down menu.
9. Expand the key **oim**.
10. Click **SOAAdminPassword**.
11. Click **Edit**.
12. Change the username to `weblogic_idm` and set the password to the accounts password.
13. Click **OK**.
14. Execute the following WLST command in order to add the `WLSAdmins` group as a member of `SOAAdmin` application role:

```
ORACLE_COMMON_HOME/wlst.sh
MW_HOME/oracle_common/modules/oracle.jps_
11.1.1/common/wlstscripts/grantAppRole.py -principalClass
weblogic.security.principal.WLSGroupImpl -appStripe soa-infra -appRoleName
SOAAdmin -principalName "WLSAdmins"
```

Where `WLSAdmins` is the group created in Section 9.4, "Preparing the Identity Store" (*`IDSTORE_WLSADMINGROUP`*).

15. Restart WLS_SOA1 and WLS_SOA2 as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."
16. Run the reconciliation process to enable the Oracle WebLogic Server administrator, `weblogic_idm`, to be visible in the OIM Identity Console. Follow these steps:
    a. Log in to the OIM Administration Console at the URL `http://ADMIN.mycompany.com/sysadmin` as the user `xelsysadm`.
    b. Click **Scheduler** under System Management.
    c. Enter **LDAP*** in the search box.
    d. Click the arrow for the **Search Scheduled Jobs** to list all the schedulers.
    e. Select **LDAP User Create and Update Full Reconciliation** and **LDAP Role Membership Full Reconciliation**.
    f. Click **Run Now** to run the job.
    g. Repeat for the job **LDAP Role Create and Update Full Reconciliation**.
    h. Log in to the OIM Identity Console at the URL listed in Section 17.2, "About Identity Management Console URLs." Perform a search to verify that the user `weblogic_idm` is visible.
17. Log in to the WebLogic Console.
18. Click **Lock & Edit** in the Change Center.
19. Navigate to `IDMDomain -> Services -> Foreign JNDI Providers`
20. Click on `ForeignJNDIProvider-SOA`
21. Under the **Configuration -> General** tab, change user `weblogic` to `weblogic_idm` and specify the corresponding password.
22. Click **Save** and **Activate Changes**.

12.21 Modifying Oracle Identity Manager to Support Active Directory

Modify Oracle Identity Manager as described in the following subsections.

This section contains the following topics:

- Section 12.21.1, "Updating the Username Generation Policy for Active Directory."
- Section 12.21.2, "Modifying the Oracle Identity Manager Properties to Support Active Directory."

12.21.1 Updating the Username Generation Policy for Active Directory

If your back end directory is Active Directory, you must update Oracle Identity Manager so that it only allows user names with a maximum of 20 characters. This is a limitation of Active Directory. Update the username generation policy from `DefaultComboPolicy` to `FirstnameLastnamepolicyforAD` as follows.

1. Log in to the OIM Administration Console at the URL listed in Section 12.2, "About Domain URLs."
2. Click on **System Configuration** under **System Management**.

3. In the **Search** box enter **Default Policy for Username Generation** and Click **Search**
4. Click **Default Policy for Username Generation**.
5. In the **Value** field, update the entry from `oracle.iam.identity.usermgmt.impl.plugins.DefaultComboPolicy` to `oracle.iam.identity.usermgmt.impl.plugins.FirstNameLastNamePolicyForAD`.
6. Click **Save**.

12.21.2 Modifying the Oracle Identity Manager Properties to Support Active Directory

When first installed, Oracle Identity Manager has a set of default system properties for its operation.

If your Identity Store is in Active Directory, you must change the System property `XL.DefaultUserNamePolicyImpl` to `oracle.iam.identity.usermgmt.impl.plugins.FirstNameLastNamePolicyForAD` or `oracle.iam.identity.usermgmt.impl.plugins.LastNameFirstNamePolicyForAD`.

To learn how to do this, see the Administering System Properties chapter of *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Manager*.

12.22 Backing Up Oracle Identity Manager

Perform a backup of the Oracle Identity Manager configuration at this point. Back up the database, the WebLogic domain, and the LDAP directories, as described in Section 17.6.3, "Performing Backups During Installation and Configuration."

12.23 Integrating Oracle Identity Manager and Oracle Access Management Access Manager

This section describes how to integrate Oracle Identity Manager and Oracle Access Management Access Manager.

This section contains the following topics:

12.23.1 Prerequisites

1. Ensure that OIM11g has been installed and configured as described in Chapter 12, "Extending the Domain to Include Oracle Identity Manager."
2. Ensure that Oracle Access Management has been installed and configured as described in Chapter 11, "Extending the Domain to Include Oracle Access Management."
3. Ensure that OHS has been installed and configured as described in Section 10.2.3, "Installing Oracle HTTP Server."

12.23.2 Adding Forgotten Password Links to the OAM Login Page

If you ran `idmConfigTool` in Section 11.6.3, "Configuring Access Manager by Using the IDM Configuration Tool" with the parameter `OAM11G_OIM_INTEGRATION_REQ` is set to `true`, you can skip this step.

If you ran the command with `OAM11G_INTEGRATION_FLAG` set to **false**, you must now rerun the command, this time setting `OAM11G_OIM_INTEGRATION_REQ` to **true** and specifying a value for `OAM11G_OIM_OHS_URL`.

12.23.3 Copying OAM Keystore Files to IDMHOST1 and IDMHOST2

If you are using Access Manager with the Simple Security Transport model, you must copy the OAM keystore files that were generated in Section 11.11, "Creating a Single Keystore for Integrating Access Manager with Other Components" to IDMHOST1 and IDMHOST2. Copy the keystore files `ssoKeystore.jks` and `oamclient-truststore.jks` to the directory *`MSERVER_HOME`*`/config/fmwconfig` on IDMHOST1 and IDMHOST2.

12.23.4 Integrating Oracle Identity Manager with Oracle Access Manager Using the idmConfigTool

Integrating Oracle Identity Manager with Access Manager using a WebGate profile employs an Access Manager Trusted Authentication Protocol (TAP) scheme. This is different from previous releases which used Network Assertion Protocol (NAP).

To integrate Access Manager with Oracle Identity Manager, perform the following steps on IDMHOST1:

1. Set `MW_HOME` to *`IAM_MW_HOME`*.

   Set `ORACLE_HOME` to *`IAM_ORACLE_HOME`*.

   Set `JAVA_HOME` to *`JAVA_HOME`*.

2. Create a properties file for the integration called `oimitg.props`, with the following contents.

```
LOGINURI: /${app.context}/adfAuthentication
LOGOUTURI: /oamsso/logout.html
AUTOLOGINURI: None
ACCESS_SERVER_HOST: IDMHOST1.mycompany.com
ACCESS_SERVER_PORT: 5575
ACCESS_GATE_ID: Webgate_IDM
COOKIE_DOMAIN: .mycompany.com
COOKIE_EXPIRY_INTERVAL: 120
OAM_TRANSFER_MODE: simple
WEBGATE_TYPE: ohsWebgate11g
SSO_ENABLED_FLAG: true
```

```
IDSTORE_PORT: 389
IDSTORE_HOST: IDSTORE.mycompany.com
IDSTORE_DIRECTORYTYPE: OUD, OID or OVD
IDSTORE_ADMIN_USER: cn=oamLDAP,cn=systemids,dc=mycompany,dc=com
IDSTORE_USERSEARCHBASE: cn=Users,dc=mycompany,dc=com
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=mycompany,dc=com
IDSTORE_LOGINATTRIBUTE: uid
MDS_DB_URL: jdbc:oracle:thin:@(DESCRIPTION=(LOAD_
BALANCE=on)(FAILOVER=on)(ADDRESS_
LIST=(ADDRESS=(protocol=tcp)(host=IDMDBHOST1-VIP.mycompany.com)(port=1521))(ADD
RESS=(protocol=tcp)(host=IDMDBHOST2-VIP.mycompany.com)(port=1521)))(CONNECT_
DATA=(SERVER=DEDICATED)(SERVICE_NAME=OIMEDG.mycompany.com)))
MDS_DB_SCHEMA_USERNAME: EDG_MDS
OIM_MANAGED_SERVER_NAME: WLS_OIM1
WLSADMIN: weblogic
WLSPORT: 7001
WLSHOST: ADMINVHN.mycompany.com
DOMAIN_NAME: IDMDomain
DOMAIN_LOCATION: ASERVER_HOME
```

where:

- ACCESS_SERVER_PORT is the Access Server Proxy port. This is *OAM_PROXY_PORT* in Section B.3.
- OAM_TRANSFER_MODE is set to simple if your access manager servers are configured to accept requests using the simple mode. Otherwise set OAM_TRANSFER_MODE to open
- SSO_ENABLED_FLAG always set to true.
- WEBGATE_TYPE is the type of WebGate agent you want to create. Valid values are ohsWebgate11g and ohsWebgate10.
- IDSTORE_HOST is the load balancer virtual host fronting your Identity store (*LDAP_LBR_HOST*)
- IDSTORE_PORT is the load balancer virtual port fronting your Identity store (*LDAP_LBR_PORT*).
- IDSTORE_DIRECTORYTYPE is set to OVD if you are using Oracle Virtual Directory server to connect to either a non-OID directory or Oracle Internet Directory. Set it to OID if your Identity Store is in Oracle Internet Directory and to OUD if you are connecting to Oracle Unified Directory.
- IDSTORE_USERSEARCHBASE is the location in the directory where Users are Stored.
- IDSTORE_GROUPSEARCHBASE is the location in the directory where Groups are Stored.
- IDSTORE_LOGINATTRIBUTE is the LDAP attribute which contains the users Login name.
- MDS_DB_URL contains the JDBC connection information for your database in the form: jdbc:oracle:thin:@(DESCRIPTION=(LOAD_ BALANCE=on)(FAILOVER=on)(ADDRESS_ LIST=(ADDRESS=(protocol=tcp)(host=IDMDBHOST1-VIP.mycompany .com)(port=1521))(ADDRESS=(protocol=tcp)(host=IDMDBHOST2-V IP.mycompany.com)(port=1521)))(CONNECT_ DATA=(SERVER=DEDICATED)(SERVICE_

NAME=OIMEDG.mycompany.com))) where 1521 is the *DB_LSNR_PORT* in Section B.3.

- MDS_DB_SCHEMA_USERNAME is the name of the schema in the Identity Management Database that holds MDS data. See Section 6.6, "Loading the Identity Management Schemas in the Oracle RAC Database by Using RCU."
- OIM_MANAGED_SERVER_NAME is the name of one of the OIM Managed Servers. It does not matter which one you use.
- WLSHOST (*ADMINVHN*) is the host of your administration server, *WLS_ADMIN_HOST* in Section B.3. This is the virtual name.
- WLSPORT is the port of your administration server, *WLS_ADMIN_PORT* in Section B.3.
- WLSADMIN is the WebLogic administrative user you use to log in to the WebLogic console.
- DOMAIN_NAME is the name of the domain that hosts Oracle Identity Manager.
- DOMAIN_LOCATION is the path to the domain on disk, that is, *ASERVER_HOME*.

3. Integrate Access Manager with Oracle Identity Manager using the command idmConfigTool, which is located at:

   IAM_ORACLE_HOME/idmtools/bin

   Note: When you run the idmConfigTool, it creates or appends to the file idmDomainConfig.param. This file is generated in the same directory that the idmConfigTool is run from. To ensure that each time the tool is run, the same file is appended to, always run the idmConfigTool from the directory:

   IAM_ORACLE_HOME/idmtools/bin

   The syntax of the command is

   ```
   idmConfigTool.sh -configOIM input_file=configfile
   ```

   For example:

   ```
   IAM_ORACLE_HOME/idmtools/bin/idmConfigTool.sh -configOIM input_
   file=oimitg.props
   ```

   When the script runs you are prompted for the following information:

   - Access Gate Password
   - SSO Keystore Password
   - Global Passphrase
   - Idstore Admin Password
   - MDS Database schema password
   - Admin Server User Password

   Sample output:

   ```
   Enter sso access gate password :
   Enter sso keystore jks password :
   Enter sso global passphrase :
   ```

```
Enter mds db schema password :
Enter idstore admin password :
Enter admin server user password :


********* Seeding OAM Passwds in OIM *********


Completed loading user inputs for - CSF Config


Completed loading user inputs for - Dogwood Admin WLS

Connecting to t3://ADMINVHN.mycompany.com:7001

Connection to domain runtime mbean server established

Seeding credential :SSOAccessKey

Seeding credential :SSOGlobalPP

Seeding credential :SSOKeystoreKey


********* ********* *********


********* Activating OAM Notifications *********


Completed loading user inputs for - MDS DB Config

Apr 3, 2012 11:56:09 PM oracle.mds
NOTIFICATION: PManager instance is created without multitenancy support as JVM
flag "oracle.multitenant.enabled" is not set to enable multitenancy support.
Initialized MDS resources

Apr 3, 2012 11:56:09 PM oracle.mds
NOTIFICATION: PManager instance is created without multitenancy support as JVM
flag "oracle.multitenant.enabled" is not set to enable multitenancy support.
Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: transfer operation started.
Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: transfer is completed. Total number of documents successfully
processed : 1, total number of documents failed : 0.
Upload to DB completed


Releasing all resources

Notifications activated.


********* ********* *********


********* Seeding OAM Config in OIM *********


Completed loading user inputs for - OAM Access Config
```

```
Validated input values

Initialized MDS resources

Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: PManager instance is created without multitenancy support as JVM
flag "oracle.multitenant.enabled" is not set to enable multitenancy support.
Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: transfer operation started.
Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: transfer is completed. Total number of documents successfully
processed : 1, total number of documents failed : 0.
Download from DB completed

Releasing all resources

Updated /u01/oracle/products/access/iam/server/oamMetadata/db/oim-config.xml

Initialized MDS resources

Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: PManager instance is created without multitenancy support as JVM
flag "oracle.multitenant.enabled" is not set to enable multitenancy support.
Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: transfer operation started.
Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: transfer is completed. Total number of documents successfully
processed : 1, total number of documents failed : 0.
Upload to DB completed


Releasing all resources

OAM configuration seeded. Please restart oim server.


********* ********* *********


********* Configuring Authenticators in OIM WLS *********


Completed loading user inputs for - LDAP connection info

Connecting to t3://ADMINVHN.mycompany.com:7001

Connection to domain runtime mbean server established

Starting edit session

Edit session started

Connected to security realm.

Validating provider configuration

Validated desired authentication providers

Created OAMIDAsserter successfuly
```

```
OAMIDAsserter is already configured to support 11g webgate

Created OIMSignatureAuthenticator successfuly

Created OVDAuthenticator successfuly

Setting attributes for OVDAuthenticator

All attributes set. Configured inOVDAuthenticatornow

LDAP details configured in OVDAuthenticator

Control flags for authenticators set sucessfully

Reordering of authenticators done sucessfully

Saving the transaction

Transaction saved

Activating the changes

Changes Activated. Edit session ended.

Connection closed sucessfully


********* ********* *********

The tool has completed its operation. Details have been logged to
automation.log
```

Note: If you have already enabled single sign-on for your WebLogic Administration Consoles as described in Section 15.3, "Configuring WebLogic Security Providers" when this script is run, you might see the following errors when this script is run:

```
ERROR: Desired authenticators already present.
[Ljava.lang.String;@7fdb492]
ERROR: Error occurred while configuration. Authentication providers
to be configured already present.
ERROR: Rolling back the operation..
```

These errors can be ignored.

4. Check the log file for errors and correct them if necessary.
5. Restart the Administration Servers as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components.".

12.23.5 Perform Bug 13824816 Workaround, if Necessary

If you did not perform the workaround described in Section 11.6.8, "Perform Bug 13824816 Workaround" when you configured Oracle Access Management Access Manager, then perform those steps now.

12.23.6 Updating Existing LDAP Users with Required Object Classes

You must update existing LDAP users with the object classes `OblixPersonPwdPolicy`, `OIMPersonPwdPolicy`, and `OblixOrgPerson`.

> **Note:** This is not required in the case of a fresh setup where you do not have any existing users.

1. On IDMHOST1, create a properties file for the integration called `user.props`, with the following contents:

```
IDSTORE_HOST: IDSTORE.mycompany.com
IDSTORE_PORT: 389
IDSTORE_ADMIN_USER: cn=oudadmin
IDSTORE_DIRECTORYTYPE: OUD, OID, or OVD
IDSTORE_USERSEARCHBASE: cn=Users,dc=mycompany,dc=com
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=mycompany,dc=com
PASSWORD_EXPIRY_PERIOD: 7300
IDSTORE_LOGINATTRIBUTE: uid
```

   Where:

   - `IDSTORE_HOST` is the name of LDAP server. For example:

     `IDSTORE.mycompany.com`

   - `IDSTORE_PORT` is the port of the LDAP server.
   - `IDSTORE_ADMIN_USER` is the bind DN of an administrative user. For example:

     `cn=orcladmin or cn=oudadmin`

   - `IDSTORE_DIRECTORYTYPE` is the type of directory, valid values are OUD, OID and OVD.
   - `IDSTORE_USERSEARCHBASE` is the location of users in the directory. For example:

     `cn=Users,dc=mycompany,dc=com`

   - `IDSTORE_GROUPSEARCHBASE` is the location of groups in the directory. For example:

     `cn=Groups,dc=mycompany,dc=com`

   - `IDSTORE_LOGINATTRIBUTE` this is the directory login attribute name. For example:

     `uid`.

   - `PASSWORD_EXPIRY_PERIOD` is the password expiry period.

2. Set `ORACLE_HOME` to *`IAM_ORACLE_HOME`*.

   Set `MW_HOME` to *`MW_HOME`*.

   Set `JAVA_HOME` to *`JAVA_HOME`*.

3. Upgrade existing LDAP, using the command `idmConfigTool`, which is located at: *`IAM_ORACLE_HOME`*`/idmtools/bin`

Note: When you run the `idmConfigTool`, it creates or appends to the file `idmDomainConfig.param`. This file is generated in the same directory that the `idmConfigTool` is run from. To ensure that each time the tool is run, the same file is appended to, always run the `idmConfigTool` from the directory:

IAM_ORACLE_HOME`/idmtools/bin`

The syntax of the command is:

```
idmConfigTool.sh -upgradeLDAPUsersForSSO input_file=configfile
```

For example:

```
idmConfigTool.sh -upgradeLDAPUsersForSSO input_file=user.props
```

When prompted, enter the password of the user you are using to connect to your Identity Store.

Sample output:

```
Enter IDSTORE_ADMIN_PASSWD :

********* Upgrading LDAP Users With OAM ObjectClasses *********

Completed loading user inputs for - LDAP connection info

Completed loading user inputs for - LDAP Upgrade

Upgrading ldap users at - cn=Users,dc=mycompany,dc=com

Parsing - cn=weblogic_idm,cn=Users,dc=mycompany,dc=com

objectclass OIMPersonPwdPolicy not present in cn=weblogic_
idm,cn=Users,dc=mycompany,dc=com. Seeding it

obpasswordexpirydate added in cn=weblogic_idm,cn=Users,dc=mycompany,dc=com

Parsing - cn=oamadmin,cn=Users,dc=mycompany,dc=com

objectclass OIMPersonPwdPolicy not present in
cn=oamadmin,cn=Users,dc=mycompany,dc=com. Seeding it

obpasswordexpirydate added in cn=oamadmin,cn=Users,dc=mycompany,dc=com

Finished parsing LDAP

LDAP Users Upgraded.
```

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

12.23.7 Update TAP Authentication Scheme

After integrating Oracle Access Management Access Manager with Oracle Identity Manager, you must update the TAP authentication scheme to perform user validation using the LDAP attribute `uid`.

Proceed as follows:

1. Log in to the OAM console at: `http://ADMIN.mycompany.com/oamconsole`
2. Click **Policy Configuration**.
3. Click **TAPResponseOnlyScheme** under **Authentication Schemes**.
4. Click **Open**.
5. Add `MatchLDAPAttribute=uid` to the **Challenge Parameters** field.
6. Click **Apply**.
7. Restart the Administration server and the Access Manager managed servers as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

12.23.8 Managing the Password of the xelsysadm User

After you integrate Oracle Identity Manager with Access Manager, two `xelsysadm` accounts exist. One is the internal account created by Oracle Identity Manager. The other is the account you created in the Identity Store in Section 9.4, "Preparing the Identity Store."

The `xelsysadm` account located in the LDAP store is the one used to access the OIM console. If you want to change the password of this account, change it in LDAP. You can use ODSM to do this. Do not change it through the OIM console.

12.23.9 Validating Integration

To validate integration, you must assign Identity Management administrators to WebLogic security groups and install WebGate as described in Chapter 15, "Configuring Single Sign-on for Administration Consoles in an Enterprise Deployment."

To validate that the wiring of Access Manager with Oracle Identity Manager 11g was successful, attempt to log in to the Oracle Identity Manager Self Service Console, as follows:

1. Using a browser, navigate to:

   `https://SSO.mycompany.com/identity`

   This redirects you to the OAM11*g* single sign-on page.
2. Log in using the `xelsysadm` user account created in Section 9.4, "Preparing the Identity Store."
3. If you see the OIM Self Service Console Page, the integration was successful.

You can perform additional validation as follows:

1. Log in to the OIM Console as the `xelsysadm` user.
2. Create a new user.
3. Log out as the `xelsysadm` user.
4. Log in as the new user you just created. As the new user, you are redirected to the Password Management page.
5. Enter the credentials and click **Submit**. If integration has been performed correctly, you arrive at the page you are trying to access.

13

Setting Up Node Manager for an Enterprise Deployment

This chapter describes how to configure Node Manager in accordance with Oracle best practice recommendations.

This chapter contains the following sections:

- Section 13.1, "Overview of the Node Manager"
- Section 13.2, "Changing the Location of the Node Manager Log"
- Section 13.3, "Enabling Host Name Verification Certificates for Node Manager"
- Section 13.4, "Starting Node Manager"

13.1 Overview of the Node Manager

Node Manager enables you to start and stop the Administration Server and the Managed Servers.

Process

The procedures described in this chapter must be performed on IDMHOST1 and IDMHOST2 for various components of the enterprise deployment topologies outlined in Chapter 2, "Introduction and Planning."

Note that the procedures in this chapter must be performed multiple times for each VIP-and-IP pair using the information provided in the component-specific chapters.

Recommendations

Oracle provides two main recommendations for Node Manager configuration in enterprise deployment topologies:

1. Oracle recommends placing the Node Manager log file in a location different from the default one (which is inside the Middleware Home where Node Manager resides). See Section 13.2, "Changing the Location of the Node Manager Log" for further details.
2. Oracle also recommends using host name verification for the communication between Node Manager and the servers in the domain. This requires the use of certificates for the different addresses used in the domain. This chapter explains the steps for configuring certificates in the hosts for host name verification. See Section 13.3, "Enabling Host Name Verification Certificates for Node Manager" for further details.

Note: The passwords used in this guide are used only as examples. Use secure passwords in a production environment. For example, use passwords that consist of random sequences of both uppercase and lowercase characters as well as numbers.

13.2 Changing the Location of the Node Manager Log

Edit the Node Manager properties file located at *IAM_MW_HOME*/wlserver_10.3/common/nodemanager/nodemanager.properties. Add the new location for the log file using the following line:

```
LogFile=ORACLE_BASE/config/nodemanager.log
```

Oracle best practice is to use a location outside the *MW_HOME* directory and inside the administration directory.

Restart Node Manager, as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components," for the change to take effect.

13.3 Enabling Host Name Verification Certificates for Node Manager

This section describes how to set up host name verification certificates for communication between Node Manager and the Administration Server. It consists of the following steps:

- Section 13.3.1, "Generating Self-Signed Certificates Using the utils.CertGen Utility"
- Section 13.3.2, "Creating an Identity Keystore Using the utils.ImportPrivateKey Utility"
- Section 13.3.3, "Creating a Trust Keystore Using the Keytool Utility"
- Section 13.3.4, "Configuring Node Manager to Use the Custom Keystores"
- Section 13.3.5, "Using a Common or Shared Storage Installation"
- Section 13.3.6, "Configuring Managed WebLogic Servers to Use the Custom Keystores"
- Section 13.3.7, "Changing the Host Name Verification Setting for the Managed Servers"

13.3.1 Generating Self-Signed Certificates Using the utils.CertGen Utility

The certificates added in this chapter (as an example) address a configuration where Node Manager listens on a physical host name (*HOST*.mycompany.com) and a WebLogic Managed Server listens on a virtual host name (*VIP*.mycompany.com). Whenever a server is using a virtual host name, it is implied that the server can be migrated from one node to another. Consequently, the directory where keystores and trust keystores are maintained ideally must reside on a shared storage that is accessible from the failover. If additional host names are used in the same or different nodes, the steps in this example must be extended to:

1. Add the required host names to the certificate stores (if they are different from *HOST*.mycompany.com and *VIP*.mycompany.com).
2. Change the identity and trust store location information for Node Manager (if the additional host names are used by Node Manager) or for the servers (if the additional host names are used by Managed Servers).

Follow these steps to create self-signed certificates on *HOST*. These certificates should be created using the network name or alias. For information on using trust CA certificates instead, see "Configuring Identity and Trust" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*. The following examples configure certificates for *HOST*.mycompany.com and *VIP*.mycompany.com; that is, it is assumed that both a physical host name (*HOST*) and a virtual host name (*VIP*) are used in *HOST*. It is also assumed that *HOST*.mycompany.com is the address used by Node Manager and *VIP*.mycompany.com is the address used by a Managed Server or the Administration Server. This is the common situation for nodes hosting an Administration Server and a Fusion Middleware component, or for nodes where two Managed Servers coexist with one server listening on the physical host name and one server using a virtual host name (which is the case for servers that use migration servers).

1. Set up your environment by running the *WL_HOME*/server/bin/setWLSEnv.sh script. In the Bourne shell, run the following commands:

   ```
   cd WL_HOME/server/bin
   . ./setWLSEnv.sh
   ```

   Verify that the *CLASSPATH* environment variable is set:

   ```
   echo $CLASSPATH
   ```

2. Create a user-defined directory for the certificates. For example, create a directory called 'certs' under the *ASERVER_HOME* directory. Note that certificates can be shared across WebLogic domains.

   ```
   cd ASERVER_HOME
   mkdir certs
   ```

   Note: The directory where keystores and trust keystores are maintained must be on shared storage that is accessible from all nodes so that when the servers fail over (manually or with server migration), the appropriate certificates can be accessed from the failover node. Oracle recommends using central or shared stores for the certificates used for different purposes (like SSL set up for HTTP invocations, for example).

3. Change directory to the directory that you just created:

   ```
   cd certs
   ```

4. Run the `utils.CertGen` tool from the user-defined directory to create the certificates for both *HOST*.mycompany.com and *VIP*.mycompany.com.

   Syntax (all on a single line):

   ```
   java utils.CertGen Key_Passphrase Cert_File_Name Key_File_Name
   [export | domestic] [Host_Name]
   ```

   Examples:

   ```
   java utils.CertGen Key_Passphrase IDMHOST1.mycompany.com_cert
   IDMHOST1.mycompany.com_key domestic IDMHOST1.mycompany.com

   java utils.CertGen Key_Passphrase IDMHOST2.mycompany.com_cert
   IDMHOST2.mycompany.com_key domestic IDMHOST2.mycompany.com

   java utils.CertGen Key_Passphrase ADMINVHN.mycompany.com_cert
   ADMINVHN.mycompany.com_key domestic ADMINVHN.mycompany.com
   ```

13.3.2 Creating an Identity Keystore Using the utils.ImportPrivateKey Utility

Follow these steps to create an identity keystore on IDMHOST1:

1. Create a new identity keystore called `appIdentityKeyStore` using the `utils.ImportPrivateKey` utility. Create this keystore under the same directory as the certificates, that is, *ASERVER_HOME*`/certs`.

   Note: The Identity Store is created (if none exists) when you import a certificate and the corresponding key into the Identity Store using the `utils.ImportPrivateKey` utility.

2. Import the certificate and private key for `IDMHOST1.mycompany.com`, `IDMHOST2.mycompany.com` and `ADMINVHN.mycompany.com` into the Identity Store. Ensure that you use a different alias for each of the certificate/key pairs imported.

   Syntax (all on a single line):

   ```
   java utils.ImportPrivateKey Keystore_File Keystore_Password
   Certificate_Alias_to_Use Private_Key_Passphrase
   Certificate_File
   Private_Key_File
   [Keystore_Type]
   ```

   Examples:

   ```
   java utils.ImportPrivateKey appIdentityKeyStore.jks Key_Passphrase
   appIdentityIDMHOST1 Key_Passphrase ASERVER_HOME/certs/IDMHOST1.mycompany.com_
   cert.pem ASERVER_HOME/certs/IDMHOST1.mycompany.com_key.pem

   java utils.ImportPrivateKey appIdentityKeyStore.jks Key_Passphrase
   appIdentityIDMHOST2 Key_Passphrase ASERVER_HOME/certs/IDMHOST2.mycompany.com_
   cert.pem ASERVER_HOME/certs/IDMHOST2.mycompany.com_key.pem

   java utils.ImportPrivateKey appIdentityKeyStore.jks Key_Passphrase
   appIdentityADMINVHN Key_Passphrase ASERVER_HOME/certs/ADMINVHN.mycompany.com_
   cert.pem ASERVER_HOME/certs/ADMINVHN.mycompany.com_key.pem
   ```

13.3.3 Creating a Trust Keystore Using the Keytool Utility

Follow these steps to create the trust keystore on each host, IDMHOST1 and IDMHOST2:

1. Copy the standard Java keystore to create the new trust keystore since it already contains most of the root CA certificates needed. Oracle does not recommend modifying the standard Java trust keystore directly. Copy the standard Java keystore CA certificates located under the *WL_HOME*/server/lib directory to the same directory as the certificates. For example:

   ```
   cp WL_HOME/server/lib/cacerts ASERVER_HOME/certs/appTrustKeyStoreIDMHOST1.jks
   ```

2. The default password for the standard Java keystore is `changeit`. Oracle recommends always changing the default password. Use the `keytool` utility to do this. The syntax is:

   ```
   keytool -storepasswd -new New_Password -keystore Trust_Keystore -storepass
   Original_Password
   ```

For example:

```
keytool -storepasswd -new Key_Passphrase -keystore appTrustKeyStoreIDMHOST1.jks
-storepass changeit
```

3. The CA certificate CertGenCA.der is used to sign all certificates generated by the utils.CertGen tool. It is located in the *WL_HOME*/server/lib directory. This CA certificate must be imported into the appTrustKeyStore using the keytool utility. The syntax is:

```
keytool -import -v -noprompt -trustcacerts -alias Alias_Name
-file CA_File_Location -keystore Keystore_Location -storepass Keystore_Password
```

For example:

```
keytool -import -v -noprompt -trustcacerts -alias clientCACert -file WL_
HOME/server/lib/CertGenCA.der -keystore appTrustKeyStoreIDMHOST1.jks -storepass
Key_Passphrase
```

13.3.4 Configuring Node Manager to Use the Custom Keystores

To configure Node Manager to use the custom keystores, add the following lines to the end of the nodemanager.properties file located in the *WL_HOME*/common/nodemanager directory:

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=Identity_Keystore
CustomIdentityKeyStorePassPhrase=Identity_Keystore_Password
CustomIdentityAlias=Identity_Keystore_Alias
CustomIdentityPrivateKeyPassPhrase=Private_Key_Used_When_Creating_Certificate
```

For example:

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=ASERVER_HOME/certs/appIdentityKeyStore.jks
CustomIdentityKeyStorePassPhrase=Key_Passphrase
CustomIdentityAlias=appIdentityIDMHOST1
CustomIdentityPrivateKeyPassPhrase=Key_Passphrase
```

The passphrase entries in the nodemanager.properties file get encrypted when you start Node Manager as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components." For security reasons, minimize the time the entries in the nodemanager.properties file are left unencrypted. After you edit the file, start Node Manager as soon as possible so that the entries get encrypted.

13.3.5 Using a Common or Shared Storage Installation

When using a common or shared storage installation for MW_HOME, Node Manager is started from different nodes using the same base configuration (nodemanager.properties). Add the certificate for all the nodes that share the binaries to the appIdentityKeyStore.jks identity store.by creating the certificate for the new node and import it to appIdentityKeyStore.jks, as described in Section 13.3.1, "Generating Self-Signed Certificates Using the utils.CertGen Utility.". Once the certificates are available in the store, each node manager must point to a different identity alias to send the correct certificate to the Administration Server.

To set different environment variables before starting Node Manager in the different nodes:

```
cd WL_HOME/server/bin
```

```
export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentityIDMHOST1

cd WL_HOME/server/bin
export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentityIDMHOST2
```

Note: Make sure to specify the custom identity alias specifically assigned to each host, for example `appIdentity1` for ...HOST1 and `appIdentity2` for ...HOST2.

13.3.6 Configuring Managed WebLogic Servers to Use the Custom Keystores

Follow these steps to configure the identity and trust keystores for *WLS_SERVER*:

1. Log in to Oracle WebLogic Server Administration Console at the URL listed in Section 17.2, "About Identity Management Console URLs."
2. Click **Lock and Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Click the name of the server for which you want to configure the identity and trust keystores (*WLS_SERVER*). The settings page for the selected server is displayed.
6. Select **Configuration**, then **Keystores**.
7. In the Keystores field, select the **Custom Identity and Custom Trust** method for storing and managing private keys/digital certificate pairs and trusted CA certificates.
8. In the Identity section, define attributes for the identity keystore:
   - **Custom Identity Keystore:** The fully qualified path to the identity keystore:

     ASERVER_HOME/certs/appIdentityKeyStore.jks

   - **Custom Identity Keystore Type:** Leave blank; it defaults to JKS.
   - **Custom Identity Keystore Passphrase:** The password (*Keystore_Password*) you provided in Section 13.3.3, "Creating a Trust Keystore Using the Keytool Utility." This attribute is optional or required depending on the type of keystore. All keystores require the passphrase to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. WebLogic Server only reads from the keystore, so whether you define this property depends on the requirements of the keystore.
9. In the Trust section, define properties for the trust keystore:
   - **Custom Trust Keystore:** The fully qualified path to the trust keystore:

     ASERVER_HOME/certs/appTrustKeyStoreIDMHOST1.jks

   - **Custom Trust Keystore Type:** Leave blank; it defaults to JKS.
   - **Custom Trust Keystore Passphrase:** The password you provided as *New_Password* in Section 13.3.3, "Creating a Trust Keystore Using the Keytool Utility." This attribute is optional or required depending on the type of keystore. All keystores require the passphrase to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. WebLogic Server only reads from the keystore, so whether you define this property depends on the requirements of the keystore.

10. Click **Save**.
11. Click **Activate Changes** in the Administration Console's Change Center to make the changes take effect.
12. Select **Configuration**, then **SSL**.
13. Click **Lock and Edit**.
14. In the **Private Key Alias** field, enter the alias you used for the host name the Managed Server listens on, for example:
    - For `WLS_OAM1`, use `appIdentityIDMHOST1`.
    - For `WLS_OAM2` use `appIdentityIDMHOST2`.
    - For `ADMINSERVER` user `appIdentityADMINVHN`.

    In the **Private Key Passphrase** and the **Confirm Private Key Passphrase** fields, enter the password for the keystore that you created in Section 13.3.2, "Creating an Identity Keystore Using the utils.ImportPrivateKey Utility."
15. Click **Save**.
16. Click **Activate Changes** in the Administration Console's Change Center to make the changes take effect.
17. Restart the server for which the changes have been applied, as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

13.3.7 Changing the Host Name Verification Setting for the Managed Servers

Once the previous steps have been performed, set host name verification for the affected Managed Servers to `BEA Hostname Verifier`. To do this, perform the following steps:

1. Log in to Oracle WebLogic Server Administration Console.
2. Select **Lock and Edit** from the change center.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Select the Managed Server in the Names column of the table. The settings page for the server is displayed.
6. Open the SSL tab.
7. Expand the **Advanced** section of the page.
8. Set host name verification to `BEA Hostname Verifier`.
9. Click **Save**.
10. Click **Activate Changes**.

13.4 Starting Node Manager

Run the following commands to start Node Manager.

```
cd WL_HOME/server/bin
./startNodeManager.sh
```

Notes:

- If you have not configured and started Node Manager for the first time yet, run the `setNMProps.sh` script as specified in section Section 8.5.4, "Starting Node Manager." This enables the use of the start script that is required for Identity Management Components.
- Verify that Node Manager is using the appropriate stores and alias from the Node Manager output. You should see the following when Node Manager starts.:

```
<Loading identity key store:
  FileName=ASERVER_HOME/certs/appIdentityKeyStore.jks,
Type=jks, PassPhraseUsed=true>
```

  Host name verification works if you apply a test configuration change to the servers and it succeeds without Node Manager reporting any SSL errors.

14

Configuring Server Migration for an Enterprise Deployment

Configuring server migration allows SOA-managed and OIM-managed servers to be migrated from one host to another, so that if a node hosting one of the servers fails, the service can continue on another node. This chapter describes how to configure server migration for an Identity Management enterprise deployment.

This chapter contains the following steps:

14.1 Overview of Server Migration for an Enterprise Deployment

Configure server migration for the WLS_OIM1, WLS_SOA1, WLS_OIM2, and WLS_SOA2 Managed Servers. The WLS_OIM1 and WLS_SOA1 Managed Server are configured to restart on IDMHOST2 should a failure occur. The WLS_OIM2 and WLS_SOA2 Managed Servers are configured to restart on IDMHOST1 should a failure occur. The WLS_OIM1, WLS_SOA1, WLS_OIM2 and WLS_SOA2 servers listen on specific floating IPs that are failed over by WebLogic Server Migration.

Perform the steps in the following sections configure server migration for the WLS_OIM1, WLS_SOA1, WLS_OIM2, and WLS_SOA2 Managed Servers.

14.2 Setting Up a User and Tablespace for the Server Migration Leasing Table

In this section, you set up a user and tablespace for the server migration leasing table:

> **Note:** If other servers in the same domain have already been configured with server migration, the same tablespace and data sources can be used. In that case, the data sources and multi data source for database leasing do not need to be re-created, but they must be retargeted to the clusters being configured with server migration.

1. Create a tablespace called `leasing`. For example, log on to SQL*Plus as the sysdba user and run the following command:

```
create tablespace leasing
logging datafile 'DB_HOME/oradata/orcl/leasing.dbf'
size 32m autoextend on next 32m maxsize 2048m extent management local;
```

2. Create a user named `leasing` and assign to it the `leasing` tablespace:

```
create user leasing identified by password;
grant create table to leasing;
grant create session to leasing;
alter user leasing default tablespace leasing;
alter user leasing quota unlimited on leasing;
```

3. Create the `leasing` table using the `leasing.ddl` script:
   a. Copy the `leasing.ddl` file located in either of the following directories to your database node:

```
WL_HOME/server/db/oracle/817
WL_HOME/server/db/oracle/920
```

   b. Connect to the database as the `leasing` user.
   c. Run the leasing.ddl script in SQL*Plus:

```
@Copy_Location/leasing.ddl;
```

   d. Currently, the script does not commit the change. Enter the following, at the SQL*Plus prompt, after the tool completes:

```
commit;
```

14.3 Creating a GridLink Data Source for Leasing Using the Oracle WebLogic Administration Console

In this section, you create a GridLink data source for the leasing table from the Oracle WebLogic Server Administration Console.

To create a GridLink data source:

1. Log in to the Oracle WebLogic Server Administration Console at the URL listed in Section 17.2, "About Identity Management Console URLs."
2. If you have not already done so, in the **Change Center**, click **Lock & Edit**.
3. In the **Domain Structure** tree, expand **Services**, then select **Data Sources**.
4. On the Summary of Data Sources page, click **New** and select **GridLink Data Source**, and enter the following:
   - **Name**: Enter a logical name for the data source. For example, `leasing`.
   - **JNDI**: Enter a name for JNDI. For example, `jdbc/leasing`.

- **Database Driver**: Select **For the Database Driver, select Oracle's Driver (Thin) for GridLink Connections Versions: 11 and later**.
- Click **Next**.

5. In the Transaction Options page, de-select **Supports Global Transactions**, and click **Next**.
6. In the GridLink Data Source Connection Properties Options screen, select **Enter individual listener information** and click **Next**.
7. Enter the following connection properties:
   - **Service Name**: Enter the service name of the database with lowercase characters. For a GridLink data source, you must enter the Oracle RAC service name. For example: `OIMEDG.mycompany.com`
   - **Host Name and Port**: Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the appropriate parameter in the database using the TCP Protocol:

     ```
     show parameter remote_listener;

     NAME                 TYPE        VALUE

     ------------------------------------------------

     remote_listener      string      DB-SCAN.mycompany.com:1521
     ```

     Note: For Oracle Database 11g Release 1 (11.1), use the virtual IP and port of each database instance listener, for example: `DBHOST1-VIP.mycompany.com` (port `1521`) and `DBHOST2-VIP.mycompany.com` (port 1521), where 1521 is *`DB_LSNR_PORT`*

     For Oracle Database 10*g*, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources see Appendix A, "Using Multi Data Sources with Oracle RAC."

   - **Database User Name**: leasing
   - **Password**: For example: welcome1
   - **Confirm Password**: Enter the password again and click **Next**.
8. On the Test GridLink Database Connection page, review the connection parameters and click **Test All Listeners**. Here is an example of a successful connection notification:

   ```
   Connection test for jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_
   LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=DB-SCAN.mycompany.com)
   (PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=OIMEDG.mycompany.com))) succeeded.
   ```

   where port 1521 is *DB_LSNR_PORT*.

   Click **Next**.
9. In the ONS Client Configuration page, do the following:
   - Select **FAN Enabled** to subscribe to and process Oracle FAN events.

- Enter here also the SCAN address for the RAC database and the ONS remote port as reported by the database (example below) and click **ADD**:

  ```
  srvctl config nodeapps -s

  ONS exists: Local port 6100, remote port 6200, EM port 2016
  ```

- Click **Next**.

  Note: For Oracle Database 11g Release 1 (11.1), use the hostname and port of each database's ONS service, for example:

  ```
  DBHOST1.mycompany.com (port 6200)
  ```

  and

  ```
  DBHOST2.mycompany.com (port 6200)
  ```

  ---

10. On the Test ONS Client Configuration page, review the connection parameters and click **Test All ONS Nodes**.

    Here is an example of a successful connection notification:

    ```
    Connection test for DB-SCAN.mycompany.com:6200 succeeded.
    ```

    Click **Next**.

11. In the Select Targets page, select **oim_cluster** and **soa_cluster** as the targets, and **All Servers in the cluster**.
12. Click **Finish**.
13. Click **Activate Changes**.

14.4 Editing Node Manager's Properties File

In this section, you edit Node Manager's properties file. This must be done for the Node Managers on the nodes where the servers are running, IDMHOST1 and IDMHOST2.

The `nodemanager.properties` file is located in the following directory:

WL_HOME`/common/nodemanager`

Add the following properties to enable server migration to work properly:

- `Interface:`

  `Interface=eth0`

  This property specifies the interface name for the floating IP (for example, eth0).

  Note: Do not specify the sub-interface, such as `eth0:1` or `eth0:2`. This interface is to be used without `:0` or `:1`. Node Manager's scripts traverse the different:*X*-enabled IPs to determine which to add or remove. For example, the valid values in Linux environments are `eth0`, `eth1`, `eth2`, `eth3`, `eth`*n*, depending on the number of interfaces configured.

- `NetMask:`

  ```
  NetMask=255.255.255.0
  ```

  This property specifies the net mask for the interface for the floating IP. The net mask should the same as the net mask on the interface.

- `UseMACBroadcast:`

  ```
  UseMACBroadcast=true
  ```

  This property specifies whether to use a node's MAC address when sending ARP packets, that is, whether to use the `-b` flag in the `arping` command.

Verify in Node Manager's output (shell where Node Manager is started) that these properties are being used, or problems may arise during migration. You should see something like this in Node Manager's output:

```
StateCheckInterval=500
eth0=*,NetMask=255.255.255.0
UseMACBroadcast=true
```

> **Note:** The following steps are not required if the server properties (start properties) have been properly set and Node Manager can start the servers remotely.

1. If not done already, set the `StartScriptEnabled` property in the `nodemanager.properties` file to true. This is required to enable Node Manager to start the managed servers.
2. Start Node Manager on IDMHOST1 and IDMHOST2 by running the `startNodeManager.sh` script, which is located in the *`WL_HOME`*/server/bin directory.

> **Note:** When running Node Manager from a shared storage installation, multiple nodes are started using the same `nodemanager.properties` file. However, each node may require different NetMask or `Interface` properties. In this case, specify individual parameters on a per-node basis using environment variables. For example, to use a different interface (`eth3`) in HOST*n*, use the `Interface` environment variable as follows:
>
> ```
> export JAVA_OPTIONS=-DInterface=eth3
> ```
>
> and start Node Manager after the variable has been set in the shell.

14.5 Setting Environment and Superuser Privileges for the wlsifconfig.sh Script

On Linux, you set environment and superuser privileges for the `wlsifconfig.sh` script:

Ensure that your PATH environment variable includes the files listed inTable 14–1.

Table 14–1 Files Required for the PATH Environment Variable

File	Located in this directory
wlsifconfig.sh	*MSERVER_HOME*/bin/server_migration
wlscontrol.sh	*WL_HOME*/common/bin
nodemanager.domains	*WL_HOME*/common/nodemanager

Grant sudo privilege to the WebLogic user ('oracle') with no password restriction, and grant execute privilege on the /sbin/ifconfig and /sbin/arping binaries.

For security reasons, sudo should be restricted to the subset of commands required to run the wlsifconfig.sh script. For example, perform the following steps to set the environment and superuser privileges for the wlsifconfig.sh script.

Note: Ask the system administrator for the appropriate sudo and system rights to perform this step.

Make sure the script is executable by the WebLogic user ('oracle'). The following is an example of an entry inside /etc/sudoers granting sudo execution privilege for oracle and also over ifconfig and arping.

To grant sudo privilege to the WebLogic user ('oracle') with no password restriction, and grant execute privilege on the /sbin/ifconfig and /sbin/arping binaries:

```
Defaults:oracle !requiretty
oracle ALL=NOPASSWD: /sbin/ifconfig,/sbin/arping
```

14.6 Configuring Server Migration Targets

In this section, you configure server migration targets. Configuring Cluster Migration sets the DataSourceForAutomaticMigration property to true.

To configure migration in a cluster:

1. Log in to the Oracle WebLogic Server Administration Console at the URL listed in Section 17.2, "About Identity Management Console URLs."
2. In the Domain Structure window, expand **Environment** and select **Clusters**. The Summary of Clusters page is displayed.
3. Click the cluster for which you want to configure migration (**oim_cluster**) in the Name column of the table.
4. Click the **Migration** tab.
5. Click **Lock and Edit**.
6. In the **Available** field, select the machines to which to allow migration, **IDMHOST1** and **IDMHOST2**, and click the right arrow.
7. Select the data source to be used for automatic migration. In this case, select the leasing data source.
8. Click **Save**.
9. In the Domain Structure window of the Oracle WebLogic Server Administration Console, expand **Environment** and select **Servers**.
10. Select the server for which you want to configure migration.

11. Click the **Migration** tab.
12. Select **Automatic Server Migration Enabled** and click **Save**.
13. Click **Activate Changes**.
14. Repeat steps 2 through 13 for the SOA cluster.
15. Restart WebLogic Administration Server, Node Managers, and the servers for which server migration has been configured, as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

14.7 Testing the Server Migration

In this section, you test the server migration. Perform these steps to verify that server migration is working properly:

To test from IDMHOST1:

1. Stop the WLS_OIM1 Managed Server. To do this, run this command:

   ```
   kill -9 pid
   ```

   where *pid* specifies the process ID of the Managed Server. You can identify the pid in the node by running this command:

   ```
   ps -ef | grep WLS_OIM1
   ```

2. Watch the Node Manager console. You should see a message indicating that WLS_OIM1's floating IP has been disabled.
3. Wait for Node Manager to try a second restart of WLS_OIM1. It waits for a fence period of 30 seconds before trying this restart.
4. Once Node Manager restarts the server, stop it again. Node Manager should now log a message indicating that the server will not be restarted again locally.

To test from IDMHOST2:

1. Watch the local Node Manager console. After 30 seconds since the last try to restart WLS_OIM1 on IDMHOST1, Node Manager on IDMHOST2 should prompt that the floating IP for WLS_OIM1 is being brought up and that the server is being restarted in this node.
2. Access the OIM Console using the Virtual Host Name, for example: `http://OIMHOST1VHN.mycompany.com:14000/identity`.

Follow the previous steps to test server migration for the WLS_OIM2, WLS_SOA1, and WLS_SOA2 Managed Servers.

Table 14–2 shows the Managed Servers and the hosts they migrate to in case of a failure.

Table 14–2 Managed Server Migration

Managed Server	Migrated From	Migrated To
WLS_OIM1	IDMHOST1	IDMHOST2
WLS_OIM2	IDMHOST2	IDMHOST1
WLS_SOA1	IDMHOST1	IDMHOST2
WLS_SOA2	IDMHOST2	IDMHOST1

Verification From the Administration Console

Migration can also be verified in the Administration Console:

1. Log in to the Administration Console.
2. Click **IDMDomain** on the left pane.
3. Click the **Monitoring** tab and then the **Migration** sub tab.

   The Migration Status table provides information on the status of the migration.

Note: After a server is migrated, to fail it back to its original node/machine, stop the Managed Server from the Oracle WebLogic Administration Console and then start it again. The appropriate Node Manager starts the Managed Server on the machine to which it was originally assigned.

14.8 Backing Up the Server Migration Configuration

Back up the database and the WebLogic domain, as described in Section 17.6.3, "Performing Backups During Installation and Configuration."

15

Configuring Single Sign-on for Administration Consoles in an Enterprise Deployment

This chapter describes how to configure single sign-on (SSO) for administration consoles in an Identity Management Enterprise deployment.

This chapter includes the following topics:

15.1 Overview of Configuring Single Sign-on for Administration Consoles in an Enterprise Deployment

If you have not integrated Oracle Access Management Access Manager with Oracle Identity Manager, you must first create WebLogic Security Providers. Then proceed as follows.

You assign WebLogic Administration groups, update boot.properties, and restart the servers. Then you install and configure WebGate and validate the setup. After WebGate is installed and configured, the Oracle HTTP Server intercepts requests for the consoles and forwards them to Access Manager for validation

The administration consoles referred to in the chapter title are:

- Oracle Enterprise Manager Fusion Middleware Control
- Oracle WebLogic Server Administration Console
- Oracle Access Management Console
- Oracle Identity Manager Console

15.2 Prerequisites

Before you attempt to integrate administration consoles with single sign-on, ensure that the following tasks have been performed in the IDMDomain:

1. Configuring Oracle HTTP Server, as described in Chapter 10, "Installing and Configuring Oracle Web Tier for an Enterprise Deployment."
2. Configuring Access Manager, as described in Chapter 11, "Extending the Domain to Include Oracle Access Management."
3. Provisioning Weblogic Administrators in LDAP as described in Section 9.4, "Preparing the Identity Store."

15.3 Configuring WebLogic Security Providers

When you run `idmConfigTool` with the `configOAM` or `configOIM` option, the tool creates security providers in the domains IDMDomain and OIMDomain. These security providers restrict access to the consoles in those domains based on the security policies of Access Manager. If you have other domains, you must create security providers in those domains manually and then update them as described in the following sections.

Note: Once you have enabled single sign-on for the administration consoles, ensure that at least one OAM Server is running to enable console access.

If you have used the Oracle Weblogic console to shut down all of the Access Manager Managed Servers, then restart one of those Managed Servers manually before using the console again.

To start WLS_OAM1 manually, use the command:

```
MSERVER_HOME/bin/startManagedWeblogic.sh WLS_OAM1
t3://ADMINVHN:7001
```

This section describes how to update the security providers to enable single sign on access to the administration consoles. This section should be performed in all domains, which have security providers including ancillary domains which have had them created manually.

This section contains the following topics:

- Section 15.3.1, "Updating Oracle Unified Directory Authenticator"
- Section 15.3.2, "Reordering the Security Providers"

15.3.1 Updating Oracle Unified Directory Authenticator

When the OUD authenticator is created, it is created with some missing information, which must be added. If you are using OUD as your identity store, you must add this information by performing the following steps.

1. Log in to the WebLogic Administration Console.
2. Click **Security Realms** from the Domain structure menu.
3. Click **Lock and Edit** in the Change Center.
4. Click **myrealm**.

5. Click on **Providers**.
6. Click on **OUDAuthenticator**.
7. Click on **Provider Specific** tab.
8. On the Provider Specific screen update the following values:
   - **All Users Filter**: `(&(uid=*)(objectclass=person))`
   - **User From Name Filter**: `(&(uid=%u)(objectclass=person))`
   - **User Name Attribute**: `uid`
   - **Static Group Object Class**: `groupofuniquenames`
   - **Static Member DN Attribute**: `uniquemember`
   - **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**.
10. Click **Activate Changes**.

15.3.2 Reordering the Security Providers

This section sets up an Access Manager asserter to enable you to delegate responsibility for credential collection to Access Manager.

1. Log in to the WebLogic Administration Console at the URL listed in Section 17.2, "About Identity Management Console URLs."
2. Click **Security Realms** from the Domain structure menu.
3. Click **Lock and Edit** in the **Change Center**.
4. Click **myrealm**.
5. Select the **Providers** tab.
6. Click **Reorder**.
7. Using the arrows on the right hand side order the providers such that the order is:
   - **OAMIDAsserter**
   - **OIM Signature Authenticator**, if present
   - **OIMAuthenticationProvider**, if present
   - **OUD Authenticator**, **OVDAuthenticator** or **OIDAuthenticator**
   - **Default Authenticator**
   - **Default Identity Asserter**

   Note: Oracle Identity Manager providers only exist if Oracle Identity Manager has been configured.

8. Click **OK**.

9. Click **Activate Changes**.
10. Restart WebLogic Administration Server and all the Managed Servers, as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

15.4 Assigning WLSAdmins Group to WebLogic Administration Groups

In an enterprise, it is typical to have a centralized Identity Management domain where all users, groups and roles are provisioned and multiple application domains (such as a SOA domain and WebCenter Portal domain). The application domains are configured to authenticate using the central Identity Management domain.

In Section 9.4, "Preparing the Identity Store" you created a user called `weblogic_idm` and assigned it to the group WLSAdmins. To be able to manage WebLogic using this account you must add the WLSAdmins group to the list of Weblogic Administration groups. This section describes how to add the WLSAdmins Group to the list of WebLogic Administrators.

Perform this step for each domain in the topology.

1. Log in to the WebLogic Administration Server Console at the URL listed in Section 17.2, "About Identity Management Console URLs.".
2. In the left pane of the console, click **Security Realms**.
3. On the Summary of Security Realms page, click **myrealm** under the **Realms** table.
4. On the Settings page for `myrealm`, click the **Roles & Policies** tab.
5. On the Realm Roles page, expand the **Global Roles** entry under the **Roles** table. This brings up the entry for Roles. Click the **Roles** link to go to the Global Roles page.
6. On the Global Roles page, click the **Admin** role to go to the Edit Global Role page:
   a. On the Edit Global Roles page, under the **Role Conditions** table, click the **Add Conditions** button.
   b. On the Choose a Predicate page, select **Group** from the list for predicates and click **Next**.
   c. On the Edit Arguments Page, Specify **WLSAdmins** in the **Group Argument** field and click **Add**.
7. Click **Finish** to return to the Edit Global Rule page.
8. The **Role Conditions** table now shows the WLSAdmins Group as an entry.
9. Click **Save** to finish adding the Admin role to the WLSAdmins Group.
10. Validate that the changes were successful by bringing up the WebLogic Administration Server Console using a web browser. Log in using the credentials for the `weblogic_idm` user.

15.5 Authorize Access Manager Administrators to Access APM Console

By default, only users in the WebLogic administrators group can access the APM console. After SSO is enabled, you will login as an Access Manager Administrator.

To enable this functionality perform the following steps:

1. Log in to the APM console at `http://ADMIN.mycompany.com/apm` as WebLogic administrator.

2. Click the **System Configuration** tab.
3. Click **Add** in the External Role Mapping box.
4. Click **Search**.
5. Select **OAMAdministrators** from the returned search results.
6. Click **Add Selected**.
7. Click **Add Principals**.

15.6 Updating the boot.properties File

Update the `boot.properties` file for the Administration Server with the WebLogic `admin` user created in LDAP.

You must update `boot.properties` on each administration server node. Follow the steps in the following sections to update the file.

This section contains the following topics:

- Section 15.6.1, "Update the Administration Servers on All Domains"
- Section 15.6.2, "Restarting the Servers"

15.6.1 Update the Administration Servers on All Domains

1. On each of the servers in the topology, go the directory:

   ASERVER_HOME/servers/*serverName*/security

   For example:

   cd *ASERVER_HOME*/servers/AdminServer/security

2. Rename the existing `boot.properties` file.
3. Use a text editor to create a file called boot.properties under the security directory. Enter the following lines in the file:

   ```
   username=adminUser
   password=adminUserPassword
   ```

   For example:

   ```
   username=weblogic_idm
   password=Password for weblogic_idm user
   ```

> **Note:** When you start the Administration Server, the username and password entries in the file get encrypted.
>
> For security reasons, minimize the time the entries in the file are left unencrypted. After you edit the file, you should start the server as soon as possible so that the entries get encrypted.

15.6.2 Restarting the Servers

Restart the WebLogic Administration server and all managed servers, as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

15.7 Installing and Configuring WebGate 11*g*

This section describes how to install and configure WebGate.

This section contains the following topics:

- Section 15.7.1, "Prerequisites"
- Section 15.7.2, "Installing Oracle WebGate on WEBHOST1 and WEBHOST2"

15.7.1 Prerequisites

Ensure that the following tasks have been performed before installing the Oracle Web Gate:

1. Install and configure the Oracle Web Tier as described in Chapter 10.
2. Ensure Oracle Access Management Access Manager has been configured as described in Chapter 11.

15.7.2 Installing Oracle WebGate on WEBHOST1 and WEBHOST2

Before starting the installer ensure that Java is installed on your machine.

1. Start the WebGate installer by issuing the command:

   ```
   ./runInstaller
   ```

   You are asked to specify the location of the Java Development Kit for example:

   WEB_MW_HOME`/jrockit_`*version*

2. On the Welcome screen, click **Next**.
3. On the Prerequisites screen, after all the checks have successfully completed, click **Next**.
4. On the Installation Location Screen, enter the following information:
   - **Oracle Middleware Home**: *WEB_MW_HOME*
   - **Oracle Home Directory**: `webgate`

   Click **Next**.
5. On the installation summary screen, click **Install**.
6. Click **Next**.
7. Click **Finish**.

Deploy WebGate to Oracle HTTP, as follows:

1. Execute the command `deployWebGateInstance.sh` which is located in:

   WEBGATE_ORACLE_HOME`/webgate/ohs/tools/deployWebGate`

   The command takes the following arguments:

   Oracle HTTP Instance configuration Directory

   WebGate Home Directory

For example:

```
./deployWebGateInstance.sh -w WEB_ORACLE_INSTANCE/config/OHS/component_name -oh
WEBGATE_ORACLE_HOME
```

2. Set the library path and change directory.

   On Linux systems, set the library path to include the *WEB_ORACLE_HOME*/lib directory, for example:

   ```
   export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:WEB_ORACLE_HOME/lib
   ```

   Change directory:

   Change directory to: *WEBGATE_ORACLE_HOME*/webgate/ohs/tools/setup/InstallTools

3. Run the following command to copy the file apache_webgate.template from the WebGate home directory to the WebGate instance location (renamed to webgate.conf) and update the httpd.conf file to add one line to include the name of webgate.conf.

   ```
   ./EditHttpConf -w WEB_ORACLE_INSTANCE/config/OHS/component_name -oh WEBGATE_
   ORACLE_HOME
   ```

4. Copy the files ObAccessClient.xml, cwallet.sso, and password.xml, which were generated when you created the agent from the directory *ASERVER_HOME*/output/Webgate_IDM_11g on IDMHOST1, to the directory *WEB_ORACLE_INSTANCE*/config/OHS/*component_name*/webgate/config

5. The files aaa_key.pem and aaa_cert.pem were generated when you created the agent from the directory *ASERVER_HOME*/output/Webgate_IDM_11g on IDMHOST1. Copy the files aaa_key.pem and aaa_cert.pem to the WebGate instance directory *WEB_ORACLE_INSTANCE*/config/OHS/*component_name*/webgate/config/simple.

6. Restart the Oracle HTTP Server as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

15.8 Validating WebGate and the Access Manager Single Sign-On Setup

To validate that WebGate is functioning correctly, open a web browser and go the OAM console URL listed in Section 17.2, "About Identity Management Console URLs."

You now see the Oracle Access Management Login page displayed. Enter your OAM administrator user name (for example, oamadmin) and password and click **Login**. Then you see theOracle Access Management console displayed.

To validate the single sign-on setup, open a web browser and go the WebLogic Administration Console and to Oracle Enterprise Manager Fusion Middleware Control at the URLs listed in Section 17.2, "About Identity Management Console URLs."

The Oracle Access Management Single Sign-On page displays. Provide the credentials for the weblogic_idm user to log in.

15.9 Backing Up Single Sign-on

Back up the Web Tier and WebLogic domain, as described in Section 17.6.3, "Performing Backups During Installation and Configuration."

16

Creating a Split Domain Topology

This chapter describe additional procedures and modifications to procedures that are necessary to deploy a split domain topology.

This chapter contains the following topics:

16.1 Introduction to Split Domain Topology

As described in Section 2.2.2.3, "Architecture Notes," in a split domain configuration, Oracle Identity Manager is installed into a separate domain from other components. The decision to use the split domain topology requires several modifications and additions to the procedures in this Guide. This chapter describes these changes and additions.

16.2 Additional Network Requirements

In addition to preparing the network as described in Chapter 3, perform the following additional preparations for split domain implementations.

This section contains the following topics:

- Section 16.2.4, "Configuring Servers to Listen on Virtual and Physical IP Addresses"
- Section 16.2.5, "Firewalls and Ports"

16.2.1 Virtual Server Names

In addition to the virtual server names listed in Section 3.3, "About Virtual Server Names Used by the Topologies," you need the following additional virtual server for the split domain topology:

OIMADMIN.mycompany.com

- This virtual server is only required when a split domain topology is being used.
- This virtual server is enabled on LBR1. It acts as the access point for all internal HTTP traffic that gets directed to the administration services in the OIM Domain. The incoming traffic from clients is non-SSL enabled. Thus, the clients access this service using the address `OIMADMIN.mycompany.com:80` (*HTTP_PORT*) and in turn forward these to ports *7777* (*OHS_PORT*) on WEBHOST1 and WEBHOST2. The services accessed on this virtual host include the WebLogic Administration Server Console, and Oracle Enterprise Manager Fusion Middleware Control.
- 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 `OIMADMIN.mycompany.com` virtual host.

16.2.2 Load Balancer Configuration

In Section 3.4.3, "Load Balancer Configuration," Table 3–1 describes the configuration of your load balancer. For a split domain topology, configure the following additional virtual host:

Table 16–1 Additional Load Balancer Configuration for Split Domain

Virtual Host	Server Pool	Protocol	SSL Termination	External
`OIMADMIN.mycompany.com:80` (*HTTP_PORT*)	`WEBHOST1.mycompany.com:7777` `WEBHOST2.mycompany.com:7777`	HTTP	No	No

16.2.3 Virtual IP Addresses

Section 3.5, "About IP Addresses and Virtual IP Addresses," lists Virtual IP addresses required by Oracle Identity Management. The following additional Virtual IP address is required for a split domain topology:

OIMADMINVHN.mycompany.com

You need this virtual IP address when you are using a split domain topology. It serves a similar function to `ADMINVHN.mycompany.com`. This virtual IP address fails over along with the Administration Server from IDMHOST1 to IDMHOST2, or vice versa.

16.2.4 Configuring Servers to Listen on Virtual and Physical IP Addresses

Section 3.5, "About IP Addresses and Virtual IP Addresses," also describes how to configure the administration server and the managed servers to listen on different virtual IPs and physical IPs. This illustrated in Figure 3–1. The following figure is a modified version of Figure 3–1, listing all the IP addresses and VIP Addresses, including those for OIMDomain.

Figure 16–1 IP Addresses and VIP Addresses

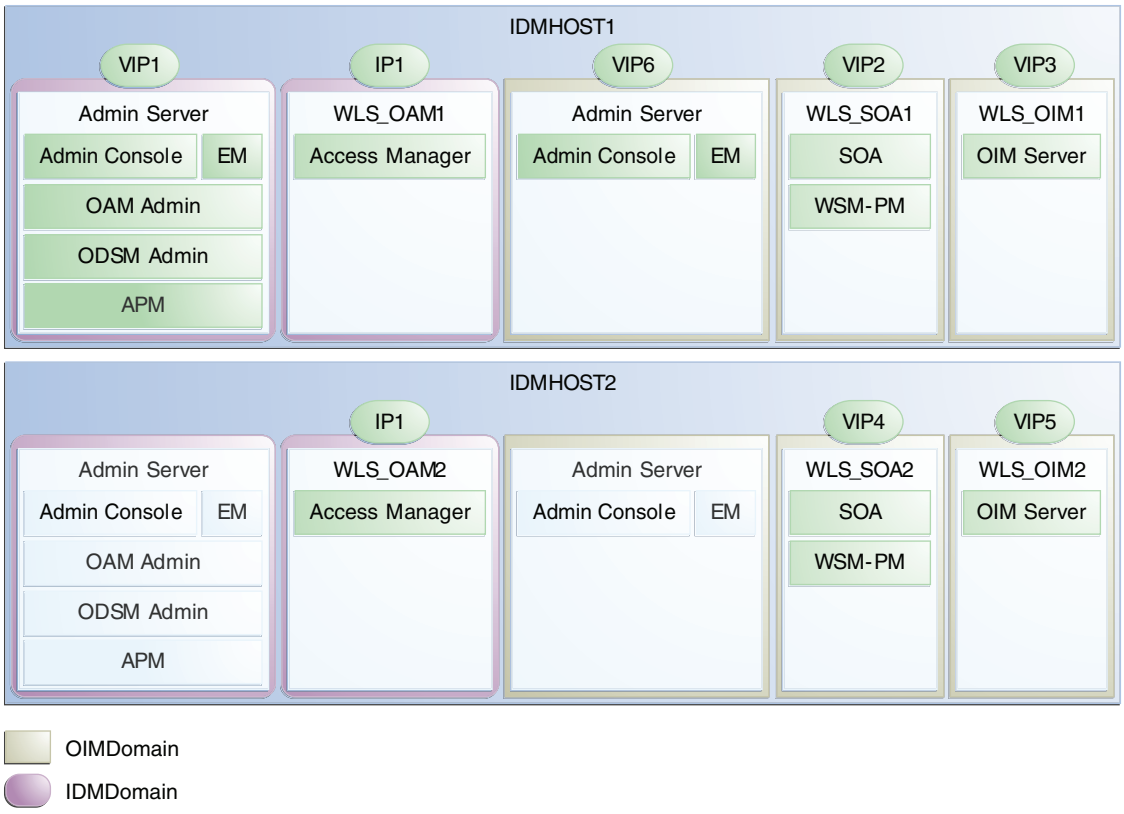


Table 3–2 provides a mapping of VIP addresses to Virtual hosts. All of the VIP addresses in that table are required. For a split domain topology, the VIP address VIP6, which maps to OIMADMINVHN, must be enabled as shown:

Table 16–2 VIP Addresses and Virtual Hosts

Virtual IP	VIP Maps to...	Description
VIP6	OIMADMINVHN	OIMADMINVHN is the virtual host name that is the listen address for the Oracle Identity Manager Administration Server. It fails over with manual failover of the Administration Server. It is enabled on the node where the Oracle Identity Manager Administration Server process is running (IDMHOST1 by default).

16.2.5 Firewalls and Ports

Section 3.6, "About Firewalls and Ports," lists the firewall ports used by the topology in Table 3–3. The following additional port is required for Oracle WebLogic Administration Server access from the Web Tier.

Table 16–3 Ports Used in the Oracle Identity Management Enterprise Deployment topologies

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Timeout
Oracle WebLogic Administration Server access from web tier	FW1	7101 (*SPLIT_WLS_ADMIN_PORT*)	HTTP / Oracle HTTP Server and Administration Server	Inbound	N/A

16.3 Additional Requirements for Preparing the File System

Section 4.4, "About Recommended Locations for the Different Directories" describes directory usage. Among other things, it recommends that you have more than one Middleware home (*MW_HOME*).

When you are implementing a split domain topology, you need yet another Middleware home, a separate *MW_HOME* for the second domain. This facilitates independent patching.

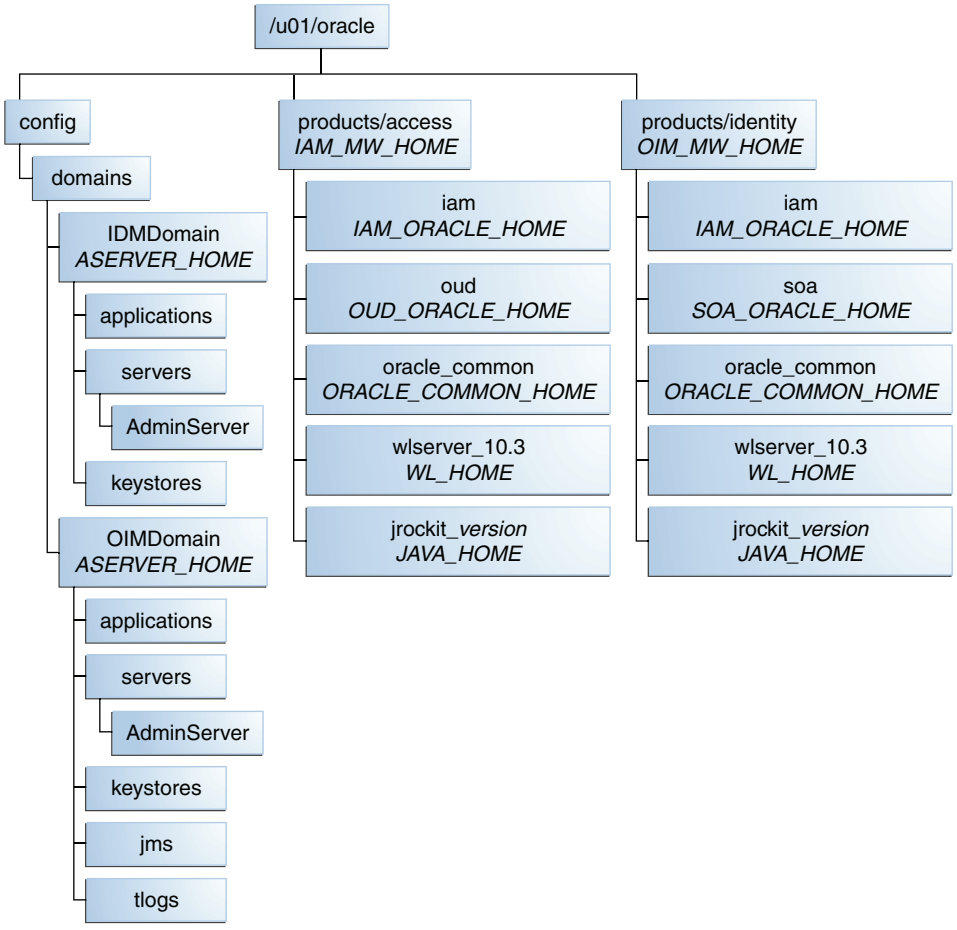
The details for shared and local storage must be modified as follows.

- Section 4.4.4.1, "Shared Storage" discusses the volumes that should be kept on shared storage. When you are using a split domain topology, there are additional volumes to be kept on shared storage. The recommended layout is described in Table 16–4 and shown in Figure 16–2.

Table 16–4 Volumes on Shared Storage

Environment Variable	Volume	Mount Point	Mounted on Hosts	Exclusive
OIM_MW_HOME	VOL1/OIM_MW_HOME	`/u01/oracle/products/identity`	IDMHOST1 IDMHOST2	No
ASERVER_HOME	VOL1/ADMIN2	`/u01/oracle/config/domains/OIMDomain`	IDMHOST1 IDMHOST2	Yes

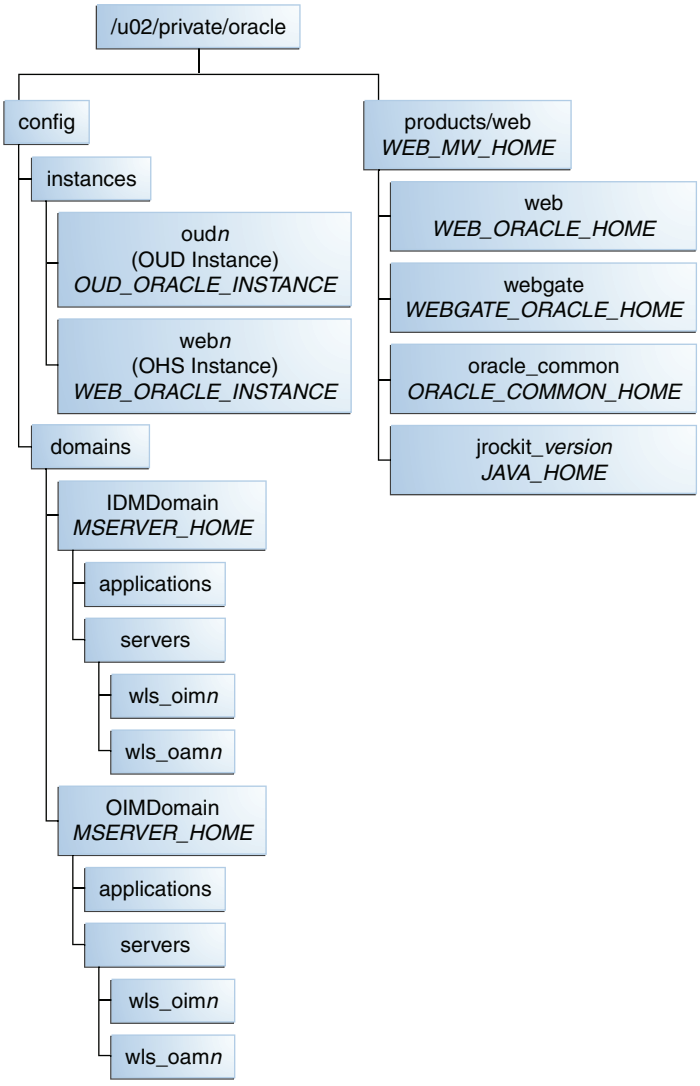
Figure 16–2 Shared Storage for Split Domain Topology



- Section 4.4.4.2, "Local Storage" lists the directories to be created on local storage.

  The recommended layout for local storage is described in Table 4–1. The split domain differs in that it includes an additional MSERVER_HOME in OIMDomain, as shown in Figure 16–3.

Figure 16–3 Local Storage



16.4 Additional Requirement for Preparing the Servers

Table 5–2 in Chapter 5, "Preparing the Servers for an Enterprise Deployment," lists additional virtual hosts that you must enable. For a split domain topology, you must also enable virtual host OIMADMINVHN.mycompany.com as shown:

Table 16–5 Virtual Hosts

VIP	Enabled on Host
OIMADMINVHN.mycompany.com	IDMHOST1

16.5 Requirements for Creating the Additional Domain

In a split domain topology, you must create two domains, one for Access Manager and the other for Oracle Identity Manager.

> **Note:** There is no need to create a separate node manager for this domain. Only one Node Manager is deployed per host, and that is used to manage WebLogic components in both domains.

Perform the instructions in Section 8.4, "Running the Configuration Wizard to Create a Domain" to create a domain for Access Manager called IDMDomain. Then repeat the instructions with the following modifications to create a second domain called OIMDomain to host the Oracle Identity Manager components.

- In Step 5, on the Select Domain Source screen, select only the following products:
  - **Oracle Enterprise Manager [oracle_common]**
  - **Oracle Platform Security Service [iam]**
  - **Oracle JRF [oracle_common]**
- In Step 14, on the Configure the Administration Server screen, enter the following values for **Listen Address** and **Listen Port**:
  - **Listen Address**: `OIMADMINVHN.mycompany.com`
  - **Listen Port**: `7101` (*SPLIT_WLS_ADMIN_PORT*)
  - **SSL Listen Port**: `7102` (*SPLIT_WLS_ADMIN_SSL_PORT*)
  - **SSL Enabled**: Selected
- In Step 17, on the Configure Machines screen, specify the Name value `OIMADMINVHN` instead `ADMINVHN`.

As described in Section 8.5.1, "Copying OIM Adapter Template," you must perform the following steps if you are using Oracle Unified Directory in active-active mode:

1. After installing Oracle Identity and Access Management, apply Patch 16943171 .
2. Manually copy the file `adapter_template_oim.xml` from *ORACLE_COMMON_HOME*`/modules/oracle.ovd_11.1.1/templates/` to *IAM_ORACLE_HOME*`/libovd/`. For example:

   ```
   cp ORACLE_COMMON_HOME/modules/oracle.ovd_11.1.1/templates/adapter_template_
   oim.xml IAM_ORACLE_HOME/libovd/
   ```

In Section 8.5.3, "Reassociate the Domain with the Existing OPSS Policy Store," you associate the first domain with the OPSS policy store. For a split domain topology, you must also associate the additional domain with the existing policy store. To do that, you must first export the encryption key from the policy store and then join the new domain to the policy store using the generated encryption key.

To generate the OIMDomain with the existing OPSS data store, proceed as follows:

1. Start `wlst` using the command:

   ```
   ORACLE_COMMON_HOME/common/bin/wlst.sh
   ```

   Then execute this command to generate the encryption key:

   ```
   exportEncryptionKey
   (jpsConfigFile="wls-domain-path/config/fmwconfig/jps-config.xml",keyFilePath="o
   ```

```
pss-keyfile-path",keyFilePassword="opss-keyfile-password" )
```

For Example:

```
exportEncryptionKey(jpsConfigFile="ASERVER_
HOME/config/fmwconfig/jps-config.xml",keyFilePath="/u01/oracle/opss_keystore",
keyFilePassword="password" )
```

where *ASERVER_HOME* is the ASERVER_HOME of IDMDomain. This creates a file called ewallet.p12 in the location specified in *opss-keyfile-path*.

Note: Before you run the command, the directory specified in the *opss-keyfile-path* must exist.

2. Associate OIMDomain with the policy store by running the following command:

```
ORACLE_COMMON_HOME/common/bin/wlst.sh IAM_ORACLE_
HOME/common/tools/configureSecurityStore.py -d ASERVER_HOME -c IAM -m join -p
opss_schema_password -k opss-keyfile-path -w opss-keyfile-password
```

where *ASERVER_HOME* is the ASERVER_HOME of OIMDomain, *opss_schema_password* is the password of the schema EDG_OPSS and *opss-keyfile-path* and *opss-keyfile-password* are the values you supplied to the export command. For example:

```
ORACLE_COMMON_HOME/common/bin/wlst.sh IAM_ORACLE_
HOME/common/tools/configureSecurityStore.py -d ASERVER_HOME -c IAM -m join -p
opss_schema_password -k /u01/oracle/opss_keystore -w password
```

To associate subsequent domains with the existing policy store, you must first export the encryption key from the policy store and then join the new domain to the policy store using the generated encryption key.

When performing a backup after creating domains, back up the OIMDomain as well as the IDMDomain. See Section 17.6.3, "Performing Backups During Installation and Configuration."

16.6 Additional Web Tier Requirements

Section 10.5, "Setting the Front End URL for the Administration Console" describes how to change the Administration Server's front end URL so that the user's browser is redirected to the appropriate load balancer address. When you have a split domain topology, in Step 8, set the **Front End Host** and **Front End HTTP PORT** fields to the OIMADMIN load balancer address, as follows.

- **Front End Host**: OIMADMIN.mycompany.com
- **Front End HTTP PORT**: 80 (*HTTP_PORT*)

Section 10.3.3.2, "Create Virtual Host Definitions" describes the files to create on each web host in *WEB_ORACLE_INSTANCE*/config/OHS/*component_name*/moduleconf. When you are using a split domain, you must make the following changes:

- **Create Virtual Host for ADMIN.mycompany.com**–The file admin_vh.conf must not contain the Oracle Identity Manager configuration. The file should contain only the following lines:

```
<VirtualHost *:7777>

   ServerName ADMIN.mycompany.com:80
```

```
   RewriteEngine On
   RewriteOptions inherit
   ServerAdmin you@your.address

###################################
## General Domain Configuration
###################################

   # Admin Server and EM
   <Location /console>
      SetHandler weblogic-handler
      WebLogicHost ADMINVHN.mycompany.com
      WebLogicPort 7001
   </Location>

   <Location /consolehelp>
      SetHandler weblogic-handler
      WebLogicHost ADMINVHN.mycompany.com
      WebLogicPort 7001
   </Location>

   <Location /em>
      SetHandler weblogic-handler
      WebLogicHost ADMINVHN.mycompany.com
      WebLogicPort 7001
   </Location>

################################################
## Entries Required by Oracle Entitlements Server
################################################

   # APM
   <Location /apm>
      SetHandler weblogic-handler
      WebLogicHost ADMINVHN.mycompany.com
      WebLogicPort 7001
   </Location>

################################################
## Entries Required by Oracle Unified Directory
################################################

   # OUD ODSM
   <Location /odsm>
      SetHandler weblogic-handler
      WebLogicHost ADMINVHN.mycompany.com
      WebLogicPort 7001
   </Location>

############################################
## Entries Required by Oracle Access Manager
############################################

   # OAM Console
   <Location /oamconsole>
      SetHandler weblogic-handler
      WebLogicHost ADMINVHN
      WebLogicPort 7001
   </Location>
```

```
</VirtualHost>
```

- **Create Virtual Host for OIMADMIN.mycompany.com**–For split domain only, create a file called `oimadmin_vh.conf`. This will contain a list of locations which are supported by clients accessing the domain using `OIMADMIN.mycompany.com`. These entries are used by clients accessing the administration components in the OIMDomain.

```
<VirtualHost *:7777>
   ServerName http://OIMADMIN.mycompany.com:80
   RewriteEngine On
   RewriteOptions inherit
   ServerAdmin you@your.address

############################
## Generic Entries
############################

   # Admin Server and EM
   <Location /console>
      SetHandler weblogic-handler
      WebLogicHost OIMADMINVHN.mycompany.com
      WebLogicPort 7101
   </Location>

   <Location /consolehelp>
      SetHandler weblogic-handler
      WebLogicHost OIMADMINVHN.mycompany.com
      WebLogicPort 7101
   </Location>

   <Location /em>
      SetHandler weblogic-handler
      WebLogicHost OIMADMINVHN.mycompany.com
      WebLogicPort 7101
   </Location>

################################################
## Entries required by Oracle Identity Manager
################################################

   # OIM self and advanced admin webapp consoles(canonic webapp)

   <Location /oim>
     SetHandler weblogic-handler
     WLProxySSL ON
     WLProxySSLPassThrough ON
     WLCookieName    oimjsessionid
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
   </Location>

   # OIM, xlWebApp - Legacy 9.x webapp (struts based)
   <Location /xlWebApp>
     SetHandler weblogic-handler
     WLProxySSL ON
     WLProxySSLPassThrough ON
     WLCookieName    oimjsessionid
     WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
     WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
```

```
    </Location>

    # OIM self service console
    <Location /identity>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WLCookieName oimjsessionid
      WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
      WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
    </Location>

    # OIM Nexaweb WebApp - used for workflow designer and DM
    <Location /Nexaweb>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WLCookieName    oimjsessionid
      WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
      WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
    </Location>

    # OIM System Admin Console
    <Location /sysadmin>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicCluster OIMHOST1VHN:14000,OIMHOST2VHN:14000
      WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/OHS/oim_component.log"
    </Location>

</VirtualHost>
```

16.7 Additional Access Manager Requirements

When updating the OAM agent, as described in Section 11.6.5, "Updating Newly-Created Agent," enter the following values for **Host Name** and **Port** in addition to the values listed in Step 16:

- **Host Name**: `OIMADMIN.mycompany.com`
- **Port**: `80` (*HTTP_PORT*)

16.8 Additional Oracle Identity Manager Requirements

Chapter 12, "Extending the Domain to Include Oracle Identity Manager," describes how to install and configure Oracle Identity Manager for the single domain topology. In general, when you are building a split domain topology, the procedures are similar, but you specify the OIMDomain instead of IDMDomain and the virtual host OIMADMINVHN instead of ADMINVHN. The specific changes are described in the following sections.

This section contains the following topics:

- Section 16.8.1, "Domain URLs"
- Section 16.8.2, "Provisioning the Login Modules and Creating the wlfullclient.jar"
- Section 16.8.3, "Extending the Domain to Configure Oracle Identity Manager and Oracle SOA Suite"

- Section 16.8.4, "Configuring Oracle Identity Manager"
- Section 16.8.5, "Deploying Oracle Identity Manager and Oracle SOA"
- Section 16.8.6, "Enabling Oracle Identity Manager to Connect to SOA"
- Section 16.8.7, "Configuring Access Manager for Oracle Identity Manager Integration"
- Section 16.8.8, "Backing Up Oracle Identity Manager"

16.8.1 Domain URLs

In Section 12.2, "About Domain URLs," Table 12–1 list the URLs for Oracle Identity Manager that become available after the installation and configuration are complete. When you are configuring a split domain topology, the Oracle Identity Manager URL will be as follows, rather than the one listed in Table 12–1:

Table 16–6 OIM URLs

Component	URLs	SSO User
OIM Administration Console	`http://OIMADMIN.mycompany.com/sysadmin`	xelsysadm

16.8.2 Provisioning the Login Modules and Creating the wlfullclient.jar

In a split doamin topology, perform the steps in Section 12.4, "Provisioning the OIM Login Modules Under the WebLogic Server Library Directory," but in *OIM_MW_HOME* rather than *IAM_MW_HOME*.

As described in Section 12.5, "Creating the wlfullclient.jar File," you must create the `wlfullclient.jar` library this library under the *IAM_MW_HOME*`/wlserver_10.3/server/lib` directory on all the machines hosting Oracle Identity Manager in the application tier of your environment. In a split domain topology, create the library in *OIM_MW_HOME* rather than *IAM_MW_HOME*. Specifically, in Step 1, you must navigate to the *OIM_MW_HOME*`/wlserver_10.3/server/lib` directory.

16.8.3 Extending the Domain to Configure Oracle Identity Manager and Oracle SOA Suite

You must extend the domain with Oracle Identity Manager components as described in Section 12.7, "Extending the Domain to Configure Oracle Identity Manager and Oracle SOA Suite." When you are creating a split domain topology, you extend the OIMDomain rather than the IDMDomain.

In Step 2, on the Select WebLogic Domain Directory screen, select the location of the domain directory for the OIMDomain, *ASERVER_HOME*, for example: `/u01/oracle/config/domains/`*domain_name*

16.8.4 Configuring Oracle Identity Manager

When configuring Oracle Identity Manager, as described in Section 12.10, "Configuring Oracle Identity Manager," on the WebLogic Administration Server screen in Step 4, specify the following URL to connect to the WebLogic Administration Server:

`t3://OIMADMINVHN.mycompany.com:7101`

where `7101` is *WLS_ADMIN_PORT* in Section B.3.

16.8.5 Deploying Oracle Identity Manager and Oracle SOA

As described in Section 12.8, "Deploying Oracle Identity Manager and Oracle SOA to Managed Server Domain Directory on IDMHOST1 and IDMHOST2," once the configuration is complete, you must propagate the Oracle Identity Manager configuration to the managed server directory on IDMHOST1 and IDMHOST2. Instead of packing the domain IDMDomain, do this on OIMDomain.

16.8.6 Enabling Oracle Identity Manager to Connect to SOA

In Section 12.20, "Enabling Oracle Identity Manager to Connect to SOA Using the Administrative Users Provisioned in LDAP," you perform post installation steps to enable Oracle Identity Manager to work with the Oracle WebLogic Server administrator user provisioned in the central LDAP store. Be sure to specify the correct domain in the following steps:

- Step 2: For a split domain topology, select **Farm_OIMDomain –> Identity and Access –> OIM –> oim(11.1.2.0.0).**
- Step 7: For a split domain topology, select **Weblogic Domain –> OIMDomain** from the Navigator.
- Step 14a: Log in to the OIM Administration Console at the URL `http://OIMADMIN.mycompany.com/sysadmin`, rather than `http://ADMIN.mycompany.com/sysadmin`, as the user `xelsysadm`.

16.8.7 Configuring Access Manager for Oracle Identity Manager Integration

In Section 12.23.4, "Integrating Oracle Identity Manager with Oracle Access Manager Using the idmConfigTool," your configuration must reflect the fact that your Oracle Identity Manager components are in are in a different domain from your Access Manager components. The following changes are required:

- When you create the properties file `oimitg.props` in Step 2, use the following contents instead of the contents shown in Section 12.23.4:

```
LOGINURI: /${app.context}/adfAuthentication
LOGOUTURI: /oamsso/logout.html
AUTOLOGINURI: None
ACCESS_SERVER_HOST: IDMHOST1.mycompany.com
ACCESS_SERVER_PORT: 5575
ACCESS_GATE_ID: Webgate_IDM
COOKIE_DOMAIN: .mycompany.com
COOKIE_EXPIRY_INTERVAL: 120
OAM_TRANSFER_MODE: simple
WEBGATE_TYPE: ohsWebgate11g
SSO_ENABLED_FLAG: true
IDSTORE_PORT: 389
IDSTORE_HOST: IDSTORE.mycompany.com
IDSTORE_DIRECTORYTYPE: OUD, OID or OVD
IDSTORE_ADMIN_USER: cn=oamLDAP,cn=Users,dc=mycompany,dc=com
IDSTORE_USERSEARCHBASE: cn=Users,dc=mycompany,dc=com
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=mycompany,dc=com
IDSTORE_LOGINATTRIBUTE: uid
MDS_DB_URL: jdbc:oracle:thin:@(DESCRIPTION=(LOAD_
BALANCE=on)(FAILOVER=on)(ADDRESS_
LIST=(ADDRESS=(protocol=tcp)(host=OIDDBHOST1-VIP.mycompany.com)(port=1521))(ADD
RESS=(protocol=tcp)(host=OIDDBHOST2-VIP.mycompany.com)(port=1521)))(CONNECT_
DATA=(SERVER=DEDICATED)(SERVICE_NAME=OIDEDG.mycompany.com)))
MDS_DB_SCHEMA_USERNAME: edg_mds
```

```
OIM_MANAGED_SERVER_NAME: WLS_OIM1
WLSADMIN: weblogic
WLSPORT: 7101
WLSHOST: OIMADMINVHN.mycompany.com
OAM11G_WLS_ADMIN_HOST: ADMINVHN.mycompany.com
OAM11G_WLS_ADMIN_PORT: 7001
OAM11G_WLS_ADMIN_USER: weblogic
DOMAIN_NAME: OIMDomain
DOMAIN_LOCATION: ASERVER_HOME
```

Additional and changed values, relative to those in Step 2 of Section 12.23.4, are shown in bold. The property definitions are the same, with one addition:

`OAM11G_WLS_ADMIN_HOST`, `OAM11G_WLS_ADMIN_PORT` and `OAM11G_WLS_ADMIN_USER` specify the details of the OAM Domain if your Oracle Identity Manager components are in a separate domain from your Access Manager components.

- When the script runs, you are prompted for information, as described in Step 3. When Access Manager and Oracle Identity Manager are in separate domains, you are also prompted for:

  Access Manager domain user password

- At the end of the procedure in Section 12.23.4, restart the WebLogic Administration Servers on both of the domains.

16.8.8 Backing Up Oracle Identity Manager

When performing a backup after configuring Oracle Identity Manager, back up the OIMDomain as well as the IDMDomain. See Section 17.6.3, "Performing Backups During Installation and Configuration."

16.9 Additional Single Sign-On Requirements

When performing a backup after configuring Single Sign-On, back up the OIMDomain as well as the IDMDomain. See Section 17.6.3, "Performing Backups During Installation and Configuration."

16.10 Additional Node Manager Requirements

In Section 13.3.1, "Generating Self-Signed Certificates Using the utils.CertGen Utility," Step 4, when you run the `utils.CertGen` tool, create an additional certificate on `OIMADMINVHN.mycompany.com`. For example

```
java utils.CertGen Key_Passphrase OIMADMINVHN.mycompany.com_cert
OIMADMVHN.mycompany.com_key domestic OIMADMINVHN.mycompany.com
```

Also perform the steps in Section 13.3.2, "Creating an Identity Keystore Using the utils.ImportPrivateKey Utility," for `OIMADMINVHN.mycompany.com`.

16.11 Additional Management Requirements

Perform the following additional management tasks when you are using a split domain topology.

This section contains the following topics:

- Section 16.11.1, "Applying Patches"

- Section 16.11.2, "Performing Backups"

16.11.1 Applying Patches

Section 17.7.2, "Patching Identity and Access Management," describes how to patch Identity and Access management in a single domain topology.

In a split domain topology, where Oracle Identity Manager is located in a domain separate from other components, apply patches as follows:

IDMDomain MW_HOME

- Common patches
- Oracle Access Manager Patches
- IDM Tool Patches

OIMDomain MW_HOME

- Common patches
- Oracle Identity Manager Patches
- IDM Tool Patches

Identity Management MW_HOME

- Common patches
- Oracle Internet Directory Patches
- Oracle Virtual Directory Patches

It is not necessary to stop processes in the IDMDomain while applying patches to the OIMDomain. Similarly, it is not necessary to stop processes in the OIMDomain while applying patches to the IDMDomain.

16.11.2 Performing Backups

When performing a backup after creating a domain, configuring Oracle Identity Manager, and configuring Single Sign-On, back up the OIMDomain as well as the directories in the IDMDomain. See Section 17.6.3, "Performing Backups During Installation and Configuration."

17

Managing the Topology for an Enterprise Deployment

This chapter describes some operations that you can perform after you have set up the Identity Management topology. These operations include monitoring, scaling, backing up your topology, and troubleshooting.

This chapter includes the following topics:

17.1 Starting and Stopping Oracle Identity Management Components

This section describes how to start, stop and restart the various components of the Oracle Enterprise Deployment for Identity Management.

This section contains the following topics:

17.1.1 Startup Order

When starting up your entire infrastructure, start the components in the following order, (ignoring those not in your topology):

1. Database(s)
2. Database Listener(s)
3. LDAP Directory Server
4. Node Manager
5. Oracle Access Manager Server(s)
6. WebLogic Administration Server
7. Oracle HTTP Server(s)
8. SOA Server(s)
9. Oracle Identity Manager Server(s)

17.1.2 Starting and Stopping Oracle Unified Directory

Start and stop Oracle Unified Directory as follows:

17.1.2.1 Starting Oracle Unified Directory

To start Oracle Unified Directory issue the following command:

```
OUD_ORACLE_INSTANCE/OUD/bin/start-ds
```

17.1.2.2 Stopping Oracle Unified Directory

To stop Oracle Unified Directory issue the command:

```
OUD_ORACLE_INSTANCE/OUD/bin/stop-ds
```

17.1.3 Starting, Stopping, and Restarting Access Manager Managed Servers

Start and stop Oracle Access Manager Managed Servers as follows:

17.1.3.1 Starting an Access Manager Managed Server When None is Running

Normally, you start Access Manager managed servers by using the WebLogic console. After you have enabled Single Sign-On for the administration consoles, however, you must have at least one Access Manager Server running in order to access a console. If no Access Manager server is running, you can start one by using WLST.

To invoke WLST on Linux or UNIX, type:

```
cd ORACLE_COMMON_HOME/common/bin
./wlst.sh
```

Once you are in the WLST shell, execute the following commands:

```
nmConnect('Admin_User','Admin_Password', 'OAMHOST','Port',  'domain_
name','MSERVER_HOME')
nmStart('OAMServer')
```

where Port is *NMGR_PORT* in Section B.3, *domain_name* is the name of the domain and *Admin_User* and *Admin_Password* are the Node Manager username and

password you entered in Step 2 of Section 8.5.5, "Updating the Node Manager Credentials." For example:

```
nmConnect('weblogic','password', 'IDMHOST1','5556',  'IDMDomain','MSERVER_HOME')
nmStart('WLS_OAM1')
```

17.1.3.2 Starting an Access Manager Managed Server When Another is Running

To start an Oracle Access Manager managed server when you already have another one running, log in to the WebLogic console using the URL listed in Section 17.2, "About Identity Management Console URLs."

Then proceed as follows:

1. Select **Environment - Servers** from the Domain Structure menu.
2. Click the **Control** tab.
3. Select **OAM Servers (WLS_OAM1 and/or WLS_OAM2)**.
4. Click the **Start** button.
5. Click **Yes** when asked to confirm that you want to start the server(s).

17.1.3.3 Stopping Access Manager Managed Servers

To stop the Oracle Access Manager Managed Server(s), log in to the WebLogic console using the URL listed in Section 17.2, "About Identity Management Console URLs." Then proceed as follows:

1. Select **Environment - Servers** from the Domain Structure menu.
2. Click the **Control** tab.
3. Select **OAM Servers (WLS_OAM1 and/or WLS_OAM2)**.
4. Click the **Shutdown** button and select **Force Shutdown now**.
5. Click **Yes** when asked to confirm that you want to shut down the server(s).

17.1.3.4 Restarting Access Manager Managed Servers

Restart the server by following the `Stop` and `Start` procedures in the previous sections.

17.1.4 Starting, Stopping, and Restarting WebLogic Administration Server

Start and stop the WebLogic Administration Server as described in the following sections.

> **Note:** `Admin_User` and `Admin_Password` are only used to authenticate connections between Node Manager and clients. They are independent from the server administration ID and password and are stored in the *ASERVER_HOME*`/config/nodemanager/nm_password.properties` file.

17.1.4.1 Starting WebLogic Administration Server

The recommended way to start the Administration server is to use WLST and connect to Node Manager:

```
cd ORACLE_COMMON_HOME/common/bin
```

```
./wlst.sh
```

Once in WLST shell, execute

```
nmConnect('Admin_User','Admin_Password','ADMINVHN','5556', 'IDMDomain','ASERVER_
HOME')
nmStart('AdminServer')
```

Alternatively, you can start the Administration server by using the command:

```
ASERVER_HOME/bin/startWebLogic.sh
```

17.1.4.2 Stopping WebLogic Administration Server

To stop the Administration Server, log in to the WebLogic console using the URL listed in Section 17.2, "About Identity Management Console URLs."

Then proceed as follows:

1. Select **Environment - Servers** from the Domain Structure menu.
2. Click the **Control** tab.
3. Select **AdminServer(admin)**.
4. Click **Shutdown** and select **Force Shutdown now**.
5. Click **Yes** when asked to confirm that you want to shut down the Administration Server.

17.1.4.3 Restarting WebLogic Administration Server

Restart the server by following the `Stop` and `Start` procedures in the previous sections.

17.1.5 Starting and Stopping Node Manager

Start and stop the Node Manager as follows:

17.1.5.1 Starting Node Manager

If the Node Manager being started is the one that controls the Administration Server (IDMHOST1 or IDMHOST2), then prior to starting the Node Manager issue the command:

```
export JAVA_OPTIONS=-DDomainRegistrationEnabled=true
```

If you are using shared storage for Node Manager, set `JAVA_OPTIONS` as described in Section 13.3.5, "Using a Common or Shared Storage Installation."

To start Node Manager, issue the commands:

```
cd IAM_MW_HOME/wlserver_10.3/server/bin
./startNodeManager.sh
```

17.1.5.2 Stopping Node Manager

To stop Node Manager, kill the process started in the previous section.

17.1.5.3 Starting Node Manager for an Administration Server

```
cd WL_HOME/server/bin
export JAVA_OPTIONS=-DDomainRegistrationEnabled=true
./startNodeManager.sh
```

Note: It is important to set `-DDomainRegistrationEnabled=true` whenever you start a Node Manager that manages the Administration Server.

17.1.6 Starting, Stopping, and Restarting Oracle HTTP Server

Prior to starting/stopping the Oracle HTTP server ensure that the environment variables *WEB_ORACLE_HOME* and *ORACLE_INSTANCE* are defined and that *ORACLE_HOME*`/opmn/bin` appears in the *PATH*. For example:

```
export ORACLE_HOME=WEB_ORACLE_HOME
export ORACLE_INSTANCE=WEB_ORACLE_INSTANCE
export PATH=$ORACLE_HOME/opmn/bin:$PATH
```

17.1.6.1 Starting Oracle HTTP Server

Start the Oracle web tier by issuing the command:

```
opmnctl startall
```

17.1.6.2 Stopping Oracle HTTP Server

Stop the web tier by issuing the command

```
opmnctl stopall
```

to stop the entire Web tier or

```
opmnctl stoproc process-type=OHS
```

to stop Oracle HTTP Server only.

17.1.6.3 Restarting Oracle HTTP Server

You can restart the web tier by issuing a `Stop` followed by a `Start` as described in the previous sections.

To restart the Oracle HTTP server only, use the following command.

```
opmnctl restartproc process-type=OHS
```

17.1.7 Starting, Stopping, and Restarting Oracle Identity Manager

Start and stop Oracle Identity Manager and Oracle SOA Suite servers as follows:

17.1.7.1 Starting Oracle Identity Manager

To start the Oracle Identity Manager Managed Server(s), log in to the WebLogic console using the URL listed in Section 17.2, "About Identity Management Console URLs."

Then proceed as follows:

1. Select **Environment - Servers** from the Domain Structure menu.
2. Click the **Control** tab.
3. Select **SOA Servers (WLS_SOA1 and/or WLS_SOA2)**.

Note: You can start the Oracle Identity Manager and Oracle SOA Suite servers independently of each other. There is no dependency in their start order. However, the SOA server must be up and running for all of the Oracle Identity Manager functionality to be available.

4. Click the **Start** button.
5. Click **Yes** when asked to confirm that you want to start the server(s).
6. After WLS_SOA1 and/or WLS_SOA2 have started, select WLS_OIM1 and/or WLS_OIM2
7. Click **Start**.
8. Click **Yes** when asked to confirm that you want to start the server(s).

17.1.7.2 Stopping Oracle Identity Manager

To stop the Oracle Identity Manager Managed Server(s), log in to the WebLogic console using the URL listed in Section 17.2, "About Identity Management Console URLs." Then proceed as follows:

1. Select **Environment - Servers** from the Domain Structure menu.
2. Click the **Control** tab.
3. Select **OIM Servers (WLS_OIM1 and/or WLS_OIM2)** and **(WLS_SOA1 and/or WLS_SOA2)**.
4. Click the **Shutdown** button and select **Force Shutdown now**.
5. Click **Yes** when asked to confirm that you want to shutdown the server(s).

17.1.7.3 Restarting Oracle Identity Manager

Restart the server by following the `Stop` and `Start` procedures in the previous sections.

17.2 About Identity Management Console URLs

Table 17–1 lists the administration consoles used in this guide and their URLs.

Table 17–1 Console URLs

Domain	Console	URL
IDMDomain	WebLogic Administration Console	`http://ADMIN.mycompany.com/console`
	Enterprise Manager FMW Control	`http://ADMIN.mycompany.com/em`
	OAM Console	`http://ADMIN.mycompany.com/oamconsole`
	ODSM	`http://ADMIN.mycompany.com/odsm`
OIMDomain	OIM Console	`https://SSO.mycompany.com/identity`
OIMDomain	WebLogic Administration Console	`http://OIMADMIN.mycompany.com/console`

Table 17–1 (Cont.) Console URLs

Domain	Console	URL
OIMDomain	Enterprise Manager FMW Control	`http://OIMADMIN.mycompany.com/em`
OIMDomain	Authorization Policy Manager	`http://OIMADMIN.mycompany.com/apm`

17.3 Monitoring Enterprise Deployments

This section provides information about monitoring the Identity Management enterprise deployment described in this manual.

This section contains the following topics:

- Section 17.3.1, "Monitoring WebLogic Managed Servers"
- Section 17.3.2, "Monitoring Oracle Unified Directory"

17.3.1 Monitoring WebLogic Managed Servers

You can use Oracle Enterprise Manager Fusion Middleware Control to monitor Managed Servers and other Fusion Middleware components, such as Access Manager, Oracle Identity Manager, and SOA. For more information, see the administrator guides listed in the Preface under "Related Documents" on page xvii.

17.3.2 Monitoring Oracle Unified Directory

You can check the status of Oracle Unified Directory by issuing the command:

```
OUD_ORACLE_INSTANCE/OUD/bin/status
```

This command accesses the locally running Oracle Unified Directory instance and reports the status of the directory, including whether or not replication and LDAP or LDAPS is enabled.

17.4 Scaling Enterprise Deployments

The reference enterprise topology discussed in this manual is highly scalable. It can be scaled up and or scaled out. When the topology is scaled up, a new server instance is added to a node already running one or more server instances. When the topology is scaled out, new servers are added to new nodes.

This section contains the following topics:

- Section 17.4.1, "Scaling Up the Topology"
- Section 17.4.2, "Scaling Out the Topology"

17.4.1 Scaling Up the Topology

The Oracle Identity Management topology described in the guide has three tiers: the database tier, application tier and web tier. The components in all the three tiers can be scaled up by adding a new server instance to a node that already has one or more server instances running.

The procedures described in this section show you how to create a new managed server or directory instance. If you add a new managed server, after adding the

managed server you must update your Oracle HTTP Server configuration files (on all nodes) and add the new server to the existing WebLogic cluster directives.

For example if you add a new Oracle Access Manager server, you must update *WEB_ORACLE_INSTANCE*/config/OHS/*component_name*/moduleconf/sso_vh.conf to include the new managed server.

Update sso_vh.conf as follows:

```
<Location /oam>
   SetHandler weblogic-handler
   WebLogicCluster
IDMHOST1.mycompany.com:14100,IDMHOST2.mycompany.com:14100,IDMHOST3.mycompany.com:1
4100
</Location>
```

Once you have updated sso_vh.conf, restart the Oracle HTTP server(s) as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components." Oracle recommends that you do this sequentially to prevent loss of service.

This section contains the following topics:

- Section 17.4.1.1, "Scaling Up Oracle Unified Directory"
- Section 17.4.1.2, "Scaling Up Oracle Access Manager 11g"
- Section 17.4.1.3, "Scaling Up Oracle Identity Manager"
- Section 17.4.1.4, "Scaling Up Oracle HTTP Server"

17.4.1.1 Scaling Up Oracle Unified Directory

The directory has two Oracle Unified Directory nodes, IDMHOST1 and IDMHOST2, each running an Oracle Unified Directory instance. The Oracle Unified Directory binaries on either node can be used for creating the new Oracle Unified Directory instance.

To add a new Oracle Unified Directory instance to either Oracle Unified Directory host, follow the steps in Section 7.4.3, "Configuring an Additional Oracle Unified Directory Instance on IDMHOST2," with the following variations:

- In Step 2 and Step 4, choose ports other than 1389 (*LDAP_DIR_PORT*), 1636 (*LDAP_DIR_SSL_PORT*), or 4444 (*LDAP_DIR_ADMIN_PORT*), as those ports are being used by the existing Oracle Unified Directory instance on the node.
- Use the location for the new Oracle Unified Directory instance as the value for *ORACLE_INSTANCE*.
- Reconfigure the load balancer with the host and port information of the new Oracle Unified Directory instance.

17.4.1.2 Scaling Up Oracle Access Manager 11g

Scale up Oracle Access Manager as follows:

Log in to the Oracle WebLogic Server Administration Console at the URL listed in Section 17.2, "About Identity Management Console URLs."

1. From the Domain Structure window of the Oracle WebLogic Server Administration Console, expand the **Environment** node and then **Servers**. The Summary of Servers page appears.
2. Click **Lock & Edit** from the Change Center menu.
3. Select an existing server on the host you want to extend, for example: WLS_OAM1.

4. Click **Clone**.
5. Enter the following information:
   - **Server Name**: A new name for the server, for example: `WLS_OAM3`.
   - **Server Listen Address**: The name of the host on which the Managed Server runs.
   - **Server Listen Port**: The port the new Managed Server uses. This port must be unique within the host.
6. Click **OK**.
7. Click the newly created server **WLS_OAM3**
8. Click **Save**.
9. Disable host name verification for the new Managed Server. Before starting and verifying the `WLS_OAM3` Managed Server, you must disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Oracle WebLogic Administration Server and the Node Manager in `IDMHOST`*n*.

   If the source server from which the new one was cloned had already disabled host name verification, these steps are not required, as the host name verification settings were propagated to the cloned server. To disable host name verification:

   a. Log in to **Oracle WebLogic Server Administration Console**.
   b. Expand the **Environment** node in the Domain Structure window.
   c. Click **Servers**. The Summary of Servers page appears.
   d. Select **WLS_OAM3** in the Names column of the table. The Settings page for server appears.
   e. Click the **SSL** tab.
   f. Click **Advanced**.
   g. Set **Hostname Verification** to `None`.
   h. Click **Save**.
10. Click **Activate Changes** from the Change Center menu.

Register the new Managed Server with Oracle Access Manager. You now must configure the new Managed Server as an Oracle Access Manager server. You do this from the Oracle OAM console. Proceed as follows:

1. Log in to the OAM console as the `oamadmin` user. Use the URL listed in Section 17.2, "About Identity Management Console URLs."
2. Click the **System Configuration** tab.
3. Click **Server Instances**.
4. Select **Create** from the Actions menu.
5. Enter the following information:
   - **Server Name**: `WLS_OAM3`
   - **Host**: Host that the server runs on
   - **Port**: Listen port that was assigned when the Managed Server was created

- **OAM Proxy Port**: Port you want the Oracle Access Manager proxy to run on. This is unique for the host
- **Proxy Server ID**: `AccessServerConfigProxy`
- **Mode**: Set to `Open` or `Simple`, depending on the mode your existing Oracle Access Manager servers are operating in.
- **Coherence Local Port**: Set Local Port to a unique value on the host

6. Click **Apply**.
7. Restart the WebLogic Administration Server as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

Add the newly created Oracle Access Manager server to all WebGate Profiles that might be using it, such as `Webgate_IDM` and `IAMSuiteAgent`

For example, to add the Oracle Access Manager server to `Webgate_IDM`, access the OAM console at the URL listed in Section 17.2, "About Identity Management Console URLs." Then proceed as follows:

1. Log in as the Oracle Access Manager Admin User you created inSection 9.4, "Preparing the Identity Store."
2. Click the **System Configuration** tab.
3. Expand **Access Manager** - **SSO Agents** - **OAM Agents**.
4. Click the open folder icon, then click **Search**.

   You should see the WebGate agent **Webgate_IDM**.
5. Click the agent **Webgate_IDM**.
6. Select **Edit** from the **Actions** menu.
7. Click **+** in the **Primary Server** list (or the **Secondary Server** list if this is a secondary server).
8. Select the newly created managed server from the **Server** list.
9. Set **Maximum Number of Connections** to `10`.
10. Click **Apply**.

Repeat Steps 5 through 10 for **IAMSuiteAgent** and all other WebGates that might be in use.

Update the Web Tier. Once the new Managed Server has been created and started, the web tier starts to direct requests to it. Best practice, however, is to inform the web server that the new Managed Server has been created.

You do this by updating the file `sso_vh.conf` on each of the web tiers. This file resides in the directory: *`WEB_ORACLE_INSTANCE`*`/config/OHS/`*`component_name`*`/moduleconf`.

Add the new server to the `WebLogicCluster` directive in the file, for example, change:

```
<Location /oam>
   SetHandler weblogic-handler
   WebLogicCluster IDMHOST1.mycompany.com:14100,IDMHOST2.mycompany.com:14100
</Location>
```

to:

```
<Location /oam>
```

```
   SetHandler weblogic-handler
   WebLogicCluster
IDMHOST1.mycompany.com:14100,IDMHOST2.mycompany.com:14100,IDMHOST1.mycompany.com:1
4101
</Location>
```

Save the file and restart the Oracle HTTP server, as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

You can now start the new Managed Server, as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

17.4.1.3 Scaling Up Oracle Identity Manager

In this case, you already have a node that runs a Managed Server configured with Oracle SOA Suite and Oracle Identity Manager components. The node contains a Middleware home, a SOA Oracle home, an Oracle Identity Manager Oracle home, and a domain directory for existing Managed Servers.

You can use the existing installations (the Middleware home, and domain directories) for creating new WLS_OIM and WLS_SOA servers. There is no need to install the Oracle Identity and Access Management or Oracle SOA Suite binaries in a new location, or to run pack and unpack.

Follow these steps for scaling up the topology:

1. Log in to the Administration Console at the URL listed in Section 17.2, "About Identity Management Console URLs." Clone either the **WLS_OIM1** or the **WLS_SOA1** into a new Managed Server. The source Managed Server to clone should be one that already exists on the node where you want to run the new Managed Server.

   To clone a Managed Server:

   a. Select **Environment** -> **Servers** from the Administration Console.

   b. From the Change Center menu, click **Lock and Edit**.

   c. Select the Managed Server that you want to clone (for example, **WLS_OIM1** or **WLS_SOA1**).

   d. Select **Clone**.

   Name the new Managed Server WLS_OIM*n* or WLS_SOA*n*, where *n* is a number to identify the new Managed Server.

   The rest of the steps assume that you are adding a new server to IDMHOST1, which is already running WLS_SOA1 and WLS_OIM1.

2. For the listen address, assign the host name or IP address to use for this new Managed Server. If you are planning to use server migration as recommended for this server, this should be the VIP (also called a floating IP) to enable it to move to another node. The VIP should be different from the one used by the Managed Server that is already running.

3. Create JMS Servers for SOA, Oracle Identity Manager, UMS, and BPM on the new Managed Server. You do this as follows:

   a. Log in to the WebLogic Administration Console and navigate to **Services -> Messaging -> JMS Servers**.

   b. Click **New**.

   c. Enter a value for **Name**, such as `BPMJMSServer_auto_3`.

d. Click **Create New File Store**.

e. Select `FileStore` from the list

f. Click **Next**.

g. Enter a value for **Name**, such as `BPMJMSFileStore_auto_3`

h. Enter the following values:

Target: The new server you are creating.

Directory: *`ASERVER_HOME`*`/jms/BPMJMSFileStore_auto_3`

Note: Ensure that the directory exists. If it does not, you will not be able to start WLS_OIM3 or WLS_SOA3.

---

i. Click **OK**.

j. When you are returned to the JMS Server screen, select the newly created file store from the list.

k. Click **Next**.

l. On the next screen set the Target to the server you are creating.

m. Click **Finish**.

Create the following JMS Queues depending on the managed server you are creating:

Server	JMS Server Name	File Store Name	Directory	Target
WLS_SOA*n*	BPMJMSServer_auto_*n*	BPMJMSFileStore_auto_*n*	*`ASERVER_HOME`*`/jms/BPMJMSFileStore_auto_`*n*	WLS_SOA*n*
WLS_SOA*n*	SOAJMSServer_auto_*n*	SOAJMSFileStore_auto_*n*	*`ASERVER_HOME`*`/jms/SOAJMSFileStore_auto_`*n*	WLS_SOA*n*
WLS_SOA*n*	UMSJMSServer_auto_*n*	UMSJMSFileStore_auto_*n*	*`ASERVER_HOME`*`/jms/UMSJMSFileStore_auto_`*n*	WLS_SOA*n*
WLS_OIM*n*	OIMJMSServer_auto_*n*	OIMJMSFileStore_auto_*n*	*`ASERVER_HOME`*`/jms/OIMJMSFileStore_auto_`*n*	WLS_OIM*n*
WLS_OIM*n*	JRFWSAsyncJmsServer_auto_*n*	JRFWSAsyncFileStore_auto_*n*	*`ASERVER_HOME`*`/jms/JRFWSAsyncFileStore_auto_`*n*	WLS_OIM*n*

4. Add the newly created JMS Queues to the existing JMS Modules by performing the following steps:

a. Log in to the WebLogic Administration Console

b. Navigate to **Services -> Messaging -> JMS Modules**

c. Click a JMSModule, such as **SOAJMSModule**

d. Click the **Sub Deployments** tab.

e. Click the listed sub deployment.

> **Note:** This subdeployment module name is a random name in the form of **JMSServerName*XXXXXX*** resulting from the Configuration Wizard JMS configuration.

f. Assign the newly created JMS server, for example **SOAJMSServer_auto_3**.

g. Click **Save**.

Perform this for each of the JMS modules listed in the following table:

JMS Module	JMS Server
BPMJMSModule	BPMJMSServer_auto_*n*
JRFWSAsyncJmsModule	JRFWSAsyncJmsServer_auto_*n*
OIMJMSModule	OIMJMSServer_auto_*n*
SOAJMSModule	SOAJMSServer_auto_*n*
UMSJMSSystemResource	UMSJMSServer_auto_*n*

5. Configure Oracle Coherence, as described in Section 12.9, "Configuring Oracle Coherence for Deploying Composites."

> **Note:** Only the **localhost** field must be changed for the server. Replace the localhost with the listen address of the new server added:

6. Reconfigure the JMS Adapter with the new server using the **FactoryProperties** field in the Administration Console. Click on the corresponding cell under the Property value and enter the following:

```
java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;
java.naming.provider.url=t3://soahostvhn1:8001,soahos2tvhn1:8001;
java.naming.security.principal=weblogic;
java.naming.security.credentials=weblogic1
```

7. Configure TX persistent store for the new server. This should be a location visible from other nodes as indicated in the recommendations about shared storage.

a. Log in to the Oracle WebLogic Server Administration Console.

b. Click **Lock and Edit**.

c. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node.

The Summary of Servers page is displayed.

d. Click the name of either the Oracle Identity Manager or the SOA server (represented as a hyperlink) in the **Name** column of the table.

e. The Settings page for the selected server is displayed, and defaults to the **Configuration** tab.

f. Open the **Services** sub tab.

g. Under the **Default Store** section of the page, provide the path to the default persistent store on shared storage. The directory structure of the path is as follows:

For Oracle Identity Manager Servers: *ASERVER_HOME*/tlogs

For SOA Servers: *ASERVER_HOME*/tlogs

h. Click **Save and Activate**.

8. Disable host name verification for the new Managed Server. Before starting and verifying the WLS_SOA*n* Managed Server, you must disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Oracle WebLogic Administration Server and the Node Manager in IDMHOST*n*. If the source server from which the new one has been cloned had already disabled host name verification, these steps are not required (the host name verification settings is propagated to the cloned server).

   To disable host name verification:

   a. Log in to **Oracle WebLogic Server Administration Console**.

   b. Expand the **Environment** node in the Domain Structure window.

   c. Click **Servers**. The Summary of Servers page appears.

   d. Select **WLS_SOA*n*** in the Names column of the table. The Settings page for the server appears.

   e. Click the **SSL** tab.

   f. Click **Advanced**.

   g. Set **Hostname Verification** to None.

   h. Click **Save**.

9. Repeat Steps 6a through 6h to disable host name verification for the WLS_OIM*n* Managed Servers. In Step d, select **WLS_OIM*n*** in the Names column of the table.

10. Add new server to Cluster Address List.

    a. In the WebLogic Administration Console, expand the **Environment** node in the Domain Structure window.

    b. Click **Clusters**. The Summary of Clusters page appears.

    c. Click on the cluster name **oim_cluster**. The Cluster Properties page is displayed.

    d. Add the new OIM managed server to the cluster address list.

    e. Click **Save**.

11. Click **Activate Changes** from the Change Center menu.

12. Update the SOA host and port using Oracle Enterprise Manager Fusion Middleware Control. Follow these steps:

    a. Open a browser and go to Oracle Enterprise Manager Fusion Middleware Control at the URL listed in Section 17.2, "About Identity Management Console URLs."

    b. Log in to Oracle Enterprise Manager Fusion Middleware Control using the Admin user credentials.

Note: At least one of the Oracle Identity Manager Managed Servers must be running when these steps are executed.

---

c. Select **Farm_IDMDomain –> Identity and Access –> OIM –> oim(11.1.2.0.0).**

d. Select **System MBean Browser** from the menu or right click to select it.

e. Select **Application defined Mbeans –> oracle.iam –> Server: WLS_OIM1 –> Application: oim –> XML Config –> Config –> XMLConfig.SOAConfig –> SOAConfig**

f. Update the value for the **Rmiurl** attribute with the host and port of the new SOA server. Click **Apply** to save the changes.

g. The **Rmiurl** attribute is used for accessing SOA EJBs deployed on SOA Managed Servers. This is the application server URL. The following is an example value for this attribute:

```
cluster:t3://soa_cluster
```

13. Restart the WebLogic Administration Server as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

14. Start and test the new Managed Server from the Administration Console.

a. Shut down the existing Managed Servers in the cluster.

b. Ensure that the newly created Managed Server, `WLS_SOA`*n*, is up.

c. Access the application on the newly created Managed Server (`http://vip:port/soa-infra`). The application should be functional.

15. Configure the newly created managed server for server migration. Follow the steps in Section 14.6, "Configuring Server Migration Targets" to configure server migration.

16. Test server migration for this new server. Follow these steps from the node where you added the new server:

a. Stop the `WLS_SOA`*n* Managed Server.

To do this, run:

```
kill -9 pid
```

on the process ID (PID) of the Managed Server. You can identify the PID of the node using

```
ps -ef | grep WLS_SOAn
```

b. Watch the Node Manager Console. You should see a message indicating that the floating IP address for WLS_SOA1 has been disabled.

c. Wait for the Node Manager to try a second restart of WLS_SOA*n*. Node Manager waits for a fence period of 30 seconds before trying this restart.

d. Once Node Manager restarts the server, stop it again. Now Node Manager should log a message indicating that the server will not be restarted again locally.

17.4.1.4 Scaling Up Oracle HTTP Server

The web tier already has a node running an instance of the Oracle HTTP Server. The existing Oracle HTTP Server binaries can be used for creating the new Oracle HTTP Server instance. To scale up the Oracle HTTP Server, follow the steps in Chapter 10, "Installing and Configuring Oracle Web Tier for an Enterprise Deployment."

1. Use the Oracle Fusion Middleware 11*g* Web Tier Utilities Configuration Wizard to scale up the topology, as described in Chapter 10, "Installing and Configuring Oracle Web Tier for an Enterprise Deployment."
2. Copy all files created in *ORACLE_INSTANCE*/config/OHS/*component*/moduleconf from the existing web tier configuration to the new one.
3. Reconfigure the load balancer with the host and port information of the new Oracle HTTP Server instance.

17.4.2 Scaling Out the Topology

In scaling out a topology, new servers are added to new nodes. The components in the Oracle Identity Management topology described in this manual can be scaled out by adding a new server instance to a new node.

Some of the procedures described in this section show you how to create a new WebLogic managed server. If you add a new managed server to your topology, after adding the managed server you must update your Oracle HTTP Server configuration files (on all nodes) and add the new server to the existing WebLogic cluster directives.

For example if you add a new Oracle Access Manager server, you must update *WEB_ORACLE_INSTANCE*/config/OHS/*component_name*/moduleconf/sso_vh.conf to include the new managed server.

Update sso_vh.conf as follows:

```
<Location /oam>
SetHandler weblogic-handler
WebLogicCluster
IDMHOST1.mycompany.com:14100,IDMHOST2.mycompany.com:14100,IDMHOST3.mycompany.com:1
4100
</Location>
```

Once you have updated sso_vh.conf, restart the Oracle HTTP server(s) as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components." Oracle recommends that you do this sequentially to prevent loss of service.

This section contains the following topics:

- Section 17.4.2.1, "Scaling Out Oracle Unified Directory"
- Section 17.4.2.2, "Scaling Out Oracle Access Manager 11g"
- Section 17.4.2.3, "Scaling Out Oracle Identity Manager"
- Section 17.4.2.4, "Scaling Out the Oracle HTTP Server"

17.4.2.1 Scaling Out Oracle Unified Directory

The directory has two Oracle Unified Directory nodes, IDMHOST1 and IDMHOST2, each running an Oracle Unified Directory instance. The Oracle Unified Directory instances can be scaled out by adding a new node to the configuration, as follows:

1. Follow the steps in Section 7.3, "Installing Oracle Unified Directory" to install the OUD binaries on the new host.
2. Follow the steps in Section 7.4.3, "Configuring an Additional Oracle Unified Directory Instance on IDMHOST2."
3. Reconfigure the load balancer with the host and port information of the new Oracle Unified Directory instance.

17.4.2.2 Scaling Out Oracle Access Manager 11g

Scale out is very similar to scale up but first requires the software to be installed on the new node.

Use the existing installations in shared storage for creating the new Managed Servers. You do not need to install WebLogic Server or Identity Management binaries in a new location but you do need to run pack and unpack to bootstrap the domain configuration in the new node.

> **Note:** If you are using shared storage, allow the new host access to that shared storage area.

1. On the new node, mount the existing Middleware home, which should include the Identity and Access Management installation and the domain directory, and ensure that the new node has access to this directory, just like the rest of the nodes in the domain.
2. To update the Middleware home list, create (or edit, if another WebLogic installation exists in the node) the *IAM_MW_HOME*`/bea/beahomelist` file and add *IAM_MW_HOME*`/product/fmw` to it.
3. Log in to the Oracle WebLogic Server Administration Console at the URL listed in Section 17.2, "About Identity Management Console URLs."
4. From the Domain Structure window of the Oracle WebLogic Server Administration Console, expand the **Environment** node and then **Servers**. The Summary of Servers page appears.
5. Click **Lock & Edit** from the Change Center menu.
6. Select an existing server on the host you want to extend, for example: **WLS_OAM1**.
7. Click **Clone**.
8. Enter the following information:
   - **Server Name**: A new name for the server, for example: `WLS_OAM3`.
   - **Server Listen Address**: The name of the host on which the Managed Server runs.
   - **Server Listen Port**: The port the new Managed Server uses. This port must be unique within the host.
9. Click **OK**.
10. Click the newly created server **WLS_OAM3**.
11. Set the SSL listen port. This should be unique on the host that the Managed Server runs on.
12. Click **Save**.
13. Disable host name verification for the new Managed Server. Before starting and verifying the WLS_OAM3 Managed Server, you must disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Oracle WebLogic Administration Server and the Node Manager in IDMHOST*n*.

    If the source server from which the new one was cloned had already disabled host name verification, these steps are not required, as the host name verification

settings was propagated to the cloned server. To disable host name verification, proceed as follows:

a. Log in to **Oracle WebLogic Server Administration Console**.

b. Expand the **Environment** node in the Domain Structure pane.

c. Click **Servers**. The Summary of Servers page appears.

d. Select **WLS_OAM3** in the Names column of the table. The Settings page for server appears.

e. Click the **SSL** tab.

f. Click **Advanced**.

g. Set **Hostname Verification** to **None**.

h. Click **Save**.

14. Click **Activate Changes** from the Change Center menu.

15. Restart the WebLogic Administration Server as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

16. Pack the domain on IDMHOST1 using the command:

```
pack.sh -domain=ASERVER_HOME -template=/tmp/IDMDomain.jar -template_name="OAM
Domain" -managed=true
```

The `pack.sh` script is located in *ORACLE_COMMON_HOME*`/common/bin`.

17. Unpack the domain on the new host using the command:

```
unpack.sh -domain=MSERVER_HOME -template=/tmp/IDMDomain.jar -app_dir=MSERVER_
HOME/applications
```

The `unpack.sh` script is located in *ORACLE_COMMON_HOME*`/common/bin`.

18. Start Node Manager and update the property file.

a. Start and stop Node Manager as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

b. Run the script `setNMProps.sh`, which is located in *ORACLE_COMMON_HOME*`/common/bin`, to update the node manager properties file, for example:

```
cd ORACLE_COMMON_HOME/common/bin
./setNMProps.sh
```

c. Start Node Manager once again as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

Register the new Managed Server with Oracle Access Manager. The new Managed Server now must be configured as an Oracle Access Manager server. You do this from the Oracle OAM console, as follows:

1. Log in to the OAM console as the `oamadmin` user. Use the URL listed in Section 17.2, "About Identity Management Console URLs."

2. Click the **System Configuration** tab.

3. Click **Server Instances**.

4. Select **Create** from the Actions menu.

5. Enter the following information:

- **Server Name**: `WLS_OAM3`
- **Host**: Host that the server is running on, `IDMHOST3`.
- **Port**: Listen port that was assigned when the Managed Server was created.
- **OAM Proxy Port**: Port you want the Oracle Access Manager proxy to run on. This is unique for the host.
- **Proxy Server ID:** `AccessServerConfigProxy`
- **Mode**: Set to `Open` or `Simple`, depending on the mode your existing Oracle Access Manager servers are operating in.

6. Click **Apply**.

Add the newly created Oracle Access Manager server to all WebGate profiles that might be using it, such as `Webgate_IDM` and `IAMSuiteAgent`.

For example, to add the Oracle Access Manager server to Webgate_IDM, access the OAM console at the URL listed in Section 17.2, "About Identity Management Console URLs." Then proceed as follows:

1. Log in as the Oracle Access Manager admin user you created in Section 9.4, "Preparing the Identity Store."
2. Click the **System Configuration** tab.
3. Expand **Access Manager** - **SSO Agents** - **OAM Agents**.
4. Click the open folder icon, then click **Search**.

   You should see the WebGate agent **Webgate_IDM**.
5. Click the agent **Webgate_IDM**.
6. Select **Edit** from the **Actions** menu.
7. Click **+** in the **Primary Server** list (or the secondary server list if this is a secondary server).
8. Select the newly created managed server from the **Server** list.
9. Set **Max Connections** to `4`.
10. Click **Apply**

Repeat Steps 5 through 10 for `IAMSuiteAgent` and other WebGates that are in use.

Update the Web Tier. Now that the new Managed Server has been created and started, the web tier starts to direct requests to it. Best practice, however, is to inform the web server that the new Managed Server has been created.

You do this by updating the file `sso_vh.conf` on each of the web tiers. This file resides in the directory: *`WEB_ORACLE_INSTANCE`*`/config/OHS/`*`component name`*`/moduleconf`.

Add the new server to the `WebLogicCluster` directive in the file, for example, change:

```
<Location /oam>
   SetHandler weblogic-handler
   WebLogicCluster IDMHOST1.mycompany.com:14100,IDMHOST2.mycompany.com:14100
</Location>
```

to:

```
<Location /oam>
```

```
   SetHandler weblogic-handler
   WebLogicCluster
IDMHOST1.mycompany.com:14100,IDMHOST2.mycompany.com:14100,IDMHOST3.mycompany.com:1
4100
</Location>
```

17.4.2.3 Scaling Out Oracle Identity Manager

When you scale out the topology, you add new Managed Servers configured with OIM and SOA to new nodes.

Before performing the steps in this section, check that you meet these requirements:

- There must be existing nodes running Managed Servers configured with OIM and SOA within the topology.
- The new node can access the existing home directories for WebLogic Server, OIM, and SOA.

  Use the existing installations in shared storage for creating a new WLS_SOA or WLS_OIM Managed Server. You do not need to install WebLogic Server, OIM, or SOA binaries in a new location but you do need to run pack and unpack to bootstrap the domain configuration in the new node.

Notes:

- If there is no existing installation in shared storage, installing WebLogic Server, IAM, and SOA in the new nodes is required as described in Section 12.9, "Configuring Oracle Coherence for Deploying Composites."
- When an *ORACLE_HOME* or *WL_HOME* is shared by multiple servers in different nodes, Oracle recommends keeping the Oracle Inventory and Middleware home list in those nodes updated for consistency in the installations and application of patches. To update the oraInventory in a node and attach an installation in a shared storage to it, use:

  OIM_ORACLE_HOME/oui/bin/attachHome.sh

- To update the Middleware home list to add or remove a *WL_HOME*, edit the *user_home*/bea/beahomelist file. See the following steps.

Follow these steps for scaling out the topology:

1. On the new node, mount the existing Middleware home, which should include the Oracle Fusion Middleware installation and the domain directory, and ensure that the new node has access to this directory, just like the rest of the nodes in the domain.
2. To attach *ORACLE_BASE* in shared storage to the local Oracle Inventory, execute the following command:

   ```
   cd IAM_MW_HOME/product/fmw/iam/oui/bin
   /attachHome.sh -jreLoc JAVA_HOME
   ```

3. To update the Middleware home list, create (or edit, if another WebLogic installation exists in the node) the *IAM_MW_HOME*/bea/beahomelist file and add *IAM_MW_HOME*/product/fmw to it.

4. Log in to the Oracle WebLogic Administration Console at the URL listed in Section 17.2, "About Identity Management Console URLs."
5. Create a new machine for the new node to be used, and add the machine to the domain.
6. Update the machine's Node Manager's address to map the IP address of the node that is being used for scale out.
7. Use the Oracle WebLogic Server Administration Console to clone the managed servers `WLS_OIM1` and `WLS_SOA1` into new Managed Servers. Name them `WLS_SOA`*n* and `WLS_OIM`*n*, respectively, where *n* is a number.

   Note: These steps assume that you are adding a new server to node *n*, where no Managed Server was running previously.

8. Assign the host names or IP addresses to the listen addresses of the new Managed Servers.
9. Assign the managed server to the newly created machine.
10. If you are planning to use server migration for this server (which Oracle recommends) this should be the VIP address (also called a floating IP address) for the server. This VIP address should be different from the one used for the existing Managed Server.
11. Create JMS Servers for SOA, Oracle Identity Manager, UMS, and BPM on the new Managed Server. You do this as follows:
    a. Log in to the WebLogic Administration Console and navigate to **Services -> Messaging -> JMS Servers**.
    b. Click **New**.
    c. Enter a value for **Name**, such as `BPMJMSServer_auto_3`.
    d. Click **Create New File Store**.
    e. Select `FileStore` from the list
    f. Click **Next**.
    g. Enter a value for **Name**, such as `BPMJMSFileStore_3`
    h. Enter the following values:

       Target: The new server you are creating.

       Directory: *`ASERVER_HOME`*`/jms/BPMJMSFileStore_3`
    i. Click **OK**.
    j. When you are returned to the JMS Server screen, select the newly created file store from the list.
    k. Click **Next**.
    l. On the next screen set the Target to the server you are creating.
    m. Click **Finish**.

    Create the following JMS Queues depending on the managed server you are creating:

Server	JMS Server Name	File Store Name	Directory	Target
WLS_SOA*n*	BPMJMSServer_auto_*n*	BPMJMSFileStore_*n*	*ASERVER_HOME*/jms/BPMJMSFileStore_*n*	WLS_SOA*n*
WLS_SOA*n*	SOAJMSServer_auto_*n*	SOAJMSFileStore_*n*	*ASERVER_HOME*/jms/SOAJMSFileStore_*n*	WLS_SOA*n*
WLS_SOA*n*	UMSJMSServer_auto_*n*	UMSJMSFileStore_*n*	*ASERVER_HOME*/jms/UMSJMSFileStore_*n*	WLS_SOA*n*
WLS_OIM*n*	OIMJMSServer_auto_*n*	OIMJMSFileStore_*n*	*ASERVER_HOME*/jms/OIMJMSFileStore_*n*	WLS_OIM*n*
WLS_OIM*n*	JRFWSAsyncJmsServer_auto_*n*	JRFWSAsyncJmsServer_*n*	*ASERVER_HOME*/jms/JRFWSAsyncJmsServerJMSFileStore_*n*	WLS_OIM*n*

12. Add the newly created JMS Queues to the existing JMS Modules by performing the following steps:
    a. Log in to the WebLogic Administration Console
    b. Navigate to **Services -> Messaging -> JMS Modules**
    c. Click a JMSModule, such as **SOAJMSModule**
    d. Click the **Sub Deployments** tab.
    e. Click the listed sub deployment.

    Note: This subdeployment module name is a random name in the form of **JMSServerName*XXXXXX*** resulting from the Configuration Wizard JMS configuration.

    f. Assign the newly created JMS server, for example **SOAJMSServer_auto_3**.
    g. Click **Save**.

    Perform this for each of the JMS modules listed in the following table:

JMS Module	JMS Server
BPMJMSModule	BPMJMSServer_auto_*n*
JRFWSAsyncJmsModule	JRFWSAsyncJmsServer_auto_*n*
OIMJMSModule	OIMJMSServer_auto_*n*
SOAJMSModule	SOAJMSServer_auto_*n*
UMSJMSSystemResource	UMSJMSServer_auto_*n*

13. Click **Activate Changes** from the Change Center menu.
14. Use the `pack`, `scp`, and `unpack` commands to create a backup of the domain. To do this perform the following steps:
    a. Run the `pack` command on IDMHOST1 to create a template:

    ```
    cd ORACLE_COMMON_HOME/common/bin
    ```

```
./pack.sh -managed=true -domain=ASERVER_HOME -template=/templates/oim_
domain.jar -template_name="OIM Domain"
```

b. Run the `scp` command on IDMHOST1 to copy the template file created to IDMHOST*n*. For example:

```
scp /templates/oim_domain.jar IDMHOSTN:/templates/oim_domain.jar
```

c. Run the `unpack` command on IDMHOST*n* to unpack the template in the Managed Server domain directory as follows:

```
cd ORACLE_COMMON_HOME/oracle_common/bin
./unpack.sh -domain=MSERVER_HOME -template=/templates/oim_domain.jar -app_
dir=MSERVER_HOME/applications
```

15. Configure Oracle Coherence, as described in Section 12.9, "Configuring Oracle Coherence for Deploying Composites."

Note: Only the **localhost** field must be changed for the server. Replace the localhost with the listen address of the new server added:

---

16. Update the SOA host and port using Oracle Enterprise Manager Fusion Middleware Control. Follow these steps:

a. Open a browser and go to Oracle Enterprise Manager Fusion Middleware Control at the URL listed in Section 17.2, "About Identity Management Console URLs."

b. Log in to Oracle Enterprise Manager Fusion Middleware Control using the WebLogic adminsitration account `weblogic_idm`.

Note: At least one of the Oracle Identity Manager Managed Servers must be running for when these steps are executed.

---

c. Select **Farm_IDMDomain** –> **Identity and Access** –> **OIM** –> **oim(11.1.2.0.0).**

d. Select **System MBean Browser** from the menu or right click to select it.

e. Select **Application defined Mbeans** –> **oracle.iam** –> **Server: WLS_OIM1** –> **Application: oim** –> **XML Config** –> **Config** –> **XMLConfig.SOAConfig** –> **SOAConfig**

f. Update the value for the **Rmiurl** attribute with the host and port of the new SOA server. Click **Apply** to save the changes.

g. The **Rmiurl** attribute is used for accessing SOA EJBs deployed on SOA Managed Servers. This is the application server URL. The following is an example value for this attribute:

```
cluster:t3://soa_cluster
```

17. Add new server to Cluster Address List.

a. In the WebLogic Administration Console, expand the **Environment** node in the Domain Structure window.

b. Click **Clusters**. The Summary of Clusters page appears.

c. Click on the cluster name **oim_cluster**. The Cluster Properties page is displayed.

d. Add the new OIM managed server to the cluster address list.

e. Click **Save**.

f. Repeat this step to add any new SOA managed servers to the soa_cluster address list.

18. Configure TX persistent store for the new server. This should be a location visible from other nodes as indicated in the recommendations about shared storage.

    From the Administration Console, select **Server_name** > **Services** tab. Under **Default Store**, in **Directory**, enter the path to the folder where you want the default persistent store to store its data files.

19. Disable host name verification for the new Managed Server. Before starting and verifying the WLS_SOA*n* and WLS_OIM*n* Managed Server, you must disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Oracle WebLogic Administration Server and the Node Manager in IDMHOST*n*. If the source server from which the new one has been cloned had already disabled host name verification, these steps are not required (the host name verification setting is propagated to the cloned server).

    To disable host name verification for WLS_SOA*n*:

    a. Expand the **Environment** node in the **Domain Structure** window.

    b. Log in to **Oracle WebLogic Server Administration Console**.

    c. Click **Servers**. The Summary of Servers page appears.

    d. Select **WLS_SOA*n*** in the **Names** column of the table.

       The Settings page for server appears.

    e. Click the **SSL** tab.

    f. Click **Advanced**.

    g. Set **Hostname Verification** to `None`.

    h. Click **Save**.

    To disable host name verification for WLS_OIM*n*, repeat the same steps, but select **WLS_OIM*n*** in the **Names** column in Step d.

20. Click **Activate Changes** from the Change Center menu.

21. Start the Node Manager on the new node. To start the Node Manager, use the installation in shared storage from the existing nodes, and start Node Manager by passing the host name of the new node as a parameter as follows:

    ```
    WL_HOME/server/bin/startNodeManager.sh
    ```

22. Start and test the new Managed Server from the Oracle WebLogic Server Administration Console:

    a. Shut down all the existing Managed Servers in the cluster.

    b. Ensure that the newly created Managed Servers, WLS_SOA*n* and WLS_OIM*n*, are running.

    c. Access the applications on the newly created Managed Servers (`http://vip:port/soa-infra` and `http://vip:port/oim`). The applications should be functional.

23. Configure server migration for the new Managed Server.

Note: Since this new node is using an existing shared storage installation, the node is already using a Node Manager and an environment configured for server migration that includes netmask, interface, `wlsifconfig` script superuser privileges. The floating IP addresses for the new Managed Servers are already present in the new node.

Configure server migration following these steps:

a. Log in to the Administration Console.

b. In the left pane, expand **Environment** and select **Servers**.

c. Select the server (represented as hyperlink) for which you want to configure migration from the **Names** column of the table. The Setting page for that server appears.

d. Click the **Migration** tab.

e. In the **Available** field, in the **Migration Configuration** section, select the machines to which to enable migration and click the right arrow.

Note: Specify the least-loaded machine as the migration target for the new server. The required capacity planning must be completed so that this node has enough available resources to sustain an additional Managed Server.

f. Select the **Automatic Server Migration Enabled** option. This enables the Node Manager to start a failed server on the target node automatically.

g. Click **Save**.

h. Restart the Administration Server, Managed Servers, and Node Manager.

i. Test server migration for the new servers WLS_SOA*n* and WLS_OIM*n*, as follows.

1. Determine the PID of the WLS_SOA*n* Managed Server by typing

```
ps -ef | grep WLS_SOAn
```

2. From the node where you added the new server, abruptly stop the WLS_SOA*n* Managed Server by typing:

```
kill -9 pid
```

3. Watch the Node Manager Console. You should see a message indicating that floating IP address for WLS_SOA1 has been disabled.

4. Wait for the Node Manager to try a second restart of WLS_SOA*n*. Node Manager waits for a fence period of 30 seconds before trying this restart.

5. Once Node Manager restarts the server, stop it again. Now Node Manager should log a message indicating that the server will not be restarted again locally.

6. Repeat Steps 1-5 for WLS_OIM*n*.

17.4.2.4 Scaling Out the Oracle HTTP Server

The web tier has two nodes each running an instance of the Oracle HTTP Server. The Oracle HTTP Server components can be scaled out by adding a new node configured to run Oracle HTTP Server to the web tier. To scale out Oracle HTTP Server, proceed as follows:

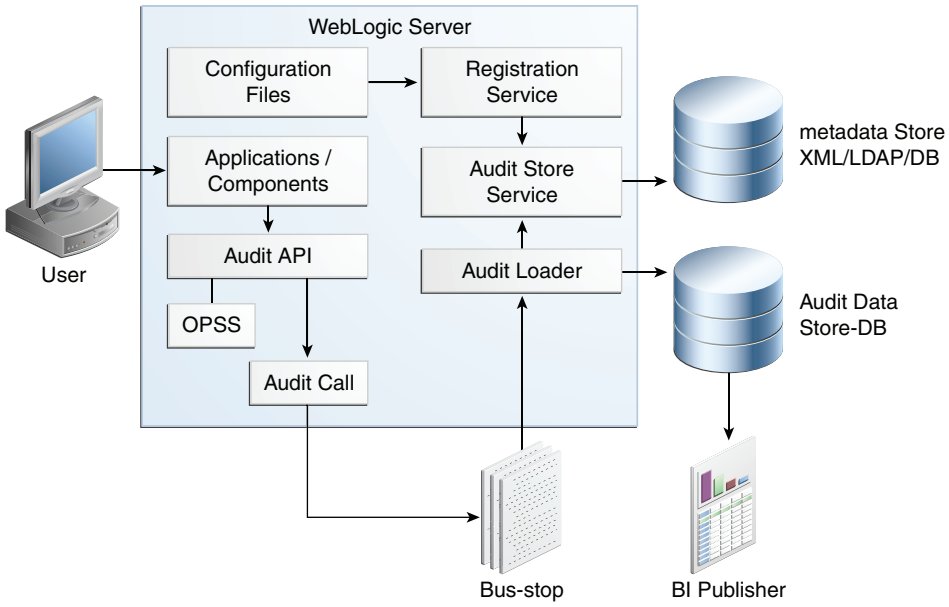
1. Follow the steps in Section 10.2.3, "Installing Oracle HTTP Server." Alternatively, on the new node, mount the existing Middleware home, if you are using shared storage.
2. Follow the steps in Chapter 10, "Installing and Configuring Oracle Web Tier for an Enterprise Deployment."
3. Copy all files created in *WEB_ORACLE_INSTANCE*/config/OHS/*component_name*/moduleconf from the existing web tier configuration to the new one.
4. If you have enabled Single Sign-on in the topology, you must update the Web Tier configuration for Single Sign-on as described in Section 15.7, "Installing and Configuring WebGate 11g."
5. Reconfigure the load balancer with the host and port information of the new Oracle HTTP Server instance.

17.5 Auditing Identity Management

Oracle Fusion Middleware Audit Framework is a new service in Oracle Fusion Middleware 11*g*, designed to provide a centralized audit framework for the middleware family of products. The framework provides audit service for platform components such as Oracle Platform Security Services (OPSS) and Oracle Web Services. It also provides a framework for JavaEE applications, starting with Oracle's own JavaEE components. JavaEE applications are able to create application-specific audit events. For non-JavaEE Oracle components in the middleware such as C or JavaSE components, the audit framework also provides an end-to-end structure similar to that for JavaEE applications.

Figure 17–1 is a high-level architectural diagram of the Oracle Fusion Middleware Audit Framework. For more information, see *Oracle Fusion Middleware Application Security Guide*.

Figure 17–1 Audit Event Flow



The Oracle Fusion Middleware Audit Framework consists of the following key components:

- Audit APIs

  These are APIs provided by the audit framework for any audit-aware components integrating with the Oracle Fusion Middleware Audit Framework. During run-time, applications may call these APIs where appropriate to audit the necessary information about a particular event happening in the application code. The interface enables applications to specify event details such as username and other attributes needed to provide the context of the event being audited.

- Audit Events and Configuration

  The Oracle Fusion Middleware Audit Framework provides a set of generic events for convenient mapping to application audit events. Some of these include common events such as authentication. The framework also enables applications to define application-specific events.

  These event definitions and configurations are implemented as part of the audit service in Oracle Platform Security Services. Configurations can be updated through Enterprise Manager (UI) and WLST (command-line tool).

- The Audit Bus-stop

  Bus-stops are local files containing audit data before they are pushed to the audit repository. In the event where no database repository is configured, these bus-stop files can be used as a file-based audit repository. The bus-stop files are simple text files that can be queried easily to look up specific audit events. When a DB-based repository is in place, the bus-stop acts as an intermediary between the component and the audit repository. The local files are periodically uploaded to the audit repository based on a configurable time interval.

- Audit Loader

  As the name implies, audit loader loads the files from the audit bus-stop into the audit repository. In the case of platform and JavaEE application audit, the audit

loader is started as part of the JavaEE container start-up. In the case of system components, the audit loader is a periodically spawned process.

- Audit Repository

  Audit Repository contains a pre-defined Oracle Fusion Middleware Audit Framework schema, created by Repository Creation Utility (RCU). Once configured, all the audit loaders are aware of the repository and upload data to it periodically. The audit data in the audit repository is expected to be cumulative and grow over time. Ideally, this should not be an operational database used by any other applications - rather, it should be a standalone RDBMS used for audit purposes only. In a highly available configuration, Oracle recommends that you use an Oracle Real Application Clusters (Oracle RAC) database as the audit data store.

- Oracle Business Intelligence Publisher

  The data in the audit repository is exposed through pre-defined reports in Oracle Business Intelligence Publisher. The reports enable users to drill down the audit data based on various criteria. For example:

  - Username
  - Time Range
  - Application Type
  - Execution Context Identifier (ECID)

For more introductory information for the Oracle Fusion Middleware Audit Framework, see the "Introduction to Oracle Fusion Middleware Audit Framework" chapter in the *Oracle Fusion Middleware Application Security Guide*.

For information on how to configure the repository for Oracle Fusion Middleware Audit Framework, see the "Configuring and Managing Auditing" chapter in the *Oracle Fusion Middleware Application Security Guide*.

The EDG topology does not include Oracle Fusion Middleware Audit Framework configuration. The ability to generate audit data to the bus-stop files and the configuration of the audit loader are available once the products are installed. The main consideration is the audit database repository where the audit data is stored. Because of the volume and the historical nature of the audit data, it is strongly recommended that customers use a separate database from the operational store or stores being used for other middleware components.

17.6 Performing Backups and Recoveries

You can use the UNIX `tar` command for most backups. Typical usage is:

```
tar -cvpf BACKUP_LOCATION/backup_file.tar directories
```

You can use the UNIX `tar` command for recovery. Typical usage is:

```
tar -xvf BACKUP_LOCATION/backup_file.tar
```

For database backup and recovery, you can use the database utility RMAN. See the *Oracle Database Backup and Recovery Reference* for more information on using this command.

This section contains the following topics:

- Section 17.6.1, "Performing Baseline Backups"

- Section 17.6.2, "Performing Runtime Backups"
- Section 17.6.3, "Performing Backups During Installation and Configuration"

17.6.1 Performing Baseline Backups

Perform baseline backups when building a system and when applying patches that update static artifacts, such as the Oracle binaries.

After performing a baseline backup, also perform a runtime backup.

Table 17–2 Static Artifacts to Back Up in the Identity Management Enterprise Deployment

Type	Location	Type	Comments
Database	ORACLE_HOME Grid ORACLE_HOME	File Copy	See *Oracle Database Backup and Recovery Reference* for more information.
Middleware Homes	*IAM_MW_HOME* *OIM_MW_HOME*[1] *DIR_MW_HOME*[2]	File Copy	Located on shared storage, so only requires backup from one host.
Web Middleware Home	Middleware Home: *WEB_MW_HOME*	File Copy	Located on local storage. Back up each WEBHOST individually.
Install Related	*ORACLE_BASE*/orainventory /etc/oratab, /etc/oraInst.loc *HOME*/bea/beahomelist	File Copy	Located on local storage. Back up each host individually.
Load Balancer		File Copy	Back up the load balancer configuration. See your vendor documentation.
Servers			Back up the operating systems. See your vendor documentation.

[1] Split domain only

[2] Middleware home of your Oracle Internet Directory or Oracle Virtual Directory Installation. Installation and configuration of Oracle Identity Management is not covered in this Guide.

Note: It is also recommended that you back up your load balancer configuration. Refer to your vendor documentation on how to do this.

For more information on backup and recovery of Oracle Fusion Middleware components, see *Oracle Fusion Middleware Administrator's Guide*.

17.6.2 Performing Runtime Backups

Perform runtime backups on an ongoing basis. These backups contain information on items that can change frequently, such as data in the database, domain configuration information, and identity information in LDAP directories.

Table 17–3 Run-Time Artifacts to Back Up in the Identity Management Enterprise Deployments

Type	Location	Type	Comments
Database	Data	RMAN	See *Oracle Database Backup and Recovery Reference* for more information.
Oracle Unified Directory	*OUD_ORACLE_INSTANCE* Oracle Unified Directory data	File Copy Oracle Unified Directory backup	Located on local storage. Back up each LDAPHOST individually. See *Oracle Fusion Middleware Administrator's Guide for Oracle Unified Directory* for more information. When performing a cold backup, backing up *OUD_INSTANCE_HOME* is sufficient.
Oracle Internet Directory	Oracle Internet Directory instance Oracle Internet Directory data in database	File Copy RMAN	See *Oracle Fusion Middleware Administrator's Guide* for more information.
Oracle Virtual Directory	Oracle Virtual Directory Instance	File Copy	See *Oracle Fusion Middleware Administrator's Guide* for more information.
Third Party LDAP	Binaries LDAP data		See your vendor documentation.
Domain Home	*ASERVER_HOME* *MSERVER_HOME*	File Copy	*ASERVER_HOME* is on shared storage, so only requires backup from one host. *MSERVER_HOME* is on local storage. Back up each host individually.
Oracle HTTP Server	*WEB_ORACLE_INSTANCE*	File Copy	*WEB_ORACLE_INSTANCE* is on local storage. Back up each WEBHOST individually.

17.6.3 Performing Backups During Installation and Configuration

It is an Oracle best practices recommendation to create a backup after successfully completing the installation and configuration of each tier, or at another logical point. Create a backup after verifying 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 more details, see the *Oracle Fusion Middleware Administrator's Guide*.

For information on database backups, refer to the *Oracle Database Backup and Recovery User's Guide*.

This section contains the following topics:

- Section 17.6.3.1, "Backing Up Middleware Home"
- Section 17.6.3.2, "Backing Up LDAP Directories"
- Section 17.6.3.3, "Backing Up the Database"
- Section 17.6.3.4, "Backing Up the WebLogic Domain"
- Section 17.6.3.5, "Backing Up the Web Tier"

17.6.3.1 Backing Up Middleware Home

Back up the middleware home whenever you create a new one or add components to it.

17.6.3.2 Backing Up LDAP Directories

Whenever you perform an action which updates the data in LDAP, back up the directory contents.

This section contains the following topics:

- Section 17.6.3.2.1, "Backing Up Oracle Unified Directory"
- Section 17.6.3.2.2, "Backing up Oracle Internet Directory"
- Section 17.6.3.2.3, "Backing up Oracle Virtual Directory"
- Section 17.6.3.2.4, "Backing Up Third-Party Directories"

17.6.3.2.1 Backing Up Oracle Unified Directory To backup Oracle Unified Directory, perform the following steps:

1. Shut down the Oracle Unified Directory Instances as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."
2. Back up Oracle Unified Directory instance directories on each host.
3. Restart the Oracle Unified Directory instances as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

17.6.3.2.2 Backing up Oracle Internet Directory To back up an Oracle Internet Directory instance:

1. Shut down the instance using `opmnctl` located under the *`OID_ORACLE_INSTANCE`*`/bin` directory:

   ```
   OID_ORACLE_INSTANCE/bin/opmnctl stopall
   ```

2. Back up the Database hosting the Oracle Internet Directory data and the Oracle Internet Directory instance home on each host.
3. Start up the instance using `opmnctl` located under the *`OID_ORACLE_INSTANCE`*`/bin` directory:

   ```
   OID_ORACLE_INSTANCE/bin/opmnctl startall
   ```

17.6.3.2.3 Backing up Oracle Virtual Directory To back up an Oracle Virtual Directory instance:

1. Shut down the instance using `opmnctl` located under the *`OVD_ORACLE_INSTANCE`*`/bin` directory:

   ```
   OVD_ORACLE_INSTANCE/bin/opmnctl stopall
   ```

2. Back up the Oracle Virtual Directory Instance home on each LDAP host.
3. Start up the instance using `opmnctl` located under the *`OVD_ORACLE_INSTANCE`*`/bin` directory:

   ```
   OVD_ORACLE_INSTANCE/bin/opmnctl startall
   ```

17.6.3.2.4 Backing Up Third-Party Directories Refer to your operating system vendor's documentation for information about backing up directories.

17.6.3.3 Backing Up the Database

Whenever you create add a component to the configuration, back up the IDMDB database. Perform this backup after creating domains or adding components such as Access Manager.

17.6.3.4 Backing Up the WebLogic Domain

To back up the WebLogic domain, perform these steps:

1. Shut down the WebLogic administration server and any managed servers running in the domain as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."
2. Back up the *ASERVER_HOME* directory from shared storage.
3. Back up the *MSERVER_HOME* directory from each host.

17.6.3.5 Backing Up the Web Tier

To back up the web tier, perform these steps:

1. Shut down the Oracle HTTP Server as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."
2. Back up the *WEB_ORACLE_INSTANCE*.
3. Start the Oracle HTTP Server as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."

17.7 Patching Enterprise Deployments

This section describes how to apply an Oracle Fusion Middleware patch file and how to patch Oracle Identity Management components with minimal down time.

This section contains the following topics:

- Section 17.7.1, "Patching an Oracle Fusion Middleware Source File"
- Section 17.7.2, "Patching Identity and Access Management"
- Section 17.7.3, "Patching Oracle Unified Directory Components"

17.7.1 Patching an Oracle Fusion Middleware Source File

For information on patching an Oracle Fusion Middleware source file, see the *Oracle Fusion Middleware Administrator's Guide*.

17.7.2 Patching Identity and Access Management

In a single domain topology, apply patches as follows:

IDMDomain MW_HOME

- Common patches
- Oracle Access Manager Patches
- Oracle Identity Manager Patches

- IDM Tool Patches

17.7.3 Patching Oracle Unified Directory Components

To patch Oracle Identity Management components with minimal down time, it is recommended that you follow these guidelines:

1. Route the LDAP traffic from IDMHOST1 to IDMHOST2.
2. Bring down the Oracle Unified Directory or Oracle Virtual Directory server on the host you want to patch (IDMHOST1).
3. Apply the patch.
4. Restart the Oracle Unified Directory.
5. Test the patch.
6. Route the traffic to LDAPHOST1 again.
7. Verify the applications are working properly.
8. Route the LDAP traffic on IDMHOST2 to IDMHOST1.
9. Repeat Steps 2-5 on IDMHOST2
10. Route the traffic to both hosts on which the patch has been applied (IDMHOST1 and IDMHOST2).

17.8 Preventing Timeouts for SQL

Most of the production deployment involves firewalls. Because database connections are made across firewalls, Oracle recommends that the firewall be configured so that the database connection is not timed out. For Oracle Real Application Clusters (Oracle RAC), the database connections are made on Oracle RAC VIPs and the database listener port. You must configure the firewall so it does not time out these connections. If such a configuration is not possible, set the SQLNET.EXPIRE_TIME=n parameter in the *ORACLE_HOME*`/network/admin/sqlnet.ora` file on the database server, where `n` is the time in minutes. Set this value to less than the known value of the timeout for the network device (that is, a firewall). For Oracle RAC, set this parameter in all of the Oracle home directories.

17.9 Manually Failing Over the WebLogic Administration Server

This section discusses how to fail over the Administration Server to IDMHOST2 and how to fail it back to IDMHOST1.

The same procedure can be applied to each domain you have created.

This section contains the following topics:

- Section 17.9.1, "Failing over the Administration Server to IDMHOST2"
- Section 17.9.2, "Starting the Administration Server on IDMHOST2"
- Section 17.9.3, "Validating Access to IDMHOST2 Through Oracle HTTP Server"
- Section 17.9.4, "Failing the Administration Server Back to IDMHOST1"

17.9.1 Failing over the Administration Server to IDMHOST2

If a node fails, you can fail over the Administration Server to another node. This section describes how to fail over the Administration Server from IDMHOST1 to IDMHOST2.

Assumptions:

- The Administration Server is configured to listen on `ADMINVHN.mycompany.com`, and not on `ANY` address. See Section 8.4, "Running the Configuration Wizard to Create a Domain."
- The Administration Server is failed over from IDMHOST1 to IDMHOST2, and the two nodes have these IP addresses:
  - IDMHOST1: `100.200.140.165`
  - IDMHOST2: `100.200.140.205`
  - ADMINVHN: `100.200.140.206`

    This is the Virtual IP address where the Administration Server is running, assigned to *interface*:*index* (for example, eth1:2), available in IDMHOST1 and IDMHOST2.
- The domain directory where the Administration Server is running in IDMHOST1 is on a shared storage and is mounted also from IDMHOST2.

  Note: NM in IDMHOST2 does not control the domain at this point, since `unpack/nmEnroll` has not been run yet on IDMHOST2. But for the purpose of AdminServer failover and control of the AdminServer itself, Node Manager is fully functional
- Oracle WebLogic Server and Oracle Fusion Middleware Components have been installed in IDMHOST2 as described in previous chapters. That is, the same path for *`IDM_ORACLE_HOME`* and *`MW_HOME`* that exists in IDMHOST1 is available in IDMHOST2.

The following procedure shows how to fail over the Administration Server to a different node, IDMHOST2.

Linux

1. Stop the Administration Server as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."
2. Migrate the IP address to the second node.
   a. Run the following command as root on IDMHOST1 (where *x*:*y* is the current interface used by `ADMINVHN.mycompany.com`):

      ```
      /sbin/ifconfig x:y down
      ```

      For example:

      ```
      /sbin/ifconfig eth0:1 down
      ```

   b. Run the following command on IDMHOST2:

      ```
      /sbin/ifconfig interface:index IP_Address netmask netmask
      ```

      For example:

```
/sbin/ifconfig eth0:1 10.0.0.1 netmask 255.255.255.0
```

Note: Ensure that the netmask and interface to be used match the available network configuration in IDMHOST2.

3. Update routing tables by using `arping`, for example:

```
/sbin/arping -q -U -c 3 -I eth0 10.0.0.1
```

17.9.2 Starting the Administration Server on IDMHOST2

Perform the following steps to start Node Manager on IDMHOST2.

1. On IDMHOST1, unmount the Administration Server domain directory. For example:

```
umount /u01/oracle
```

2. On IDMHOST2, mount the Administration Server domain directory. For example:

```
mount /u01/oracle
```

3. Start Node Manager by using the following commands:

```
cd WL_HOME/server/bin
./startNodeManager.sh
```

4. Stop the Node Manager by killing the Node Manager process.

Note: Starting and stopping Node Manager at this point is only necessary the first time you run Node Manager. Starting and stopping it creates a property file from a predefined template. The next step adds properties to that property file.

5. Run the setNMProps.sh script to set the `StartScriptEnabled` property to `true` before starting Node Manager:

```
cd MW_HOME/oracle_common/common/bin
./setNMProps.sh
```

Note: You must use the `StartScriptEnabled` property to avoid class loading failures and other problems.

6. Start the Node Manager as described in Section 17.1.5.3, "Starting Node Manager for an Administration Server."

7. Start the Administration Server on IDMHOST2.

```
cd ORACLE_COMMON_HOME/common/bin
./wlst.sh
```

Once in the WLST shell, execute the following commands:

```
nmConnect('Admin_User','Admin_Password', 'IDMHOST2','5556',
'IDMDomain','ASERVER_HOME')
nmStart('AdminServer')
```

Note: Use the full path name for *ASERVER_HOME* in the WLST `nmConnect` command.

17.9.3 Validating Access to IDMHOST2 Through Oracle HTTP Server

1. Test that you can access the Oracle WebLogic Server Administration Console at:

   `http://ADMINVHN.mycompany.com/console`

   where `7001` is *WLS_ADMIN_PORT* in Section B.3.

2. Check that you can access and verify the status of components in the Oracle Enterprise Manager at:

   `http://ADMINVHN.mycompany.com/em`

If you are using a split domain topology, perform the same steps to check that you can Access the Administration Server when it is running on IDMHOST2.

17.9.4 Failing the Administration Server Back to IDMHOST1

This step checks that you can fail back the Administration Server, that is, stop it on IDMHOST2 and run it on IDMHOST1. To do this, migrate ADMINVHN back to IDMHOST1 node as described in the following steps.

1. Ensure that the Administration Server is not running. If it is, stop it from the WebLogic console, or by running the command `stopWebLogic.sh` from *ASERVER_HOME*`/bin`.
2. On IDMHOST2, unmount the Administration server domain directory. For example:

   ```
   umount /u01/oracle
   ```

3. On IDMHOST1, mount the Administration server domain directory. For example:

   ```
   mount /u01/oracle
   ```

4. Disable the `ADMINVHN.mycompany.com` virtual IP address on IDMHOST2 and run the following command as `root` on IDMHOST2:

   `/sbin/ifconfig` *x*`:`*y* `down`

   For example:

   ```
   /sbin/ifconfig eth0:1 down
   ```

5. Run the following command on IDMHOST1:

   `/sbin/ifconfig interface:index` *IP_Address* `netmask` *netmask*

   For example:

   ```
   /sbin/ifconfig interface:index 100.200.140.206 netmask 255.255.255.0
   ```

Note: Ensure that the netmask and interface to be used match the available network configuration in IDMHOST1

6. Update routing tables by running

   ```
   /sbin/arping -q -U -c 3 -I interface 100.200.140.206
   ```

   For example, run the following command from IDMHOST1:

   ```
   /sbin/arping -q -U -c 3 -I eth0 100.200.140.206
   ```

7. If Node Manager is not already started on IDMHOST1, start it, as described in Section 17.1, "Starting and Stopping Oracle Identity Management Components."
8. Start the Administration Server again on IDMHOST1.

   ```
   cd ORACLE_COMMON_HOME/common/bin
   ./wlst.sh
   ```

   Once in the WLST shell, execute

   ```
   nmConnect(Admin_User,'Admin_Password, IDMHOST1,'5556','IDMDomain','ASERVER_
   HOME')
   nmStart('AdminServer')
   ```

9. Test that you can access the Oracle WebLogic Server Administration Console at:

   ```
   http://ADMINVHN.mycompany.com:7001/console
   ```

   where `7001` is *WLS_ADMIN_PORT* in Section B.3.

10. Check that you can access and verify the status of components in the Oracle Enterprise Manager at:

    ```
    http://ADMINVHN.mycompany.com:7001/em
    ```

17.10 Troubleshooting

This section describes how to troubleshoot common issues that can arise with the Identity Management enterprise deployment described in this manual.

This section contains the following topics:

- Section 17.10.1, "Troubleshooting Oracle Internet Directory"
- Section 17.10.2, "Troubleshooting Oracle Virtual Directory"
- Section 17.10.3, "Troubleshooting Access Manager 11g"
- Section 17.10.4, "Troubleshooting Oracle Identity Manager"
- Section 17.10.5, "Troubleshooting Oracle SOA Suite"
- Section 17.10.6, "Using My Oracle Support for Additional Troubleshooting Information"

17.10.1 Troubleshooting Oracle Internet Directory

This section describes some common problems that can arise with Oracle Internet Directory and the actions you can take to resolve the problem. It contains the following topics:

- Section 17.10.1.1, "Oracle Internet Directory Server is Not Responsive."
- Section 17.10.1.2, "SSO/LDAP Application Connection Times Out"
- Section 17.10.1.3, "LDAP Application Receives LDAP Error 53 (DSA Unwilling to Perform)"

- Section 17.10.1.4, "TNSNAMES.ORA, TAF Configuration, and Related Issues"

17.10.1.1 Oracle Internet Directory Server is Not Responsive.

Problem

The Oracle Internet Directory server is not responsive. When the load balancing router is configured to send an ICMP message to the LDAP SSL port for monitoring, the Oracle Internet Directory server starting SSL negotiation sometimes hangs, and thus it is required that the load balancing router not use ICMP messages for monitoring the LDAP SSL port.

Solution

Use an alternative such as TCP or the LDAP protocol itself.

Also, monitoring the LDAP non-SSL port is sufficient to detect LDAP availability.

17.10.1.2 SSO/LDAP Application Connection Times Out

Problem

The SSO/LDAP Application connection is lost to Oracle Internet Directory server

Solution

Verify the load balancing router timeout and SSO/Application timeout configuration parameter. The SSO/LDAP application timeout value should be less than LBR IDLE time out.

17.10.1.3 LDAP Application Receives LDAP Error 53 (DSA Unwilling to Perform)

Problem

The LDAP application is receiving LDAP Error 53 (DSA Unwilling to Perform). When one of the database nodes goes down during the middle of the LDAP transaction, the Oracle Internet Directory server sends error 53 to the LDAP client

Solution

To see why the Oracle Internet Directory database node went down, see the Oracle Internet Directory logs in this location:

```
ORACLE_INSTANCE/diagnostics/logs/OID/oidldapd01s*.log
```

17.10.1.4 TNSNAMES.ORA, TAF Configuration, and Related Issues

Problem

Issues involving TNSNAMES.ORA, TAF configuration, and related issues.

Solution

See the *Oracle Database High Availability Overview* manual.

17.10.2 Troubleshooting Oracle Virtual Directory

This section describes some common problems that can arise with Oracle Virtual Directory and the actions you can take to resolve the problem. It contains the following topics:

17.10.2.1 Command Not Found Error When Running SSLServerConfig.sh

Problem

You get a `command not found` error when you run `SSLServerConfig.sh`, for example:

```
./SSLServerConfig.sh: line 169: 20110520125611: command not found
```

Solution

Edit the file `orapki.sh` (on Linux) and remove any blank lines at the end of the file. Save the file and run `SSLServerConfig.sh` again.

17.10.2.2 Oracle Virtual Directory is Not Responsive

Problem

Oracle Virtual Directory is not responsive. When the load balancing router is configured to send an ICMP message to the LDAP SSL port for monitoring, the Oracle Virtual Directory server starting SSL negotiation sometimes hangs, and thus it is required that the load balancing router not use ICMP messages for monitoring the LDAP SSL port.

Solution

Use an alternative such as TCP or the LDAP protocol itself.

Also, monitoring the LDAP non-SSL port is sufficient to detect LDAP availability.

17.10.2.3 SSO/LDAP Application Connection Times Out

Problem

The SSO/LDAP Application connection is lost to the Oracle Virtual Directory server.

Solution

Verify the load balancing router timeout and SSO/Application timeout configuration parameter. The SSO/LDAP application timeout value should be less than LBR IDLE time out.

17.10.2.4 TNSNAMES.ORA, TAF Configuration, and Related Issues

Problem

Issues involving TNSNAMES.ORA, TAF configuration, and related issues.

Solution

See the *Oracle Database High Availability Overview* manual.

17.10.2.5 SSLServerConfig.sh Fails with Error

Problem

When you run `SSLServerConfig.sh` for component OVD, sometime it fails with an error similar to this:

```
>>>Enter password for weblogic:
>>>Enter your keystore name [ovdks1.jks]:
Checking the existence of ovdks1.jks in the OVD...

>>>Failed to configure your SSL server wallet
>>>Please check /scratch/aime1/edgfa/idm//rootCA/keystores/ovd/ks_check.log for
more information
```

In the log file, you see an error message like this:

```
Problem invoking WLST - Traceback (innermost last):
File "/scratch/aime1/edgfa/idm/rootCA/keystores/ovd/ovdssl-check.py", line 8, in ?
File "<iostream>", line 182, in cd
File "<iostream>", line 1848, in raiseWLSTException
WLSTException: Error occured while performing cd : Attribute
oracle.as.ovd:type=component.listenersconfig.sslconfig,name=LDAP SSL
Endpoint,instance=ovd1,component=ovd1 not found. Use ls(a) to view the
attributes
```

Solution

The problem is intermittent. To work around the issue, re-run the script.

17.10.3 Troubleshooting Access Manager 11g

This section describes some common problems that can arise with Access Manager and the actions you can take to resolve the problem. It contains the following topics:

- Section 17.10.3.1, "User Reaches the Maximum Allowed Number of Sessions"
- Section 17.10.3.2, "Policies Do Not Get Created When Oracle Access Manager is First Installed"
- Section 17.10.3.3, "You Are Not Prompted for Credentials After Accessing a Protected Resource"
- Section 17.10.3.4, "Cannot Log In to OAM Console"

17.10.3.1 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 OAM Administration Console.

To modify the configuration by using the OAM Administration Console, proceed as follows:

1. Go to **System Configuration** -> **Common Settings** -> **Session**
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.

17.10.3.2 Policies Do Not Get Created When Oracle Access Manager is First Installed

Problem

The Administration Server takes a long time to start after configuring Oracle Access Manager.

Solution

Tune the OAM database. When the Administration server first starts after configuring Oracle 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.

17.10.3.3 You Are Not Prompted for Credentials After Accessing a Protected Resource

Problem

When you access a protected resource, Oracle 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 IDMDomain have been set. You should have aliases for `IDMDomain`:80, `IDMDomain`:Null, `ADMIN.mycompany.com:80`, and `SSO.mycompany.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 *`ASERVER_HOME`*`/output` to the WebGate Lib directory and that OHS was restarted.

4. When OBAccessClient.xml was first created, the file was not formatted. When the OHS is restarted, reexamine the file to ensure that it is now formatted. OHS gets a new version of the file from Oracle Access Manager when it first starts.
5. Shut down the Oracle Access Manager servers and try to access the protected resource. You should see an error saying Oracle Access Manager servers are not available. If you do not see this error, re-install WebGate.

17.10.3.4 Cannot Log In to OAM Console

Problem

You cannot log in to the OAM 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.

17.10.4 Troubleshooting Oracle Identity Manager

This section describes some common problems that can arise with Oracle Identity Manager and the actions you can take to resolve the problem. It contains the following topics:

- Section 17.10.4.1, "java.io.FileNotFoundException When Running Oracle Identity Manager Configuration"
- Section 17.10.4.2, "ResourceConnectionValidationxception When Creating User in Oracle Identity Manager"

17.10.4.1 java.io.FileNotFoundException When Running Oracle Identity Manager Configuration

Problem

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/oaconfigplan.xml`.
2. Start the configuration again (*`IAM_ORACLE_HOME`*`/bin/config.sh`).

17.10.4.2 ResourceConnectionValidationxception When Creating User in Oracle Identity Manager

Problem

If you are creating a user in Oracle Identity Manager (by logging into Oracle Identity Manager, 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.

17.10.5 Troubleshooting Oracle SOA Suite

This section describes some common problems that can arise with Oracle SOA Suite and the actions you can take to resolve the problem. It contains the following topics:

17.10.5.1 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 Server Administration 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.

17.10.6 Using My Oracle Support for Additional Troubleshooting Information

You can use My Oracle Support (formerly MetaLink) to help resolve Oracle Fusion Middleware problems. My Oracle Support contains several useful troubleshooting resources, such as:

- Knowledge base articles
- Community forums and discussions
- Patches and upgrades
- Certification information

> **Note:** You can also use My Oracle Support to log a service request.

You can access My Oracle Support at https://support.oracle.com.

A

Using Multi Data Sources with Oracle RAC

Oracle recommends using GridLink data sources when developing new Oracle RAC applications. However, if you are using legacy applications and databases that do not support GridLink data sources, refer to the information in this appendix.

This appendix provides the following topics:

- Section A.1, "About Multi Data Sources and Oracle RAC"
- Section A.2, "Typical Procedure for Configuring Multi Data Sources for an EDG Topology"

A.1 About Multi Data Sources and Oracle RAC

A multi data source provides an ordered list of data sources to use to satisfy connection requests. Normally, every connection request to this kind of multi data source is served by the first data source in the list. If a database connection test fails and the connection cannot be replaced, or if the data source is suspended, a connection is sought sequentially from the next data source on the list.

For more information about configuring Multi Data Sources with Oracle RAC, see "Using Multi Data Sources with Oracle RAC" in the *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server*.

A.2 Typical Procedure for Configuring Multi Data Sources for an EDG Topology

You configure data sources when you configure a domain. For example, when you are configuring the initial Administration domain for an Enterprise Deployment reference topology, you use the configuration wizard to define the characteristics of the domain, as well as the data sources.

The procedures for configuring the topologies in this Enterprise Deployment Guide include specific instructions for defining GridLink data sources with Oracle RAC. If you want to use Multi Data Sources instead of GridLink data sources, replace the GridLink instructions with the following:

1. In the Configure JDBC Component Schema screen:
   a. Select the appropriate schemas.
   b. For the RAC configuration for component schemas, **Convert to RAC multi data source**.
   c. Ensure that the following data source appears on the screen with the schema prefix when you ran the Repository Creation Utility.

d. Click **Next**.

2. The Configure RAC Multi Data Sources Component Schema screen appears In this screen, do the following:

   a. Enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU.

      – **Driver:** Select **Oracle driver (Thin) for RAC Service-Instance connections, Versions:10, 11**.

      – **Service Name:** Enter the service name of the database.

      – **Username:** Enter the complete user name (including the prefix) for the schemas.

      – **Password:** Enter the password to use to access the schemas.

   b. Enter the host name, instance name, and port.

   c. Click **Add**.

   d. Repeat this for each Oracle RAC instance.

   e. Click **Next**.

3. In the Test JDBC Data Sources screen, the connections are tested automatically. The **Status** column displays the results. Ensure that all connections were successful. If not, click **Previous** to return to the previous screen and correct your entries.

   Click **Next** when all the connections are successful.

B

Worksheets for Identity Management Topology

This appendix contains worksheets to help you keep track of machine names, IP addresses, directories, and other important data.

We recommend that you open the PDF version if this Guide in a PDF reader and print out this appendix. Update these worksheet as you set up your enterprise deployment.

This chapter contains the following worksheets:

B.1 Hosts, Virtual Hosts, and Virtual IP Addresses for Identity Management

Use this worksheet to record information about hosts and IP addresses.

Table B–1 Hosts, Virtual Hosts, and Virtual IP Addresses

Documented Alias	Type	Your Host Name	IP Address	Operating System and Version
WEBHOST1	Host			
WEBHOST2	Host			
IDMHOST1	Host			
IDMHOST2	Host			
IDMDBHOST1	Database Host			
IDMDBHOST2	Database Host			
OIMADMINVHN	Virtual Host			

Table B–1 (Cont.) Hosts, Virtual Hosts, and Virtual IP Addresses

Documented Alias	Type	Your Host Name	IP Address	Operating System and Version
ADMINVHN	Virtual Host			
SOAHOST1VHN	Virtual Host			
SOAHOST2VHN	Virtual Host			
OIMHOST1VHN	Virtual Host			
OIMHOST2VHN	Virtual Host			
IDMINTERNAL.mycompany.com	Load Balancer Virtual Name			
SSO.mycompany.com	Load Balancer Virtual Name			
ADMIN.mycompany.com	Load Balancer Virtual Name			
OIMADMIN.mycompany.com	Load Balancer Virtual Name			
IDSTORE.mycompany.com	Load Balancer Virtual Name			
IDMDomain	Domain Name			
OIMDomain	Domain Name			

B.2 Directory Mapping

Use this worksheet to keep track of directories.

Table B–2 Directory Mapping

Documented Variable	Sample Directory Path	Your Directory Path
IAM_MW_HOME	/u01/oracle/products/access	
IAM_ORACLE_HOME	/u01/oracle/products/access/iam	
WEB_MW_HOME	/u02/private/oracle/products/web	
SOA_ORACLE_HOME	/u01/oracle/products/access/soa	
OUD_ORACLE_HOME	/u01/oracle/products/access/oud	
WEB_ORACLE_HOME	/u02/private/oracle/products/web/web	
WEBGATE_ORACLE_HOME	/u02/private/oracle/products/web/webgate	
ORACLE_COMMON_HOME	/u01/oracle/products/access/oracle_common	
WL_HOME	/u01/oracle/products/access/wlserver_10.3	
JAVA_HOME	/u01/oracle/products/access/jrockit_version	

Table B–2 (Cont.) Directory Mapping

Documented Variable	Sample Directory Path	Your Directory Path
OUD_ORACLE_INSTANCE	/u02/private/oracle/config/instances/oud*n*	
WEB_ORACLE_INSTANCE	/u02/private/oracle/config/instances/web*n*	
ASERVER_HOME (IDMDomain)	/u01/oracle/config/domains/IDMDomain	
MSERVER_HOME (IDMDomain)	/u02/private/oracle/config/domains/IDMDomain	
ASERVER_HOME (OIMDomain)	/u01/oracle/config/domains/OIMDomain	
MSERVER_HOME (OIMDomain)	/u02/private/oracle/config/domains/OIMDomain	

B.3 Port Mapping

Use this worksheet to keep track of ports.

Table B–3 Port Mapping

Documented Variable	Documented Port	Description	Your Port
HTTP_SSL_PORT	443	SSL Port for accessing the site externally	
HTTP_PORT	80	Non SSL Port used for accessing admin functions internally	
LDAP_LBR_PORT	389	LDAP Access Port on Load Balancer	
LDAP_LBR_SSL_PORT	636	LDAPS Access Port on Load Balancer	
LDAP_DIR_PORT	1389	OUD Access port	
LDAP_DIR_SSL_PORT	1636	OUD SSL Access port	
LDAP_DIR_ADMIN_PORT	4444	OUD Admin Port	
LDAP_DIR_REPL_PORT	8989	OUD Replication Port	
OHS_PORT	7777	Oracle HTTP Server Listen Port	
OAM_PROXY_PORT	5575	OAM Listen Port	
ONS_PORT	6200	ONS Port	
DB_LSNR_PORT	1521	Listener Port	
OAM_PORT	14100	OAM Managed Server Port	
OIM_PORT	14000	OIM Managed Server Port	
WLS_ADMIN_PORT	7001	WLS Administration Port	
WLS_ADMIN_SSL_PORT	7002	WLS Administration SSL Port	
NMGR_PORT	5556	Node Manager Listen Port	

Table B–3 (Cont.) Port Mapping

Documented Variable	Documented Port	Description	Your Port
SOA_PORT	8001	SOA Port	
SPLIT_WLS_ADMIN_PORT	7101	Split Domain WLS Administration Port	
SPLIT_WLS_ADMIN_SSL_PORT	7102	Split Domain WLS Administration SSL Port	

B.4 LDAP Directory Details

Use this worksheet to keep track of LDAP information.

Table B–4 LDAP Directory Details

Description	Documented Value	Customer Value
LDAP Directory Hosts	IDMHOST1 IDMHOST2	
LDAP Directory SSL Port	1636	
LDAP Directory Non SSL Port	1389	
LDAP Administration Port	4444	
Back end Directory Type	OUD	
LDAP Virtual host	IDSTORE.mycompany.com	
LDAP Load Balanced SSL Port	636	
LDAP Load Balanced Non-SSL Port	389	
LDAP Administration User	cn=oudadmin	
OUD_ORACLE_INSTANCE	`/u02/private/oracle/config/instances/oud1` `/u02/private/oracle/config/instances/oud2`	
LDAP Directory Tree	`dc=mycompany,dc=com`	
LDAP Group Search Base	`cn=Groups,dc=mycompany,dc=com`	
LDAP User Search Base	`cn=Users,dc=mycompany,dc=com`	
LDAP Reserve Location	`cn=Reserve,dc=mycompany,dc=com`	
LDAP System ID Location	`cn=systemids,dc=mycompany,dc=com`	

B.5 Database Details

Use this worksheet to keep track of database information.

Table B–5 Database Details

Description	Documented Value	Customer Value
Database Hosts	IDMDBHOST1 IDMDBHOST2	
Scan Address Name	DB-SCAN.mycompany.com	
Database Name	IDMDB.mycompany.com	
Database Service Names defined	OAMEDG.mycompany.com OIMEDG.mycompany.com OESEDG.mycompany.com	
System Account Name and Password	system/xxxxx	
RCU Schema Prefix	EDG	
ONS Port	6200	
Listener Port	1521	

B.6 Web Tier Details

Use this worksheet to keep track of Web Tier information.

Table B–6 Web Tier Details

Description	Documented Value	Customer Value
Web Tier Hosts	WEBHOST1 WEBHOST2	
Oracle HTTP Server Listen Port	7777	
WEB_ORACLE_HOME	`/u02/private/oracle/products/web/web`	
WEBGATE_ORACLE_HOME	`/u02/private/oracle/products/web/webgate`	
WEB_ORACLE_INSTANCE	`/u02/private/oracle/config/instances/web1` `/u02/private/oracle/config/instances/web2`	
Virtual Hosts	ADMIN.mycompany.com SSO.mycompany.com IDMINTERNAL.mycompany.com	

B.7 Application Tier Details

Use this worksheet to keep track of Application Tier information

Table B–7 Application Tier Details

Description	Documented Value	Customer Value
Host (Virtual Hosts)	ADMINVHN (IDMHOST1)	
	OIMADMINVHN (IDMHOST1)	
	OIMHOST1VHN (IDMHOST1)	
	SOAHOST1VHN (IDMHOST1)	
	OIMHOST2VHN (IDMHOST2)	
	OIMADMINVHN (IDMHOST2)	
	OIMHOST2VHN (IDMHOST2)	
	SOAHOST2VHN (IDMHOST2)	
Domain Name	IDMDomain	
ASERVER_HOME	/u01/oracle/config/domains/IDM Domain	
MSERVER_HOME	u02/private/oracle/config/domains /IDMDomain	
Domain Name	OIMDomain	
ASERVER_HOME	/u01/oracle/config/domains/OIM Domain	
MSERVER_HOME	u02/private/oracle/config/domains /OIMDomain	
Components Installed	OAM Console, OES Console, OAM, OIM	
OAM Managed Server Names	WLS_OAM1	
	WLS_OAM2	
OIM Managed Server Names	WLS_OIM1	
	WLS_OIM2	
OAM Managed Server Port	14100	
OIM Managed Server Port	14000	

B.8 User and Group Mapping

Use this worksheet to keep track of administrative accounts.

Table B–8 User Mapping

configTool Parameter	Documented Value	Customer Value
IDSTORE_OAMADMINUSER	oamadmin	
IDSTORE_OAMSOFTWAREUSER	oamLDAP	
IDSTORE_OIMADMINUSER	oimLDAP	
IDSTORE_WLSADMINUSER	weblogic_idm	

Table B–9 Group Mapping

configTool Parameter	Documented Value	Customer Value
OAM11G_IDSTORE_ROLE_ SECURITY_ADMIN	OAMAdministrators	
IDSTORE_OIMADMINGROUP	OIMAdministrators	
IDSTORE_WLSADMINGROUP	WLSAdmins	

Index

F

G

H

I

K

L

M

N

O

P

R

S

T

U

V

W