Oracle® Fusion Middleware

Enterprise Deployment Guide for Oracle Identity Management (Oracle Fusion Applications Edition)

11*g* Release 7 (11.1.7)

E21032-21

October 2013

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

ORACLE

Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management (Oracle Fusion Applications Edition), 11*g* Release 7 (11.1.7)

E21032-21

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

Contributors: Pradeep Bhat, Bruce Jiang, 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 for Provisioning

7 Preparing the Database for an Enterprise Deployment

12 Setting Up Node Manager for an Enterprise Deployment

13 Configuring Server Migration for an Enterprise Deployment

14 Validating Provisioning

15 Scaling Enterprise Deployments

16 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* (*Fusion Applications Edition*).

Audience

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

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 Integration Overview for Oracle Identity Management Suite*
- *Oracle Fusion Middleware Application Security Guide*
- *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 Manager with Oracle Security Token Service*
- *Oracle Fusion Middleware Integration Guide for Oracle Access Manager*
- *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 Fusion Middleware Administrator's Guide for Oracle Identity Federation*
- *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*
- *Oracle Fusion Middleware Integration Overview for Oracle Identity Management Suite*

For additional information about Oracle Fusion Applications, consult the following documents in the Oracle Fusion Applications 11.1.4 library:

- *Oracle Fusion Applications Administrator and Implementor Roadmap*
- *Oracle Fusion Applications Administrator's Guide*
- *Oracle Fusion Applications Enterprise Deployment Guide for Customer Relationship Management*

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 changes in this guide and provide pointers to additional information.

New and Changed Features for 11*g* Release 7 (11.1.7)

Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management (Oracle Fusion Applications Edition) 11g Release 7 (11.1.7) includes the following changes:

- Use of the Identity Management Provisioning Wizard to create a provisioning response file.
- Use of the `runIDMProvisioning` command-line tool to perform Identity Management Provisioning.

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.

Deploying Oracle Identity Management as described in this guide is a prerequisite for deploying Oracle Fusion Applications as described in *Oracle Fusion Applications Enterprise Deployment Guide for Customer Relationship Management*.

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 (Oracle Fusion Applications Edition) focuses on enterprise deployments in Linux environments. However, you can also implement enterprise deployments using UNIX and Windows environments.

1.2 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 Oracle, 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.

1.3 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 1.3.1, "Built-in Security"
- Section 1.3.2, "High Availability"

1.3.1 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 the Web Tier, Application Tier, and the Directory 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.
- Oracle Internet Directory is isolated in the Directory 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.

1.3.2 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 to the Enterprise Deployment Reference Topologies

This chapter describes and illustrates the enterprise deployment reference topologies described in this guide and helps you plan your deployment.

The key to a successful Enterprise Deployment is planning and preparation. The road map for installation and configuration in this chapter directs you to the appropriate chapters for the tasks you need to perform. Use this chapter to help you plan your Oracle Identity Management enterprise deployment.

This chapter contains the following topics:

2.1 Overview of Enterprise Deployment Reference Topologies

This section describes diagrams used to illustrate the enterprise deployment possibilities described in this guide. Use this section to plan your enterprise deployment topology.

This section covers these topics:

2.1.1 Reference Topologies Documented in the Guide

Oracle Identity Management consists of a number of products, which can be used either individually or collectively. The Enterprise Deployment Guide for Identity Management (Fusion Applications Edition) enables you to build two different enterprise topologies for Fusion Applications.

In the diagrams, active nodes are shown in color, and passive nodes are shown in white.

See Also: The supported platforms documentation for Oracle Fusion Applications.

This section contains the following topologies:

- Section 2.1.1.1, "Oracle Access Manager 11g and Oracle Identity Manager 11g for Fusion Applications"
- Section 2.1.1.2, "Oracle Identity Federation 11g for Fusion Applications"

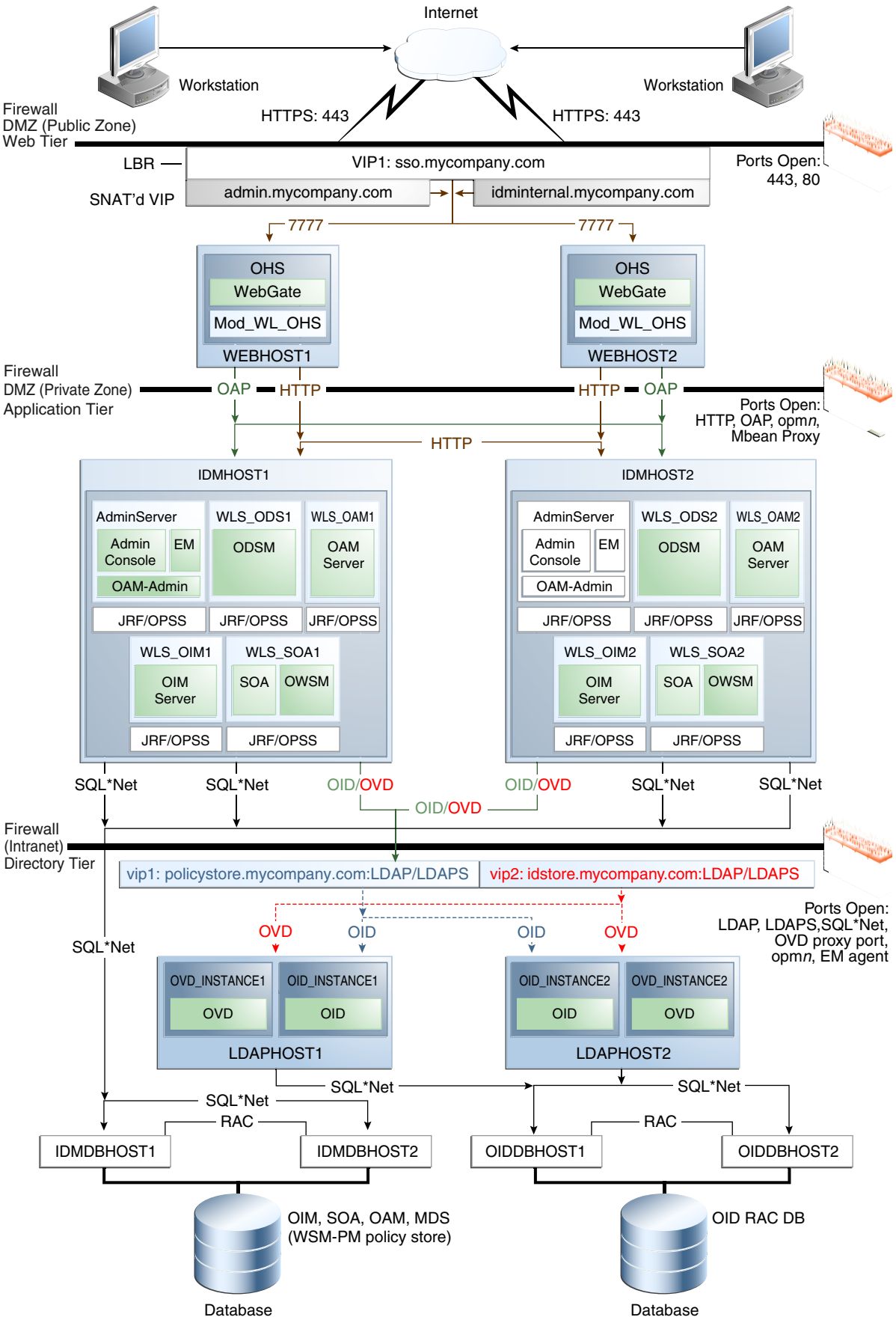
2.1.1.1 Oracle Access Manager 11g and Oracle Identity Manager 11g for Fusion Applications

Figure 2–1 is a diagram of the Oracle Access Manager 11*g* and Oracle Identity Manager 11*g* topology. This is the topology that the vast majority of customers will use. It enables you to add Identity Management functionality to Oracle Applications deployments.

Oracle 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.

This topology will not service federated requests. If you want to use Oracle Identity Federation as well, then, in addition to this topology, you must also deploy a separate federated topology similar to that shown in Section 2.1.1.2, "Oracle Identity Federation 11g for Fusion Applications."

Figure 2–1 Oracle Access Manager 11g and Oracle Identity Manager 11g Topology for Fusion Applications


Internet
Workstation
Workstation
Firewall
DMZ (Public Zone)
Web Tier
HTTPS: 443
HTTPS: 443
LBR
VIP1: sso.mycompany.com
SNAT'd VIP
admin.mycompany.com
idminternal.mycompany.com
Ports Open:
443, 80
7777
7777
OHS
WebGate
Mod_WL_OHS
WEBHOST1
OHS
WebGate
Mod_WL_OHS
WEBHOST2
Firewall
DMZ (Private Zone)
Application Tier
OAP
HTTP
HTTP
OAP
HTTP
Ports Open:
HTTP, OAP, opmn,
Mbean Proxy
IDMHOST1
AdminServer
Admin Console
EM
OAM-Admin
WLS_ODS1
ODSM
WLS_OAM1
OAM Server
JRF/OPSS
JRF/OPSS
JRF/OPSS
WLS_OIM1
OIM Server
WLS_SOA1
SOA
OWSM
JRF/OPSS
JRF/OPSS
IDMHOST2
AdminServer
Admin Console
EM
OAM-Admin
WLS_ODS2
ODSM
WLS_OAM2
OAM Server
JRF/OPSS
JRF/OPSS
JRF/OPSS
WLS_OIM2
OIM Server
WLS_SOA2
SOA
OWSM
JRF/OPSS
JRF/OPSS
SQL*Net
SQL*Net
OID/OVD
OID/OVD
OID/OVD
SQL*Net
SQL*Net
Firewall
(Intranet)
Directory Tier
vip1: policystore.mycompany.com:LDAP/LDAPS
vip2: idstore.mycompany.com:LDAP/LDAPS
Ports Open:
LDAP, LDAPS,SQL*Net,
OVD proxy port,
opmn, EM agent
OVD
OID
OID
OVD
SQL*Net
OVD_INSTANCE1
OVD
OID_INSTANCE1
OID
LDAPHOST1
OID_INSTANCE2
OID
OVD_INSTANCE2
OVD
LDAPHOST2
SQL*Net
SQL*Net
SQL*Net
RAC
RAC
IDMDBHOST1
IDMDBHOST2
OIDDBHOST1
OIDDBHOST2
OIM, SOA, OAM, MDS
(WSM-PM policy store)
OID RAC DB
Database
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 Load Balancer: The load balancer receives user requests on Port 80 (HTTP) and Port 443 (HTTPS), strips out the SSL where appropriate and passes the requests onto the Oracle HTTP servers using Port 7777.
- The Web Tier: There are two WebLogic servers, each of which hosts an Oracle HTTP Server and Oracle WebGate.
- The Application Tier: There are two servers, IDMHOST1 and IDMHOST2. Requests are received from the Web Tier. Each server contains managed servers for the following products:
  - Oracle Access Manager, which hosts the Access Server
  - Oracle Identity Manager, which hosts an OIM Server and corresponding JRF/OPSS processes
  - SOA, which hosts a SOA Server and corresponding JRF/OPSS processes

  IDMHOST1 also contains the WebLogic Administration Server, which hosts the WebLogic Console, Enterprise Manager Fusion Middleware Control, OAM Console, and ODSM (for Oracle Internet Directory and Oracle Virtual Directory). In the event of the failure of IDMHOST1, the WebLogic Administration Server can be started on IDMHOST2.
- The Directory Tier: There are two hosts, LDAPHOST1 and LDAPHOST2. These hosts contain the Oracle Internet Directory instances.

  This is also where the databases reside. These contain the Oracle Internet Directory schemas, 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.

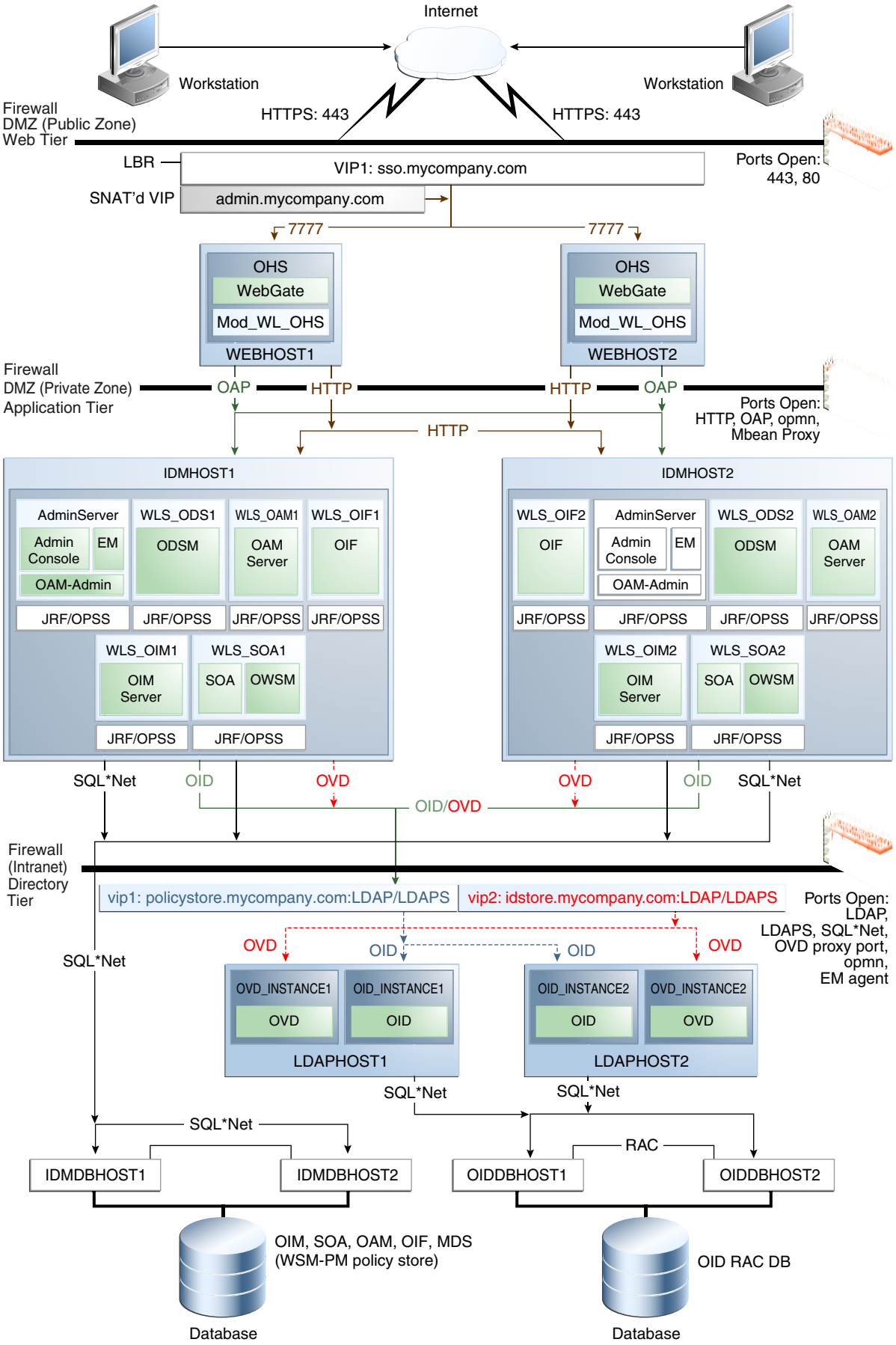
  In addition, the load balancer distributes LDAP requests among the Oracle Unified Directory instances on IDMHOST1 and IDMHOST2, using the load balancer virtual host IDSTORE.mycompany.com
- Firewalls: These are used to separate the Web, Application, and Directory tiers into different zones. WEBHOST1 and 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 Oracle Identity Federation 11g for Fusion Applications

Figure 2–2 is a diagram of the Oracle Identity Federation 11*g* topology for Fusion Applications. This topology is only relevant to users who are deploying Oracle Identity Federation. An enterprise deployment containing Oracle Identity Federation can only be used for that purpose. That is, it cannot be used to service both federated and non-federated requests.

Figure 2–2 Oracle Identity Federation 11g Topology for Fusion Applications


Internet
Workstation
Workstation
HTTPS: 443
HTTPS: 443
Firewall
DMZ (Public Zone)
Web Tier
LBR
VIP1: sso.mycompany.com
SNAT'd VIP
admin.mycompany.com
Ports Open:
443, 80
7777
7777
OHS
WebGate
Mod_WL_OHS
WEBHOST1
OHS
WebGate
Mod_WL_OHS
WEBHOST2
Firewall
DMZ (Private Zone)
Application Tier
OAP
HTTP
HTTP
OAP
HTTP
Ports Open:
HTTP, OAP, opmn,
Mbean Proxy
IDMHOST1
AdminServer
Admin Console
EM
OAM-Admin
WLS_ODS1
ODSM
WLS_OAM1
OAM Server
WLS_OIF1
OIF
JRF/OPSS
JRF/OPSS
JRF/OPSS
JRF/OPSS
WLS_OIM1
OIM Server
WLS_SOA1
SOA
OWSM
JRF/OPSS
JRF/OPSS
IDMHOST2
WLS_OIF2
OIF
AdminServer
Admin Console
EM
OAM-Admin
WLS_ODS2
ODSM
WLS_OAM2
OAM Server
JRF/OPSS
JRF/OPSS
JRF/OPSS
JRF/OPSS
WLS_OIM2
OIM Server
WLS_SOA2
SOA
OWSM
JRF/OPSS
JRF/OPSS
SQL*Net
OID
OVD
OVD
OID
SQL*Net
OID/OVD
Firewall
(Intranet)
Directory
Tier
vip1: policystore.mycompany.com:LDAP/LDAPS
vip2: idstore.mycompany.com:LDAP/LDAPS
Ports Open:
LDAP,
LDAPS, SQL*Net,
OVD proxy port,
opmn,
EM agent
SQL*Net
OVD
OID
OID
OVD
OVD_INSTANCE1
OVD
OID_INSTANCE1
OID
LDAPHOST1
OID_INSTANCE2
OID
OVD_INSTANCE2
OVD
LDAPHOST2
SQL*Net
SQL*Net
SQL*Net
RAC
IDMDBHOST1
IDMDBHOST2
OIDDBHOST1
OIDDBHOST2
OIM, SOA, OAM, OIF, MDS
(WSM-PM policy store)
OID RAC DB
Database
Database

This topology is similar to the one shown in Figure 2–1. It differs from that figure in that the IDMHOSTs in the Application Tier include the additional product, Oracle Identity Federation.

2.1.2 About the Directory Tier

The Directory Tier is in the Intranet Zone. The Directory Tier is the deployment tier where all the LDAP services reside. This tier includes products such as Oracle Internet Directory and Oracle Virtual Directory. The Directory Tier is managed by directory administrators providing enterprise LDAP service support.

The Directory Tier is closely tied with the Data Tier. Access to the Data Tier is important 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.

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.

Typically protected by firewalls, applications above the Directory Tier access LDAP services through a designated LDAP host port. The standard LDAP port is 389 for the non-SSL port and 636 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.

The Directory Tier stores two types of information:

- Identity Information: Information about users and groups
- Oracle Platform Security Services (OPSS): Information about security policies and about configuration.

Extending the domain to include Oracle Internet Directory includes the following steps:

- Configure two instances of Oracle Internet Directory by using the Oracle Identity Management 11*g* Configuration Wizard
- Register the instances with the WebLogic Server Domain (IDMDomain).
- Validate the instances
- Tune Oracle Internet Directory

Although the topology diagrams do not show LDAP directories other than Oracle Internet Directory, you can use Microsoft Active Directory to store identity information. You must always store policy information in Oracle Internet Directory. You may store identity information in Oracle Internet Directory or in another directory.

If you store the Identity details in a directory other than Oracle Internet Directory, you can use Oracle Virtual Directory to present that information

Oracle Fusion Middleware Integration Overview for Oracle Identity Management Suite describes how to configure Oracle Virtual Directory for two multidirectory scenarios.

- A split profile, or split directory configuration, is one where identity data is stored in multiple directories, possibly in different locations. A split profile is used to store custom attributes required for Fusion Application Deployment. Use this kind

of deployment when you do not want to modify the existing Identity Store by extending the schema. In that case, deploy a new Oracle Internet Directory instance to store the extended attributes. Alternatively, you can use the Oracle Internet Directory instance deployed for Policy Store for this purpose.

- Another multidirectory scenario is one where you have distinct user and group populations. In this configuration, Oracle-specific entries and attributes are stored in Oracle Internet Directory. Enterprise-specific entries that might have Fusion Applications-specific attributes are stored in Active Directory.

In both multidirectory scenarios, you use Oracle Virtual Directory to present all the identity data in a single consolidated view that Oracle Identity Management components can interpret.

Although you can use a single Oracle Internet Directory instance for storing both the identity and policy information, in some cases you might need to use two separate Oracle Internet Directory installations, one for the Policy Store and another for Identity Store. For example, this might be necessary due to throughput or enterprise directory requirements. You might also need to use separate Oracle Internet Directory installations if you have a shared Identity Management deployment with multiple Oracle Fusion Applications pods pointing to it.

If your policy store is in a separate Oracle Internet Directory instance, this instance is created on the same host as the Oracle Internet Directory instance used for identity information, although it will use a different database to hold its information. As a result, the Oracle Internet Directory containing policy information will use different ports from those used by the primary Oracle Internet Directory.

If you intend to separate your identity and policy information, you must create two separate clusters of highly available Oracle Internet Directory. These Oracle Internet Directory clusters can share the same machines but they should use separate Real Application Clusters databases as their data store.

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

This guide assumes that you are creating two virtual names: one for your Policy Store (`POLICYSTORE.mycompany.com`) and one for your Identity Store (`IDSTORE.mycompany.com`). When using a single Oracle Internet Directory for both your identity and policy information, you can either create two virtual host names, both pointing to the same directory, or combine them into a single suitable virtual host name in the load balancer.

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.

2.1.2.1 Considering Oracle Internet Directory Password Policies

By default, Oracle Internet Directory passwords expire in 120 days. Users who do not reset their passwords before expiration can no longer authenticate to Oracle Internet Directory. This includes administrative users, such as `oamadmin`. Your Identity Management environment cannot work properly unless these users can authenticate. See *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory* for information about changing Oracle Internet Directory password policies.

2.1.2.2 Using Different Directory Configurations

The enterprise deployment described in this guide shows Oracle Access Manager using Oracle Internet Directory as the only LDAP repository. Oracle Access Manager uses a single LDAP for policy and configuration data. It is possible to configure another LDAP as the Identity Store where users, organizations and groups reside. For example, an Oracle Access Manager instance may use Oracle Internet Directory as its policy and configuration store and point to an instance of Microsoft Active Directory for users and groups.

In addition, the Identity Stores can potentially be front-ended by Oracle Virtual Directory to virtualize the data sources.

To learn more about the different types of directory configuration for Oracle Access Manager, consult the 11g Oracle Access Manager documentation at Oracle Technology Network. Customers considering these variations should adjust their Directory Tier and Oracle Access Manager deployment accordingly.

2.1.2.3 High Availability Provisions

- Oracle Internet Directory Instances are active/active deployments.
- Oracle Virtual Directory Instances are active/active deployments.
- If the Oracle Internet Directory fails on the LDAPHOST, Oracle Process Management and Notification (OPMN) server attempts to restart it.
- If the Oracle Virtual Directory fails on the LDAPHOST, Oracle Process Management and Notification (OPMN) server attempts to restart it.

2.1.3 About the Application Tier

The Application Tier is the tier where Java EE applications are deployed. Products such as Oracle Identity Manager, Oracle Identity Federation, 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 have the WebLogic Server with the Administration Console, Oracle Enterprise Manager Fusion Middleware Control, Oracle Directory Services Manager, Oracle Identity Federation, and Oracle Access Management Server configured. IDMHOST1 and IDMHOST2 run both the WebLogic Server

Administration Servers and Managed Servers. Note that the Administration Server is configured to be active-passive, that is, although it is installed on both nodes, only one instance is active at any time. If the active instance goes down, then the passive instance starts up and becomes the active instance.

The Oracle Access Management Server communicates with the Directory Tier to verify user information.

- On the firewall protecting the Application Tier, the HTTP ports, and OAP port are open. The OAP (Oracle Access Protocol) port is for the WebGate module running in Oracle HTTP Server in the Web Tier to communicate with Oracle Access Manager to perform operations such as user authentication.
- In the OAM and OIM topology, IDMHOST1 and IDMHOST2 have Oracle Identity Manager and Oracle SOA installed. Oracle Identity Manager is user provisioning application. Oracle SOA deployed in this topology is exclusively used for providing workflow functionality for Oracle Identity Manager.

2.1.3.1 Architecture Notes

- Oracle Enterprise Manager Fusion Middleware Control is integrated with Oracle Access Manager using the Oracle Platform Security Services (OPSS) agent.
- 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 WLS_ODS1 Managed Server on IDMHOST1 and WLS_ODS2 Managed Server on IDMHOST2 are in a cluster and the Oracle Directory Services Manager applications are targeted to the cluster.
- The WLS_OAM1 Managed Server on IDMHOST1 and WLS_OAM2 Managed Server on IDMHOST2 are in a cluster and the Oracle Access Manager applications are targeted to the cluster.
- Oracle Directory Services Manager are bound to the listen addresses of the WLS_ODS1 and WLS_ODS2 Managed Servers. By default, the listen address for these Managed Servers is set to IDMHOST1 and IDMHOST2 respectively.
- The WLS_OIM1 Managed Server on IDMHOST1 and WLS_OIM2 Managed Server on IDMHOST2 are in a cluster and the Oracle Identity Manager applications are targeted to the cluster.
- The WLS_SOA1 Managed Server on IDMHOST1 and WLS_SOA2 Managed Server on IDMHOST2 are in a cluster and the Oracle SOA applications are targeted to the cluster
- The WLS_OIF1 Managed Server on IDMHOST1 and WLS_OIF2 Managed Server on IDMHOST2 are in a cluster and the Oracle Identity Federation applications are targeted to the cluster.

2.1.3.2 High Availability Provisions

- The OAM Servers are active-active deployments.
- Oracle Access Manager, 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 Identity Federation Servers are active-active deployments; the Oracle Identity Federation Server may communicate with the Data Tier at run time.
- 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.

2.1.3.3 Security Provisions

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

2.1.3.4 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.

2.1.4 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 Oracle Access Manager, the Web Tier is required.

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 Access Manager component) in Oracle HTTP Server uses Oracle Access Protocol (OAP) to communicate with Oracle Access Manager running on IDMHOST1 and IDMHOST2, in the Identity Management DMZ. WebGate and Oracle 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.

2.1.4.1 Architecture Notes

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

2.1.4.2 High Availability Provisions

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

2.1.4.3 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 Hardware Requirements for an Enterprise Deployment

The minimum hardware requirements for the Enterprise Deployment on Linux operating systems are listed in Table 2–1. The memory figures represent the physical memory required to install and run an Oracle Fusion Middleware server.

For detailed requirements, or for requirements for other platforms, see the *Oracle Fusion Middleware System Requirements and Specifications*. Also see the *Technical Release Notes* for Oracle Fusion Applications 11g Release 7 (11.1.7), especially the sections with titles such as "System Requirements" and "Pre-Installation Known Issues."

Table 2–1 Typical Hardware Requirements

Server	Processor	Disk	Memory	TMP Directory	Swap
`Database Hosts` OIDDBHOST*n*, 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	40 GB	4 GB	Default	Default
IDMHOST*n*	4 or more X Pentium 1.5 GHz or greater	30 GB	16 GB	Default	Default
LDAPHOST*n*	2 or more X Pentium 1.5 GHz or greater	30 GB	4 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; however, this 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 servers to the configuration by following the instructions in Chapter 15, "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.

2.3 Software Components Installed as Part of the Provisioning Process

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

For complete information about downloading Oracle Fusion Middleware software, see Section 6.1.2, "Software to Install" and the "Preparing for an Installation" chapter in *Oracle Fusion Applications Installation Guide*.

Table 2–2 Software Versions Used

Short Name	Product	Version
OHS11G	Oracle HTTP Server	11.1.1.7.0
JRockit	Oracle JRockit	JDK 6 (jrockit 1.6.0_37-b06 or newer)
WLS	Oracle WebLogic Server	10.3.6.0
IAM	Oracle Identity and Access Management	11.1.1.7.0
SOA	Oracle SOA Suite	11.1.1.7.0
IDM	Oracle Identity Management	11.1.1.7.0
WebGate	WebGate 11*g*	11.1.1.7.0
RCU	Repository Creation Assistant	11.1.1.7.0

2.4 Road Map for the Reference Topology Installation and Configuration

In this current release of the Guide, the configuration of the enterprise deployment topology is largely automatic.

Configuration is performed by using the Oracle Identity Management Provisioning Wizard to create a response file, then using the Identity Management Provisioning Tools to perform the provision process.

There are two main phases:

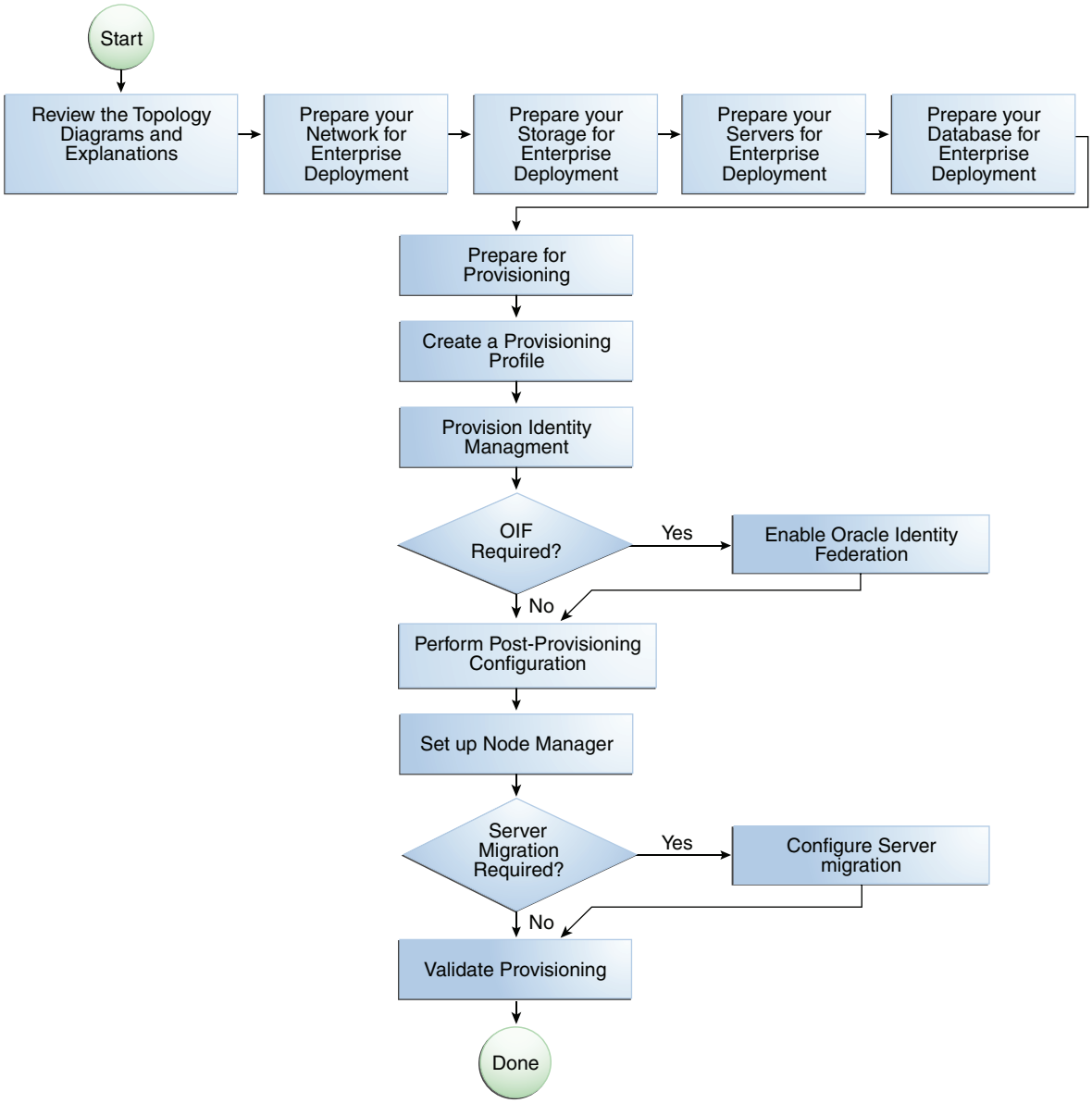
- Information Gathering Phase - This is where the wizard collects information about your target environment.
- Provisioning Phase - This is where the tool takes the information gleaned in the Information Gathering Phase and creates the deployment, installing and configuring all of the software required.

> **See Also:** *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`

2.4.1 Flow Chart of the Oracle Identity Management Enterprise Deployment Process for Oracle Fusion Applications

Figure 2–3, "Flow Chart of the Oracle Identity Management Enterprise Deployment Process for Oracle Fusion Applications" 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 2–3 Flow Chart of the Oracle Identity Management Enterprise Deployment Process for Oracle Fusion Applications



2.4.2 Steps in the Oracle Identity Management Enterprise Deployment Process

Table 2–3 describes each of the steps in the enterprise deployment process flow chart for Oracle Identity Management, shown in Figure 2–3. The table also provides information on where to obtain more information about each step in the process.

Table 2–3 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 File System 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 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 7, "Preparing the Database for an Enterprise Deployment"
Prepare for Provisioning	Perform prerequisite tasks.	Chapter 6, "Preparing for Provisioning"
Create a Provisioning Profile	Run the Oracle Identity Management Provisioning Wizard to create a provisioning profile	Chapter 8, "Creating a Provisioning Profile"
Provision Identity Management	Run the provisioning commands on the host machines, using the provisioning profile as input.	Chapter 9, "Provisioning Identity Management"
Perform Post-Provisioning Configuration	Perform additional configuration.	Chapter 10, "Performing Post-Provisioning Configuration"
Enable Oracle Identity Federation?	Enable and start Oracle Identity Configuration	Chapter 11, "Enabling Oracle Identity Federation"
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 12, "Setting Up Node Manager for an Enterprise Deployment"

Table 2–3 (Cont.) Steps in the Oracle Identity Management Enterprise Deployment Process

Step	Description	More Information
Configure Server Migration	Configure server migration for the WLS_OIM1, WLS_SOA1, WLS_OIM2, and WLS_SOA2 Managed Servers. Configure the WLS_OIM1 and WLS_SOA1 Managed Server to restart on IDMHOST2 if a failure occurs. Configure the WLS_OIM2 and WLS_SOA2 Managed Servers to restart on IDMHOST1 if a failure occurs.	Chapter 13, "Configuring Server Migration for an Enterprise Deployment"
Validate Provisioning	Perform additional sanity checks, in addition to those provided by the provisioning process.	Chapter 14, "Validating Provisioning"

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, 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, three 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 private zone–This is where the application servers reside. This zone cannot be accessed directly from the outside world. It can be accessed only by the web servers that reside in the public zone. This access is determined by the use of a firewall

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

- 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

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.

3.3.2 Virtual Server names

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

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

Some of the virtual server names are used by some topologies and not others.

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 document assumes that the deployment is one of those shown in Section 2.1.1, "Reference Topologies Documented in the Guide."

3.3.2.1 POLICYSTORE.mycompany.com

- This virtual server is enabled on LBR2. It acts as the access point for all policy-based LDAP traffic, which is stored in the Oracle Internet Directory servers in the Directory Tier. Traffic to both the SSL and non-SSL is configured. The clients access this service using the address `POLICYSTORE.mycompany.com:636` for SSL and `POLICYSTORE.mycompany.com:389` for non-SSL.

  Note: Oracle recommends that you configure the same port for SSL connections on the LDAP server and Oracle Internet Directory on the computers on which Oracle Internet Directory is installed.

  This is a requirement for most Oracle 11*g* products that use Oracle Internet Directory through the load balancing router.

- This virtual server directs traffic received on port 389 to each of the Oracle Internet Directory instances on port 3060.
- This virtual server directs traffic received on port 636 to each of the Oracle Internet Directory instances on port 3131.
- Monitor the heartbeat of the Oracle Internet Directory processes on LDAPHOST1 and LDAPHOST2. If an Oracle Internet Directory process stops on LDAPHOST1 or LDAPHOST2, or if either host LDAPHOST1 or LDAPHOST2 is down, the load balancer must continue to route the LDAP traffic to the surviving computer.

3.3.2.2 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.
- Monitor the heartbeat of the Oracle Virtual Directory processes on LDAPHOST1 and LDAPHOST2. If an Oracle Virtual Directory process stops on LDAPHOST1 or LDAPHOST2, or if either host LDAPHOST1 or LDAPHOST2 is down, the load balancer must continue to route the LDAP traffic to the surviving computer.
- If your implementation uses a third-party directory as an identity store, you access that directory through Oracle Virtual Directory. The provisioning process initially uses Oracle Internet Directory as its identity store. After provisioning is completed, you change the Oracle Virtual Directory configuration to point to your actual identity store, the third party directory, as a post provisioning task. After that, you still access identity information through Oracle Virtual Directory using the endpoint `IDSTORE.mycompany.com`, but now it points to the third-party directory.
- This virtual server directs traffic received on port 389 *(LDAP_LBR_PORT)* to each of the Oracle Virtual Directory instances on port 6501.
- This virtual server directs traffic received on port 636 (*LDAP_LBR_SSL_PORT*) to each of the Oracle Virtual Directory instances on port 7501.

3.3.2.3 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 ports 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, 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.

3.3.2.4 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.

3.3.2.5 SSO.mycompany.com

- This is the virtual name which fronts all Identity Management components, including Oracle Identity Federation, Oracle Access Manager, 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 ports 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.

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.

- Configuration of the virtual server(s) in the load balancer for the Directory Tier with a high value for the connection timeout for TCP connections. This value should be more than the maximum expected time over which no traffic is expected between Oracle Access Manager and the Directory Tier.
- 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
SSO.mycompany.com:443	WEBHOST1.mycompany.com:7777 WEBHOST2.mycompany.com:7777	HTTPS	Yes	Yes	Identity Management requires that the following be added to the HTTP header: Header Name: IS_SSL Header Value: ssl
IDMINTERNAL.mycompany.com:80	WEBHOST1:7777 WEBHOST2:7777	HTTP	No	No	
ADMIN.mycompany.com:80	WEBHOST1.mycompany.com:7777 WEBHOST2.mycompany.com:7777	HTTP	No	No	
POLICYSTORE.mycompany.com:389	LDAPHOST1.mycompany.com:3060 LDAPHOST2.mycompany.com:3060	LDAP	No	No	Only required if policy store is different from idstore. This is true if you use either Active Directory or Oracle Internet Directory with a split profile.
POLICYSTORE.mycompany.com:636	LDAPHOST1.mycompany.com:3131 LDAPHOST2.mycompany.com:3131	LDAP	No	No	

Table 3–1 (Cont.) Load Balancer Configuration

Virtual Host	Server Pool	Protocol	SSL Termination	External	Other Required Configuration/Comments
IDSTORE.mycompany.com:389	LDAPHOST1.mycompany.com:6501 LDAPHOST2.mycompany.com:6501	LDAP	No	No	If you have an Oracle Internet Directory-only topology and are not using Oracle Virtual Directory, the server pool must contain Oracle Internet Directory servers.
IDSTORE.mycompany.com:636	LDAPHOST1.mycompany.com:7501 LDAPHOST2.mycompany.com:7501	LDAP	No	No	If you have an Oracle Internet Directory-only topology and are not using Oracle Virtual Directory, the server pool must contain Oracle Internet Directory servers.

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

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 IPs and VIPs Mapped to Administration Server and Managed Servers

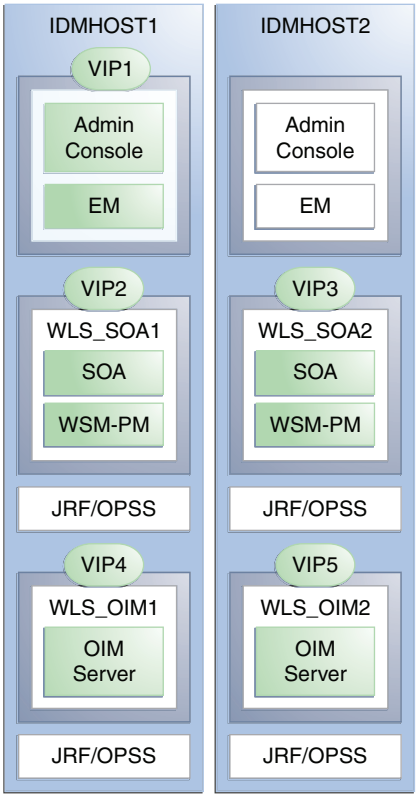


Table 3–2 provides descriptions of the various virtual hosts.

Table 3–2 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.) 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	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).
VIP4	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)
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 Directory Tier.

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	HTTP / 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).
IDMDomain 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_ODS	FW1	7006	HTTP / Oracle HTTP Server to WebLogic Server	Inbound	Timeout depends on the `mod_weblogic` parameters used.
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

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
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.
Oracle Access Manager Server 11*g*	FW1	5575	OAP	Both	N/A
Oracle 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
Enterprise Manager Agent - Directory Tier to Enterprise Manager	FW2	5160	HTTP / Enterprise Manager Agent and Enterprise Manager	Both	N/A
OPMN access in Directory Tier	FW2	OPMN remote port	HTTP / Administration Server to OPMN	Inbound	N/A
Oracle Virtual Directory proxy port	FW2	8899	HTTP / Administration Server to Oracle Virtual Directory	Inbound	N/A
Database access	FW2	1521	SQL*Net	Both	Timeout depends on all database content and on the type of process model used for Oracle Identity Management.
Oracle Internet Directory access	FW2	3060	LDAP	Inbound	Tune the directory server's parameters based on the load balancer, and not the other way around. Ideally, these connections should be configured not to time out.
Oracle Internet Directory access	FW2	3131	LDAP SSL	Inbound	Tune the directory server's parameters based on the load balancer, and not the other way around. Ideally, these connections should be configured not to time out.
Oracle Virtual Directory access	FW2	6501	LDAP	Inbound	Ideally, these connections should be configured not to time out.

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
Oracle Virtual Directory access	FW2	7501	LDAP SSL	Inbound	Ideally, these connections should be configured not to time out.
Oracle Identity Federation	FW2	7499	HTTP	Both	N/A
Session replication within a WebLogic Server cluster	N/A	N/A	N/A	N/A	By default, this communication uses the same port as the server's listen address.
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 Fixed Ports Used by the Provisioning Wizard

The Identity Management Provisioning Wizard uses some ports that you cannot change. These are shown in Table 3–4. When scaling the topology, you can use different ports.

Table 3–4 Fixed Ports Used by the Provisioning Wizard

Description	Name	Value
Oracle Virtual Directory proxy port	*OVD_ADMIN_PORT*	8899
	OVD_PORT	6501
	OVD_SSL_PORT	7501
Load Balancer Policy Store Port	*LDAP_POLICY_LBR_PORT*	389
Load Balancer Policy Store SSL Port	*LDAP_POLICY_LBR_SSL_PORT*	636
Oracle Access Manager proxy port	*OAM_PROXY_PORT*	5575
Oracle WebLogic Server administration port	*WLS_ADMIN_PORT*	7001

3.8 Managing Oracle 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 3.8.1, "Oracle Access Manager Protocols."
- Section 3.8.2, "Overview of Integration Requests."
- Section 3.8.3, "Overview of User Request."

- Section 3.8.4, "About the Unicast Requirement for Communication."

3.8.1 Oracle 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.

3.8.2 Overview of Integration Requests

Oracle Access Manager is responsible for creating sessions for users. When Oracle Access Manager is integrated with another Identity Management component, such as Oracle Identity Manager, authentication is delegated to those components.

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. Oracle Access Manager credential collector is invoked and the user enters a user name and password in response to a prompt. Oracle 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 Oracle 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 Oracle Access Manager can read.

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

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

3.8.3 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.

3.8.4 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 this directory structure and directory terminology. Other directory layouts are possible and supported.

This chapter contains the following topics:

4.1 Overview of Preparing the File System for Enterprise Deployment

It is important to set up your file system in a way that makes the enterprise deployment easier to understand, configure, and manage. Oracle recommends setting up your file system 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`

- **SHARED_ROOT**: This environment variable and related directory path refer to the root directory on shared storage where the binaries and configuration information ARE stored.
- **LOCAL_ROOT**: This environment variable and related directory path refer to the root directory on local storage where the binaries and configuration information are stored.
- **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_HOME*s. An example of a typical *MW_HOME* is:

  ORACLE_BASE/product/fmw

  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*, *DIR_MW_HOME*, *WEB_MW_HOME*. For more information about homes, see Table 4–3, " Summary of Homes".
- **WL_HOME**: This variable and related directory path contains installed files necessary to host a WebLogic Server, for example *MW_HOME*/wlserver_10.3.
- **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. For example: IAM_ORACLE_HOME, OIM_ORACLE_HOME, WEB_ORACLE_HOME
- **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 4.3, "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/local/config/instances/ohs1
- **JAVA_HOME**: This is the location where JRockit is installed.
- **ASERVER_HOME**: This is the primary location of the domain configuration. A typical example is: /u01/oracle/config/domains/IDMDomain
- **MSERVER_HOME**: This is a copy of the domain configuration used to start and stop managed servers. A typical example is: /u02/local/oracle/config/domains/IDMDomain
- **WEBGATE_ORACLE_HOME**: This is the location of the WebGate installation.
- **DB_ORACLE_HOME**: This is the location of the Oracle Database installation.

4.3 About Recommended Locations for the Different Directories

This section contains the following topics:

4.3.1 Shared Storage Recommendations for Binary (Oracle Home) Directories

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

4.3.1.1 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 Concepts*.

4.3.1.2 About Sharing a Single Middleware Home Across Multiple Hosts

Sharing a Middleware home across different hosts enables you to install the Middleware home in a single location on a shared volume and use 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.*

4.3.1.3 About Using Redundant Binary (Middleware Home) Directories

Note: This is a manual process which you would perform after provisioning. You would do this by backing up the directories that provisioning creates and then restoring them to a new volume on the storage. You could then mount the new volume onto servers with the same mount point. For example:

- Volume 1 mounted to hosts 1, 3 and 5
- Volume 2 mounted to hosts 2, 4, and 6

Both would have the same mount point: `/u01`

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.

4.3.2 Shared Storage Recommendations for Provisioning Repository

The Identity Management Provisioning Wizard requires that each host in the topology have access to the Fusion Applications Provisioning Repository in the same location.

It is recommended that, for the duration of the provisioning, the provisioning repository be located on shared storage that is made available to each host.

After provisioning is complete the Provisioning Repository can be dismounted from each host.

4.3.3 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 4.3.3.1, "About Oracle WebLogic Server Administration and Managed Server Domain Configuration Files"
- Section 4.3.3.2, "Shared Storage Requirements for Administration Server Domain Configuration Files"
- Section 4.3.3.3, "Local Storage Requirements for Managed Server Domain Configuration Files"

4.3.3.1 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*.

4.3.3.2 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.

4.3.3.3 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.

4.3.4 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*.

4.3.5 Recommended Directory Locations

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

This section includes the following topics:

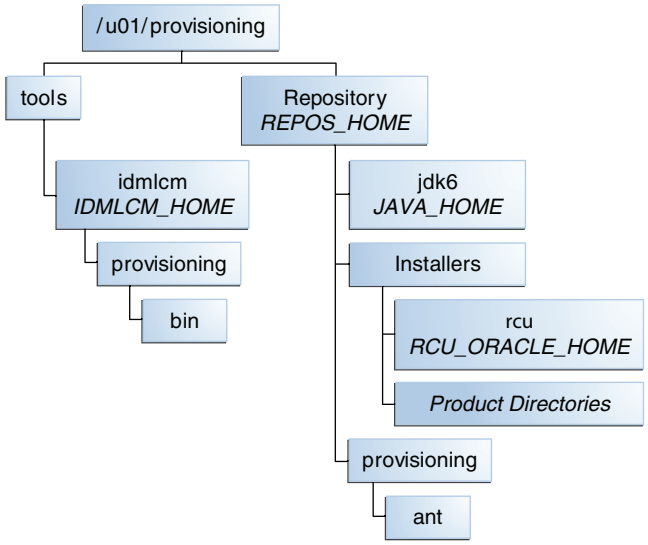
- Section 4.3.5.2, "Shared Storage"
- Section 4.3.5.3, "Local Storage"

4.3.5.1 Provisioning Repository

The provisioning repository is located on shared storage and is required only for the provisioning process. It is recommended that this share be made available to all hosts in the topology during provisioning. Once provisioning is complete, this share is no longer required.

If you cannot share this directory with your Web Tier, you must create a local copy of it in the DMZ.

Figure 4–1 Provisioning Repository



4.3.5.2 Shared Storage

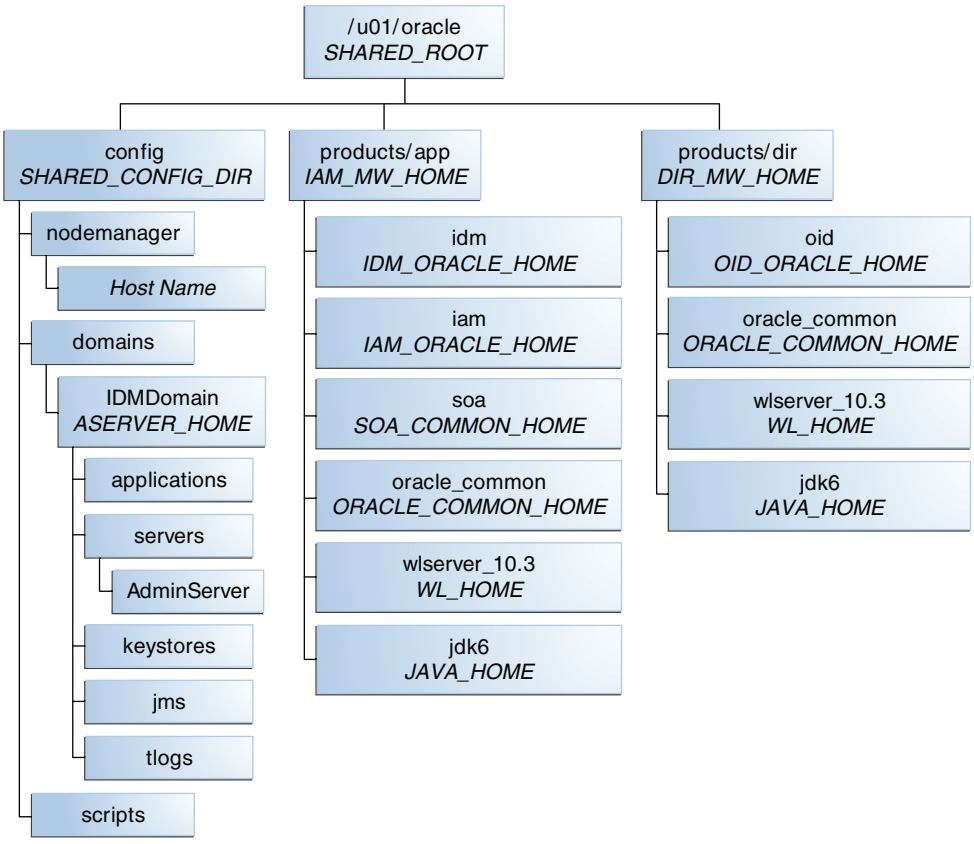
In an Enterprise Deployment, it is recommended that the volumes shown in Table 4–1 be created on shared Storage. You can mount shared storage either exclusively or shared.

When scaling out or scaling up, you can use the shared *MW_HOME* for additional servers of the same type without performing more software installations.

Table 4–1 Volumes on Shared Storage

Volume	Mount Point	Mounted on Hosts
`VOL1/OracleIDM`	`/u01/oracle`	IDMHOST1 IDMHOST2 LDAPHOST1 LDAPHOST2

Figure 4–2 Shared Storage



4.3.5.3 Local Storage

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

Table 4–2 Local Storage Directories

Tier	Environment Variable	Directory	Hosts
Web Tier	*WEB_MW_HOME*	`/u01/local/oracle/products/ohs`*n*	WEBHOST1 WEBHOST2
Web Tier	*WEB_ORACLE_INSTANCE*	`/u02/local/oracle/ohs`*n*	WEBHOST1 WEBHOST2
Directory Tier	*OID_ORACLE_INSTANCE*	`/u02/local/oracle/config/instances/oid`*n*	LDAPHOST1 LDAPHOST2
Directory Tier	*OVD_ORACLE_INSTANCE*	`/u02/local/oracle/config/instances/ovd`*n*	LDAPHOST1 LDAPHOST2
Application Tier	*MSERVER_HOME*	`/u02/local/oracle/config/domains/IDMDomain`	IDMHOST1 IDMHOST2

Figure 4–3 Local Storage Part 1

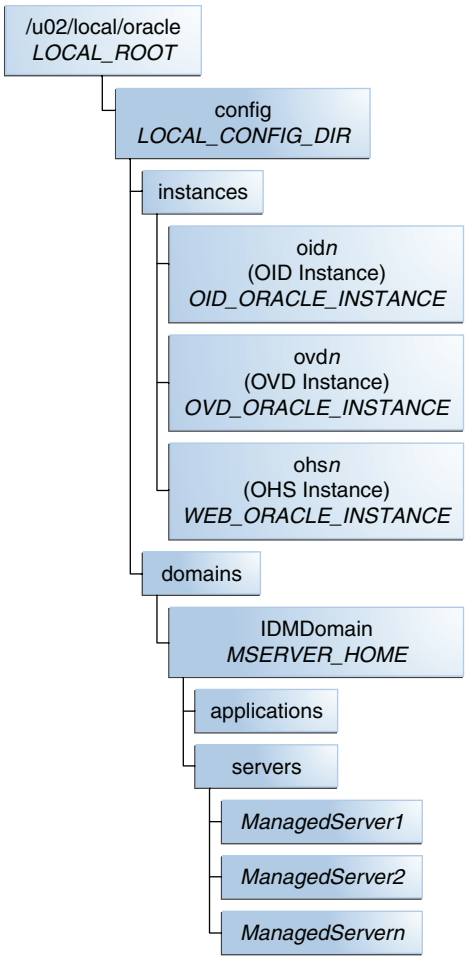
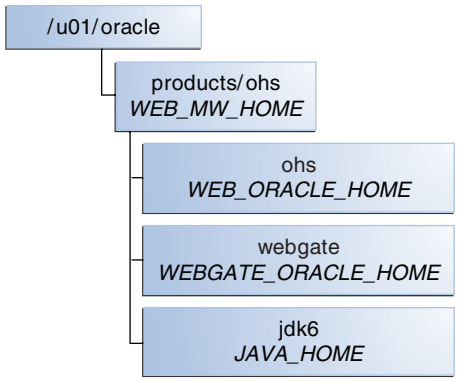


Figure 4–4 Local Storage Part 2



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.

4.4 Oracle Fusion Middleware Homes

During the provisioning process, the following Middleware homes and Oracle homes are created.

Table 4–3 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
OID_ORACLE_HOME	Contains the binary files for Oracle Internet Directory and Oracle Virtual Directory.	Oracle Internet Directory Oracle Virtual Directory
IDM_ORACLE_HOME	Contains the binary and library files for Oracle Identity Management and is located in: *MW_HOME*/idm	Oracle Directory Services Manager Oracle Identity Federation
IAM_ORACLE_HOME	Contains the binary and library files required for Oracle Identity and Access Management and is located in *MW_HOME*/iam.	Oracle Access Manager Oracle Identity Management
WEB_ORACLE_HOME	Contains the binary and library files required for OHS and is located in *WEB_MW_HOME*/ohs.	
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 *IAM_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
JAVA_HOME	This is the location where JRockit is installed	
ASERVER_HOME	This is the primary location of the domain configuration.	
MSERVER_HOME	This is a copy of the domain configuration used to start and stop managed servers.	
WEBGATE_ORACLE_HOME	This is the location of the WebGate installation.	
REPOS_HOME	Fusion applications Software Repository.	Location of the Software Repository
IDMLCM_HOME	Contains the provisioning software.	Location of the Provisioning Tool

For a list of directory variables used in this guide, see Section 4.2, "Terminology for Directories and Directory Variables."

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 2.2, "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 provisioning you must perform the following tasks:

1. Install a certified operating system.
2. Install all necessary patches and packages as listed in *Oracle Fusion Applications Installation Guide*.

5.4.1 Meeting UNIX and Linux Requirements

This section includes the following topics:

- Section 5.4.1.1, "Configuring 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 Configuring 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 cluster.

The values in the following table are the current Linux recommendations. For more information, refer to *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 150000.

> **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  150000
* hard  nofile  150000
* soft  nproc   16384
* hard  nproc   16384
```

If you are installing on Oracle Linux Server Release 6, also edit the `/etc/security/limits.d/90-nproc.conf` file, and ensure it has the following line:

```
* soft nproc 16384
```

After editing the files, reboot the machine.

See the *Oracle Fusion Middleware System Requirements and Specifications* for the latest suggested value.

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.

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	IDMHOST1

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, perform the steps in this section. For operating systems other than Linux, refer to your manufacturer documentation.

To enable 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
```

Enable your network to register the new location of the virtual IP address:

```
/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
```

In the following example, the IP address is enabled on the interface `Local Area Connection`.

```
netsh interface ip add address "Local Area connection" 100.200.140.206
255.255.255.0
```

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. For more information, see Chapter 4.3.5.2, "Shared Storage."

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 users on each node.

- `oracle`–The user that owns 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.
- `nobody`–An unprivileged user.

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
```

5.10 Synchronize Oracle Internet Directory Nodes

Synchronize the time on the individual Oracle Internet Directory nodes using Greenwich Mean Time so that there is a discrepancy of no more than 250 seconds between them.

Note: If OID Monitor detects a time discrepancy of more than 250 seconds between the two nodes, the OID Monitor on the node that is behind stops all servers on its node. To correct this problem, synchronize the time on the node that is behind in time. The OID Monitor automatically detects the change in the system time and starts the Oracle Internet Directory servers on its node.

6 Preparing for Provisioning

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

This chapter contains the following topics:

6.1 Assembling Information for Identity Management Provisioning

Assemble the following information prior to provisioning. You can print out the table from the PDF version of this Guide and record your own values.

Table 6–1 Provisioning Information

Description	Variable	Documented Value	Customer Value
Software Repository Location	*REPOS_HOME*	`/u01/provisioning/Repository`	
Software Installation Location	*SHARED_ROOT*	`/u01/oracle`	
Shared Configuration Location	*SHARED_CONFIG_DIR*		
Local Configuration Location	*LOCAL_ROOT*	`/u02/local/oracle`	
Common IDM Password for IDM provisioning wizard	*COMMON_IDM_PASSWORD*		
Identity & Access Management Host 1		IDMHOST1.mycompany.com	
Admin Server virtual host		ADMINVHN.mycompany.com	
Identity & Access Management Host 2		IDMHOST2.mycompany.com	

Table 6–1 (Cont.) Provisioning Information

Description	Variable	Documented Value	Customer Value
ODSM Port, Second ODSM PORT	*ODSM_PORT*, *Second_ODSM_PORT*	7005	
Directory Host 1		LDAPHOST1.mycompany.com	
Directory Host 2		LDAPHOST2.mycompany.com	
OID Realm DN,	*REALM_DN*	dc=mycompany,dc=com	
OID Identity Store Service Name	*IDSTORE_SERVICENAME*	OIDEDG.mycompany.com	
OID Identity Store Schema Password	*IDSTORE_PASSWORD*		
OID Identity Store Host VIP Names/SCAN Address	*SCAN_ADDRESS*	DB-SCAN.mycompany.com	
OID Identity Store Listener Port	*DB_LSNR_PORT*	1521	
OID Identity Store Instance Names		OIDDB1, OIDDB2	
OID Policy Store Realm DN	*POLICY_STORE_REALM_DN*	dc=mycompany,dc=com	
First OIM Server virtual host		OIMHOST1VHN.mycompany.com	
Second OIM Server virtual host		OIMHOST2VHN.mycompany.com	
OIM Port, Second OIM Port	*OIM_PORT*	14000	
Email Outgoing Server Name	*EMAIL_SERVER*	EMAIL.mycompany.com	
Email Outgoing Server Port	*EMAIL_PORT*	465	
Email User Name	*EMAIL_USER*	username	
Email Password	*EMAIL_PASSWORD*		
First SOA Server virtual host		SOAHOST1VHN.mycompany.com	
Second SOA Server virtual host		SOAHOST2VHN.mycompany.com	
SOA Ports, Hosts 1 and 2	*SOA_PORT*	8001	
OIM DB Service Name	*IDSTORE_SERVICENAME*	OIMEDG.mycompany.com	
OIM DB Schema Password	*IDSTAORE_PASSWORD*		
IAM DB VIP Names/SCAN Address	*SCAN_ADDRESS*	MDB-SCAN.mycompany.com	
IAM Listener Port	*DB_LSNR_PORT*	1521	
IAM DB Instance Name		MDB1, MDB2	
Outgoing Email Server Name	*EMAIL_SERVER*		
Outgoing Email Server Port	*EMAIL_PORT*		

Table 6–1 (Cont.) Provisioning Information

Description	Variable	Documented Value	Customer Value
Outgoing Email Security		SSL	
Email Username	*EMAIL_USER*,		
Email Password	*EMAIL_PASSWORD*		
OAM Port, Second OAM Port	*OAM_PORT*	14100	
OAM Transfer Mode		Simple. (Open on AIX.)	
OAM Cookie Domain	*OAM_COOKIE_ DOMAIN*	`.mycompany.com`	
OAM WebGate			
First OHS host		WEBHOST1.mycompany.com	
Second OHS host		WEBHOST2.mycompany.com	
OHS Port, Second OHS Port	*OHS_PORT*	7777	
OHS SSL Port, Second OHS SSL Port	*OHS_SSL_PORT*	4443	
Load Balancer Admiin Virtual Host Name		ADMIN.mycompany.com	
Load Balancer Admin Port	*HTTP_PORT*	80	
Load Balancer Admin Port is SSL?		No	
Load Balancer Internal Callbacks Virtual Host Name		IDMINTERNAL.mycompany.com	
Load Balancer Internal Callbacks Port	*HTTP_PORT*	80	
Load Balancer Internal Callbacks Port is SSL?		No	
Load Balancer SSO Virtual Host Name		SSO.mycompany.com	
Load Balancer SSL Port	*HTTP_SSL_PORT*	443	
Load Balancer ID Store Virtual Host Name		IDSTORE.mycompany.com	
Load Balancer ID Store Port (OVD)	*LDAP_LBR_PORT*	389	
Load Balancer ID Store SSL Port (OVD)	*LDAP_LBR_SSL_ PORT*	636	
Load Balancer Policy Store Virtual Host Name		POLICYSTORE.mycompany.com	
LDAP Port	*OID_LDAP_PORT*	389	
LDAP SSL Port	*OID_LDAP_SSL_ PORT*	636	

6.2 Disable Oracle Internet Directory Monitoring

Before beginning Identity Management Provisioning, disable Oracle Internet Directory monitoring on the load balancer and leave it disabled until provisioning is complete.

6.3 Creating an Oracle Fusion Applications Provisioning Repository

The software required by Oracle Identity Management is located in the Oracle Fusion Applications Provisioning Repository. If you have not already done so then you need to create an Oracle Fusion Applications Provisioning Repository as described in "Creating the Provisioning Repository" in *Oracle Fusion Applications Installation Guide*.

Unzip the RCU zip file *REPOS_HOME*/installers/fmw_rcu/linux/rcuHome.zip to:

REPOS_HOME/installers/rcu

6.4 Verifying Java

Make sure that your Provisioning Repository contains Java. It should reside in a directory called jdk6.

6.5 Installing the IDM Provisioning Wizard

The IDM Provisioning Wizard must be visible to each host in the topology. The wizard is only required during the provisioning process, and can be removed after provisioning.

The installation script for the IDM Lifecycle Tools (IDM Provisioning Wizard and IDM Patching Tools) resides in the directory:

REPOS_HOME/installers/idmlcm/idmlcm/Disk1

To begin installing the tools, change to that directory and start the script.

```
cd REPOS_HOME/installers/idmlcm/idmlcm/Disk1
./runInstaller -jreLoc REPOS_HOME/jdk6
```

Then proceed as follows:

1. On the Welcome screen, click **Next**.
2. If you are running the Wizard on a UNIX platform, you are prompted for the location of the **Inventory Directory**, which is used to keep track of all Oracle products installed on this host.

   In the **Operating System Group ID** field, select the group whose members you want to grant access to the inventory directory. All members of this group can install products on this host. Click **OK** to continue.

   The **Inventory Location Confirmation** dialog prompts you to run the *inventory_directory*/createCentralInventory.sh script as root to create the /etc/oraInst.loc file. This file is a pointer to the central inventory and must be present for silent installations. It contains two lines:

   inventory_loc=*path_to_central_inventory*

   inst_group=*install_group*

   The standard location for this file is /etc/oraInst.loc, but it can be created anywhere. If you create it in a directory other than /etc, you must include the

-invPtrLoc argument and enter the location of the inventory when you run the Identity Management Provisioning Wizard or the runIDMProvisioning script.

If you do not have root access on this host but want to continue with the installation, select **Continue installation with local inventory**.

Click **OK** to continue.

3. On the Prerequisite Checks screen, verify that checks complete successfully, then click **Next**.
4. On the Specify Install Location screen, enter the following information:
   a. Oracle Middleware Home - This is the parent directory of the directory where the Identity Management Provisioning Wizard will be installed. This must be on shared storage for example:

      ```
      /u01/provisioning/tools
      ```

   b. Oracle Home Directory - This is a subdirectory of the above directory where the wizard will be installed. For example:

      ```
      idmlcm
      ```

   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**.

6.6 Applying Patch 17434914

Without this patch, the IDM Provisioning tools set up JMS queues incorrectly. Apply this patch before performing provisioning.

6.7 Checking Port Availability

Before starting to provision your environment, you must ensure that none of the ports you intend to use is already in use.

To do this, perform the following steps:

1. Log on to the machine that the component will run on.
2. Check that no process is running using that port using the command:

   ```
   netstat -an | grep port
   ```

   where *port* is the port number you are checking for.

   For example, for Oracle HTTP server the command is:

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

For a full list of the default ports, see Table 3–3, " Ports Used in the Oracle Identity Management Enterprise Deployment Topologies".

7

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:

7.1 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 7.2, "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 7.3, "Installing the Database for an Enterprise Deployment."
- Create database services, as described in Section 7.4, "Creating Database Services."
- Create the required Oracle schemas in the database using the Repository Creation Utility (RCU). See Section 7.6, "Loading the Identity Management Schemas in the Oracle RAC Database by Using RCU."

7.2 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:

7.2.1 Databases Required

For Oracle Identity management, a number of separate databases are recommended. Table 7–1 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 7–1 Mapping between Topologies, Databases and Schemas

Topology Type	Database Names	Database Hosts	Service Names	Schemas in Database
Oracle Access Manager 11*g* and Oracle Identity Manager 11*g* (OAM11g/OIM11g)	OIDDB	OIDDBHOST1 OIDDBHOST2	`OIDEDG.mycompany.com`	ODS
	IDMDB	IDMDBHOST1 IDMDBHOST2	`IAMEDG.mycompany.com`	OAM, IAU, ORASDPM
			`IAMEDG.mycompany.com`	MDS[1], OIM, SOAINFRA
Oracle Identity Federation 11g (OIF11g/OAM11g)	IDMDB	IDMDBHOST1 IDMDBHOST2	`IAMEDG.mycompany.com`	OAM, IAU, ORASDPM
			`IAMEDG.mycompany.com`	MDS, OIM, SOAINFRA
			`IAMEDG.mycompany.com`	OIF
	OIDDB	OIDDBHOST1 OIDDBHOST2	`OIDEDG.mycompany.com`	ODS

[1] The SOA and Oracle Identity Manager components share the MDS repository.

Notes: Two databases are recommended for Oracle Internet Directory and Oracle Access Manager/Oracle Identity Manager because they are likely to have different configuration requirements. If desired, they can be combined into a single database.

The following sections apply to all the databases listed in Table 7–1.

7.2.2 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.

7.2.3 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%';
```

7.2.4 Patching the Oracle Database

Patches are required for some versions of Oracle Database.

7.2.4.1 Patch Requirements for Oracle Database 11g (11.1.0.7)

Table 7–2 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 7–2 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

7.2.4.2 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 7–3 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 7–3 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 metalink 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]" 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>.

7.2.5 About Initialization Parameters

The databases must have the following minimum initialization parameters defined:

Table 7–4 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.

If the database is being used for Oracle Internet Directory, it must have the following minimum initialization parameters defined:

Table 7–5 Minimum Initialization Parameters for Oracle RAC Oracle Internet Directory Databases

Parameter	Value
aq_tm_processes	1
dml_locks	200
job_queue_processes	10
open_cursors	800
session_max_open_files	50
sessions	500
processes	2500
sga_target	4G
pga_aggregate_target	2G
sga_max_size	4G
session_cached_cursors	500
_b_tree_bitmap_plans	FALSE
parallel_max_servers[1]	1

[1] Only required for Oracle Internet Directory Databases where the Oracle RAC system has more than 32 CPUs.

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

7.3 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.

7.4 Creating Database Services

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

- Section 7.4.1, "Why Create Database Services?"
- Section 7.4.2, "Creating Database Services for 10.x and 11.1.x Databases"
- Section 7.4.3, "Creating Database Services for 11.2.x Databases"
- Section 7.4.4, "Database Tuning"

7.4.1 Why Create Database Services?

In an Oracle Grid Database, it is recommended that you create a service for each application. Although this is not a requirement, doing so provides the following benefits:

1. You can target applications to specific database instances.
2. You can move applications between instances.
3. As demand increases or decreases, you can change the number of instances an application can use.

7.4.2 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 Section 7–1, " Mapping between Topologies, 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 IAMEDG.mycompany.com for Oracle Access Manager:

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

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

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

3. Start the service using `srvctl`:

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

When creating a service in the database for Oracle Internet Directory, ensure that the service is enabled for high-availability notifications and configured with the proper server-side Transparent Application Failover (TAF) settings. Use the `DBMS_SERVICE` package to create the service to enable high availability notification to be sent through Advanced Queuing (AQ) by setting the `AQ_HA_NOTIFICATIONS` attribute to `TRUE` and configure server-side Transparent Application Failover (TAF) settings, as follows:

1. Use the CREATE_SERVICE subprogram to both create the database service and enable high-availability notification and configure server-side Transparent Application Failover (TAF) settings.

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

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

   Then execute this command:

   ```
   EXECUTE
   DBMS_SERVICE.CREATE_SERVICE(
   SERVICE_NAME => 'OIDEDG.mycompany.com',
   NETWORK_NAME => 'OIDEDG.mycompany.com',
   AQ_HA_NOTIFICATIONS => TRUE,
   FAILOVER_METHOD => DBMS_SERVICE.FAILOVER_METHOD_BASIC,
   FAILOVER_TYPE => DBMS_SERVICE.FAILOVER_TYPE_SELECT,
   FAILOVER_RETRIES => 5, FAILOVER_DELAY => 5);
   ```

   Note: The `EXECUTE DBMS_SERVICE` command shown must be entered on a single line to execute properly.

   For more information about the `DBMS_SERVICE` package, see the *Oracle Database PL/SQL Packages and Types Reference*.

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

   ```
   srvctl add service -d oiddb -s OIDEDG.mycompany.com -r oiddb1,oiddb2
   ```

3. Start the service using srvctl:

   ```
   srvctl start service -d oiddb -s OIDEDG.mycompany.com
   ```

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

7.4.3 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 7–1, " Mapping between Topologies, Databases and Schemas".

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

```
srvctl add service -d oiddb -s OIDEDG.mycompany.com -r idmdb1,idmdb2 -q TRUE -m
BASIC -e SELECT -w 5 -z 5
```

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)

Note: Transparent Application Failover (TAF) and AQ settings are only required when creating a service for Oracle Internet Directory.

2. Start the Service using `srvctl start service`

```
srvctl start service -d oiddb -s OIDEDG.mycompany.com
```

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

```
srvctl status service -d oiddb -s OIDEDG.mycompany.com
Service OIDEDG.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 oiddb -s OIDEDG.mycompany.com
Service name: OIDEDG.mycompany.com
Service is enabled
Server pool: OIDDB_OIDEDG.mycompany.com
Cardinality: 2
Disconnect: false
Service role: PRIMARY
Management policy: AUTOMATIC
DTP transaction: false
AQ HA notifications: true
Failover type: SELECT
Failover method: BASIC
TAF failover retries: 5
TAF failover delay: 5
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*.

7.4.4 Database Tuning

The database parameters defined in Section 7.3, "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);
```

7.5 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
```

7.6 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 one of the Oracle RAC nodes. 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 `OIDEDG.mycompany.com`.
- **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, for example, enter `FA`.

   Note: All schemas except for the ODS schema are required to have a prefix. In this release, the RCU prefix must be `FA`.

   Components: Select the schemas shown in the following table:

Product	RCU Option	Service Name	Comments
Oracle Internet Directory	Identity Management–Oracle Internet Directory	`OIDEDG.mycompany.com`	
Oracle Access Manager	Identity Management–Oracle Access Manager	`IAMEDG.mycompany.com`	Audit Services will also be selected.
Oracle Identity Manager	Identity Management–Oracle Identity Manager	`IAMEDG.mycompany.com`	Metadata Services, SOA infrastructure, and User Messaging will also be selected.
Oracle Identity Federation	Identity Management–Oracle Identity Federation	`IAMEDG.mycompany.com`	

   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 7–1 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.

7.7 Backing up the Database

After you have prepared your database, back it up as described in Section 16.5.3.3, "Backing Up the Database."

8

Creating a Provisioning Profile

This chapter describes how to create a provisioning profile by using the Identity Management Provisioning Wizard.

Before you can perform provisioning, you must provide information about your topology to the Identity Management Provisioning Wizard. Once you have provided all the necessary input, the wizard will create a provisioning file that you can use to perform the provisioning operation.

Refer to the information you assembled in Section 6.1, "Assembling Information for Identity Management Provisioning."

This chapter contains the following sections:

8.1 Running the Identity Management Provisioning Wizard to Create a Profile

To start the Identity Management Provisioning Wizard, execute the following commands from: *IDMCLM_HOME*/provisioning/bin

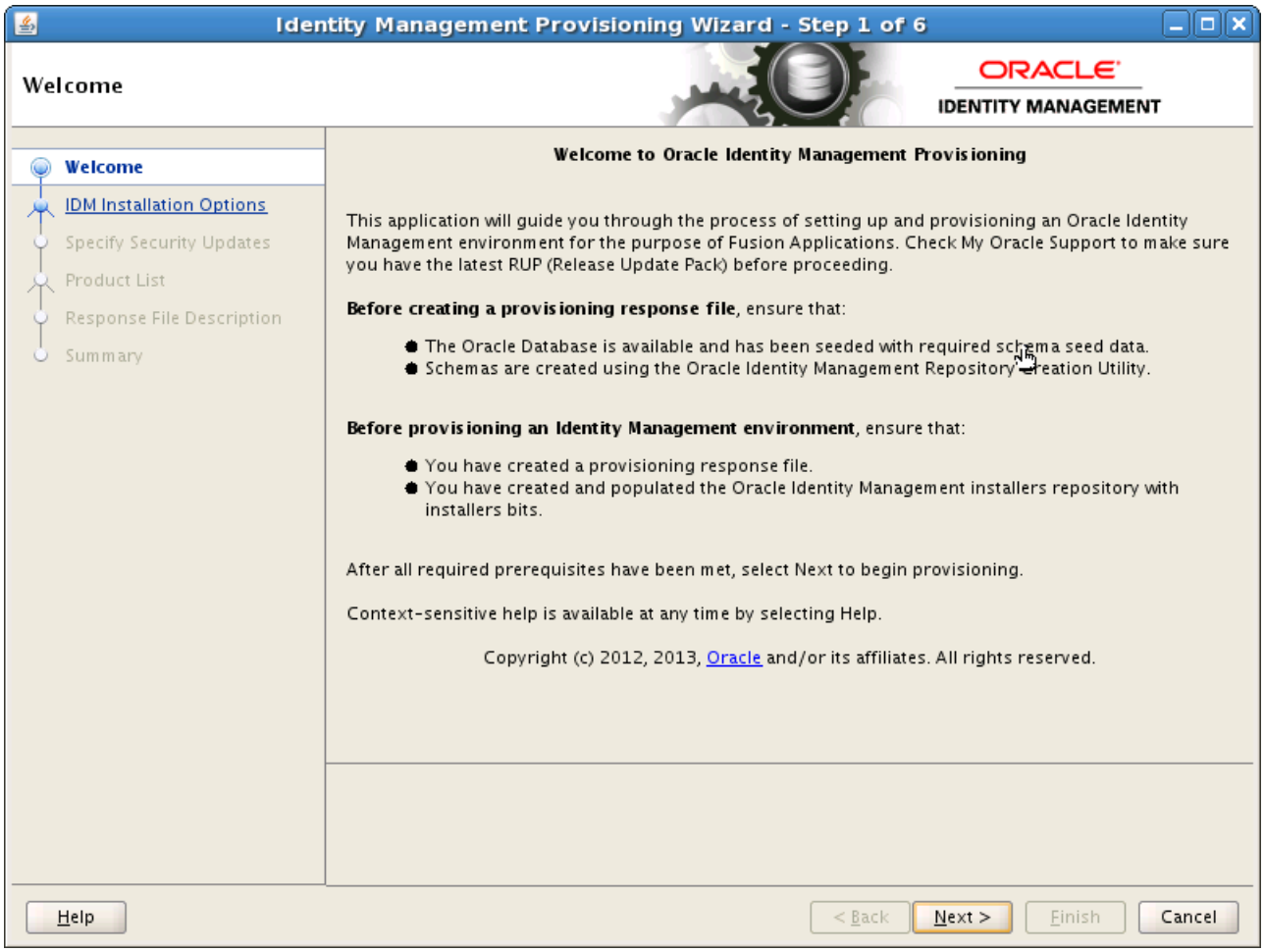
Set JAVA_HOME to *REPOS_HOME*/jdk6.

Issue the command:

```
./idmProvisioningWizard.sh
```

When the wizard starts, proceed as follows:

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

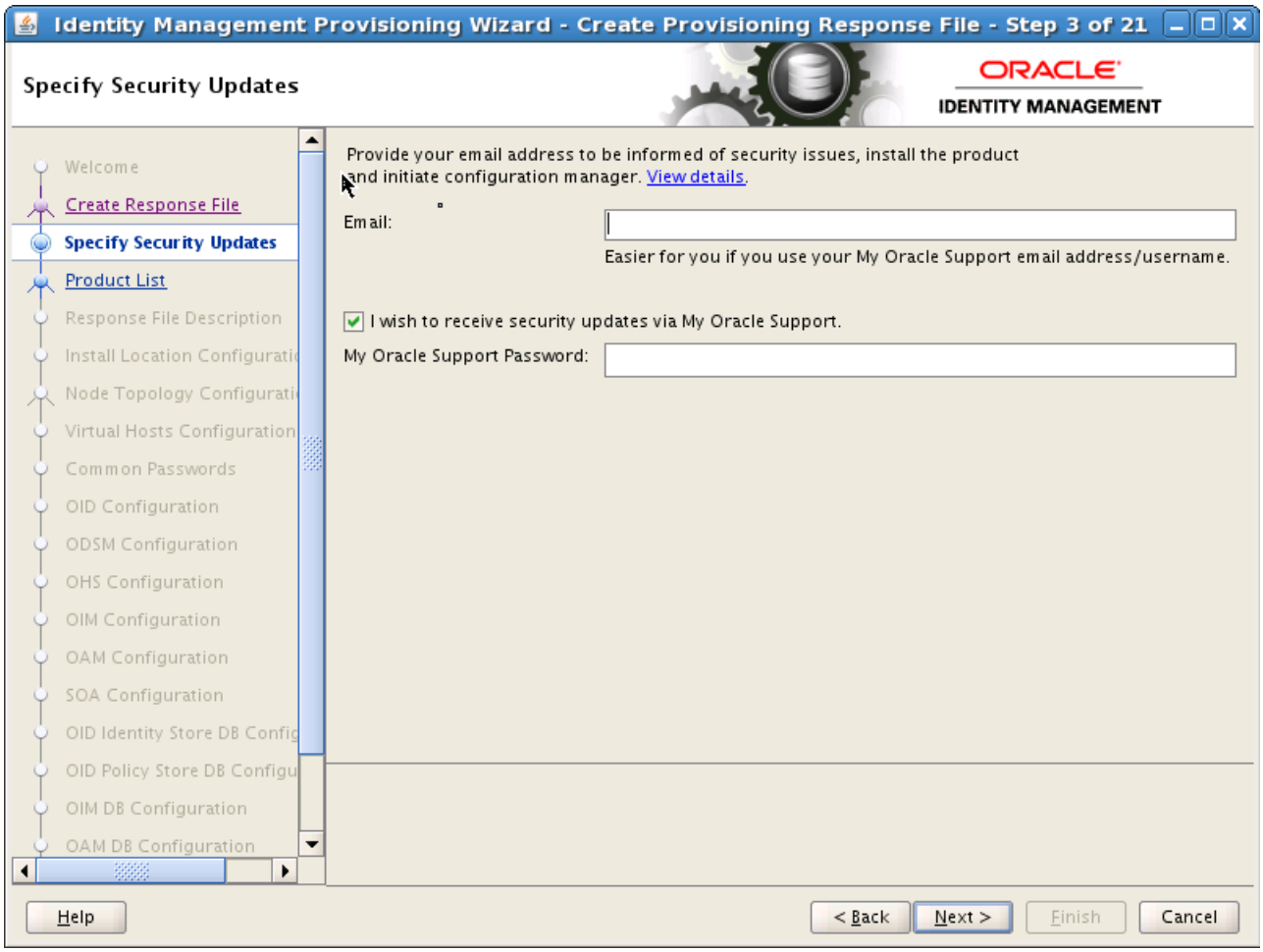


2. If you are prompted for the location of the **Inventory Directory**, proceed as described in Step 2 in Section 6.5, "Installing the IDM Provisioning Wizard."

   Click **OK** to continue.

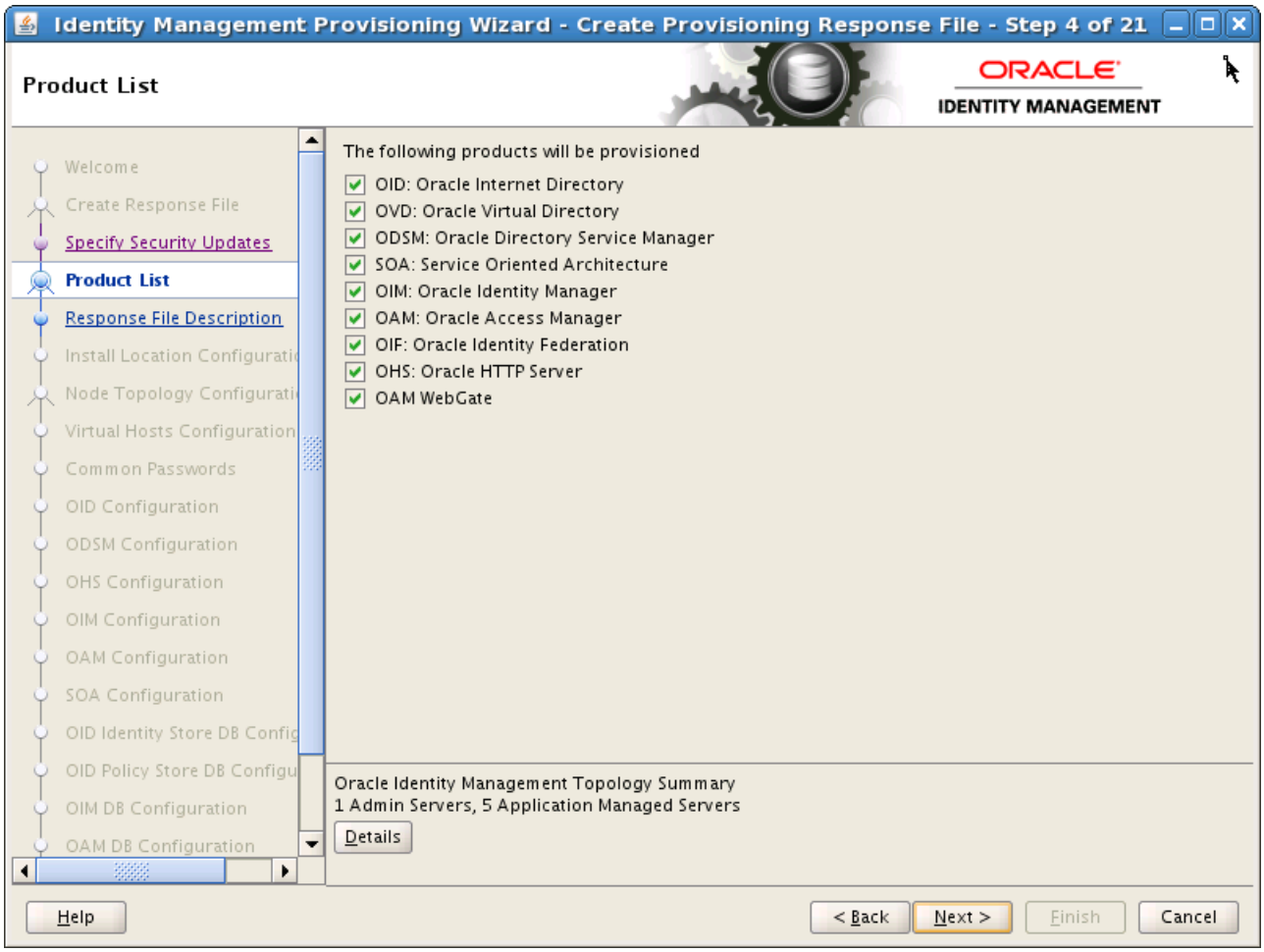
3. On the IDM Installation Options screen, select **Create a New Identity Management Environment Response File**, and click **Next**.

4. On the Specify Security Updates screen, choose whether to register with Oracle Support for updates or search for updates locally.



Click **Next**.

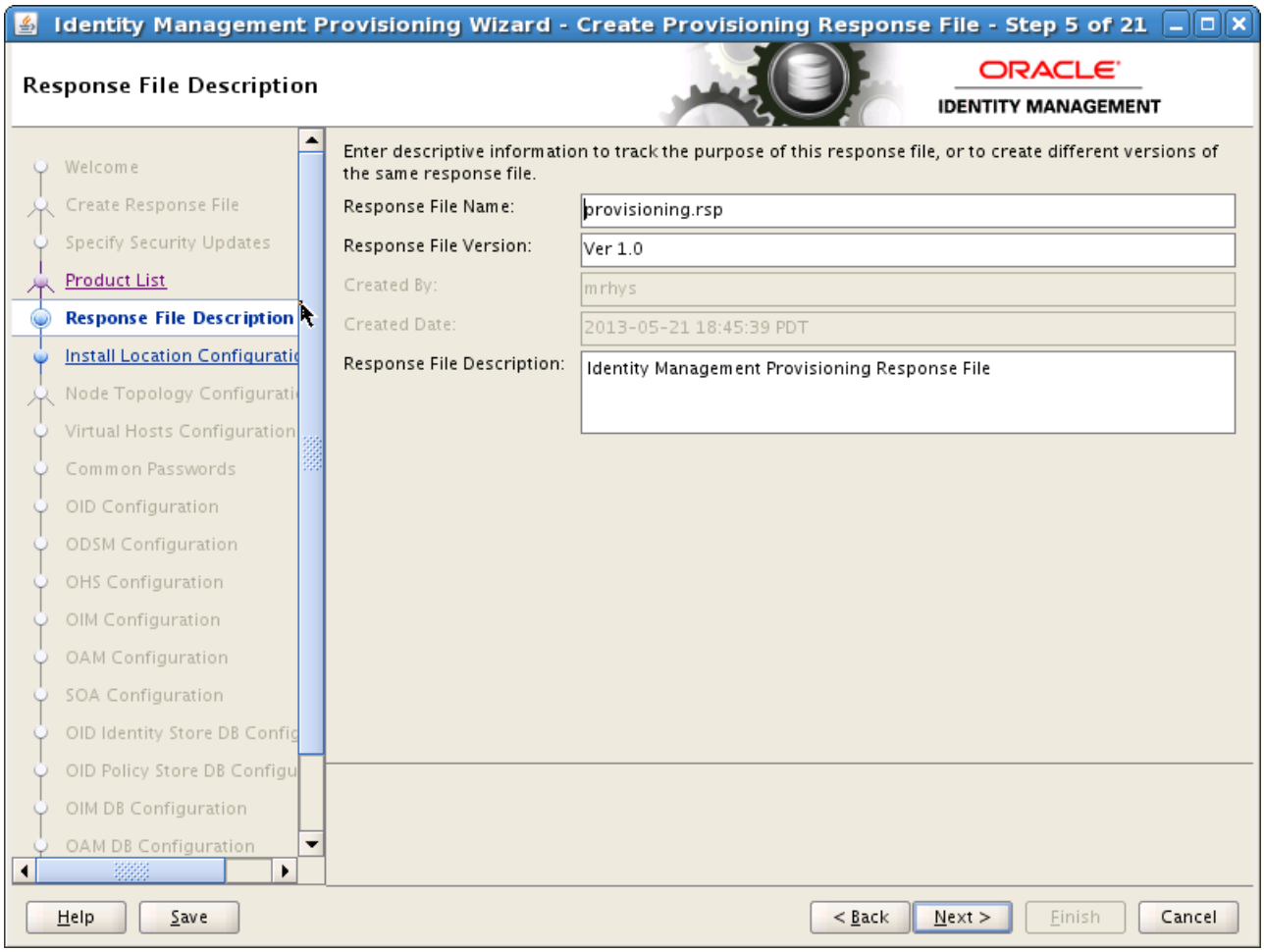
5. The Product List screen is purely informational. Click **Next**:



6. On the Response File Description Screen, enter the following information:

   - **Response File Name**: `provisioning.rsp`
   - **Response File Version**: `Ver 1.0`
   - **Response File Description**: A description such as `Provisioning Response File`

   Click **Next**.

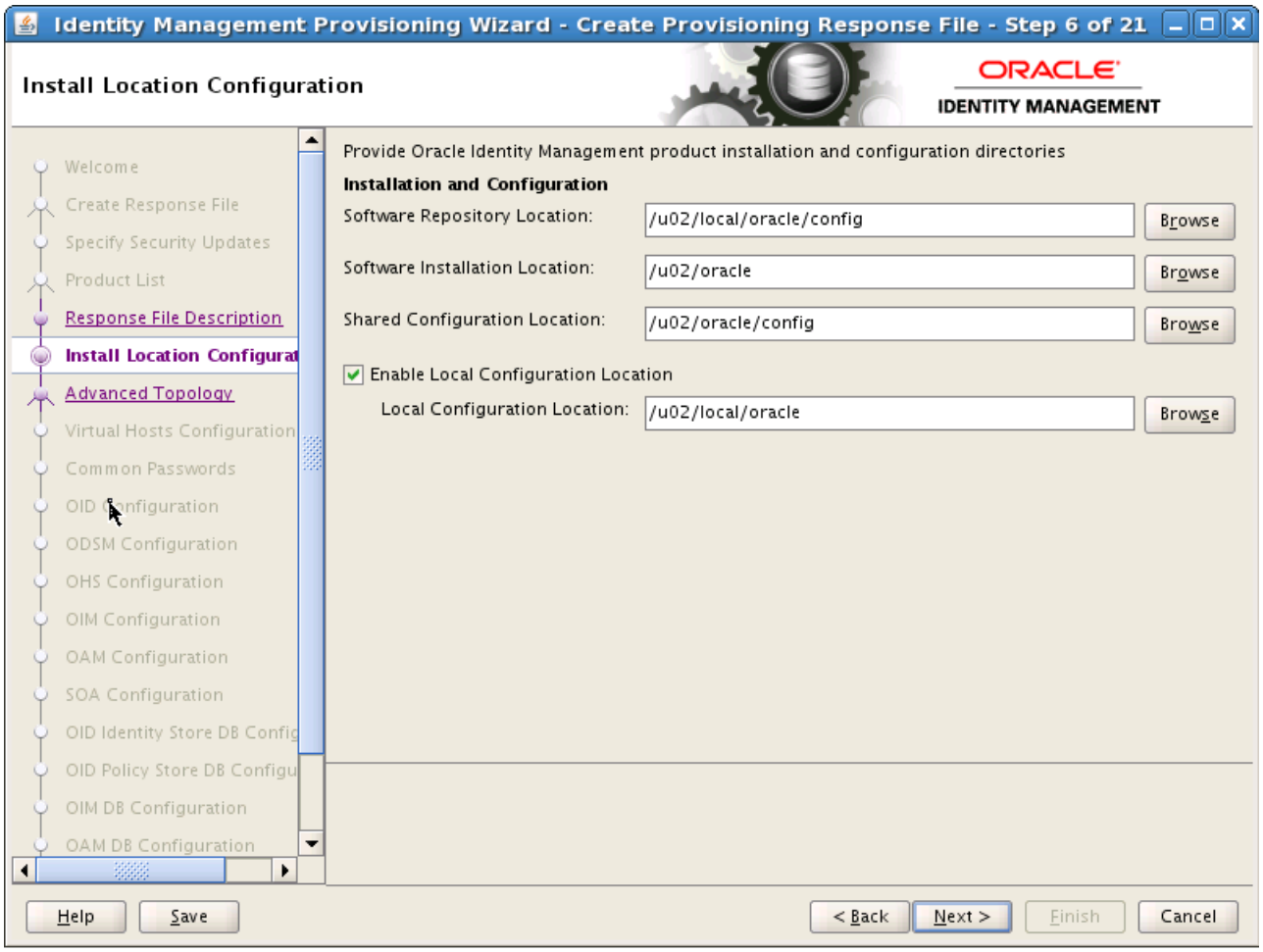


7. On the Install Location Configuration Screen, enter the following information:

   - **Software Repository Location**: This is the location of the Provisioning repository, for example: *REPOS_HOME* in the worksheet.
   - **Software Installation Location**: This is the location on shared storage under where you want the Middleware Home to be placed, for example: *SHARED_ROOT*

     Note: Note: The maximum length of this location is 59 characters in this release.

   - **Shared Configuration Location** Enter the location of shared configuration, for example *SHARED_CONFIG_DIR*.
   - **Enable Local Applications Configuration** Select this for Enterprise Deployments.
   - **Local Configuration Location**: This is the location on local storage where you want the Oracle HTTP Server Middleware home and local configuration files to be stored, for example: *LOCAL_CONFIG_DIR*. Click **Next**.



8. On the Node Topology Configuration screen, select **EDG Topology** and provide the following information:

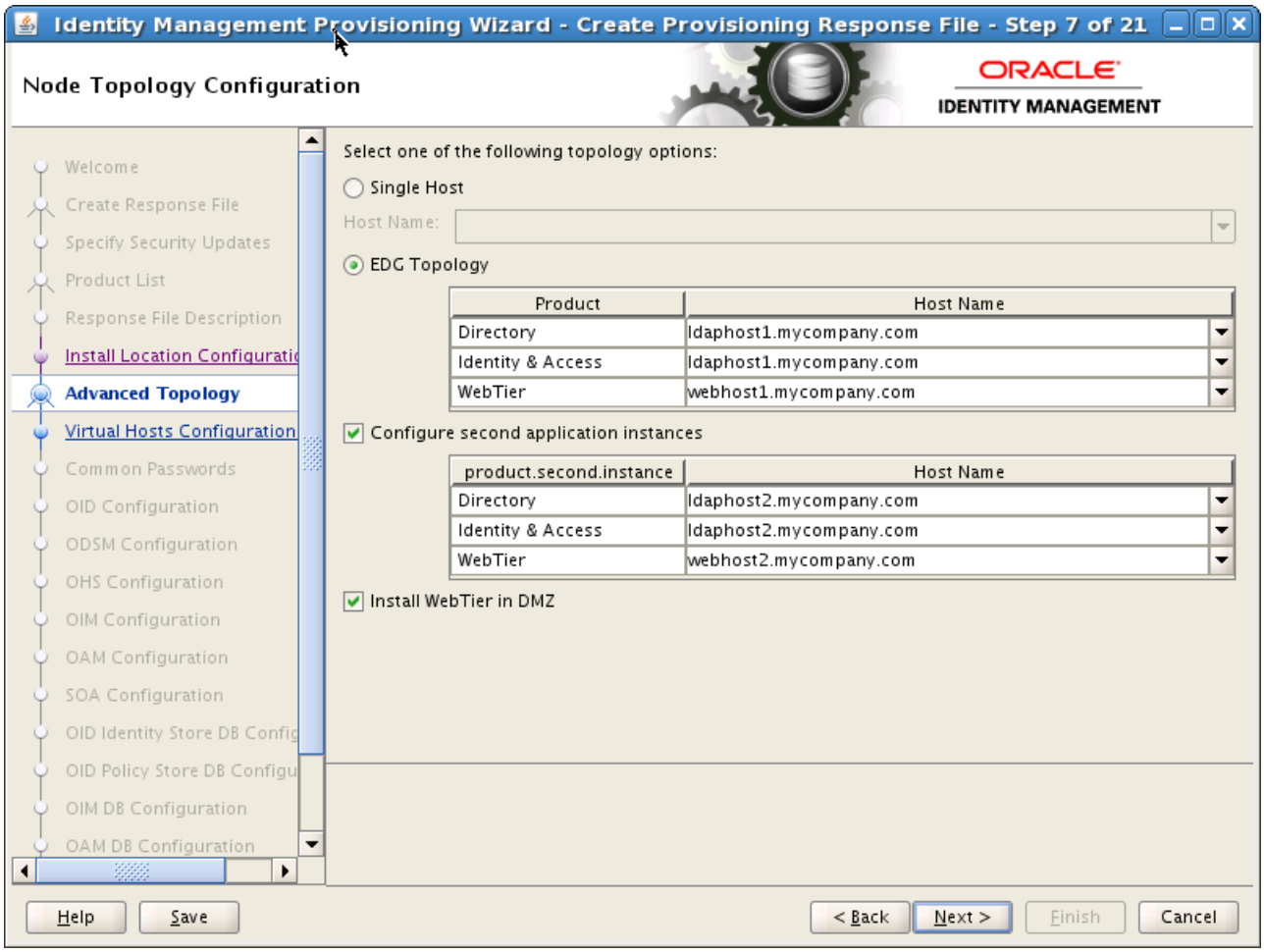
> **Note:** All host names must be fully qualified.

Ensure **Configure Secondary Instances** is selected and enter:

- **Directory**: `LDAPHOST1.mycompany.com`
- **Identity & Access**: `IDMHOST1.mycompany.com`
- **Web Tier**: `WEBHOST1.mycompany.com`

Ensure **Install WebTier in DMZ** is selected.

- **Directory**: `LDAPHOST2.mycompany.com`
- **Identity & Access**: `IDMHOST2.mycompany.com`
- **Web Tier**: `WEBHOST2.mycompany.com`



Notes:

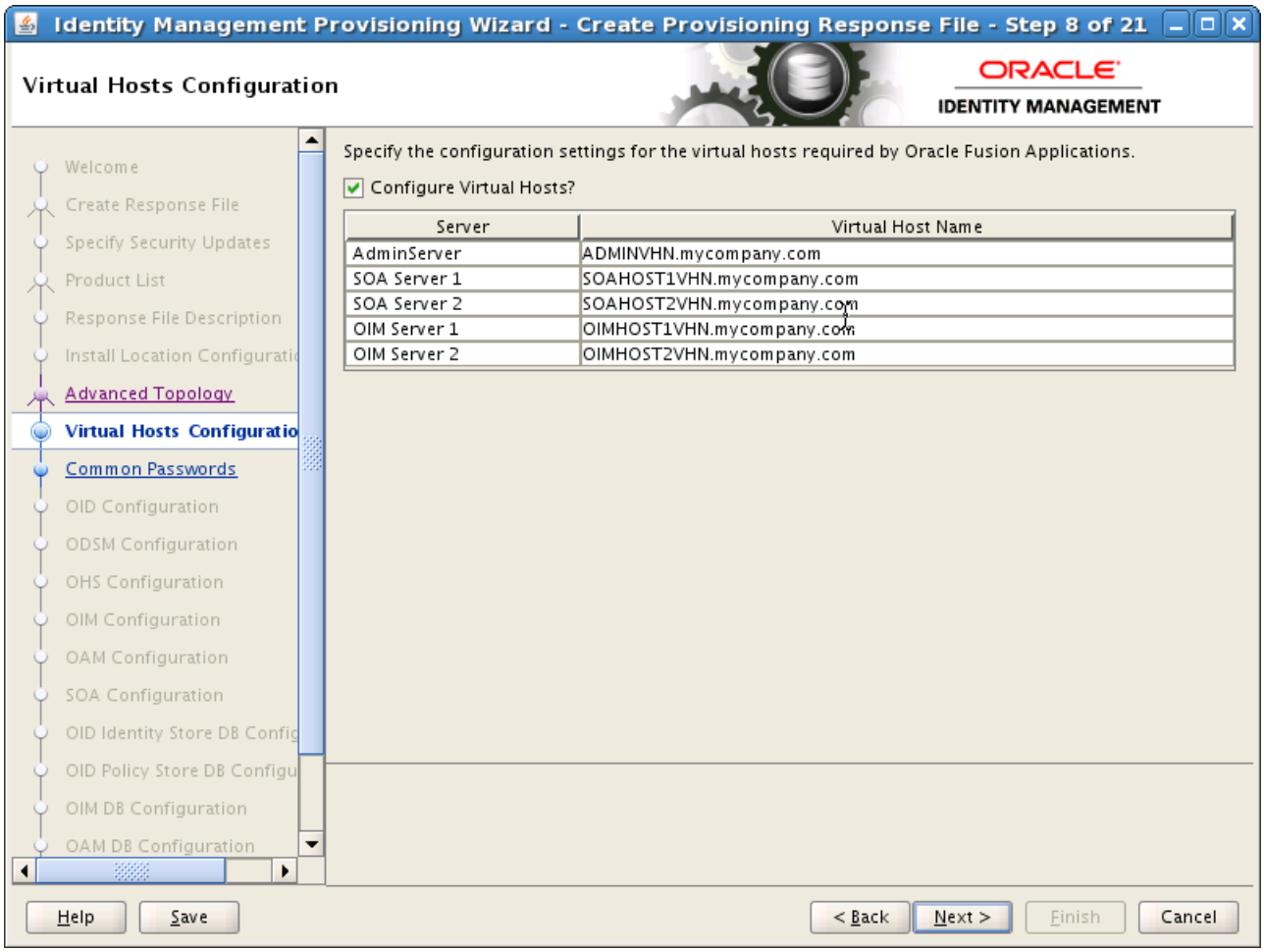
- OHS is not placed on the same host as a mid tier or LDAP component. In the topologies described in this guide, OHS is located in a DMZ for added security.
- OHS cannot be located on an LDAP host

Click **Next**.

9. On the Virtual Hosts Configuration screen, select **Configure Virtual Hosts**.

   Enter the **Virtual Host Name** for each managed **Server** in the topology, for example:

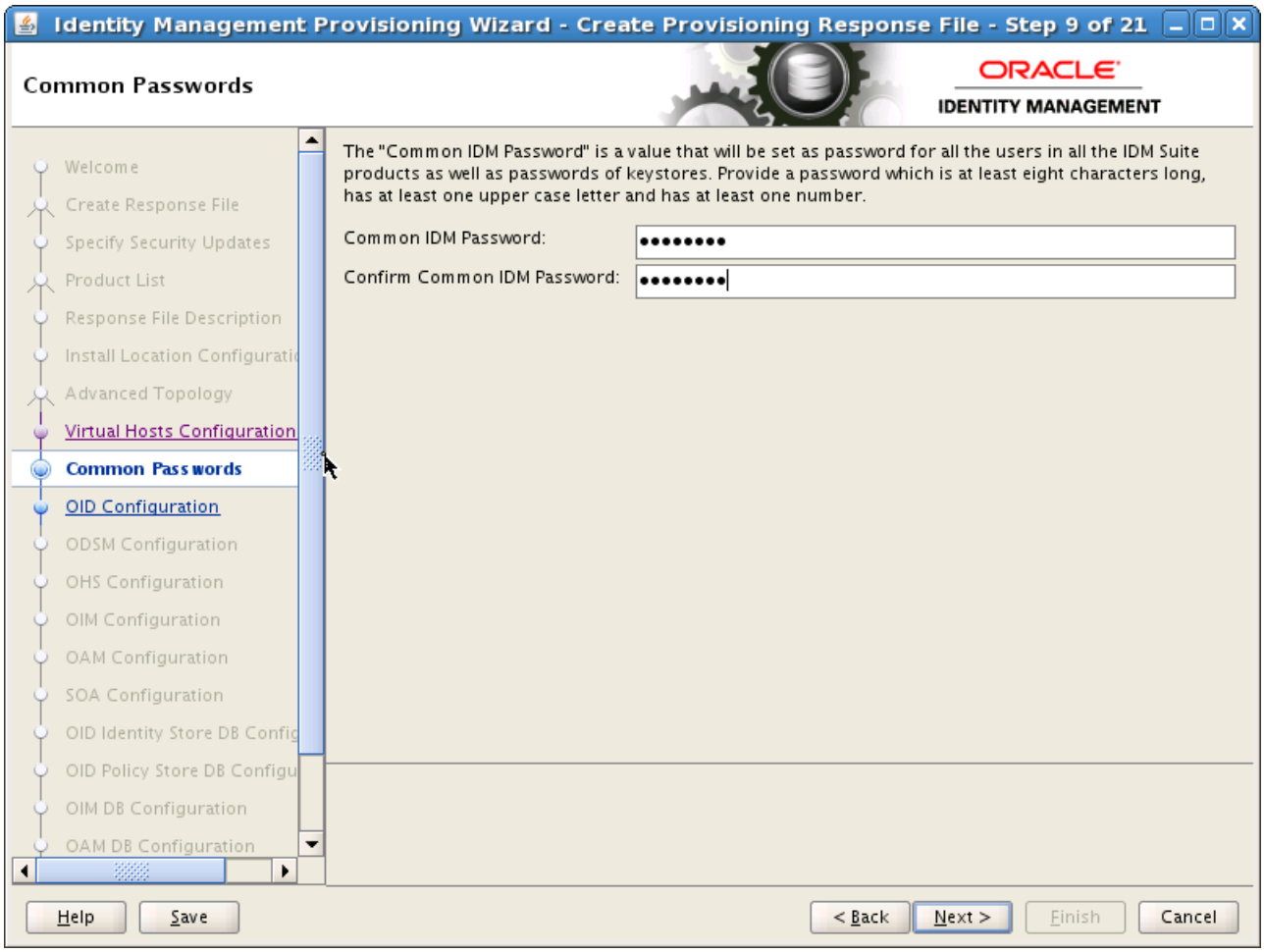
   - **Admin Server**: `ADMINVHN.mycompany.com`
   - **SOA Server**: `SOAHOST1VHN.mycompany.com`
   - **SOA Server 2**: `SOAHOST2VHN.mycompany.com`
   - **OIM Server**: `OIMHOST1VHN.mycompany.com`
   - **OIM Server 2**: `OIMHOST2VHN.mycompany.com`



Click **Next**.

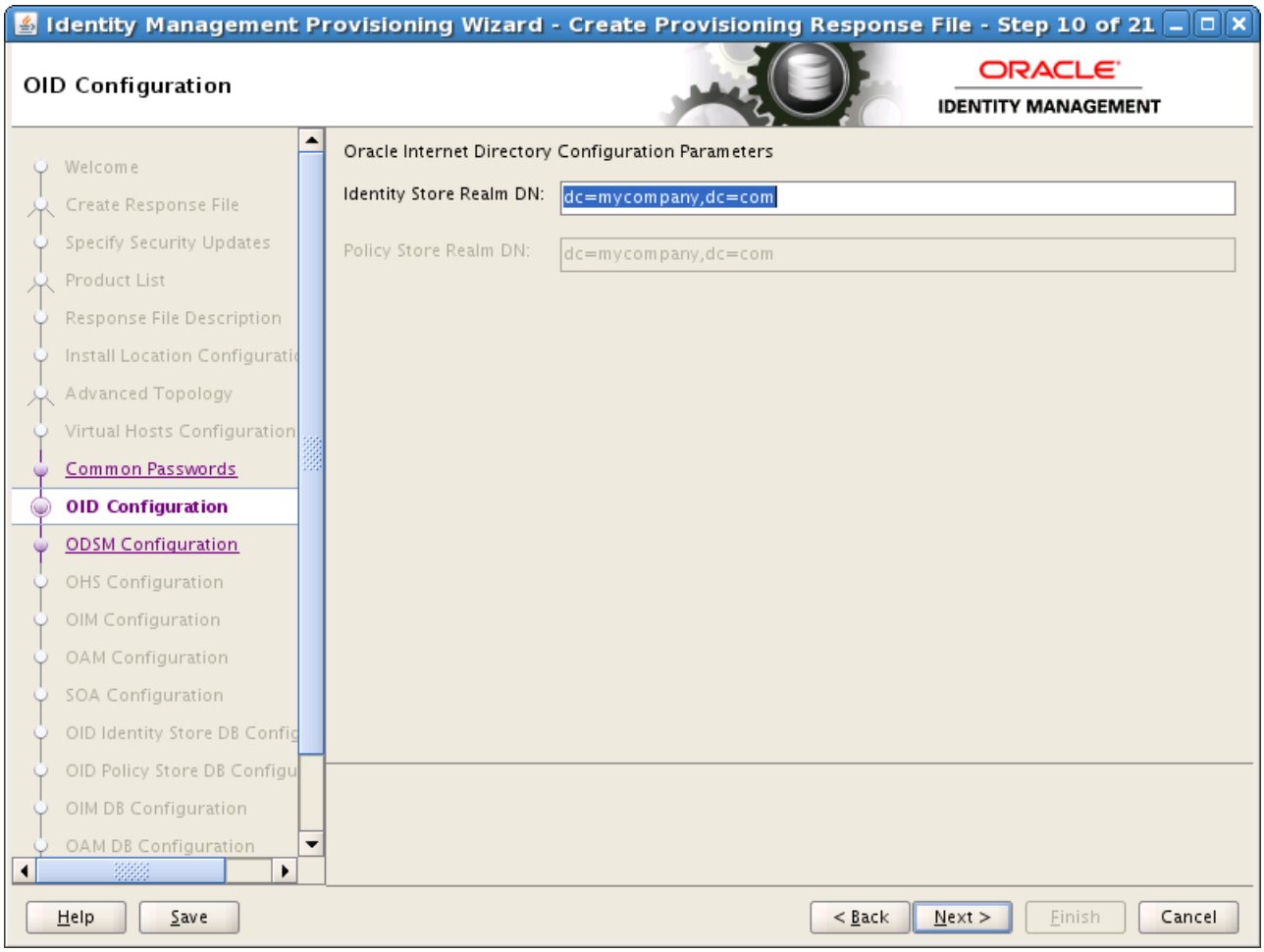
10. On the Common Passwords screen, enter a **Common IDM Password** (*COMMON_IDM_PASSWORD*) to be used for all accounts created as part of the provisioning. This password must be eight characters long and have at least one number and one uppercase letter.

    Confirm the password in **Confirm Common IDM Password**



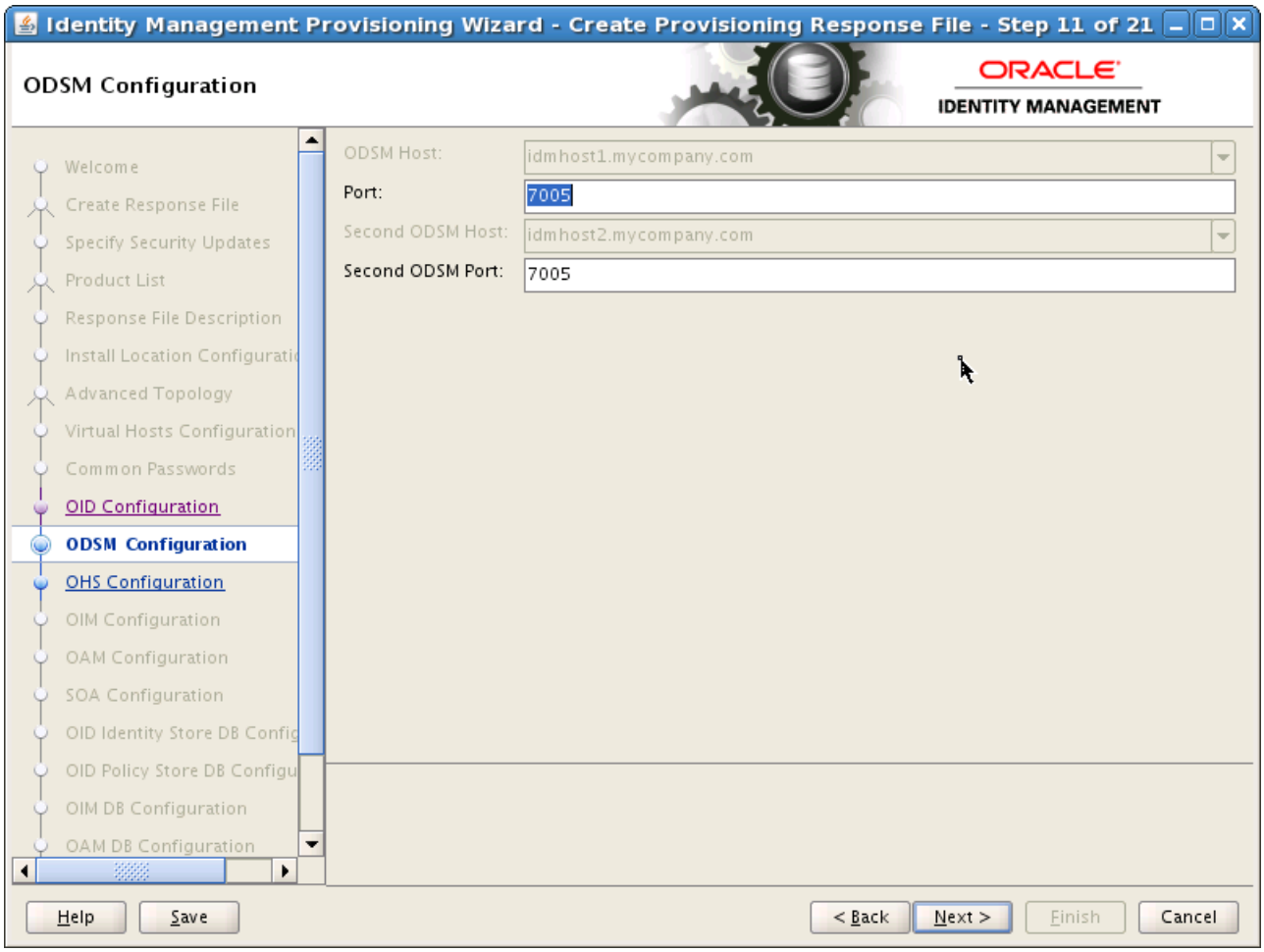
Click **Next**.

11. On the OID Configuration screen, enter the security **Realm DN**, for example: `dc=mycompany,dc=com` (*REALM_DN*).

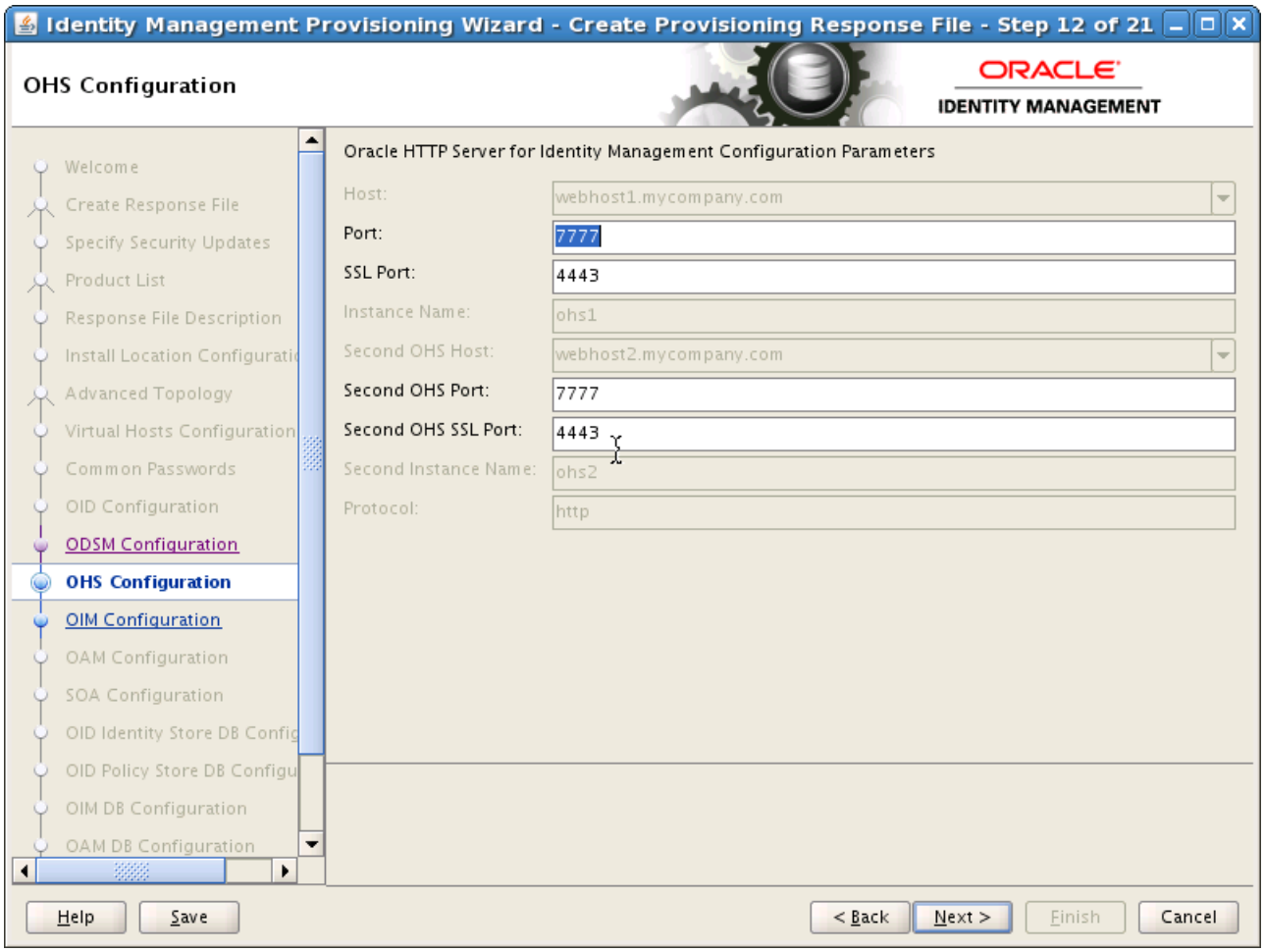


Click **Next**.

12. On the ODSM Configuration screen, change the ports that the ODSM managed servers will use (*ODSM_PORT*, *Second_ODSM_PORT*) if required, and click **Next**.



13. On the OHS Configuration screen, change the ports (*OHS_PORT* and *OHS_SSL_PORT*) that the Oracle HTTP Server managed servers will use, if required, and click **Next**.



14. On the OIM Configuration screen, under **Oracle Identity Manager Configuration Parameters**, enter the ports to be used by the Oracle Identity Manager managed servers into the **Port** and **Second OIM Port** fields (*OIM_PORT*)

    If you want to set up an email server then proceed as follows:

    a. Select **Configure Email Server**

    b. Select **Custom Email Server**

    c. Enter:

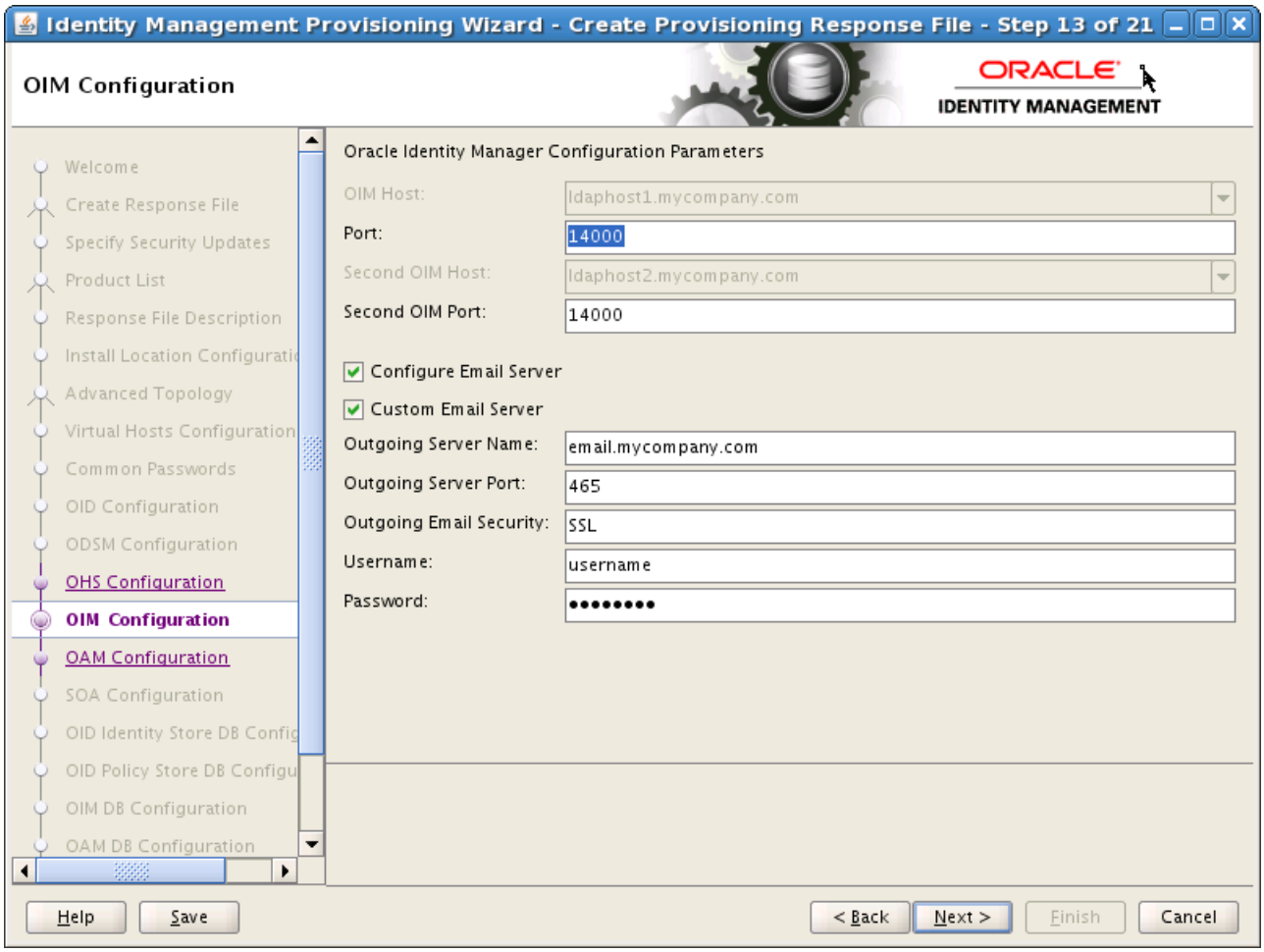
    Outgoing Server Name: The name of your outgoing email server, for example: `mail.mycompany.com` (*EMAIL_SERVER*)

    Outgoing Server Port: The port your email server uses (*EMAIL_PORT*).

    Outgoing Email Security: If this port is SSL enabled, enter `SSL`

    Username: If you require a username to authenticate with the email server, enter that username (*EMAIL_USER*) here.

    Password: Password (*EMAIL_PASSWORD*) for the above user.

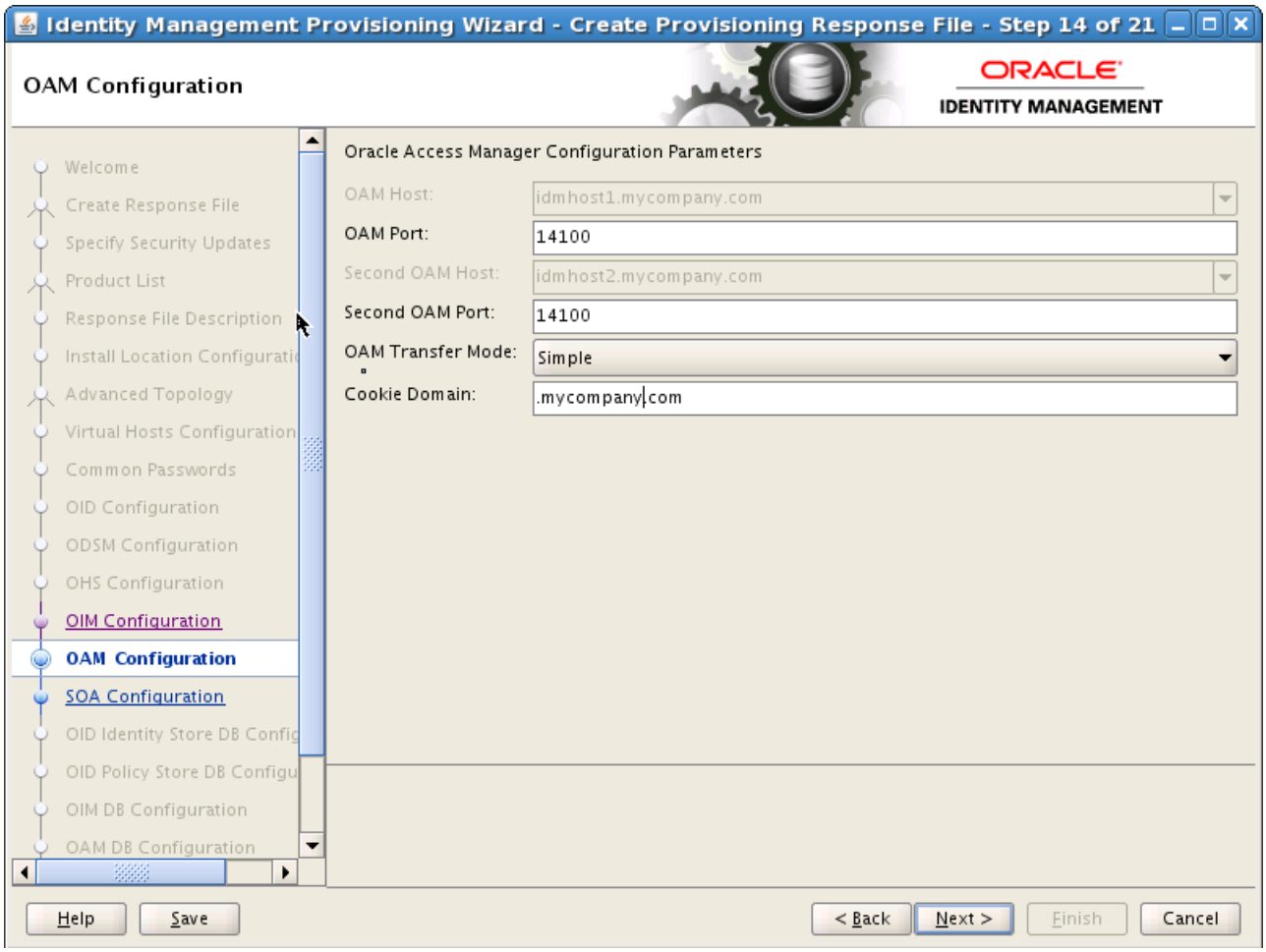


Click **Next**.

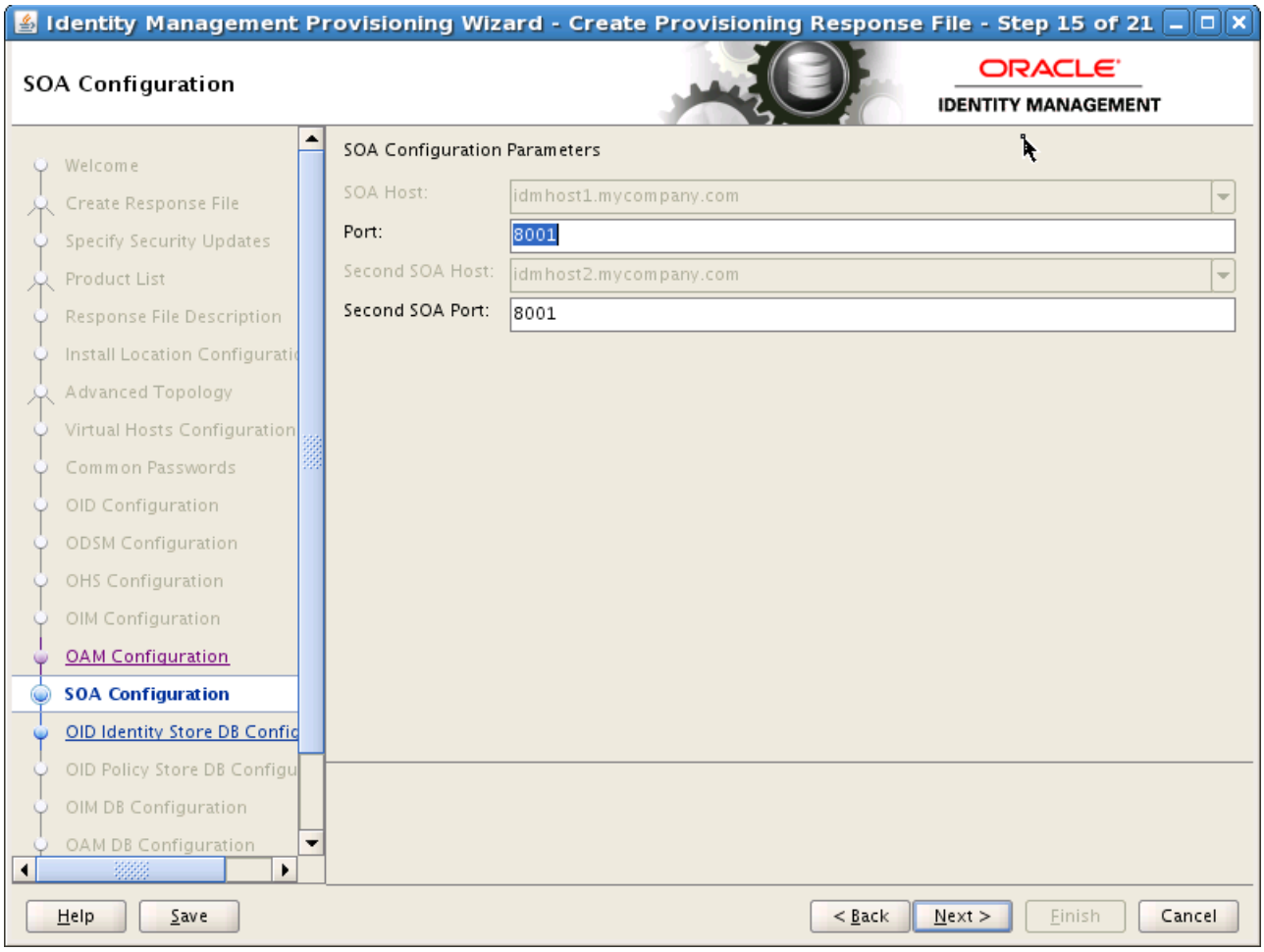
15. On the OAM Configuration screen, enter the following information:

    - Change the ports that the OAM managed servers will use (*OAM_PORT*), if required.
    - Specify the **OAM Transfer Mode**. This must be `Open` on AIX and `Simple` on other platforms.
    - Enter a value for **Cookie Domain**, for example: `.mycompany.com` (*OAM_COOKIE_DOMAIN*).

    Click **Next**.



16. On the SOA Configuration screen, enter the ports to be used by the SOA Managed servers and click **Next**.



17. On the **OID Identity Store DB Configuration** screen, you enter the database connection details for your Oracle Internet Directory Database. Enter the following values:

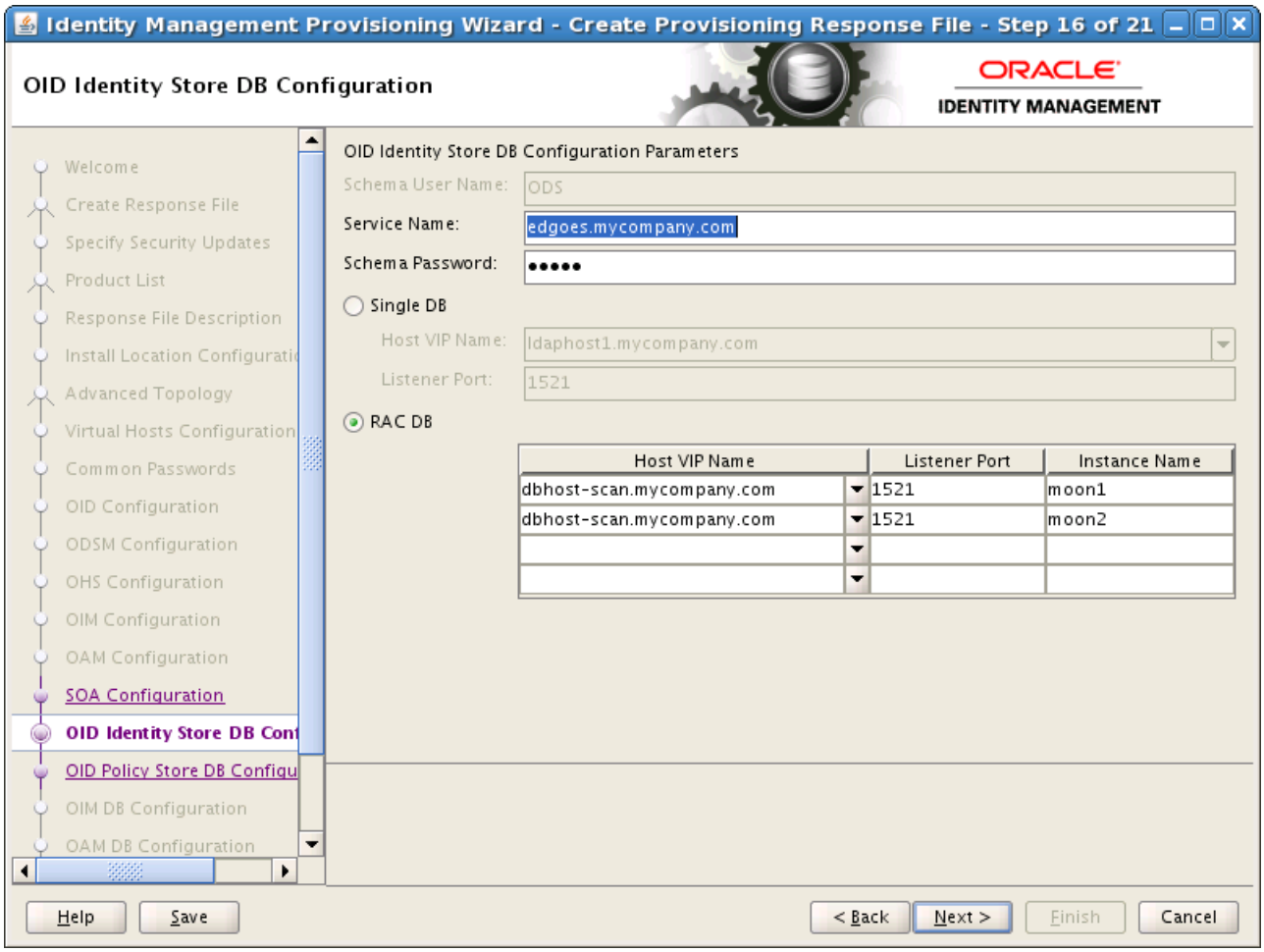
    Service Name: The service name of the database service, for example: oidedg.mycompany.com

    Schema Password: The password you used when creating the OID schema in RCU

    Select **RAC DB**.

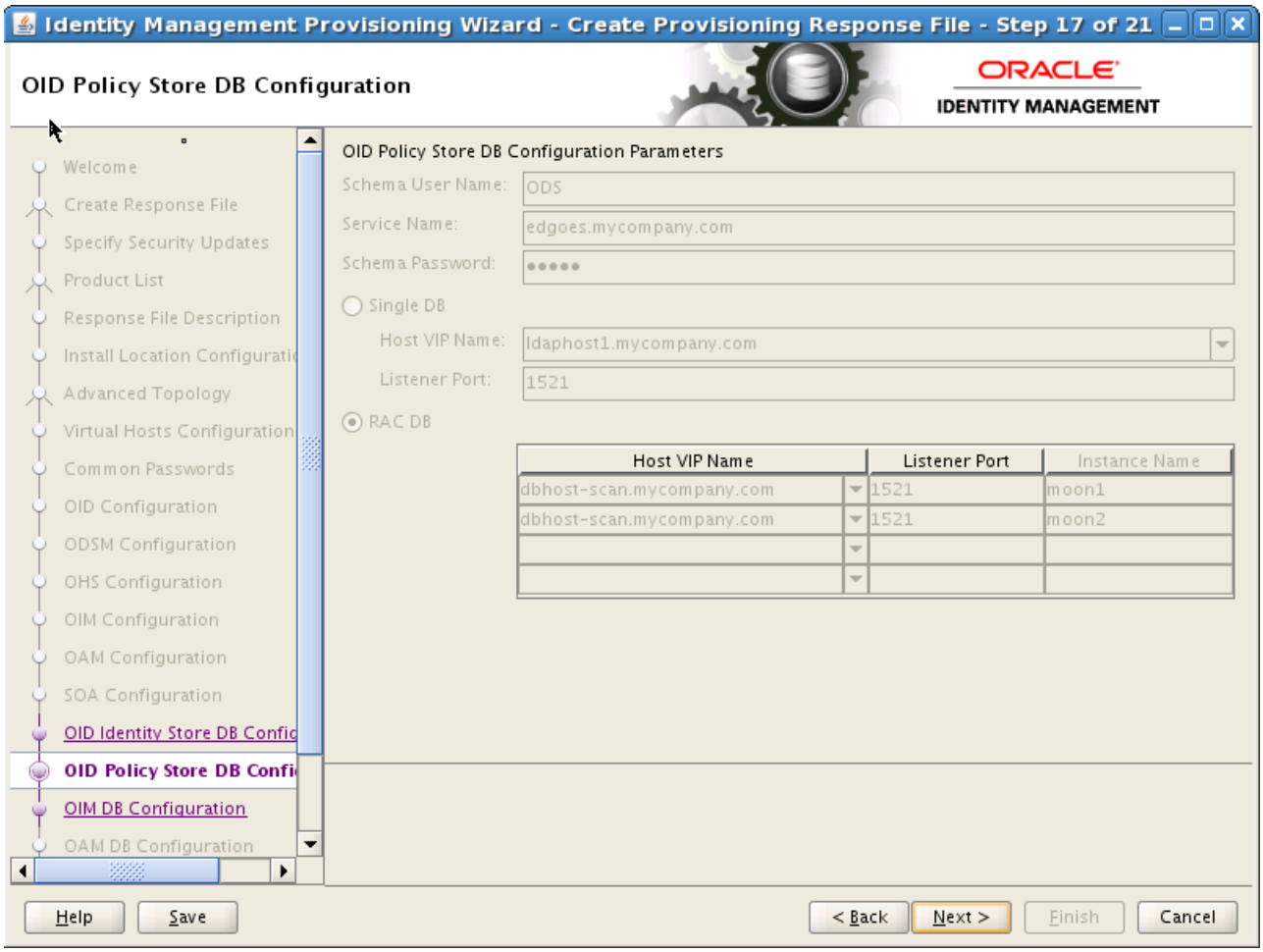
    Currently the wizard supports two RAC database instances. Provide the following information for each database instance.

    - **Host VIP Name**: This is the VIP name of the RAC database instance. If you are using Oracle Database 11.2, this must be the SCAN address.
    - **Listener Port**: This is the Listener Port *DB_LSNR_PORT*.
    - **Instance Name**: This is the instance name of the database instance. Use a different instance name for each entry.



Click **Next**

18. The OID Policy Store DB Configuration screen is greyed out.



Click **Next**

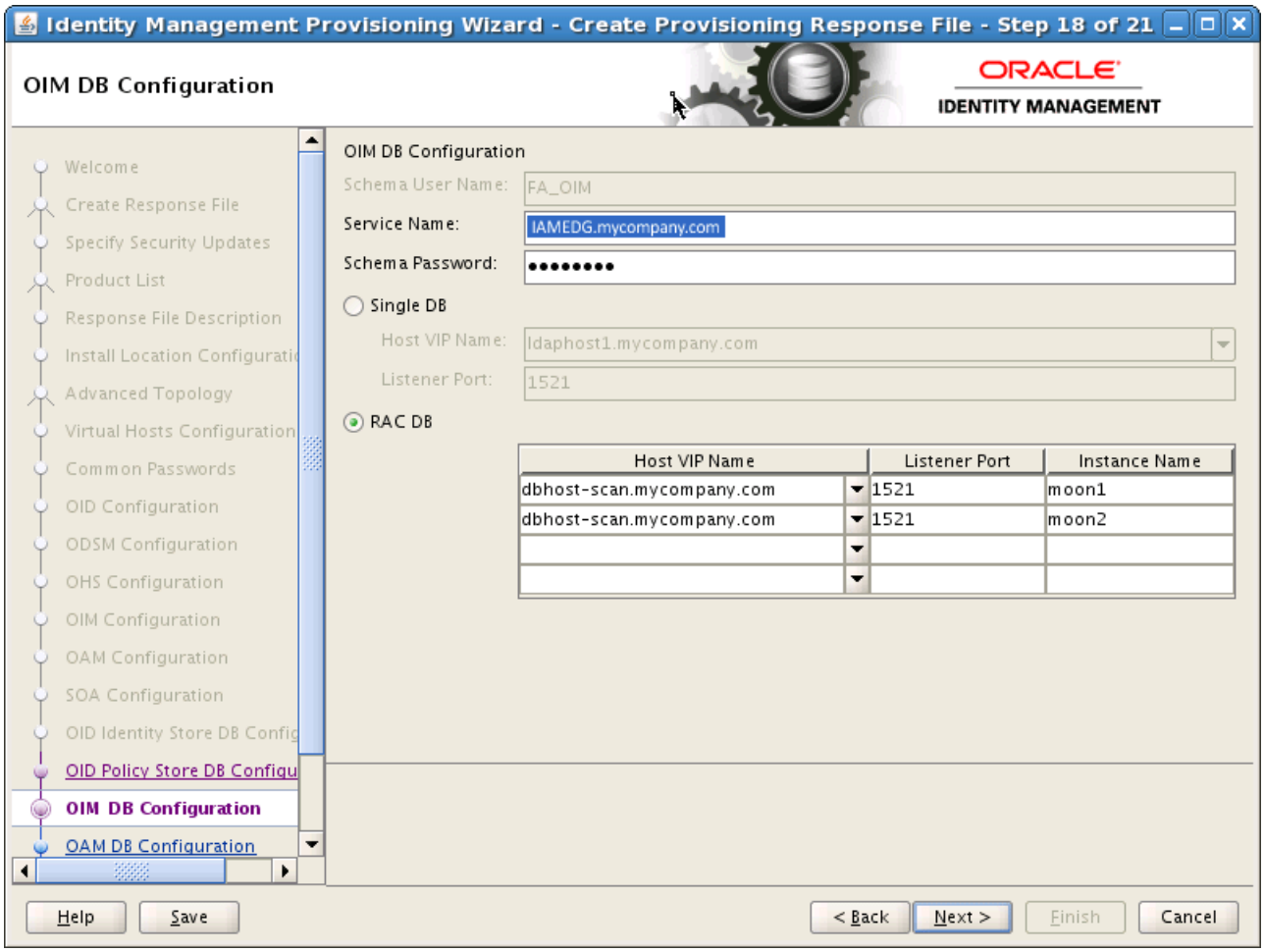
19. On the OIM DB Configuration screen, enter the details about the Oracle Database where Oracle Identity Manager information will be stored.

    OIM DB Service Name: The service name of the database service, for example: `IAMEDG.mycompany.com`

    OIM Schema Password: The password you used when creating the Oracle Identity Manager schema in RCU.

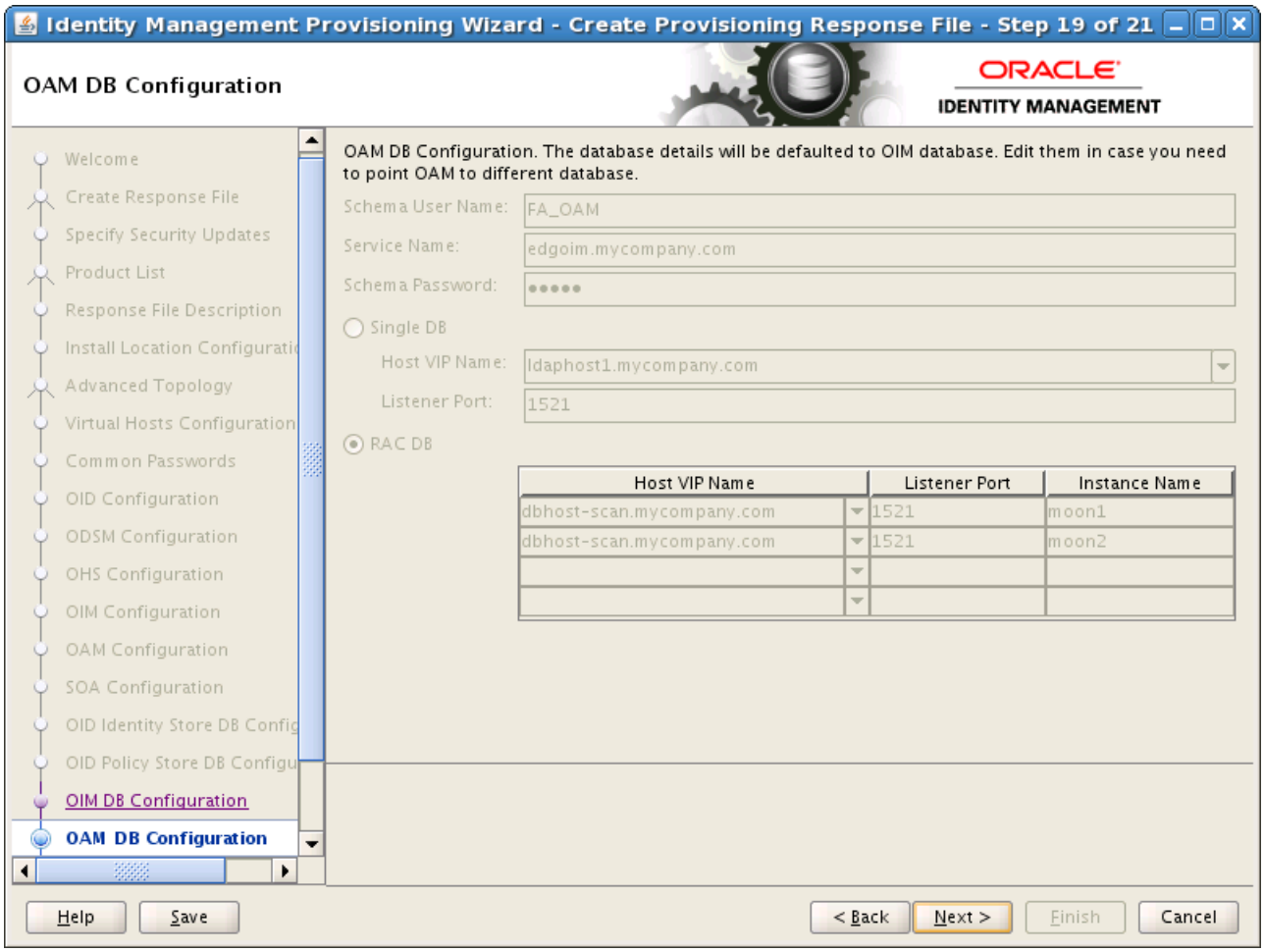
    Select **RAC DB**. Currently the wizard supports two RAC database instances. Provide the following information for each database instance.

    - **Host Name**: This is the VIP name of the RAC database instance. If you are using Oracle Database 11.2, this must be the SCAN address.
    - **Port**: This is the Listener Port
    - **Instance Name**: This is the instance name of the database instance. Use a different instance name for each entry.



Click **Next**

20. The OAM DB Configuration screen is greyed out. The information on the screen is the same as on the OIM DB Configuration screen, except for the Schema User Name.



Click **Next**.

21. On the Load Balancer screen, enter details about your load balancer virtual hosts.

    Under **HTTP/HTTPS Load Balancer Details**, enter the **Virtual Host Name** and **Port** for each **Endpoint**.

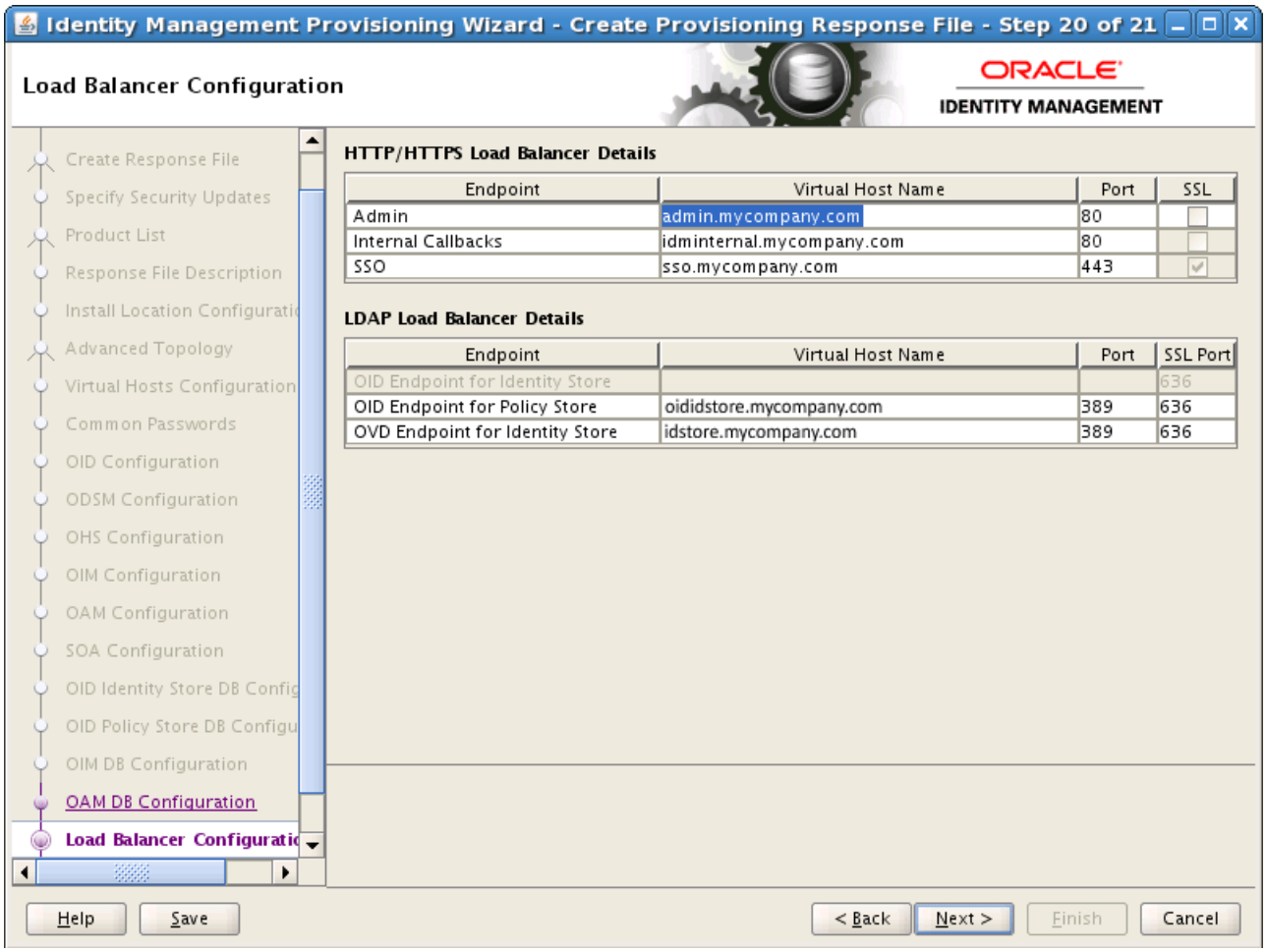
    - **Admin**: Admin Virtual Host and port, for example: `admin.mycompany.com` Port `80`, deselect **SSL**.
    - **Internal Callbacks**: This is the internal call back virtual host and port, for example: `idminternal.mycompany.com`, Port `80`
    - **SSO**: This is the main application entry point, for example: sso.mycompany.com Port 443

    Under **LDAP (OID) Load Balancer Details**, enter the **Virtual Host Name**, **Port**, and **SSL Port** for each **Endpoint**.

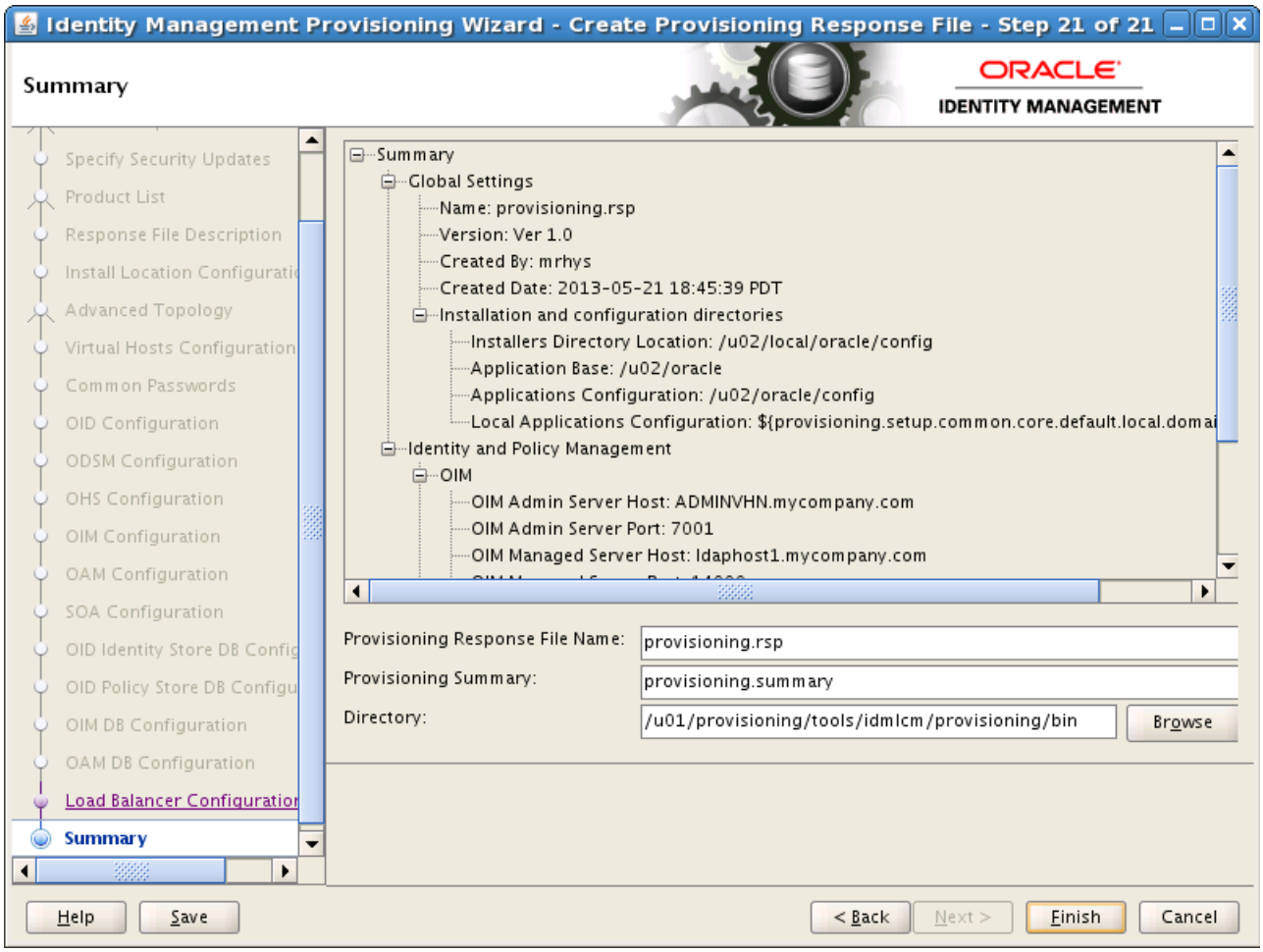
    - **ID Store**: This is the virtual host of the Identity store, for example: `idstore.mycompany.com`, Port: `389`
    - **Policy Store**: This is the virtual host of the Policy store, for example: `oididstore.mycompany.com`, Port `389`

Note: If your identity store and policy store are in the same internet directory, you can use the same virtual host name for both the Identity Store and the Policy Store End Points.

If you plan to use a different identity store, such as split profile or Active Directory, you must use different end points. The end point for your policy store must be the name of the load balancer that distributes requests across your Oracle Internet Directory instances. The Identity Store end point must be the name of the load balancer that distributes requests across your Oracle Virtual Directory instances.



22. On the Summary screen, enter the **Provisioning Response File Name** and the **Directory** where it is to be stored. Leave the **Provisioning Summary** field at the default value.



Click **Finish** to generate the provisioning response file.

8.2 Update User Names in Provisioning Response File

Provisioning creates a number of users, with default user names. You can change three of these user names to more sensible names by editing the `provisioning.rsp` file created above.

In keeping with user names used in previous releases of this guide, change these entries as follows.

> **Note:** You can change these values to anything that matches your requirements.

```
#IDStore UserNames Configuration
IDSTORE_OAMADMINUSER=oamadmin
IDSTORE_OAMSOFTWAREUSER=oamLDAP
IDSTORE_OIMADMINUSER=oimLDAP
```

Save the file after you make the changes.

8.3 Copy Provisioning File to DMZ Hosts

The process described in this chapter creates a provisioning file in the directory you specified on the Summary screen in Step 22. This file must be available to each host in the topology. If you have a shared provisioning directory, then this file is automatically available. If, however, you have not shared your provisioning directory, you must manually copy the file to the same location on the DMZ hosts, WEBHOST1 and WEBHOST2.

9 Provisioning Identity Management

This chapter describes how to provision Identity Management.

It contains the following sections:

9.1 Introduction to the Provisioning Process

This section introduces the provisioning process.

9.1.1 Provisioning Stages

There are eight stages to provisioning. These stages are:

1. preverify - This checks that each of the servers being used in the topology satisfies the minimum requirements of the software being installed and configured.
2. install - This installs all of the software required by the installation.
3. preconfigure - This does the following:
   - Creates OID and seeds it with Users/Groups.
   - Creates OVD
   - Configures ODSM
   - Creates the WebLogic Domain
   - Creates OHS instance
4. configure - This does the following:
   - Associates the Policy Store to OID
   - Starts managed servers as necessary
   - Associates OAM with OID
   - Configure OIM
5. configure-secondary - This does the following:
   - Integrates Weblogic Domain with Webtier
   - Register webtier with domain
   - Integrate OAM and OIM

6. postconfigure - This does the following:
   - Register OID with Weblogic Domain
   - SSL Enable OID and OVD
   - Tune OID
   - Run OIM Reconciliation
   - Configure UMS Mail Server
   - Generate OAM Keystore
   - Configure OIF
   - Configure Webgates
7. startup - This starts up all components in the topology
8. validate - This performs a number of checks on the built topology to ensure that everything is working as it should be.

Each stage must be completed on each host in the topology before the next stage can begin. Failure of a stage will necessitate a cleanup and restart.

9.1.2 Processing Order

You must process hosts in the following order:

1. LDAP Host 1
2. LDAP Host 2
3. Identity and Access Management Host 1
4. Identity and Access Management Host 2
5. Web Host 1
6. Web Host 2

This equates to the following order for hosts in this guide:

1. LDAPHOST1
2. LDAPHOST2
3. IDMHOST1
4. IDMHOST2
5. WEBHOST1
6. WEBHOST2

9.2 Provisioning Procedure

The following sections describe the procedure for performing provisioning.

- Section 9.2.1, "Running the Provisioning Commands"
- Section 9.2.2, "Creating Backups"
- Section 9.2.3, "Apply Patch 16708003"
- Section 9.2.4, "Copy Provisioning Files to WEBHOST1 and WEBHOST2"

- Section 9.2.5, "Copying WebGate Configuration Files to WEBHOST1 and WEBHOST2"

9.2.1 Running the Provisioning Commands

Provisioning is accomplished by running the command `runIDMProvisioning.sh` a number of times on each host in the topology.

BEFORE embarking on the provisioning process, read this entire section. There are extra steps detailed below which must be performed during the process.

You MUST run each command on each host in the topology before running the next command.

Before running the provisioning tool, set the following environment variables:

- Set `ANT_HOME` to: *`REPOS_HOME`*`/provisioning/ant`
- Set `JAVA_HOME` to: *`REPOS_HOME`*`/jdk6`

The commands you must run are:

```
runIDMProvisioning.sh -responseFile IDMLCM_HOME/provisioning/bin/provisioning.rsp
-target preverify

runIDMProvisioning.sh -responseFile IDMLCM_HOME/provisioning/bin/provisioning.rsp
-target install

runIDMProvisioning.sh -responseFile IDMLCM_HOME/provisioning/bin/provisioning.rsp
-target preconfigure

runIDMProvisioning.sh -responseFile IDMLCM_HOME/provisioning/bin/provisioning.rsp
-target configure

runIDMProvisioning.sh -responseFile IDMLCM_HOME/provisioning/bin/provisioning.rsp
-target configure-secondary

runIDMProvisioning.sh -responseFile IDMLCM_HOME/provisioning/bin/provisioning.rsp
-target postconfigure

runIDMProvisioning.sh -responseFile IDMLCM_HOME/provisioning/bin/provisioning.rsp
-target startup

runIDMProvisioning.sh -responseFile IDMLCM_HOME/provisioning/bin/provisioning.rsp
-target validate
```

9.2.2 Creating Backups

It is important that you take a backup of the file systems and databases at the following points:

1. Prior to starting provisioning.
2. At the end of the installation phase.
3. Upon completion of provisioning

It is not supported to restore a backup at any phase other than those three.

9.2.3 Apply Patch 16708003

After performing the install phase on the primoridial host (IDMHOST1), you must download Patch 16708003and apply it on IDMHOST1.

9.2.4 Copy Provisioning Files to WEBHOST1 and WEBHOST2

If you are not sharing your provisioning directory onto the WEBHOSTs, you must manually copy the following directories from IDMHOST1 to the local provisioning directories on those hosts. You must do this BEFORE running the install on those hosts.

SHARED_CONFIG_DIR/lcmconfig/topology

SHARED_CONFIG_DIR/lcmconfig/credconfig

For example:

```
scp -r SHARED_CONFIG_DIR/lcmconfig/topology WEBHOST1:SHARED_CONFIG_DIR/lcmconfig/
scp -r SHARED_CONFIG_DIR/lcmconfig/credconfig WEBHOST1:SHARED_CONFIG_
DIR/lcmconfig/
```

9.2.5 Copying WebGate Configuration Files to WEBHOST1 and WEBHOST2

When configuring WebGate during the postconfigure stage, the provisioning tool requires access to files created on the primordial host. So BEFORE postconfigure is run on WEBHOST1 and WEBHOST2, you must copy the entire directory *ASERVER_HOME*/output to the same location on WEBHOST1 and WEBHOST2.

For example:

```
scp -r IDMHOST1:$ASERVER_HOME/output  WEBHOST1:$ASERVER_HOME
```

Note: Before making the copy, you might need to manually create the directory *ASERVER_HOME* on WEBHOST1 and WEBHOST2. After provisioning is complete, you can remove this directory from WEBHOST1 and WEBHOST2.

9.3 Check List

To help keep track of the provisioning process, print this check list from the PDF version of this guide. Run each stage on the hosts shown, and add a check mark to the corresponding row when that run is complete.

Provisioning Stage	Host	Complete
Preverify	LDAPHOST1	
	LDAPHOST2	
	IDMHOST1	
	IDMHOST2	
	WEBHOST1	
	WEBHOST2	
Install	LDAPHOST1	

Provisioning Stage	Host	Complete
	LDAPHOST2	
	IDMHOST1	
Copy Provisioning Files	IDMHOST1	
	WEBHOST1	
	WEBHOST2	
Install	IDMHOST2	
	WEBHOST1	
	WEBHOST2	
Preconfigure	LDAPHOST1	
	LDAPHOST2	
	IDMHOST1	
	IDMHOST2	
	WEBHOST1	
	WEBHOST2	
Configure	LDAPHOST1	
	LDAPHOST2	
	IDMHOST1	
	IDMHOST2	
	WEBHOST1	
	WEBHOST2	
Configure Secondary	LDAPHOST1	
	LDAPHOST2	
	IDMHOST1	
	IDMHOST2	
	WEBHOST1	
	WEBHOST2	
Post Configure	LDAPHOST1	
	LDAPHOST2	
	IDMHOST1	
	IDMHOST2	
Copy WebGate Files	WEBHOST1	
	WEBHOST2	
Post Configure	WEBHOST1	
	WEBHOST2	
Startup	LDAPHOST1	
	LDAPHOST2	
	IDMHOST1	

Provisioning Stage	Host	Complete
	IDMHOST2	
	WEBHOST1	
	WEBHOST2	
Validate	LDAPHOST1	
	LDAPHOST2	
	IDMHOST1	
	IDMHOST2	
	WEBHOST1	
	WEBHOST2	

10

Performing Post-Provisioning Configuration

This chapter describes tasks you must perform after provisioning.

It contains the following sections:

10.1 Correcting Datasource Configuration

Due to Bugs 17075699 and 17076033 in Identity Management Provisioning, you must make changes to the following datasources:

- **EDNLocalTxDataSource-rc*n***
- **mds-oim-rc*n***
- **mds-owsm-rc*n***
- **mds-soa-rc*n***
- **oamDS-rc*n***
- **oimJMSStoreDS-rc*n***
- **OraSDPMDataSource-rc*n***
- **SOALocalTxDataSource-rac*n***

To make the changes, proceed as follows:

1. Log in to the WebLogic Administration Console at the URL listed in Section 16.2, "About Identity Management Console URLs."
2. Click **Lock & Edit**.
3. Navigate to **Services -> Data Sources**.
4. Click on the data source to be updated, for example, **mds-soa-rc0**
5. Click the **Transaction** tab.
6. Deselect **Supports Global Transactions**.
7. Click **Save**.

8. Repeat Steps 4 through 7 for all the listed datasources.
9. Click **Activate Changes**.
10. Restart all servers.

10.2 Updating 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, on both WEBHOST1 and WEBHOST2.

Proceed as follows:

1. Edit the file `httpd.conf`, which is located in:

   WEB_ORACLE_INSTANCE`/config/OHS/`*component_name*

2. Find the entry that looks like this:

   ```
   <IfModule mpm_worker_module>
   ```

3. Update the values in this section as follows:

   ```
   <IfModule mpm_worker_module>
     ServerLimit 20
     MaxClients 1000
     MinSpareThreads 200
     MaxSpareThreads 800
     ThreadsPerChild 50
     MaxRequestsPerChild 10000
     AcceptMutex fcntl
   </IfModule>
   ```

4. Leave all remaining values unchanged.
5. Save the file.

10.3 Creating ODSM Connections to Oracle Virtual Directory

Before you can manage Oracle Virtual Directory you must create connections from ODSM to each of your Oracle Virtual Directory instances. To do this, proceed as follows:

1. Access ODSM through the load balancer at: `http://ADMIN.mycompany.com/odsm`
2. Follow these steps to create connections to Oracle Virtual Directory:

   To create connections to Oracle Virtual Directory, follow these steps. Create connections to each Oracle Virtual Directory node separately. Using the Oracle Virtual Directory load balancer virtual host from ODSM is not supported:

   a. Create a direct connection to Oracle Virtual Directory on LDAPHOST1 providing the following information in ODSM:

      `Host`: `LDAPHOST1.mycompany.com`

      `Port`: `8899` (The Oracle Virtual Directory proxy port, *OVD_ADMIN_PORT* in Section 3.7, "Fixed Ports Used by the Provisioning Wizard.")

      Enable the SSL option.

      `User`: `cn=orcladmin`

      `Password`: password_to_connect_to_OVD

b. Create a direct connection to Oracle Virtual Directory on LDAPHOST2 providing the following information in ODSM:

`Host`: `LDAPHOST2.mycompany.com`

`Port`: `8899` (The Oracle Virtual Directory proxy port)

Enable the SSL option.

`User`: `cn=orcladmin`

`Password`: password_to_connect_to_OVD

10.4 Post-Provisioning Steps for Oracle Identity Manager

Perform the following task to ensure that Oracle Identity Manager works correctly after provisioning.

- Section 10.4.1, "Add an Oracle Identity Manager Property"

10.4.1 Add an Oracle Identity Manager Property

As a workaround for a bug in the Identity Management Provisioning tools (Bug 16667037), you must add an Oracle Identity Manager property. Perform the following steps:

1. Log in to the WebLogic Console. (The Console URLs are provided in Section 16.2, "About Identity Management Console URLs.")
2. Navigate to **Environment -> Servers**.
3. Click **Lock and Edit**.
4. Click on the server **WLS_OIM1**.
5. Click on the **Server Start** subtab
6. Add the following to the **Arguments** field:

```
-Djava.net.preferIPv4Stack=true
```

7. Click **Save**.
8. Repeat Steps 4-7 for the managed server **WLS_OIM2**.
9. Click **Activate Changes**.
10. Restart the managed servers WLS_OIM1 and WLS_OIM2, as described in Section 16.1, "Starting and Stopping Components."

10.5 Post-Provisioning Steps for Oracle Access Manager

Perform the tasks in the following sections.

- Section 10.5.1, "Updating Existing WebGate Agents"
- Section 10.5.2, "Update WebGate Configuration"
- Section 10.5.3, "Creating Oracle Access Manager Policies for WebGate 11g"

The Identity Management Console URLs are provided in Section 16.2, "About Identity Management Console URLs."

10.5.1 Updating Existing WebGate Agents

Update the OAM Security Model of all WebGate profiles, with the exception of Webgate_IDM and Webgate_IDM_11g, which should already be set

To do this, perform the following steps:

1. Log in to the Oracle Access Manager Console as the Oracle Access Manager administration user identified by the entry in Section 8.2, "Update User Names in Provisioning Response File."
2. Click the **System Configuration** tab.
3. Expand **Access Manager Settings - SSO Agents**.
4. Click **OAM Agents** and select **Open** from the **Actions** menu.
5. In the Search window, click **Search**.
6. Click an Agent, for example: **IAMSuiteAgent**.
7. Set the Security value to the security model in the **OAM Configuration** screen of the Identity Management Provisioning Wizard, as described in Section 8.1, "Running the Identity Management Provisioning Wizard to Create a Profile.".

   Click **Apply**.
8. Restart the managed servers WLS_OAM1 and WLS_OAM2 as described in Section 16.1, "Starting and Stopping Components."

10.5.2 Update WebGate Configuration

To update the maximum number of WebGate connections, proceed as follows.

1. In the Oracle Access Manager Console, select the **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 `20`. (This is the total maximum number of connections for the primary servers, which is 10 wls_oam1 connections plus 10 wls_oam2 connections.)
7. Set **AAA Timeout Threshold** to `5`.
8. In the **User Defined Parameters** box, set `client_request_retry_attempts` to `11`.
9. If the following Logout URLs are not listed, add them:
   - `/oamsso/logout.html`
   - `/console/jsp/common/logout.jsp`
   - `/em/targetauth/emaslogout.jsp`
10. Click **Apply**.

Repeat Steps 4 through 7 for each WebGate

10.5.3 Creating Oracle Access Manager Policies for WebGate 11*g*

In order to allow WebGate 11*g* to display the credential collector, you must add `/oam` to the list of public policies.

Proceed as follows:

1. Log in to the OAM console at: `http://ADMIN.mycompany.com/oamconsole`
2. Select the **Policy Configuration** tab.
3. Expand **Application Domains - IAM Suite**
4. Click **Resources**.
5. Click **Open**.
6. Click **New resource**.
7. Provide the following values:
   - **Type**: `HTTP`
   - **Description**: `OAM Credential Collector`
   - **Host Identifier**: `IAMSuiteAgent`
   - **Resource URL**: `/oam`
   - **Protection Level**: `Unprotected`
   - **Authentication Policy**: `Public Policy`
8. Leave all other fields at their default values.
9. Click Apply.

10.6 Passing Configuration Properties File to Oracle Fusion Applications

Oracle Fusion Applications requires a property file which details the IDM deployment. After provisioning, this file can be found at the following location:

SHARED_CONFIG_DIR`/fa/idmsetup.properties`

11

Enabling Oracle Identity Federation

The Identity Management provisioning tools create, but do not start, Oracle Identity Federation. This chapter explains how to enable Oracle Identity Federation after provisioning has completed.

Oracle Identity Federation is an optional component. If you are not planning to use Oracle Identity Federation, skip this chapter. This chapter describes how to extend the Identity Management domain to include Oracle Identity Federation in an enterprise deployment.

This chapter contains the following topics:

11.1 Starting OIF Managed Servers

Start the managed servers wls_oif1 and wls_oif2 as follows:

1. Run `stopall.sh` as described in Section 16.1, "Starting and Stopping Components."
2. Update the Oracle Identity Federation Property File `oif_startup.conf` to automatically start Oracle Identity Federation. To do this, edit the file `oif_startup.conf` which is located in the directory: *SHARED_CONFIG_DIR*`/scripts`

   Edit the file so that it looks like this:

```
#
# OIF is enabled OOTB for Shared IDM
#
# OIF_ENABLED indicates whether or not OIF should be started/stopped
# as part of the startoif.sh/stopoif.sh scripts. Valid values are true or false
# If false, the OIF will not be started or stopped
```

```
OIF_ENABLED=true
# OPMN_EMAGENT_MANAGED_BY_OIF_SCRIPT indicates whether or not OPMN and
# the EMAgent components for the OIM domain should be started, when OIF is
enabled.
# Valid values are true or false. If false, OPMN and the EMAgent components
will not
# be started or stopped when OIF is enabled.
# If OIF is disabled, OPMN and the EMAgent components will not be started or
stopped
OPMN_EMAGENT_MANAGED_BY_OIF_SCRIPT=true
```

Save the file.

3. Run `startall.sh` as described in Section 16.1, "Starting and Stopping Components."

11.2 Updating OIF Web Configuration

Edit the file `idminternal_vh.conf` which is located in *`WEB_ORACLE_INSTANCE`*`/config/OHS/`*`component`*`/modultconf`

Add the following lines inside the VirtualHost block:

```
################################################
## Entries Required by Oracle Identity Federation
################################################

   #OIF
   <Location /fed>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WebLogicCluster IDMHOST1.mycompany.com:7499,IDMHOST2.mycompany.com:7499
   </Location>
```

Save the file and restart the Oracle HTTP Server as described in Section 16.1, "Starting and Stopping Components."

Repeat this for each Oracle HTTP Server instance.

11.3 Validating Oracle Identity Federation

Validate the configuration of Oracle Identity Federation on IDMHOST1 and IDMHOST2 by accessing the SP metatadata on each host.

On IDMHOST1, access the SP metadata by going to:

`http://IDMHOST1.mycompany.com:7499/fed/sp/metadata`

On IDMHOST2, access the SP metadata by going to:

`http://IDMHOST2.mycompany.com:7499/fed/sp/metadata`

If the configuration is correct, you can access the following URL from a web browser:

`https://SSO.mycompany.com/fed/sp/metadata`

You should see metadata.

11.4 Configuring the Enterprise Manager Agents

All the Oracle Fusion Middleware components deployed in this enterprise deployment are managed by using Oracle Enterprise Manager Fusion Middleware Control. To manage Oracle Identity Federation with this tool, you must configure the EM agents with the correct monitoring credentials. Update the credentials for the EM agents associated with IDMHOST1 and IDMHOST2. Follow these steps to complete this task:

1. Use a web browser to access Oracle Enterprise Manager Fusion Middleware Control at `http://ADMINVHN.mycompany.com:7001/em`. Log in as the WebLogic user.
2. From the Domain Home Page, navigate to the Agent-Monitored Targets page using the menu under **Farm -> Agent-Monitored Targets**.
   - Click the **Configure** link for the Target Type Identity Federation Server to go to the Configure Target Page.
   - On the Configure Target Page, click **Change Agent** and choose the correct agent for the host.

     Note: If you are unsure about which agent to update, execute the command:

     ```
     OIF_ORACLE_INSTANCE/EMAGENT/EMAGENT/bin/emctl status agent
     ```

   - Update the **WebLogic monitoring user name** and the **WebLogic monitoring password**. Enter `weblogic_idm` as the WebLogic monitoring user name and the password for the weblogic user as the WebLogic monitoring password.
   - Click **OK** to save your changes.

11.5 Enabling Oracle Identity Federation Integration with LDAP Servers

By default, Oracle Identity Federation is not configured to be integrated with LDAP Servers deployed in a high availability configuration. To integrate Oracle Identity Federation with highly available LDAP Servers to serve as user data store, federation data store, or authentication engine, you must configure Oracle Identity Federation based on the LDAP server's function.

Proceed as follows to integrate Oracle Identity Federation with an LDAP Server deployed in a high availability configuration

1. On IDMHOST1, set environment variables as follows:

   Set `DOMAIN_HOME` to *`MSERVER_HOME`*.

   Set `IDM_ORACLE_HOME` to *`IDM_ORACLE_HOME`*.
2. Set Oracle Identity Federation-specific environment variables by executing the `setOIFEnv.sh` script. This script is located under the *`IDM_ORACLE_HOME`*`/fed/scripts` directory.

   For example:

   ```
   cd IDM_ORACLE_HOME/fed/scripts
   . setOIFEnv.sh
   ```

3. On IDMHOST1, run the `WLST` script located under the *`ORACLE_COMMON_HOME`*`/bin` directory.

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

4. Connect to one of the Oracle Identity Federation Managed Servers:

   ```
   connect()
   ```

   Enter the username and password to connect to the Oracle Identity Federation Managed Servers. This is the same as the WebLogic Administration user name and password.

   Enter the URL to connect to the Oracle Identity Federation Managed Server:

   ```
   t3://IDMHOST1.mycompany.com:7499
   ```

5. Then enter the following properties, as needed:

   - To integrate the user data store with a highly available LDAP Server, set the `userldaphaenabled` boolean property from the `datastore` group to `true`:

     ```
     setConfigProperty('datastore','userldaphaenabled', 'true', 'boolean')
     Update was successful for: userldaphaenabled
     ```

   - Validate the user data store is integrated with a highly available LDAP store by running:

     ```
     getConfigProperty('datastore', 'userldaphaenabled')
     Value(s) for property: true
     ```

     The userldaphaenabled property must return `true`.

   - To integrate the LDAP authentication engine with a highly available LDAP Server, set the `ldaphaenabled` boolean property from the `authnengines` group to `true`:

     ```
     setConfigProperty('authnengines','ldaphaenabled', 'true', 'boolean')
     Update was successful for: ldaphaenabled
     ```

   - Validate the LDAP authentication engine is integrated with a highly available LDAP store by running:

     ```
     getConfigProperty('authnengines','ldaphaenabled')
     Value(s) for property: true
     ```

     The `ldaphaenabled` property for the `authnengines` group must return `true`.

   Note: On IDMHOST1, delete the following directories:

   - *ASERVER_HOME*/config/fmwconfig/servers/wls_oif1/applications
   - *ASERVER_HOME*/config/fmwconfig/servers/wls_oif2/applications

   ---

11.6 Updating the Oracle Identity Federation Authentication Scheme in Oracle Access Manager

Oracle Access Manager ships with an Oracle Identity Federation Authentication Scheme. This scheme needs to be updated before it can be used. To update the scheme, log in to the OAM console as the OAM administration user identified by the entry in

Section 8.2, "Update User Names in Provisioning Response File." The URL is:
`http://ADMIN.mycompany.com/oamconsole`

Then perform the following steps:

1. Click the **Policy Configuration** tab.
2. Expand **Authentication Schemes** under the Shared Components tree.
3. Select **OIFScheme** from under the Authentication Schemes and then select **Open** from the menu.
4. On the Authentication Schemes page, provide the following information
   - **Challenge URL**:
     `https://SSO.mycompany.com:443/fed/user/spoam11g`
   - **Context Type**: Select **external** from the list.

   Accept the defaults for all other values
5. Click **Apply** to update the `OIFScheme`.

11.7 Setting Oracle Identity Federation Authentication Mode and Enabling Password Policy Profile

Proceed as follows:

1. On IDMHOST1, run the `WLST` script located under the *ORACLE_COMMON_HOME*/common/bin directory.

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

2. Connect to the WebLogic Adminiswtration Server:

   ```
   connect()
   ```

   Enter the username and password to connect to the Oracle Identity Federation Managed Servers. This is the same as the WebLogic Administration user name and password.

   Enter the URL to connect to the Oracle Identity Federation Managed Server:

   ```
   t3://IDMHOST1.mycompany.com:7001
   ```

3. Execute the following WLST command:

   ```
   domainRuntime()
   ```

4. To configure the Oracle Access Manager for Oracle Identity Federation SSO flows, execute the following WLST commands:

   ```
   configOAMOIFSaaS(fedMode="Dedicated")

   enablePasswordPolicyProfile()
   ```

11.8 Enabling and Disabling Oracle Identity Federation

In Service Provider (SP) mode, Oracle Access Manager delegates user authentication to Oracle Identity Federation, which uses the Federation Oracle Single Sign-On protocol with a remote Identity Provider. Once the Federation Oracle Single Sign-On flow is performed, Oracle Identity Federation will create a local session and then

propagates the authentication state to Oracle Access Manager, which maintains the session information.

This section provides the steps to integrate Oracle Identity Federation with Oracle Identity Manager in authentication mode and SP mode.

> **Note:** Federation Trust must be established prior to enabling Oracle Identity Federation.

This section contains the following topics:

- Section 11.8.1, "Enabling Oracle Identity Federation"
- Section 11.8.2, "Disabling Oracle Identity Federation"

11.8.1 Enabling Oracle Identity Federation

This section describes how to switch the authentication of the Oracle Access Manager security domain from local authentication to Federation SSO.

Perform the following operations to switch from local authentication to Federation SSO for Browser Based Schemes:

1. In a browser, go to the OAM Console, at:

   `http://ADMINVHN.mycompany.com:7001/oamconsole`

   Log in as the Oracle Access Manager user identified by the entry in Section 8.2, "Update User Names in Provisioning Response File."
2. Navigate to **Policy Configuration -> Shared Components -> Authentication Schemes -> FAAuthScheme**.
3. Set the **Challenge Method** to **FORM**.
4. Set the **Authentication Module** to `SaaSModule`.
5. Set the **Challenge URL** to `/pages/oamLogin.jsp`.
6. Set the **Context Type** to **customWar**.
7. Set the **Context Value** to `/fusion_apps`.
8. Set the Challenge Parameters field with the following entries:
   - `federationEnabled=true`
   - `ssoChooserEnabled=false`[1]
   - `fedSSOEnabled=true`
   - `initial_command=NONE`
   - `TAPPartnerId=OIFDAPPartner`
   - `TAPChallengeURL=https://SSO.mycompany.com:443/fed/user/spoam11g`
9. Click **Apply**.

[1] If dual authentication mode is required, set `ssoChooserEnabled=true` instead of `ssoChooserEnabled=false`. Dual authentication mode is required when some users in the Oracle Fusion Applications LDAP directory do not exist in the Identity Provider's directory. Those users cannot be authenticated with Federation SSO and must be challenged locally.

11.8.2 Disabling Oracle Identity Federation

This section describes how to switch the authentication of the OAM security domain from Federation SSO to local authentication.

Perform the following operations to switch from local authentication to Federation SSO for Browser Based Schemes:

1. In a browser, go to the OAM Console, at:

   `http://ADMINVHN.mycompany.com:7001/oamconsole`

   Log in as the Oracle Access Manager user identified by the entry in Section 8.2, "Update User Names in Provisioning Response File."

2. Navigate to **Policy Configuration -> Shared Components -> Authentication Schemes -> FAAuthScheme**.
3. Set the **Challenge Method** to **FORM**.
4. Set the **Authentication Module** to `SaaSModule`.
5. Set the **Challenge URL** to `/pages/oamLogin.jsp`.
6. Set the **Context Type** to **customWar**.
7. Set the **Context Value** to `/fusion_apps`.
8. Set the Challenge Parameters field with the following entries:
   - `federationEnabled=false`
   - `ssoChooserEnabled=false`
   - `fedSSOEnabled=false`
   - `initial_command=NONE`
   - `TAPPartnerId=OIFDAPPartner`
   - `TAPChallengeURL=https://SSO.mycompany.com:443/fed/user/spoam11g`
9. Click **Apply**.

12

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:

12.1 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 for various components of the enterprise deployment topologies outlined in Section 2.1.1, "Reference Topologies Documented in the Guide." The topologies and hosts are shown in Table 12–1.

Table 12–1 Hosts in Each Topology

Topology	Hosts
OAM11g/OIM11g	IDMHOST1
	IDMHOST2
OIF11g	IDMHOST1
	IDMHOST2

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).
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 12.5, "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.

12.2 Configuring Node Manager to Use SSL

By default, provisioning does not configure Node Manager in SSL mode. You must configure each node manager in the topology to use SSL.

For each node manager that has its configuration in *SHARED_CONFIG_DIR*`/nodemanager/hostname`, perform the following steps:

1. Edit the file `nodemanager.properties`.
2. Change the line `SecureListener=false` to `SecureListener=true`.
3. Save the file.
4. Restart Node Manager by killing the `nodemanager` process, and restart as follows.

   Execute the command:

   ```
   startNodeManagerWrapper.sh
   ```

Repeat these steps for each node manager.

12.3 Update Domain to Access Node Manager Using SSL

1. Log in to the WebLogic Administration console as the user `weblogic_idm`. (Console URLs are provided in Section 16.2, "About Identity Management Console URLs.")
2. Select **IDMDomain > Environment > Machines** from the Domain Structure menu.
3. Click on **Lock and Edit**.
4. Click on one of the machines, for example **idmhost1.mycompany.com**.
5. Click on the **Node Manager** tab.
6. Change **Type** from **Plain** to **SSL**.
7. Click **Save**

8. Repeat Steps 4-7 for each machine
9. Click **Activate Changes**.

12.4 Update Start and Stop Scripts to Use SSL

You must update the following files, which are generated by the provisioning tool. Each of these files is located in the directory *SHARED_CONFIG_DIR*/scripts/basescripts:

- stop_nodemanager_template.py
- stop_adminserver_template.py
- start_adminserver_template.py

Proceed as follows for each of the files:

1. Edit the file.
2. Locate the line which starts with nmConnect.
3. Change the last parameter from plain to SSL.

   For example in start_adminserver_template.py, change the line:

   ```
   nmConnect('admin', nmpwd, 'localhost', '5556', 'IDMDomain' ,
   '/u01/oracle/config/domains/IDMDomain' , 'Plain')
   ```

   to

   ```
   nmConnect('admin', nmpwd, 'localhost', '5556', 'IDMDomain' ,
   '/u01/oracle/config/domains/IDMDomain' , 'SSL')
   ```

4. Save the file.

12.5 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 12.5.1, "Generating Self-Signed Certificates Using the utils.CertGen Utility"
- Section 12.5.2, "Creating an Identity Keystore Using the utils.ImportPrivateKey Utility"
- Section 12.5.3, "Creating a Trust Keystore Using the Keytool Utility"
- Section 12.5.4, "Configuring Node Manager to Use the Custom Keystores"
- Section 12.5.5, "Configuring Managed WebLogic Servers to Use the Custom Keystores"
- Section 12.5.6, "Changing the Host Name Verification Setting for the Managed Servers"

12.5.1 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 'keystores' under the *ASERVER_HOME* directory. Note that certificates can be shared across WebLogic domains.

   ```
   cd ASERVER_HOME
   mkdir keystores
   ```

   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 keystores
   ```

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
```

Also create a certificate for the Admin Server virtual host.

```
java utils.CertGen Key_Passphrase ADMINVHN.mycompany.com_cert
ADMINVHN.mycompany.com_key domestic ADMINVHN.mycompany.com
```

12.5.2 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*`/keystores`).

   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` 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/keystores/IDMHOST1.mycompany.com_cert.pem ASERVER_
   HOME/keystores/IDMHOST1.mycompany.com_key.pem

   java utils.ImportPrivateKey appIdentityKeyStore.jks Key_Passphrase
   appIdentityIDMHOST2 Key_Passphrase ASERVER_
   HOME/keystores/IDMHOST2.mycompany.com_cert.pem ASERVER_
   HOME/keystores/IDMHOST2.mycompany.com_key.pem

   java utils.ImportPrivateKey appIdentityKeyStore.jks Key_Passphrase
   appIdentityADMVHN Key_Passphrase ASERVER_HOME/keystores/ADMINVNH.mycompany.com_
   cert.pem ASERVER_HOME/keystores/ADMINVNH.mycompany.com_key.pem
   ```

12.5.3 Creating a Trust Keystore Using the Keytool Utility

Follow these steps to create the trust keystore on each host, for example 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/keystores/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
   ```

12.5.4 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 *SHARED_CONFIG_DIR*/`nodemanager`/*hostname* directory, where *hostname* is the name of the host where nodemanager runs:

```
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/keystores/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 16.1, "Starting and Stopping 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.

12.5.5 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: `http://ADMIN.mycompany.com/console`
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`/keystores/appIdentityKeyStore.jks`

   - **Custom Identity Keystore Type:** Leave blank; it defaults to JKS.
   - **Custom Identity Keystore Passphrase:** The password (*`Keystore_Password`*) you provided in Section 12.5.3, "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`/keystores/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 12.5.3, "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_ods1`, use `appIdentityIDMHOST1`.
    - For `wls_ods2` use `appIdentityIDMHOST2`.
    - For `ADMINSERVER` use `appIdentityADMVHN`.

    In the **Private Key Passphrase** and the **Confirm Private Key Passphrase** fields, enter the password for the keystore that you created in Section 12.5.2, "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 16.1, "Starting and Stopping Components."

12.5.6 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. (Console URLs are provided in Section 16.2, "About Identity Management Console URLs.")
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**.

12.6 Update boot.properties Files

Each managed server has a `boot.properties` file which is created as part of the process described in previous sections. In order to start managed servers using the provisioning start script, you must update each of these files and comment out following line:

```
TrustKeyStore=DemoTrust
```

That is, when you have finished updating the file, the line should look like this:

```
#TrustKeyStore=DemoTrust
```

The files you must update are:

ASERVER_HOME`/servers/AdminServer/security/boot.properties`

and each managed server `boot.properties` file. These have path names of the form:

MSERVER_HOME`/servers/`*servername*`/security/boot.properties`

and

ASERVER_HOME`/servers/AdminServer/data/nodemanager/boot.properties`

12.7 Starting Node Manager

Run the following commands to start Node Manager.

```
cd SHARED_CONFIG_DIR/nodemanager/hostname
./startNodeManagerWrapper.sh
```

Note: 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/keystores/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.

13

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:

13.1 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.

13.2 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;
      ```

13.3 Creating a Multi Data Source Using the Oracle WebLogic Administration Console

The second step is to create a multi data source for the leasing table from the Oracle WebLogic Server Administration Console. (Console URLs are provided in Section 16.2, "About Identity Management Console URLs.") You create a data source to each of the Oracle RAC database instances during the process of setting up the multi data source, both for these data sources and the global leasing multi data source. When you create a data source:

- Ensure that this is a non-XA data source.
- The names of the multi data sources are in the format of `MultiDS-rac0`, `MultiDS-rac1`, and so on.
- Use Oracle's Driver (Thin) Version 9.0.1, 9.2.0, 10, 11.
- Data sources do not require support for global transactions. Therefore, do *not* use any type of distributed transaction emulation/participation algorithm for the data source (do not choose the **Supports Global Transactions** option, or the **Logging Last Resource**, **Emulate Two-Phase Commit**, or **One-Phase Commit** options of the **Supports Global Transactions** option), and specify a service name for your database.
- Target these data sources to the oim_cluster and the soa_cluster.

- Ensure the data source's connection pool initial capacity is set to 0 (zero). To do this, select **Services**, **JDBC**, and then **Datasources**. In the Datasources screen, click the **Datasource Name**, then click the **Connection Pool** tab, and enter **0** (zero) in the **Initial Capacity** field.

Creating a Multi Data Source

Perform these steps to create a multi data source:

1. From Domain Structure window in the Oracle WebLogic Server Administration Console, expand the **Services** node. The Summary of JDBC Data Source page appears.
2. Click **Data Sources**. The Summary of JDBC Multi Data Source page is displayed.
3. Click **Lock and Edit**.
4. Click **New Multi Data Source**. The Create a New JDBC Multi Data Source page is displayed.
5. Enter `leasing` as the name.
6. Enter `jdbc/leasing` as the JNDI name.
7. Select **Failover** as algorithm (default).
8. Click **Next.**
9. Select **oim_cluster** and **soa_cluster** as the targets.
10. Click **Next**.
11. Select **non-XA driver** (the default).
12. Click **Next**.
13. Click **Create New Data Source**.
14. Enter `leasing-rac0` as the name. Enter `jdbc/leasing-rac0` as the JNDI name. Enter `oracle` as the database type.

    Note: When creating the multi data sources for the leasing table, enter names in the format of *`MultiDS`*`-rac0`, *`MultiDS`*`-rac1`, and so on.

15. Click **Next**.
16. On JDBC Data Source Properties, select **Database Driver: Oracle's Driver (Thin) for RAC Service-Instance connections**.
17. Deselect **Supports Global Transactions**.
18. Click **Next**.
19. Enter the service name, database name, host port, and password for your leasing schema.
20. Click **Next**.
21. Click **Test Configuration** and verify that the connection works.
22. Click **Next**.
23. Target the data source to **oim_cluster** and **SOA cluster**.
24. Click **Finish**.

25. Select the data source you just created, for example `leasing-rac0`, and add it to the right screen.
26. Click **Create a New Data Source** for the second instance of your Oracle RAC database, target it to the oim_cluster and soa_cluster, repeating the steps for the second instance of your Oracle RAC database.
27. Add the second data source to your multi data source.
28. Click **Activate Changes**.

13.4 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 directory

SHARED_CONFIG_DIR/nodemanager/*hostname*

where *hostname* is the name of the host where the node manager is running.

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
```

Notes:

- `LogToStderr` must be set to true (in `node.propertes`), in order for you to see the properties in the output.
- 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 `startNodeManagerWrapper.sh` script, which is located in the *SHARED_CONFIG_DIR*/nodemanager/hostname 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 by setting `JAVA_OPTIONS` to: `-DInterface=eth3`

Then start Node Manager.

13.5 Setting Environment and Superuser Privileges for the wlsifconfig.sh Script

On Linux, you must set environment and superuser privileges for the `wlsifconfig.sh` script:

Ensure that your PATH environment variable includes the files listed in Table 13–1.

Table 13–1 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`	*SHARED_CONFIG_DIR*/nodemanager/HostName

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.

Grant `sudo` privilege to the WebLogic user `oracle` with no password restriction, and grant execute privilege on the `/sbin/ifconfig` and `/sbin/arping binaries`.

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
```

13.6 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: `http://ADMIN.mycompany.com/console`
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 **Candidate Machines for Migratable Servers** field, select the machine to which to allow migration and click the right arrow. In this case, select **IDMHOST1** and **IDMHOST2**.
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 16.1, "Starting and Stopping Components.".

Note: If migration is only going to be allowed to specific machines, do not specify candidates for the cluster, but rather specify candidates only on a server per server basis.

13.7 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: OIMVH1. (Console URLs are provided in Section 16.2, "About Identity Management Console URLs.")

Follow the previous steps to test server migration for the WLS_OIM2, WLS_SOA1, and WLS_SOA2 Managed Servers.

Table 13–2 shows the Managed Servers and the hosts they migrate to in case of a failure.

Table 13–2 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. (Console URLs are provided in Section 16.2, "About Identity Management Console URLs.")
2. Click **IDMDomain** on the left console.
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.

13.8 Backing Up the Server Migration Configuration

Back up the database and the WebLogic domain, as described in Section 16.5.3, "Performing Backups During Installation and Configuration."

14 Validating Provisioning

The provisioning process includes several validation checks to ensure that everything is working correctly. This chapter describes additional checks that you can perform for additional sanity checking.

This chapter contains the following sections:

14.1 Validating the Administration Server

Validate the WebLogic Administration Server as follows.

14.1.1 Verify Connectivity

Verify that you can access the administration console by accessing the URL:

`http://admin.mycompany.com/console and logging in as the user weblogic_idm`

Verify that all managed servers are showing a status of Running.

Verify that you can access Oracle Enterprise Manager Fusion Middleware Control by accessing the URL:

`http://admin.mycompany.com/em and logging in as the user weblogic_idm`

14.1.2 Validating Failover

Test failover of the Administration Server to IDMHOST2 and then fail back to IDMHOST1, as described in Section 16.8, "Manually Failing Over the WebLogic Administration Server."

14.2 Validating the Oracle Access Manager Configuration

To Validate that this has completed correctly.

1. Access the OAM console at: `http://ADMIN.mycompany.com/oamconsole`

2. Log in as the user identified by the entry in Section 8.2, "Update User Names in Provisioning Response File."
3. Click the **System Configuration** tab
4. Expand **Access Manager Settings - SSO Agents - OAM Agents**.
5. Click the open folder icon, then click **Search**.
6. You should see the WebGate agents `Webgate_IDM`, `Webgate_IDM_11g` and `IAMSuiteAgent`.

14.3 Validating Oracle Directory Services Manager (ODSM)

Validate the Application Tier configuration as follows:

14.3.1 Validating Browser Connection to ODSM Site

Follow these steps to validate that you can connect the Oracle Directory Services Manager site in a browser:

1. In a web browser, verify that you can connect to Oracle Directory Services Manager (ODSM) at:

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

   For example, on IDMHOST1, enter this URL, where `7005` is *`ODSM_PORT`* in Section 6.1, "Assembling Information for Identity Management Provisioning."

   ```
   http://IDMHOST1.mycompany.com:7005/odsm
   ```

   and on IDMHOST2, enter this URL:

   ```
   http://IDMHOST2.mycompany.com:7005/odsm
   ```

2. In a web browser, verify that you can access ODSM through the load balancer address:

   ```
   http://ADMIN.mycompany.com/odsm
   ```

14.3.2 Validating ODSM Connections to Oracle Internet Directory

Validate that Oracle Directory Services Manager can create connections to Oracle Internet Directory.

Create a connection to the Oracle Internet Directory on each ODSM instance separately. Even though ODSM is clustered, the connection details are local to each node. Proceed as follows:

1. Set environment variables. Set `ORACLE_HOME` to *`IDM_ORACLE_HOME`*, `ORACLE_INSTANCE` to *`OID_ORACLE_INSTANCE`*, and `JAVA_HOME` to *`JAVA_HOME`*. Set `PATH` to include *`JAVA_HOME`*.
2. Launch Oracle Directory Services Manager from IDMHOST1:

   ```
   http://IDMHOST1.mycompany.com:7005/odsm
   ```

3. Create a connection to the Oracle Internet Directory virtual host by providing the following information in ODSM:
   - Server: `OIDSTORE.mycompany.com`
   - Port: `636` (*`LDAP_LBR_SSL_PORT`*)

- Enable the SSL option
- User: `cn=orcladmin`
- Password: *`ldap-password`*

4. Launch Oracle Directory Services Manager from IDMHOST2.

   Follow Step 3 to create a connection to Oracle Internet Directory from IDMHOST2

   `http://IDMHOST2.mycompany.com:7005/odsm`

5. Create a connection to the Oracle Internet Directory virtual host by providing the corresponding information in ODSM

> **Note:** Accept the certificate when prompted.

14.4 Validating Oracle Identity Manager

Validate Oracle Identity Manager as follows.

14.4.1 Validating the Oracle Internet Directory Instances

To validate the Oracle Internet Directory instances, ensure that you can connect to each Oracle Internet Directory instance and the load balancing router using these commands:

> **Note:** Ensure that the following environment variables are set before using `ldapbind`:
>
> - *`ORACLE_HOME`* (set to *`IDM_ORACLE_HOME`*)
> - *`OID_ORACLE_INSTANCE`*
> - *`PATH`* - The following directory locations should be in your *`PATH`*:
>
>   *`ORACLE_HOME`*`/bin`
>
>   *`ORACLE_HOME`*`/ldap/bin`
>
>   *`ORACLE_HOME`*`/ldap/admin`

```
ldapbind -h LDAPHOST1.mycompany.com -p 3060 -D "cn=orcladmin" -q
ldapbind -h LDAPHOST1.mycompany.com -p 3131-D "cn=orcladmin" -q -U 1
ldapbind -h LDAPHOST2.mycompany.com -p 3060 -D "cn=orcladmin" -q
ldapbind -h LDAPHOST2.mycompany.com -p 3131-D "cn=orcladmin" -q -U 1

ldapbind -h OIDIDSTORE.mycompany.com -p 389 -D "cn=orcladmin" -q
ldapbind -h OIDIDSTORE.mycompany.com -p 636 -D "cn=orcladmin" -q -U 1
```

> **Note:** The `-q` option prompts the user for a password. LDAP Tools have been modified to disable the options `-w` *`password`* and `-P` *`password`* when the environment variable `LDAP_PASSWORD_PROMPTONLY` is set to `TRUE` or `1`. Use this feature whenever possible.

14.4.2 Validating the Oracle Virtual Directory Instances

To validate the Oracle Virtual Directory instances, ensure that you can connect to each Oracle Virtual Directory instance and the load balancing router using these `ldapbind` commands:

```
ldapbind -h LDAPHOST1.mycompany.com -p 6501 -D "cn=orcladmin" -q
ldapbind -h LDAPHOST2.mycompany.com -p 6501 -D "cn=orcladmin" -q
ldapbind -h IDSTORE.mycompany.com -p 389 -D "cn=orcladmin" -q

ldapbind -h LDAPHOST1.mycompany.com -p 7501 -D "cn=orcladmin" -q -U 1
ldapbind -h LDAPHOST2.mycompany.com -p 7501 -D "cn=orcladmin" -q -U 1
```

14.4.3 Validating SSL Connectivity

You can manually verify that the SSL connection has been set up correctly by generating a wallet and then using that wallet to access Oracle Internet Directory. Proceed as follows:

Execute the command

```
cd ORACLE_COMMON_HOME/bin
./SSLClientConfig.sh -component cacert
```

providing the following inputs:

- LDAP host name: Name of the Oracle Internet Directory server containing the Domain Certificate
- LDAP port: Port used to access Oracle Internet Directory (*OID_LDAP_PORT*), for example: `3060`
- LDAP User: Oracle Internet Directory admin user, for example: `cn=orcladmin`
- Password: Oracle Internet Directory admin user password
- SSL Domain for CA: This is `IDMDomain`.
- Password for truststore: This is the password you want to assign to your wallet.

When the command executes, it generates wallets in the directory *IDM_ORACLE_HOME*`/rootCA/keystores/common`

Now that you have a wallet, you can test that authentication is working by executing the command:

```
ldapbind -h LDAPHOST1.mycompany.com -p 3131 -U 2 -D cn=orcladmin -q -W "file:IDM_
ORACLE_HOME/rootCA/keystores/common" -Q
```

You will be prompted for your Oracle Internet Directory password and for the wallet password. If the bind is successful, the SSL connection has been set up correctly.

14.4.4 Validating Oracle Identity Manager

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/oim
```

Log in using the `xelsysadm` username and password.

14.4.5 Validating SOA Instance from the WebTier

Validate SOA by accessing the URL:

`http://IDMINTERNAL.mycompany.com:80/soa-infra`

and logging in using the `xelsysadm` username and password.

14.4.6 Validating Oracle Identity Manager Instance

Validate the Oracle Identity Manager Server Instance by bringing up the Oracle Identity Manager Console in a web browser at:

`https://SSO.mycompany.com/oim`

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 URL:

`http://IDMINTERNAL.mycompany.com/soa-infra`

Log in as the `weblogic_idm` user.

14.5 Validating WebGate and the Oracle Access Manager Single Sign-On Setup

To validate that WebGate is functioning correctly, open a web browser and go the OAM console at: `http://ADMIN.mycompany.com/oamconsole`

You now see the Oracle Access Manager Login page displayed. Enter your OAM administrator user name (for example, `oamadmin`) and password and click **Login**. Then you see the Oracle Access Manager console displayed.

To validate the single sign-on setup, open a web browser and go the WebLogic Administration Console at `http://ADMIN.mycompany.com/console` and to Oracle Enterprise Manager Fusion Middleware Control at: `http://ADMIN.mycompany.com/em`

The Oracle Access Manager Single Sign-On page displays. Provide the credentials for the `weblogic_idm` user to log in.

15

Scaling Enterprise Deployments

The reference enterprise topology discussed in this guide is highly scalable. It can be scaled up and or scaled out. This chapter explains how to do so.

To scale up the topology, you add a new component instance to a node already running one or more component instances. To scale out the topology, you add new component instances to new nodes.

This chapter contains the following topics:

15.1 Scaling Up the Topology

The Oracle Identity Management topology described in the guide has three tiers: the Directory 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.

15.2 Scaling Out the Topology

You scale out a topology by adding new components to new nodes. The components in all three tiers of the Oracle Identity Management topology described in this guide can be scaled out by adding a new component instance to a new node.

15.3 Scaling Out the Database

If you require more than the standard four database instances, then you must add additional database instances manually. The following steps assume that you have the database instances already configured.

Oracle Identity Management components interface with the database using WebLogic Datasources. In systems that use Oracle RAC, Data sources are configured as Multi Datasources in Identity Management. A multi datasource is made up of several child datasources, one for each RAC database Instance. The Identity Management Applications interface with the RAC database by accessing the parent multi data source. If you add a new database instance, then you must create new datasources for each of the existing multi datasources and then add the new data source into the pre-existing multi data source. Because different applications use different datasources, you must add the database to each data source that is using the database.

To do this perform the following steps:

1. Log In to the WebLogic console using the URL: `http://admin.mycompany.com/console`
2. Select **Services > Messaging > Data Sources** from the Domain Structure window.
3. Click on a Data Source which has an ID of the type **Multi**
4. Click on the **Targets** tab and make a note of what targets the multi data source is assigned to.
5. Click on the **Configuration** tab and the **Data Sources** subtab. The chosen box shows you what Data sources are currently part of the multi datasource.
6. Select **Services > Messaging > Data Sources** from the Domain Structure window.
7. This time click on one of the data sources that are currently part of the multi datasource.
8. Make a note of the following attributes. (The example shown is the data source EDNDDatasource-rc0, which is part of the multi data source EDNDatasource.)

   General Tab

   JNDI Name: for example, `jdbc/EDNDatasource-rc0`

   Connection Pool Tab:

   - **URL**: for example: `jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=idmdb-scan.mycompany.com)(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=soa_edg.mycpmpany.com)(INSTANCE_NAME=idmdb1)))`
   - **Driver Class**: for example: `oracle.jdbc.xa.client.OracleXADataSource`
   - **Properties**: for example:

     `user=FA_SOAINFRA`

     `oracle.net.CONNECT_TIMEOUT=10000`
   - **System Properties**: for example: `v$session.program=JDBCProgramName`
   - **Password**: This is the database password you used when you ran the RCU
9. Return to the Overview screen
10. Click **Lock and Edit**.
11. Click **New >Generic Datasource**
12. Select **Services - Messaging > Data Sources** from the Domain Structure window.
13. Provide the following:
    - **Name**: Choose a name for the datasource for example: `EDNDDatasource-rc3`

- **JNDI Name**: Enter a jndi name, for example: **jdbc/EDNDatasource-rc3**
- **Database Type**: `oracle`

14. Click **Next**.
    - **Database Driver**: This can be determined from the Driver class, that is, `xa` or `non xa`. For example: Oracle's Driver (thin XA) for RAC Service-Instance connections; Versions 10 and later
15. Click **Next**.
16. On the Transaction Options page, click **next**.
17. On the Connection Properties page enter:
    - **Service name**: Database service name for example: `soaedg.mycompany.com`
    - **Database Name**: Enter the name of the database for example: IDMDB
    - **Host Name**: If this is for an 11.2 database enter the database scan address. Otherwise enter the VIP of the host being added.
    - **Port**: Enter the listener port, for example: `1521`
    - **Database User Name**: enter the value from Properties, for example: `FA_SOAINFRA`
    - Enter the password assigned when RCU was run and confirm it.

    Click **Next**.
18. On the Test Configuration page, test the connection.

    Click Next
19. On the Targets page, assign the same targets as you noted for the mutli datasource.
20. Click **Finish**.
21. Now that the datasource has been defined, it can be added to the existing multi datasource.

    Select **Services > Messaging -> Data Sources** from the Domain Structure window.
22. Click on the multi datasource, for example: **EDNDDatasource**
23. Click on the **Targets** tab and add the newly created data source.
24. Click **Finish**
25. Click **Activate Changes**
26. Repeat for each data source that uses the database.

15.4 Scaling the Directory Tier

The Directory tier consists of two LDAP hosts, each running Oracle Internet Directory and Oracle Virtual Directory.

This section contains the following topics:

- Section 15.4.1, "Scaling Oracle Internet Directory"
- Section 15.4.2, "Scaling Oracle Virtual Directory"

15.4.1 Scaling Oracle Internet Directory

The Directory Tier has two Oracle Internet Directory nodes, LDAPHOST1 and LDAPHOST2, each running an Oracle Internet Directory instance.

When scaling up, use the existing Oracle Identity Management binaries on either node for creating the new Oracle Internet Directory instance.

To add a new Oracle Internet Directory instance to either Oracle Internet Directory node, or to scale out Oracle Internet Directory instances, perform the steps in the following subsections:

- Section 15.4.1.1, "Assembling Information for Scaling Oracle Internet Directory"
- Section 15.4.1.2, "Configuring an Additional Oracle Internet Directory Instance"
- Section 15.4.1.3, "Registering Oracle Internet Directory with the WebLogic Server Domain (IDMDomain)"
- Section 15.4.1.4, "Configuring Oracle Internet Directory to Accept Server Authentication Mode SSL Connections"
- Section 15.4.1.5, "Reconfiguring the Load Balancer."

15.4.1.1 Assembling Information for Scaling Oracle Internet Directory

Assemble the following information before scaling Oracle Internet Directory.

Description	Variable	Documented Value	Customer Value
Host Name		LDAPHOST3.mycompany.com	
OID Port	*OID_LDAP_PORT*	3060	
OID SSL Port	*OID_LDAP_SSL_PORT*	636	
Oracle Instance Location	*OID_ORACLE_INSTANCE*		
Oracle Instance/component Name	oid*n*	oid3	
OID Admin Password			
Password to protect your SSL wallet/keystore	*COMMON_IDM_PASSWORD*		
Password for the CA wallet	*COMMON_IDM_PASSWORD*		
WebLogic Admin Host		ADMINVHN.mycompany.com	
WebLogic Admin Port	*WLS_ADMIN_PORT*	7001	
WebLogic Admin User		weblogic_idm	
WebLogic Admin Password			

15.4.1.2 Configuring an Additional Oracle Internet Directory Instance

The schema database must be running before you perform this task. Follow these steps to install Oracle Internet Directory on the host:

1. Before starting the configuration, determine the ports you want to use for the new directory instance. For Scale out, these can be the same as the other instances you have. For Scale Up these ports must be unique to the new server instance.

   Ensure that ports you want are not in use by any service on the computer by issuing these commands for the operating system you are using.

   For example, on Linux, you would enter:

   ```
   netstat -an | grep "3060"
   ```

```
netstat -an | grep "3131"
```

If a port is not in use, no output is returned from the command. If the ports are in use (that is, if the command returns output identifying either port), you must free them.

Remove the entries for the ports in the `/etc/services` file and restart the services, as described in Section 16.1, "Starting and Stopping Components," or restart the computer.

2. Create a file containing the ports used by Oracle Internet Directory. On Disk1 of the installation media, locate the file `stage/Response/staticports.ini`. Copy it to a file called `oid_ports.ini`. Delete all entries in `oid_ports.ini` except for `Non-SSL Port for Oracle Internet Directory` and `SSL Port for Oracle Internet Directory`. Change the values of those ports to the ports you want to use, for example: `3060` and `3131`.

   Note: If the port names in the file are slightly different from those listed in this step, use the names in the file.

3. Start the Oracle Identity Management 11*g* Configuration Wizard by running *`OID_ORACLE_HOME`*`/bin/config.sh`.
4. On the Welcome screen, click **Next**.
5. On the Select Domain screen, select **Configure without a Domain**.

   Click **Next**.
6. On the Specify Installation Location screen, specify the following values:

   Oracle Instance Location: *`OID_ORACLE_INSTANCE`*

   Oracle Instance Name: `oid`*`n`*, where *`n`* is a sequential number for the instance. For example, if you already have two instances configured, *`n`* will be `3`, so you would enter `oid3`.

   Click **Next**.
7. On the Specify Email for Security Updates screen, specify these values:
   - Email Address: Provide the email address for your My Oracle Support account.
   - Oracle Support Password: Provide the password for your My Oracle Support account.
   - Check the check box next to the **I wish to receive security updates via My Oracle Support** field.

   Click **Next**.
8. On the Configure Components screen, select Oracle Internet Directory, deselect all the other components, and click **Next**.
9. On the Configure Ports screen, you use the `oid_ports.ini` file you created in Step 2 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 `oid_ports.ini`.
   c. Click **Save**, then click **Next**.

10. On the Specify Schema Database screen, select **Use Existing Schema** and specify the following values:

    - Connect String: `OIDDBHOST1-VIP.mycompany.com:1521:idmdb1^OIDDBHOST2-VIP.mycompany.com:1521:idmdb2@OIDEDG.mycompany.com`

    Notes:

    - The Oracle RAC database connect string information must be provided in the format:

      host1:*port1*:*instance1*^*host2*:*port2*:*instance2*@*servicename*

    - During this installation, it is not required that all the Oracle RAC instances to be up. If one Oracle RAC instance is up, the installation can proceed.

    - You must provide complete and accurate information. Specifically, you must provide the correct host, port, and instance name for each Oracle RAC instance, and the service name you provide must be configured for all the specified Oracle RAC instances.

      Any incorrect information entered in the Oracle RAC database connect string must be corrected manually after the installation.

    - User Name: `ODS`
    - Password: *password* This is the password of the ODS schema in the database as specified when RCU was run.

    Click **Next**.

11. The ODS Schema in use message appears. The ODS schema chosen is already being used by the existing Oracle Internet Directory instance. Therefore, the new Oracle Internet Directory instance being configured would reuse the same schema.

    Choose **Yes** to continue.

    A popup window with this message appears:

    ```
    "Please ensure that the system time on this Identity Management Node is
    in sync with the time on other Identity management Nodes that are part
    of the Oracle Application Server Cluster (Identity Management)
    configuration. Failure to ensure this may result in unwanted instance
    failovers, inconsistent operational attributes in directory entries and
    potential inconsistent behavior of password state policies."
    ```

    Ensure that the system time is synchronized among all the IDMHOSTs. See Section 5.10, "Synchronize Oracle Internet Directory Nodes" for more information.

    Click **OK** to continue.

12. On the Specify OID Admin Password screen, specify the Oracle Internet Directory administration password.

    Note: If you see a message saying that OID is not running, verify that the orcladmin account has not become locked and try again. Do not continue until this message is no longer displayed.

Click **Next**.

13. 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. When they are correct, click **Configure**.
14. On the Configuration screen, multiple configuration assistants are launched in succession. This process can be lengthy. Wait for the configuration process to finish.
15. On the Installation Complete screen, click **Finish** to confirm your choice to exit.
16. To validate the installation of the Oracle Internet Directory instance on the new LDAP host, issue these commands:

```
ldapbind -h LDAPHOST.mycompany.com -p 3060 -D "cn=orcladmin" -q
ldapbind -h LDAPHOST.mycompany.com -p 3131-D "cn=orcladmin" -q -U 1
```

where *LDAPHOST* is the host where the new instance is running.

Note: Ensure that the following environment variables are set before using `ldapbind`:

- *ORACLE_HOME*
- *ORACLE_INSTANCE*
- *PATH* - The following directory locations should be in your *PATH*:

  IDM_ORACLE_HOME/bin

  IDM_ORACLE_HOME/ldap/bin

  IDM_ORACLE_HOME/ldap/admin

15.4.1.3 Registering Oracle Internet Directory with the WebLogic Server Domain (IDMDomain)

All the Oracle Fusion Middleware components deployed in this enterprise deployment are managed by using Oracle Enterprise Manager Fusion Middleware Control. To manage the Oracle Internet Directory component with this tool, you must register the component and the Oracle Fusion Middleware instance that contains it with an Oracle WebLogic Server domain. A component can be registered either at install time or post-install. A previously un-registered component can be registered with a WebLogic domain by using the `opmnctl registerinstance` command.

To register the Oracle Internet Directory instances installed on the host where the new server instance is running, follow these steps for each instance:

1. On the new host:

   Set `ORACLE_HOME` to *IDM_ORACLE_HOME*.

   Set `ORACLE_INSTANCE` to *OID_ORACLE_INSTANCE*, where *OID_ORACLE_INSTANCE* is the location of the newly created instance.

2. Execute the `opmnctl registerinstance` command:

   ```
   OID_ORACLE_INSTANCE/bin/opmnctl registerinstance -adminHost WLSHostName
   -adminPort WLSPort -adminUsername adminUserName
   ```

   For example:

   ```
   OID_ORACLE_INSTANCE/bin/opmnctl registerinstance -adminHost
   ```

```
ADMINVHN.mycompany.com -adminPort 7001 -adminUsername weblogic
```

The command requires login to WebLogic Administration Server (`ADMINVHN.mycompany.com`)

Username: `weblogic`

Password: `*******`

Note: For additional details on registering Oracle Internet Directory components with a WebLogic Server domain, see the "Registering an Oracle Instance or Component with the WebLogic Server" section in *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*.

---

3. On the host where the new instance is running, update the Enterprise Manager Repository URL using the `emctl` utility with the `switchOMS` flag. This will enable the local emagent to communicate with the WebLogic Administration Server using the virtual IP address. The `emctl` utility is located under the *`OID_ORACLE_INSTANCE`*`/EMAGENT/EMAGENT/bin` directory.

   Syntax:

   ```
   ./emctl switchOMS ReposURL
   ```

   For Example:

   ```
   ./emctl switchOMS http://ADMINVHN:7001/em/upload
   ```

   Output:

   ```
   ./emctl switchOMS http://ADMINVHN.mycompany.com:7001/em/upload
   Oracle Enterprise Manager 10g Release 5 Grid Control 10.2.0.5.0.
   Copyright (c) 1996, 2009 Oracle Corporation.  All rights reserved.
   SwitchOMS succeeded.
   ```

4. Force the agent to reload its configuration by issuing the command:

   ```
   ./emctl reload
   ```

5. Check that the agent is using the correct Upload URL using the command:

   ```
   ./emctl status agent
   ```

6. Validate that the agents on the host where the new server is running are configured properly to monitor their respective targets. Follow these steps to complete this task:

   - Use a web browser to access Oracle Enterprise Manager Fusion Middleware Control at:

     `http://ADMINVHN.mycompany.com:7001/em`

     Log in as the `weblogic_idm` user.

   - From the Domain Home Page navigate to the Agent-Monitored Targets page using the menu under **Farm** -> **Agent-Monitored Targets**.

   - Update the WebLogic monitoring user name and the WebLogic monitoring password.

– Enter `weblogic_idm` as the WebLogic monitoring user name and the password for the `weblogic_idm` user as the WebLogic monitoring password.

– Click **OK** to save your changes.

15.4.1.4 Configuring Oracle Internet Directory to Accept Server Authentication Mode SSL Connections

If you are using SSL Authentication Mode, you must perform the following to ensure that your Oracle Internet Directory instances are capable of accepting requests using this mode. You must configure each Oracle Internet Directory instance independently.

15.4.1.4.1 Configuring Oracle Internet Directory for SSL To enable Oracle Internet Directory to communicate using SSL Server Authentication Mode, perform the following steps on the host where the new server is running:

> **Note:** When you perform this operation, only the Oracle Internet Directory instance you are working on should be running.

1. Set the *ORACLE_HOME*, *ORACLE_INSTANCE* and *JAVA_HOME* variables. For example:
   - Set `ORACLE_HOME` to *IDM_ORACLE_HOME*.
   - Set `ORACLE_INSTANCE` to *OID_ORACLE_INSTANCE*.
   - Set `JAVA_HOME` to *DIR_MW_HOME*`/jdk6`
   - Set the `PATH` variable to include *JAVA_HOME*.
2. To enable SSL Server Authentication use the tool SSLServerConfig which is located in:

   ORACLE_COMMON_HOME`/bin`

   For example

   ```
   $ORACLE_COMMON_HOME/bin/SSLServerConfig.sh -component oid
   ```

3. When prompted, enter the following information:
   - LDAP Hostname: Central LDAP host, for example: `POLICYSTORE.mycompany.com`
   - LDAP port: *LDAP_POLICY_LBR_PORT*, for example: `389`
   - Admin user DN: `cn=orcladmin`
   - Password: *orcladmin_password*
   - sslDomain for the CA: `IDMDomain` Oracle recommends that the SSLDomain name be the same as the Weblogic domain name to make reference easier.
   - Password to protect your SSL wallet/keystore: *COMMON_IDM_PASSWORD*
   - Enter confirmed password for your SSL wallet/keystore: *COMMON_IDM_PASSWORD*
   - Password for the CA wallet: *certificate_password*. This is the master password you used when you ran provisioning.
   - Country Name 2 letter code: Two letter country code, such as `US`
   - State or Province Name: State or province, for example: `California`

- Locality Name: Enter the name of your city, for example: `RedwoodCity`
- Organization Name: Company name, for example: `mycompany`
- Organizational Unit Name: Leave at the default
- Common Name: Name of this host, for example: `LDAPHOST3.mycompany.com`
- OID component name: Name of your Oracle Instance, for example: `oid1`. If you need to determine what your OID component name is, execute the command:

  `OID_ORACLE_INSTANCE``/bin/opmnctl status`

- WebLogic admin host: Host running the WebLogic Administration Server, for example:. `ADMINVHN.mycompany.com`
- WebLogic admin port: *`WLS_ADMIN_PORT`*, for example: `7001`
- WebLogic admin user: Name of your WebLogic administration user, for example: `weblogic`
- WebLogic password: *`password`*.
- AS instance name: Name of the new instance you entered in Step 6 of Section 15.4.1.2, "Configuring an Additional Oracle Internet Directory Instance," for example: `oid3`.
- SSL wallet name for OID component [oid_wallet1]: Accept the default
- Do you want to restart your OID component: `Yes`
- Do you want to test your SSL setup? `Yes`
- SSL Port of your OID Server: *`OID_LDAP_SSL_PORT`*, for example: `3131`

Sample output:

```
Server SSL Automation Script: Release 11.1.1.4.0 - Production
Copyright (c) 2010 Oracle. All rights reserved.

Downloading the CA wallet from the central LDAP location...
>>>Enter the LDAP Hostname [SLC00DRA.mycompany.com]: POLICYSTORE.mycompany.com
>>>Enter the LDAP port [3060]: 3060
>>>Enter an admin user DN [cn=orcladmin]
>>>Enter password for cn=orcladmin:
>>>Enter the sslDomain for the CA [idm]: IDMDomain
>>>Enter a password to protect your SSL wallet/keystore:
>>>Enter confirmed password for your SSL wallet/keystore:
>>>Enter password for the CA wallet:
>>>Searching the LDAP for the CA usercertificate ...
Importing the CA certifcate into trust stores...
>>>Searching the LDAP for the CA userpkcs12 ...

Invoking OID SSL Server Configuration Script...
Enter attribute values for your certificate DN
>>>Country Name 2 letter code [US]:
>>>State or Province Name [California]:
>>>Locality Name(eg, city) []:Redwood
>>>Organization Name (eg, company) [mycompany]:
>>>Organizational Unit Name (eg, section) [oid-20110524015634]:
>>>Common Name (eg, hostName.domainName.com) [SLC00XXX.mycompany.com]:
The subject DN is
cn=SLC00DRA.mycompany.com,ou=oid-20110524015634,l=Redwood,st=California,c=US
```

```
Creating an Oracle SSL Wallet for oid instance...
/u01/oracle/products/access/idm/../oracle_common/bin
>>>Enter your OID component name: [oid1]
>>>Enter the weblogic admin server host [SLC00XXX.mycompany.com] ADMINVHN
>>>Enter the weblogic admin port: [7001]
>>>Enter the weblogic admin user: [weblogic]
>>>Enter weblogic password:
>>>Enter your AS instance name:[asinst_1] oid1
>>>Enter an SSL wallet name for OID component [oid_wallet1]
Checking the existence of oid_wallet1 in the OID server...
Configuring the newly generated Oracle Wallet with your OID component...
Do you want to restart your OID component?[y/n]y

Do you want to test your SSL set up?[y/n]y
>>>Please enter your OID ssl port:[3131] 3131
Please enter the OID hostname:[SLC00DRA.mycompany.com] LDAPHOST3.mycompany.com
>>>Invoking IDM_ORACLE_HOM/bin/ldapbind -h LDAPHOST3.mycompany.com -p 3131-U 2 -D
cn=orcladmin ...
Bind successful

Your oid1 SSL server has been set up successfully
```

Confirm that the script has been successful.

Repeat all the steps in this section, Section 15.4.1.4, "Configuring Oracle Internet Directory to Accept Server Authentication Mode SSL Connections." for each Oracle Internet Directory instance.

15.4.1.5 Reconfiguring the Load Balancer

If you are accessing your Oracle Internet Directory instances through a load balancer, add the new Oracle Internet Directory instance to the existing server pool defined on the load balancer for distributing requests across the Oracle Internet Directory instances.

15.4.2 Scaling Oracle Virtual Directory

The Directory Tier has two nodes, LDAPHOST1 and LDAPHOST2, each running an Oracle Virtual Directory instance.

When scaling up, you can use existing Oracle Identity Management binaries on either node for creating the new Oracle Virtual Directory instance.

To add a new Oracle Virtual Directory instance to either Oracle Virtual Directory node, or to scale out Oracle Virtual Directory instances, perform the steps in the following subsections:

- Section 15.4.2.1, "Assembling Information for Scaling Oracle Virtual Directory"
- Section 15.4.2.2, "Configuring an Additional Oracle Virtual Directory"
- Section 15.4.2.3, "Post-Configuration Steps"
- Section 15.4.2.4, "Creating ODSM Connections to Oracle Virtual Directory"
- Section 15.4.2.5, "Creating Adapters in Oracle Virtual Directory"
- Section 15.4.2.6, "Reconfiguring the Load Balancer"

15.4.2.1 Assembling Information for Scaling Oracle Virtual Directory

Assemble the following information before scaling Oracle Virtual Directory.

Description	Variable	Documented Value	Customer Value
Host Name		LDAPHOST3.mycompany.com	
OVD Listen Port	*OVD_PORT*	6501	
OVD SSL Port	*OVD_SSL_PORT*	7501	
Oracle Virtual Directory Proxy Port	*OVD_ADMIN_PORT*	8899	
Oracle Instance Location	*OVD_ORACLE_INSTANCE*	/u02/local/oracle/config/instances/oid*n*	
OVD Existing Instance/Component Name	ovd*n*	ovd1	
Newly Created Instance/Component Name	ovd*n*	ovd3	
OVD Administrator Password			
WebLogic Admin Host	*WLSHostName*	ADMINVHN.mycompany.com	
WebLogic Admin Port	*WLS_PORT*	7001	
WebLogic Admin User	*adminUserName*	weblogic_idm	
WebLogicAdmin Password			
Back end Identity Store host	*OID_LBR_HOST*	OIDIDSTORE.mycompany.com	
Back end Identity Store port	*OID_LDAP_PORT*	3060	
Identity Store LDAP admin password			
Password to protect your SSL wallet/keystore	*COMMON_IDM_PASSWORD*		
Password for the CA wallet (created when you ran IdM Provisioning Wizard)	*COMMON_IDM_PASSWORD*		

15.4.2.2 Configuring an Additional Oracle Virtual Directory

Follow these steps to configure the new Oracle Virtual Directory instance:

1. Ensure that ports you are using (*OVD_PORT* and *OVD_SSL_PORT* in Section 3.7, "Fixed Ports Used by the Provisioning Wizard") are not in use by any service on the computer by issuing these commands for the operating system you are using.

   On Linux:

   ```
   netstat -an | grep "6501"
   netstat -an | grep "7501"
   ```

   If a port is not in use, no output is returned from the command. 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 used by Oracle Virtual Directory in the `/etc/services` file and restart the services, as described in Section 16.1, "Starting and Stopping Components," or restart the computer.

2. Create a file containing the ports used by Oracle Virtual Directory. On Disk1 of the installation media, locate the file `stage/Response/staticports.ini`. Copy it to a file called `ovd_ports.ini`. Delete all entries in `ovd_ports.ini` except for `Non-SSL Port for Oracle Virtual Directory` and `SSL Port for Oracle Virtual`

`Directory`. Change the values of those ports to the ports you want to use, for example: `3060` and `3131`.

> **Note:** If the port names in the file are slightly different from those listed in this step, use the names in the file.

3. Start the Oracle Identity Management 11*g* Configuration Wizard by running *`OID_ORACLE_HOME`*`/bin/config.sh`.
4. On the Welcome screen, click **Next**.
5. On the Select Domain screen, select **Configure without a Domain**.

   Click **Next**.
6. On the Specify Installation Location screen, specify the following values:

   Oracle Instance Location: *`OVD_ORACLE_INSTANCE`*

   Oracle Instance Name: `ovd`*`n`*, where *`n`* is a sequential number for the instance. For example, if you already have two instances configured, *`n`* will be `3`, so you would enter `ovd3`.

   Click **Next**.
7. On the Specify Email for Security Updates screen, specify these values:
   - **Email Address**: Provide the email address for your My Oracle Support account.
   - **Oracle Support Password**: Provide the password for your My Oracle Support account.
   - Check the check box next to the **I wish to receive security updates via My Oracle Support** field.

   Click **Next**.
8. On the Configure Components screen, select Oracle Virtual Directory, deselect all the other components, and click **Next**.
9. On the Configure Ports screen, you use the `ovd_ports.ini` file you created in Step 2 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 `ovd_ports.ini`.
   c. Click **Save**, then click **Next**.
10. On the Specify Virtual Directory screen: In the Client Listeners section, enter:
    - LDAP v3 Name Space: *`REALM_DN`* in Section 6.1, "Assembling Information for Identity Management Provisioning," for example: `dc=mycompany,dc=com`

    In the OVD Administrator section, enter:
    - Administrator User Name: `cn=orcladmin`
    - Password: *`administrator_password`*
    - Confirm Password: *`administrator_password`*

    Select **Configure the Administrative Server in secure mode**.

Click **Next**.

11. 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. When they are correct, click **Configure**.
12. On the Configuration screen, multiple configuration assistants are launched in succession. This process can be lengthy. Wait for the configuration process to finish.

    Click **Next**.
13. On the Installation Complete screen, click **Finish** to confirm your choice to exit.
14. To validate the installation of the Oracle Virtual Directory instance on the host, issue these commands:

```
ldapbind -h LDAPHOST.mycompany.com -p 6501 -D "cn=orcladmin" -q
ldapbind -h LDAPHOST.mycompany.com -p 7501 -D "cn=orcladmin" -q -U 1
```

    where *LDAPHOST* is the host where the new instance is running.

Note: Ensure that the following environment variables are set before using `ldapbind`:

- Set *ORACLE_HOME* to *OID_ORACLE_HOME*.
- Set *ORACLE_INSTANCE* to *OVD_ORACLE_INSTANCE*.
- *PATH* - The following directory locations should be in your *PATH*:

  OID_ORACLE_HOME/bin

  OID_ORACLE_HOME/ldap/bin

  OID_ORACLE_HOME/ldap/admin

15.4.2.3 Post-Configuration Steps

This section contains the following topics:

- Section 15.4.2.3.1, "Registering Oracle Virtual Directory with the Oracle WebLogic Server Domain (IDMDomain)"
- Section 15.4.2.3.2, "Configuring Oracle Virtual Directory for SSL"

15.4.2.3.1 Registering Oracle Virtual Directory with the Oracle WebLogic Server Domain (IDMDomain)

All the Oracle Fusion Middleware components deployed in this enterprise deployment are managed by using Oracle Enterprise Manager Fusion Middleware Control. To manage the Oracle Virtual Directory component with this tool, you must register the component and the Oracle Fusion Middleware instance that contains it with an Oracle WebLogic Server domain. A component can be registered either at install time or post-install. A previously un-registered component can be registered with a WebLogic domain by using the `opmnctl registerinstance` command.

To register the Oracle Virtual Directory instances, follow these steps on the host where the new instance is running:

1. Set the *ORACLE_HOME* to *OID_ORACLE_HOME*.
2. Set `ORACLE_INSTANCE` to *OVD_ORACLE_INSTANCE1*, where *OVD_ORACLE_INSTANCE1* is the location of the newly-created instance.
3. Execute the `opmnctl registerinstance` command:

```
OVD_ORACLE_INSTANCE/bin/opmnctl registerinstance -adminHost WLSHostName
-adminPort WLSPort -adminUsername adminUserName
```

For example:

```
OVD_ORACLE_INSTANCE/bin/opmnctl registerinstance \
   -adminHost ADMINVHN.mycompany.com -adminPort 7001 -adminUsername weblogic
```

The command requires login to WebLogic Administration Server.

Username: `weblogic`

Password: *`password`*

> **Note:** For additional details on registering Oracle Virtual Directory components with a WebLogic Server domain, see the "Registering an Oracle Instance Using OPMNCTL" section in *Oracle Fusion Middleware Administrator's Guide for Oracle Virtual Directory*.

4. In order to manage Oracle Virtual Directory by using Oracle Enterprise Manager Fusion Middleware Control, you must update the Enterprise Manager Repository URL to point to the virtual IP address associated with the WebLogic Administration Server. Do this using the `emctl` utility with the `switchOMS` flag. This will enable the local emagent to communicate with the WebLogic Administration Server using the virtual IP address. The `emctl` utility is located under the *`OVD_ORACLE_INSTANCE`*`/EMAGENT/EMAGENT/bin` directory.

   Syntax:

   ```
   ./emctl switchOMS ReposURL
   ```

   For Example:

   ```
   ./emctl switchOMS http://ADMINVHN:7001/em/upload
   ```

   Output:

   ```
   ./emctl switchOMS http://ADMINVHN.mycompany.com:7001/em/upload
   Oracle Enterprise Manager 10g Release 5 Grid Control 10.2.0.5.0.
   Copyright (c) 1996, 2009 Oracle Corporation.  All rights reserved.
   SwitchOMS succeeded.
   ```

5. Force the agent to reload its configuration by issuing the command:

   ```
   ./emctl reload
   ```

6. Check that the agent is using the correct Upload URL using the command:

   ```
   ./emctl status agent
   ```

7. Validate if the agents on the host where the new instance is running are configured properly to monitor their respective targets. Follow these steps to complete this task:

   a. Use a web browser to access Oracle Enterprise Manager Fusion Middleware Control at `http://ADMINVHN.mycompany.com:7001/em`. Log in as the `weblogic_idm` user.

   b. From the Domain Home Page navigate to the Agent-Monitored Targets page using the menu under **Farm** -> **Agent-Monitored Targets**

c. Update the WebLogic monitoring user name and the WebLogic monitoring password.

- Enter `weblogic_idm` as the WebLogic monitoring user name and the password for the weblogic user as the WebLogic monitoring password.
- Click **OK** to save your changes.

15.4.2.3.2 Configuring Oracle Virtual Directory for SSL Before configuring Oracle Virtual Directory for SSL, set the *ORACLE_HOME*, *ORACLE_INSTANCE* and *JAVA_HOME* variables. For example, on the new *LDAPHOST*, set ORACLE_HOME to *OID_ORACLE_HOME*, set ORACLE_INSTANCE to *OVD_ORACLE_INSTANCE*, set JAVA_HOME to *JAVA_HOME*, and add *JAVA_HOME* to your *PATH* variable.

Start the SSL Configuration Tool by issuing the command `SSLServerConfig` command which is located in the directory *ORACLE_COMMON_HOME*/bin directory.

For example:

```
ORACLE_COMMON_HOME/bin/SSLServerConfig.sh -component ovd
```

When prompted, enter the following information:

- LDAP Hostname: Central LDAP host, for example: `POLICYSTORE.mycompany.com`

> **Note:** It is recommended that you use the Policy Store directory, not the Identity Store.

- LDAP port: LDAP port, for example: `3060` (*OID_LDAP_PORT*)
- Admin user DN: `cn=orcladmin`
- Password: *administrator_password*
- sslDomain for the CA: `IDMDomain`
- Password to protect your SSL wallet/keystore: *password_for_local_keystore*
- Enter confirmed password for your SSL wallet/keystore: *password_for_local_keystore*
- Password for the CA wallet: *certificate_password*. This is the master password you crated when you ran provisioning.
- Country Name 2 letter code: Two letter country code, such as `US`
- State or Province Name: State or province, for example: `California`
- Locality Name: Enter the name of your city, for example: `RedwoodCity`
- Organization Name: Company name, for example: `mycompany`
- Organizational Unit Name: Leave at the default
- Common Name: Name of this host, for example: `LDAPHOST3.mycompany.com`
- OVD Instance Name: for example, `ovd1`. If you need to determine what your OVD component name is, execute the command:

  ```
  OVD_ORACLE_INSTANCE/bin/opmnctl status
  ```

- Oracle instance name: Name of your newly created Oracle instance, for example: `ovd3`

- WebLogic admin host: Host running the WebLogic Administration Server, for example:. `ADMINVHN.mycompany.com`
- WebLogic admin port: WebLogic Administration Server port, for example: `7001` (*`WLS_ADMIN_PORT`*)
- WebLogic admin user: Name of your WebLogic administration user, for example: `weblogic`
- WebLogic password: *`password`*.
- SSL wallet name for OVD component [ovdks1.jks]: Accept the default

When asked if you want to restart your Oracle Virtual Directory component, enter `Yes`.

When asked if you would like to test your OVD SSL connection, enter `Yes`. Ensure that the test is a success.

15.4.2.4 Creating ODSM Connections to Oracle Virtual Directory

Before you can manage Oracle Virtual Directory you must create connections from ODSM to each of your Oracle Virtual Directory instances. To do this, proceed as follows:

1. Access ODSM through the load balancer at: `http://ADMIN.mycompany.com/odsm`
2. Follow these steps to create connections to Oracle Virtual Directory:

   To create connections to Oracle Virtual Directory, follow these steps. Create connections to each Oracle Virtual Directory node separately. Using the Oracle Virtual Directory load balancer virtual host from ODSM is not supported:

   a. Create a direct connection to Oracle Virtual Directory on the new host providing the following information in ODSM:

      `Host`: *`LDAPHOST`*`.mycompany.com`

      `Port`: `8899` (The Oracle Virtual Directory proxy port, *`OVD_ADMIN_PORT`* in Section 3.7, "Fixed Ports Used by the Provisioning Wizard")

      Enable the SSL option.

      `User`: `cn=orcladmin`

      `Password`: password_to_connect_to_OVD

   b. Create a direct connection to Oracle Virtual Directory on the host where your new instance is running, providing the following information in ODSM:

      `Host`: *`LDAPHOST`*`.mycompany.com`

      `Port`: `8899` (The Oracle Virtual Directory proxy port)

      Enable the SSL option.

      `User`: `cn=orcladmin`

      `Password`: password_to_connect_to_OVD

15.4.2.5 Creating Adapters in Oracle Virtual Directory

Oracle Virtual Directory communicates with other directories through adapters.

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 15.4.2.5.1, "Creating Oracle Virtual Directory Adapters for Oracle Internet Directory and Active Directory"
- Section 15.4.2.5.2, "Validating the Oracle Virtual Directory Adapters"

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

You can use `idmConfigTool` 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 the environment variable `ORACLE_HOME` to *`IAM_ORACLE_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:LDAPHOST.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:LDAPHOST.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_ADMIN_PORT`*) in Section 3.7, "Fixed Ports Used by the Provisioning Wizard."
   - `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 (*OID_LDAP_PORT* or *OID_LDAP_SSL_PORT* in Section 6.1, "Assembling Information for Identity Management Provisioning").
- 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 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 the newly created Oracle Virtual Directory instance in your topology, with the appropriate value for ovd.host in the property file.

15.4.2.5.2 Validating the Oracle Virtual Directory Adapters Perform the following tasks by using ODSM:

1. Access ODSM through the load balancer at: http://ADMIN.mycompany.com/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 LDAPHOST.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 LDAPHOST -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 LDAPHOST3 -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.

15.4.2.6 Reconfiguring the Load Balancer

If you are accessing your Oracle Virtual Directory instances through a load balancer, add the new Oracle Virtual Directory instance to the existing server pool defined on the load balancer for distributing requests across the Oracle Virtual Directory instances.

15.5 Scaling the Application Tier

The Application Tier has two nodes (IDMHOST1 and IDMHOST2) running Managed Servers for Oracle Access Manager, Oracle Identity Federation, Oracle Directory Services Manager, and Oracle Identity Manager.

This section contains the following topics:

15.5.1 Mounting Middleware Home and Creating a New Machine when Scaling Out

When scaling out a component of the Application Tier, perform these steps first:

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. See Section 5.7, "Mounting Shared Storage Onto the Host" for more information.
2. To attach *IAM_HOME* in shared storage to the local Oracle Inventory, execute the following command:

   ```
   cd IAM_ORACLE_HOME/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*`/oui/bin` to it.
4. Log in to the WebLogic Administration Console at:
   `http://ADMIN.mycompany.com/console`
5. Create a new machine for the new node to be used, and add the machine to the domain, as follows.
   a. Select **Environment -> Machines** from the Navigation menu.
   b. Click **Lock and Edit**.
   c. Click **New** on the Machine Summary screen.
   d. Enter the following information:

      Name: Name of the machine. This is usually the host name.

      Machine OS: Select UNIX.

   e. Click **Next**.
   f. On the Node Manager Properties page, enter the following information:

Type: SSL.

Listen Address: Use the host name.

g. Click **Finish**.

h. Click **Activate Changes**.

15.5.2 Creating a New Node Manager when Scaling Out

Node Manager is used to start and stop WebLogic managed servers on the new host. In order to create a new node manager for the new host perform the following steps:

1. Create a new directory for the new node manager by copying an existing one. Copy the directory *SHARED_CONFIG_DIR*/nodemanger/idmhost1.mycompany.com to: *SHARED_CONFIG_DIR*/nodemanger/*newidmhost*.mycompany.com

   For example:

   ```
   cp -r $SHARED_CONFIG_DIR/nodemanger/idmhost1.mycompany.com $SHARED_CONFIG_
   DIR/nodemanger/newidmhost.mycompany.com
   ```

2. Change to the newly created directory.

   ```
   cd SHARED_CONFIG_DIR/nodemanger/newidmhost.mycompany.com
   ```

3. Edit the nodemanager.properties file, changing all the entries for IDMHOST1 to *newidmhost*. For example:

   ```
   DomainsFile=/u01/oracle//config/nodemanager/IDMHOST1.mycompany.com/nodemanager.
   domain
   ```

   becomes

   ```
   DomainsFile=/u01/oracle//config/nodemanager/newidmhost.mycompany.com/nodemanage
   r.domain
   ```

4. Edit the startNodeManagerWrapper.sh file, changing all the entries for IDMHOST1 to IDMHOST3. For example:

   ```
   NM_HOME=/u01/oracle/config/nodemanager/idmhost1.mycompany.com
   ```

   becomes

   ```
   NM_HOME=/u01/oracle/config/nodemanager/idmhost3.mycompany.com
   ```

5. Start the node manager by invoking the command:

   ```
   ./startNodeManagerWrapper.sh
   ```

15.5.3 Scaling ODSM

To scale up ODSM, use the existing installations (WebLogic Server home, Oracle Fusion Middleware home, and domain directories) for creating a new Managed Server for the Oracle Directory Services Manager component.

To scale out, 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 move files to *MSERVER* on the new node. (This is described in Section 15.5.5.7, "Completing the Oracle Identity Manager Configuration Steps.")

To scale ODSM instances, follow these steps:

1. Assemble the following information for scaling ODSM.

Description	Variable	Documented Value	Customer Value
Host name		IDMHOST3.mycompany.com	
ODSM Port	*ODSM_PORT*	7005	
Oracle Instance Location/Name	*ODS_ORACLE_INSTANCE*	*LOCAL_CONFIG_DIR*/instances/odsm	
Oracle Middleware Home Location	*IAM_MW_HOME*	/u01/oracle/products/app	
Oracle Home Directory		idm	
WebLogic Admin host		ADMINVHN.mycompany.com	
WebLogic Admin Port	*WLS_ADMIN_PORT*	7001	
WebLogic User Name		weblogic_idm	
WebLogic Password	*COMMON_IDM_PASSWORD*		
WebLogic Server Directory		*IAM_MW_HOME* /wlserver_10.3	

2. 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* manual in the Oracle Fusion Middleware documentation library for the platform and version you are using.
3. If you plan on provisioning the Instance Home or the Managed Server domain directory on shared storage, ensure that the appropriate shared storage volumes are mounted on the new host as described in Section 15.5.1, "Mounting Middleware Home and Creating a New Machine when Scaling Out."
4. If you are scaling out, you can use the default port (*ODSM_PORT* in Section 6.1, "Assembling Information for Identity Management Provisioning"). If you are scaling up, you must choose a unique port for this instance. Ensure that port number you are using is not in use by any service on the computer by issuing this command 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 "7005"
   ```

   If the port is in use (if the command returns output identifying the port), you must free it.

   On Linux:

   Remove the entries for port 7005 (*ODSM_PORT*) in the /etc/services file if the port is in use by a service and restart the services, as described in Section 16.1, "Starting and Stopping Components," or restart the computer.
5. Create a file containing the ports used by ODSM. On Disk1 of the installation media, locate the file stage/Response/staticports.ini. Copy it to a file called odsm_ports.ini. Delete all entries in odsm_ports.ini except for ODSM Server Port No. Change the values of ODSM Server Port No. to the value you want to use. If you are scaling out, you can use the default port, 7005 (*ODSM_PORT* in Section 6.1, "Assembling Information for Identity Management Provisioning"). If you are scaling up, choose a unique port for this instance.

Note: If the port name in the file is slightly different from those listed in this step, use the name in the file.

6. Start the Oracle Identity Management 11g Configuration Wizard by running the `config.sh` script located under the *IDM_ORACLE_HOME*`/bin` directory on the new host. For example: *IDM_ORACLE_HOME*`/bin`
7. On the Welcome screen, click **Next**.
8. On the Select Domain screen, select the **Expand Cluster** option and specify these values:
   - **Hostname:** `ADMINVHN.mycompany.com`
   - **Port**: `7001` (*WLS_ADMIN_PORT*)
   - **UserName**: `weblogic_idm`
   - **User Password**: *password for the webLogic user*

   Click **Next**.
9. A dialog box with the following message appears:

   ```
   The selected domain is not a valid Identity Management domain or the installer
   cannot determine if it is a valid domain. If you created the domain using the
   Identity Management installer, you can ignore this message and continue. If you
   did not create the domain using the Identity Management installer, refer to the
   Identity Management documentation for information on how to verify the domain
   is valid.
   ```

   Click **YES** to continue.

   This is a benign warning that you can safely ignore.
10. If you are scaling out, on the Specify Installation Location screen, specify the following values. The values for the **Oracle Middleware Home Location** and the **Oracle Home Directory** fields are prefilled. The values default to the Middleware home and Oracle home previously installed on IDMHOST1. Choose a new instance using a sequential number.
    - **Oracle Middleware Home Location**: *IAM_MW_HOME*
    - **Oracle Home Directory**: idm
    - **WebLogic Server Directory**: *IAM_MW_HOME* `/wlserver_10.3`
    - **Oracle Instance Location**: *ODS_ORACLE_INSTANCE*
    - **Oracle Instance Name**: *ODS_ORACLE_INSTANCE*

    Click **Next**.
11. On the Email for Security Updates screen, specify these values:
    - **Email Address**: Provide the email address for your My Oracle Support account.
    - **Oracle Support Password**: Provide the password for your My Oracle Support account.
    - Check the check box next to the **I wish to receive security updates via My Oracle Support** field.

Click **Next**.

12. On the Configure Components screen, de-select all the products except ODSM and then click **Next**.
13. On the Configure Ports screen, you use the `odsm_ports.ini` file you created in Step 4 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 `odsm_ports.ini`.
    c. Click **Save**, then click **Next**.
14. 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 **Configure**.
15. On the Configuration Progress screen, multiple configuration assistants are launched in succession; this process can be lengthy. Wait until it completes.
16. On the Installation Complete screen, click **Finish** to confirm your choice to exit.
17. Add the newly added Managed Server host name and port to the list WebLogicCluster parameter, as described in Section 15.5.8, "Adding New WebLogic Managed Server to Oracle HTTP Server Configuration Files."

15.5.4 Scaling Oracle Access Manager 11g

To scale up, use the existing installations (WebLogic Server home, Oracle Fusion Middleware home, and domain directories) for creating a new Managed Server for the Oracle Access Manager component.

Use the existing binaries 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.

Scale Oracle Access Manager by performing the steps in the following subsections:

- Section 15.5.4.1, "Assembling Information for Scaling Oracle Access Manager"
- Section 15.5.4.2, "Prepare New Node for Scaling Out"
- Section 15.5.4.3, "Configure New Oracle Access Manager Server"
- Section 15.5.4.4, "Run Pack/Unpack"
- Section 15.5.4.5, "Register Managed Server with Oracle Access Manager"
- Section 15.5.4.6, "Update WebGate Profiles"
- Section 15.5.4.7, "Update the Web Tier"

15.5.4.1 Assembling Information for Scaling Oracle Access Manager

Assemble the following information before scaling Oracle Access Manager.

Description	Variable	Documented Value	Customer Value
Host Name	IDMHOST*n*		
Existing OAM server		WLS_OAM1	
New OAM server name	WLS_OAM*n*	WLS_OAM3	
Server Listen Address			
Server Listen Port	*OAM_PORT*	14100	
WebLogic Admin Host		ADMINVHN.mycompany.com	
WebLogic Admin Port	*WLS_ADMIN_PORT*	7001	
WebLogic Admin User		weblogic_idm	
WebLogic Admin Password			

15.5.4.2 Prepare New Node for Scaling Out

The following steps are necessary only if you are scaling out.

1. Ensure that shared storage is mounted on the new node, as described in Section 15.5.1, "Mounting Middleware Home and Creating a New Machine when Scaling Out."
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* to it.

15.5.4.3 Configure New Oracle Access Manager Server

1. Log in to the Oracle WebLogic Administration Console at: http://ADMIN.mycompany.com/console
2. 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.
3. Click **Lock & Edit** from the Change Center menu.
4. Select an existing server on the host you want to extend, for example: WLS_OAM1.
5. Click **Clone**.
6. 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.

     If you are scaling out, you can use the default port, 14100 (*OAM_PORT* in Table 6–1). If you are scaling up, choose a unique port.
7. Click **OK**.
8. Click the newly created server **WLS_OAM3**
9. Set **Machine** to be the machine you created in Section 15.5.1, "Mounting Middleware Home and Creating a New Machine when Scaling Out."
10. Click **Save**.

11. 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 `IDMHOSTn`.

    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. In Oracle Enterprise Manager Fusion Middleware Control, select **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**.

12. Click **Activate Changes** from the Change Center menu.

15.5.4.4 Run Pack/Unpack

Run `pack` and `unpack` as described in Section 15.5.7, "Running Pack/Unpack."

15.5.4.5 Register Managed Server with Oracle Access Manager

Register the new Managed Server with Oracle Access Manager. You now must configure the new Managed Server now as an Oracle Access Manager server. You do this from the Oracle OAM console. Proceed as follows:

1. Log in to the OAM console at `http://ADMIN.mycompany.com/oamconsole` as the user identified by the entry in Section 8.2, "Update User Names in Provisioning Response File."
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 same mode as existing Oracle Access Manager servers.
6. Click **Coherence** tab.

   Set **Local Port** to a unique value on the host.

7. Click **Apply**.
8. Restart the WebLogic Administration Server as described in Section 16.1, "Starting and Stopping Components."

15.5.4.6 Update WebGate Profiles

Add the newly created Oracle Access Manager server to all WebGate Profiles that might be using it, such as `Webgate_IDM`, `Webgate_IDM_11g`, and `IAMSuiteAgent`

For example, to add the Oracle Access Manager server to `Webgate_IDM`, access the OAM console at: `http://ADMIN.mycompany.com/oamconsole`

Then proceed as follows:

1. Log in as the Oracle Access Manager Admin User.
2. Click the **System Configuration** tab.
3. Expand **Access Manager Settings** - **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 **Webgate_IDM_11g**, **IAMSuiteAgent**, and all other WebGates that might be in use.

You can now start the new Managed Server, as described in Section 16.1, "Starting and Stopping Components."

15.5.4.7 Update the Web Tier

Add the newly added Managed Server host name and port to the list WebLogicCluster parameter, as described in Section 15.5.8, "Adding New WebLogic Managed Server to Oracle HTTP Server Configuration Files."

Save the file and restart the Oracle HTTP server, as described in Section 16.1, "Starting and Stopping Components."

15.5.5 Scaling Oracle Identity Manager

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. Use the existing installations in shared storage for creating a new WLS_SOA and WLS_OIM managed server. There is no need to install the Oracle Identity and Access Management or Oracle SOA Suite binaries in a new location

When scaling up, you add WLS_SOA and WLS_OIM managed servers to existing nodes.

In either case, you must run pack and unpack.

When you scale out the topology, you add new Managed Servers configured with OIM and SOA to new nodes. First check that the new node can access the existing home directories for WebLogic Server, OIM, and SOA. You do need to run pack and unpack to bootstrap the domain configuration in the new node.

Follow the steps in the following subsections to scale the topology:

- Section 15.5.5.1, "Assembling Information for Scaling Oracle Identity Manager"
- Section 15.5.5.2, "Cloning an Existing Oracle Identity Manager Server when Scaling Up Oracle Identity Manager or SOA"
- Section 15.5.5.3, "Mounting Middleware Home and Creating a New Machine when Scaling Out"
- Section 15.5.5.4, "Configuring New JMS Servers"
- Section 15.5.5.5, "Performing Pack/Unpack When Scaling Out"
- Section 15.5.5.6, "Configuring Oracle Coherence for Deploying Composites"
- Section 15.5.5.7, "Completing the Oracle Identity Manager Configuration Steps"

15.5.5.1 Assembling Information for Scaling Oracle Identity Manager

Assemble the following information before scaling Oracle Identity Manager.

Description	Variable	Documented Value	Customer Value
Host name	`IDMHOSTn`	`IDMHOST1`	
SOA virtual server name		SOAHOST*x*VHN	
OIM virtual server name		OIMHOST*x*VHN	
SOA managed server to clone	`WLS_SOAn`	`WLS_SOA1`	
OIM managed server to clone	`WLS_OIMn`	`WLS_OIM1`	
SOA managed server name	`WLS_SOAn`	`WLS_SOA3`	
OIM managed server name	`WLS_OIMn`	`WLS_OIM3`	
Numeric extension for new JMS servers	*n*	3	
WebLogic Admin Host		ADMINVHN.mycompany.com	
WebLogic Admin Port	*WLS_ADMIN_PORT*	7001	
WebLogic Admin User		weblogic_idm	
WebLogic Admin Password			

15.5.5.2 Cloning an Existing Oracle Identity Manager Server when Scaling Up Oracle Identity Manager or SOA

Follow this procedure twice, once to clone **WLS_SOA1** and once again to clone **WLS_OIM1**.

1. Log in to the Administration Console at: `http://ADMIN.mycompany.com/console`
2. Clone 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.

3. 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.

15.5.5.3 Mounting Middleware Home and Creating a New Machine when Scaling Out

Mount the Middleware home, as described in Section 15.5.1, "Mounting Middleware Home and Creating a New Machine when Scaling Out."

15.5.5.4 Configuring New JMS Servers

Create JMS Servers for SOA, Oracle Identity Manager, UMS, and BPM on the new Managed Server. You do this as follows:

1. Log in to the WebLogic Administration Server and navigate to **Services -> Messaging -> JMS Servers**.
2. Click **New**.
3. Enter a value for **Name**, such as `BPMJMSServer_auto_3`.
4. Click **Create New Store**.
5. Select `FileStore` from the list
6. Click **Next**.
7. Enter a value for **Name**, such as `BPMJMSFileStore_auto_3`
8. Enter the following values:

   Target: The new server you are creating.

   Directory: *`ASERVER_HOME`*`/jms/BPMJMSFileStore_auto_3`
9. Click **OK**.
10. When you are returned to the JMS Server screen, select the newly created file store from the list.
11. Click **Next**.
12. On the next screen set the Target to the server you are creating.
13. 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*	PS6SOAJMSServer_auto_*n*	PS6SOAJMSFileStore_auto_*n*	*ASERVER_HOME*/jms/PS6SOAJMSFileStore_auto_*n*	wls_OIM*n*

Add the newly created JMS Queues to the existing JMS Modules by performing the following steps:

1. Log in to the WebLogic Administration Console
2. Navigate to **Services -> Messaging -> JMS Modules**
3. Click a JMSModule, such as **SOAJMSModule**
4. Click the **Sub Deployments** tab.
5. 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.

6. Assign the newly created JMS server, for example **SOAJMSServer_auto*n***.
7. Click **Save**.
8. Perform this for each of the JMS modules listed in the following table:

JMS Module	JMS Server
BPMJMSModule	BPMJMSServer_auto_*n*
JRFWSAsyncJmsModule	JRFWSAsyncJmServer_auto_*n*
OIMJMSModule	OIMJMSServer_auto_*n*
SOAJMSModule	SOAJMSServer_auto_*n*
UMSJMSSystemResource	UMSJMSServe_auto_*n*

9. Click **Activate Configuration** from the Change Center menu.

15.5.5.5 Performing Pack/Unpack When Scaling Out

This section is necessary only when you are scaling out.

Run `pack` and `unpack` as described in Section 15.5.7, "Running Pack/Unpack."

15.5.5.6 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.

15.5.5.6.1 Enabling Communication for Deployment Using Unicast Communication Specify the nodes using the `tangosol.coherence.wkan` 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, for example: SOAHOST3VHN. 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. You will also need to add the new server to the existing entries.

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: SOAHOST3VHN is the virtual host name that maps to the virtual IP where WLS_SOA3 listening (in SOAHOST3).

15.5.5.6.2 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, WLS_SOA2, and WLS_SOA3 into the Arguments field.

   For WLS_SOA1, enter the following:

   ```
   -Dtangosol.coherence.wka1=SOAHOST1VHN
   -Dtangosol.coherence.wka2=SOAHOST2VHN
   -Dtangosol.coherence.wka3=SOAHOST3VHN
   -Dtangosol.coherence.localhost=SOAHOST1VHN
   ```

   For WLS_SOA2, enter the following:

   ```
   -Dtangosol.coherence.wka1=SOAHOST1VHN
   -Dtangosol.coherence.wka2=SOAHOST2VHN
   -Dtangosol.coherence.wka3=SOAHOST3VHN
   -Dtangosol.coherence.localhost=SOAHOST2VHN
   ```

   For WLS_SOA3, enter the following:

   ```
   -Dtangosol.coherence.wka1=SOAHOST1VHN
   -Dtangosol.coherence.wka2=SOAHOST2VHN
   -Dtangosol.coherence.wka3=SOAHOST3VHN
   -Dtangosol.coherence.localhost=SOAHOST3VHN
   ```

   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.wka3=SOAHOST3VHN
-Dtangosol.coherence.localhost=SOAHOST1VHN
-Dtangosol.coherence.localport=8089
-Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
-Dtangosol.coherence.wka3.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.wka3=SOAHOST3VHN
-Dtangosol.coherence.localhost=SOAHOST2VHN
-Dtangosol.coherence.localport=8089
-Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
-Dtangosol.coherence.wka3.port=8089
```

WLS_SOA3 (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.wka3=SOAHOST3VHN
-Dtangosol.coherence.localhost=SOAHOST3VHN
-Dtangosol.coherence.localport=8089
-Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
-Dtangosol.coherence.wka3.port=8089
```

For more information about Coherence Clusters see the *Oracle Coherence Developer's Guide*.

8. Click **Save** and **Activate Changes**.

Note: 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.

Note: 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.

15.5.5.7 Completing the Oracle Identity Manager Configuration Steps

1. 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 WebLogic Administration Console, select the **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.

2. 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. In the Oracle Enterprise Manager Console, select **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**.

3. 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.

4. Click **Activate Changes** from the Change Center menu.

5. Restart the WebLogic Administration Server as described in Section 16.1, "Starting and Stopping Components."

6. 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.

7. Configure the newly created managed server for server migration. Follow the steps in Section 13.6, "Configuring Server Migration Targets" to configure server migration.

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.

8. 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.
   e. Repeat Steps a-d for WLS_OIM*n*.

15.5.6 Scaling Oracle Identity Federation

The Application Tier has two nodes (IDMHOST1 and IDMHOST2) running a Managed Server configured with Oracle Identity Federation.

The Oracle Identity Federation instances can be scaled out by adding a new node with a Managed Server to the existing cluster.

The existing installations (WebLogic Server home, Oracle Fusion Middleware home, and domain directories) can be used for creating a new Managed Server for Oracle Identity Federation.

The Oracle Identity Federation instances can be scaled out by adding a new node with a Managed Server to the existing cluster.

Perform the steps in the following sections to scale Oracle Identity Federation.

- Section 15.5.6.1, "Assembling Information for Scaling Oracle Identity Federation"
- Section 15.5.6.2, "Configuring Oracle Identity Federation"
- Section 15.5.6.3, "Performing Pack/Unpack when Scaling Out"
- Section 15.5.6.4, "Complete Oracle Identity Federation Server Configuration"
- Section 15.5.6.5, "Add New Managed Server to OHS Configuration"

15.5.6.1 Assembling Information for Scaling Oracle Identity Federation

Assemble the following information before scaling Oracle Identity Federation.

Description	Variable	Documented Value	Customer Value
Host name		IDMHOST3.mycompany.com	
OIF Port	*OIF_PORT*	7499	
Instance name	oif*n*	oif3	
WebLogic Admin Host		ADMINVHN.mycompany.com	
WebLogic Admin Port	*WLS_ADMIN_PORT*	7001	
WebLogic Admin User		weblogic_idm	
WebLogic Admin Password			

15.5.6.2 Configuring Oracle Identity Federation

1. 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.
2. Create a file containing the ports used by Oracle Internet Directory. On Disk1 of the installation media, locate the file `stage/Response/staticports.ini`. Copy it to a file called `oif_ports.ini`. Delete all entries in `oif_ports.ini` except for `Oracle Identity Federation Server Port`. Change the value of that port to the port you are using for this instance.

   If you are scaling out, you can use the default port, 7499 (*OIF_PORT*). If you are scaling up, you must choose a unique port for this instance.

   Note: If the port name in the file is slightly different from those listed in this step, use the name in the file.

3. Ensure that the appropriate shared storage volumes are mounted on the new IDMHOST, as described in Section 15.5.1, "Mounting Middleware Home and Creating a New Machine when Scaling Out."
4. Ensure that the port you want to use is not in use by any service on the computer by issuing these commands for the operating system you are using, specifying the port you want to use. If a port is not in use, no output is returned from the command.

   On UNIX:

   ```
   netstat -an | grep "7499"
   ```

   If the port is in use (if the command returns output identifying the port), you must free it.

   On UNIX:

   Remove the entries for port `7499` in the `/etc/services` file.
5. Start the Oracle Identity Management 11*g* Configuration Wizard located under the *IDM_ORACLE_HOME*`/bin` directory as follows:

   Issue this command:

   ```
   ./config.sh
   ```

6. On the Welcome screen, click **Next**.
7. On the Select Domain screen, select the **Expand Cluster** option and specify these values:
   - **HostName**: `ADMINVHN.mycompany.com`
   - **Port**: `7001`
   - **UserName**: `weblogic_idm`
   - **User Password**: *`weblogic_user_password`*

   Click **Next**.
8. A dialog box with the following message appears:

   ```
   The selected domain is not a valid Identity Management domain or the installer
   cannot determine if it is a valid domain. If you created the domain using the
   Identity Management installer, you can ignore this message and continue. If you
   did not create the domain using the Identity Management installer, refer to the
   Identity Management documentation for information on how to verify the domain
   is valid.
   ```

   This is a benign warning that you can ignore.

   Click **Yes** to continue.
9. On the Specify Installation Location screen, specify the following values:
   - **Oracle Middleware Home Location**: *`OIF_MW_HOME`* (This value is prefilled and cannot be updated.)
   - **Oracle Home Directory**: `idm` (This value is prefilled and cannot be updated.)
   - **WebLogic Server Directory**: *`OIF_MW_HOME`*`/wlserver_10.3`
   - **Oracle Instance Location**: *`OIF_ORACLE_INSTANCE`*
   - **Instance Name**: `oif`*`n`*, where *`n`* is a sequential number, for example `oif3`.

   Click **Next**.
10. On the Specify Security Updates screen (if shown), specify the values shown in this example:
    - **Email Address**: Provide the email address for your My Oracle Support account.
    - **Oracle Support Password**: Provide the password for your My Oracle Support account.
    - Select **I wish to receive security updates via My Oracle Support**.

    Click **Next**.
11. On the Configure Components screen, de-select all the components except Oracle Identity Federation components. Select only Oracle Identity Federation from the Oracle Identity Federation components. Do not select Oracle HTTP Server.

    Click **Next**.
12. On the Configure Ports screen, you use the `oif_ports.ini` file you created in Step 2 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 `oif_ports.ini`.

c. Click **Save**, then click **Next**.

13. On the Installation Summary screen, review the selections to ensure that they are correct. If they are not correct, click **Back** to modify selections on previous screens. Then click **Configure**.

14. On the Configuration Progress screen, view the progress of the configuration.

15. On the Installation Complete screen, click **Finish** to confirm your choice to exit.

15.5.6.3 Performing Pack/Unpack when Scaling Out

This section is necessary only when you are scaling out.

From IDMHOST1, copy the applications directory under the *ASERVER_HOME*`/config/fmwconfig/servers/wls_oif1` directory to the *ASERVER_HOME*`/config/fmwconfig/servers/wls_oif`*n* directory, where `wls_oif`*n* is the new server being added, for example:

```
cp -rp ASERVER_HOME/config/fmwconfig/servers/wls_oif1/applications
user@IDMHOST1:ASERVER_HOME/config/fmwconfig/servers/wls_oif3
```

Then run `pack` and `unpack` as described in Section 15.5.7, "Running Pack/Unpack."

15.5.6.4 Complete Oracle Identity Federation Server Configuration

Perform the steps in Section 11.3, "Validating Oracle Identity Federation" and Section 11.4, "Configuring the Enterprise Manager Agents" to completed the configuration of your new server.

15.5.6.5 Add New Managed Server to OHS Configuration

Add the newly added Managed Server host name and port to the list WebLogicCluster parameter, as described in Section 15.5.8, "Adding New WebLogic Managed Server to Oracle HTTP Server Configuration Files."

15.5.7 Running Pack/Unpack

Whenever you extend a domain to include a new managed server, you must extract the domain configuration needs from the *ASERVER_HOME* location to the *MSERVER_HOME* location. This applies whether you are scaling up or out. To do this perform the following steps.

1. Pack the domain on IDMHOST1 to create a template pack using the command:

   ```
   pack.sh -domain=ASERVER_HOME -template =/templates/managedServer.jar -template_
   name="template_name" -managed=true
   ```

   The `pack.sh` script is located in *ORACLE_COMMON_HOME*`/common/bin`.

2. Unpack the domain on the new host for scale out, or on the existing host for scale up, using the command:

   ```
   unpack.sh -domain=MSERVER_HOME -template=/templates/managedServer.jar -app_
   dir=MSERVER_HOME/applications
   ```

   The `unpack.sh` script is located in *ORACLE_COMMON_HOME*`/common/bin`.

3. If you are scaling out, start Node Manager and update the property file.

a. Start and stop Node Manager as described in Section 16.1, "Starting and Stopping 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 16.1, "Starting and Stopping Components."

15.5.8 Adding New WebLogic Managed Server to Oracle HTTP Server Configuration Files

Scaling an Application Tier component typically requires you 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.

In the Web tier, there are several configuration files under *WEB_ORACLE_INSTANCE*`/config/OHS/componentname/moduleconf`, including `admin_vh.conf`, `sso_vh.conf` and `idminternal_vh.conf`. Each contain a number of entries in location blocks. If a block references two server instances and you add a third one, you must update that block with the new server.

For example if you add a new Oracle Access Manager server, you must update `sso_vh.conf` to include the new managed server. You 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>

<Location /fusion_apps>
   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>

<Location /fusion_apps>
   SetHandler weblogic-handler
   WebLogicCluster
IDMHOST1.mycompany.com:14100,IDMHOST2.mycompany.com:14100,IDMHOST3.mycompany.com:1
4100
</Location>
```

Similarly, if you add a new ODSM server, you must update ODSM entries in the file `admin_vh.conf`.

Once you have updated the configuration file, restart the Oracle HTTP server(s) as described in Section 16.1, "Starting and Stopping Components." Oracle recommends that you do this sequentially to prevent loss of service.

15.6 Scaling the Web Tier

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 the Oracle HTTP Server, perform the steps in the following subsections:

- Section 15.6.1, "Assembling Information for Scaling the Web Tier"
- Section 15.6.2, "Mounting Middleware Home and Copying Oracle HTTP Server Files when Scaling Out"
- Section 15.6.3, "Running the Configuration Wizard to Configure the HTTP Server"
- Section 15.6.4, "Registering Oracle HTTP Server with WebLogic Server"
- Section 15.6.5, "Reconfiguring the Load Balancer"

15.6.1 Assembling Information for Scaling the Web Tier

Assemble the following information before scaling the Web Tier.

Description	Variable	Documented Value	Customer Value
Host name		WEBHOST1.mycompany.com	
OHS port	*OHS_PORT*	7777	
Instance Name	webn	web1 or web2	
Component Name	webn	web1 or web2	
WebLogic Admin Host		ADMINVHN.mycompany.com	
WebLogic Admin Port	*WLS_ADMIN_PORT*	7001	
WebLogic Admin User		weblogic_idm	
WebLogic Admin Password			

15.6.2 Mounting Middleware Home and Copying Oracle HTTP Server Files when Scaling Out

On the new node, mount the existing Middleware home.

Copy all files created in *ORACLE_INSTANCE*/config/OHS/*component*/moduleconf from the existing Web Tier configuration to the new one.

15.6.3 Running the Configuration Wizard to Configure the HTTP Server

Perform these steps to configure the Oracle Web Tier:

1. 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 value of OPMN Local Port to 6700. If you are scaling out, you can use the default value, 7777, for OHS PORT. If you are scaling up, you must choose a unique value for that instance on the machine.

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

2. Change the directory to the location of the Oracle Fusion Middleware Configuration Wizard:

   `cd WEB_ORACLE_HOME/bin`

3. 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 3.7, "Fixed Ports Used by the Provisioning Wizard."
   - **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 the number of the new host, for example, `3`:
   - **Instance Home Location**: *WEB_ORACLE_INSTANCE* (`/u02/local/oracle/config/instances/ohs`*n*, for example, `/u02/local/oracle/config/instances/ohs1`)
   - **Instance Name**: `web`*n*
   - **OHS Component Name**: `web`*n*

   Click **Next**.
5. On the Configure Ports screen, you use the `ohs_ports.ini` file you created in Step 1to 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 **Save**, then click **Next**.
6. On the Specify Security Updates screen, specify these values:
   - **Email Address**: The email address for your My Oracle Support account.

- **Oracle Support Password**: The password for your My Oracle Support account.

Select: **I wish to receive security updates via My Oracle Support**.

Click **Next**.

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.

15.6.4 Registering Oracle HTTP Server with WebLogic Server

For Oracle Enterprise Manager Fusion Middleware Control to be able to manage and monitor the new Oracle HTTP server, you must register the Oracle HTTP server with IDMDomain. To do this, register Oracle HTTP Server with WebLogic Server by running the following command on the host where the new server is running:

```
cd WEB_ORACLE_INSTANCE/bin
./opmnctl registerinstance -adminHost ADMINVHN.mycompany.com \
   -adminPort 7001 -adminUsername weblogic
```

15.6.5 Reconfiguring the Load Balancer

Add the new Oracle HTTP Server instance to the existing server pool defined on the load balancer for distributing requests across the HTTP instances.

15.7 Post-Scaling Steps for All Components

Provisioning creates a set of scripts to start and stop managed servers defined in the domain. When you create a new managed server in the domain you need to update the domain configuration so that these start and stop scripts can also start the newly created managed server.

To update the domain configuration, edit the file `serverInstancesCustom.txt`, which is located in the directory: *SHARED_CONFIG_DIR*`/scripts`

If you want to start a node manager on a new machine, add an entry which looks like this:

newmachine`.mycompany.com NM` *nodemanager_pathname nodemanager_port*

For example:

```
IDMHOST3.mycompany.com NM /u01/oracle/config/nodemanager/idmhost3.mycompany.com
5556
```

If you want to start a managed server called WLS_OIM3 add an entry which looks like this:

newmachine`.mycompany.com OIM` *ManagedServerName*

For example:

```
IDMHOST3 OIM WLS_OIM3
```

Save the file.

16

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:

16.1 Starting and Stopping 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:

16.1.1 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. Oracle Internet Directory

4. Oracle Virtual Directory
5. Node Manager
6. Oracle Access Manager Server(s)
7. WebLogic Administration Server
8. Oracle HTTP Server(s)
9. SOA Server(s)
10. Oracle Identity Manager Server(s)

16.1.2 Starting and Stopping Servers

During provisioning, scripts were created in the *SHARED_ROOT*/config/scripts directory to start and stop all the servers in the environment. Two of the scripts are available for you to use from the command line to start and stop all Identity Management servers. The remaining scripts are used internally and must not be invoked from the command line.

Note: These scripts do NOT stop or start the database.

16.1.2.1 Starting All Servers

Provisioning created a file called startall.sh, which is used to start all of the components on a particular server. To start everything in the correct order run the command on hosts in the following order:

- LDAPHOST1
- LDAPHOST2
- IDMHOST1
- IDMHOST2
- WEBHOST1
- WEBHOST2

If you want to start the services on a single host, execute the command on that host.

Before invoking this script, set JAVA_HOME to *JAVA_HOME*.

During exectution you will be prompted to enter the Weblogic and Node Manager administrator passwords.

The script starts the servers in the following order:

1. Node Manager1
2. AdminServer
3. wls_ods1
4. wls_soa1
5. wls_oim1
6. wls_oam1
7. wls_oif1
8. ohs1

9. oid1
10. oid2
11. ohs2
12. Node Manager 2
13. wls_ods2
14. wls_soa2
15. wls_oim2
16. wls_oam2
17. wls_oif2

16.1.2.2 Stopping All Servers:

The script to stop all servers is `stopall.sh`.

Before invoking this script, set `JAVA_HOME` to *`JAVA_HOME`*.

During exectution you will be prompted to enter the Weblogic and Node Manager administrator passwords.

16.2 About Identity Management Console URLs

Table 16–1 lists the administration consoles used in this guide and their URLs.

Table 16–1 Console URLs

Domain	Console	URL
IDMDomain	WebLogic Administration Console	`http://ADMIN.mycompany.com/console`
IDMDomain	Enterprise Manager FMW Control	`http://ADMIN.mycompany.com/em`
IDMDomain	OAM Console	`http://ADMIN.mycompany.com/oamconsole`
IDMDomain	ODSM	`http://ADMIN.mycompany.com/odsm`

16.3 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 16.3.1, "Monitoring Oracle Internet Directory"
- Section 16.3.2, "Monitoring Oracle Virtual Directory"
- Section 16.3.3, "Monitoring WebLogic Managed Servers"

16.3.1 Monitoring Oracle Internet Directory

You can use the Oracle Enterprise Manager Fusion Middleware Control to monitor Oracle Internet Directory, as follows:

1. On the Farm home page for an Identity Management domain, view the Fusion Middleware pie chart. This chart shows the status of Oracle Fusion Middleware

components. Green sections of the chart indicate components that are up and running properly, and red sections indicate components that are down.

2. The Identity and Access section below the chart includes the name of each individual Oracle Internet Directory instance (for example: oid1, oid2), its status, host name, and CPU usage percentage. A green arrow in the Status column indicates that the instance is up and running properly, and a red arrow indicates the instance is down. Click the name of an individual Oracle Internet Directory instance to view the home page for that instance.
3. The home page for an instance displays metrics for the instance such as performance, load, security, response, CPU utilization %, and memory utilization %.

16.3.1.1 Oracle Internet Directory Component Names Assigned by Oracle Identity Manager Installer

When you perform an Oracle Internet Directory installation using Oracle Identity Management 11*g* Installer, the default component name that the installer assigns to the Oracle Internet Directory instance is oid1. You cannot change this component name.

The instance specific configuration entry for this Oracle Internet Directory instance is `cn=oid1, cn=osdldapd, cn=subconfigsubentry`.

If you perform a second Oracle Internet Directory installation on another computer and that Oracle Internet Directory instance uses the same database as the first instance, the installer detects the previously installed Oracle Internet Directory instance on the other computer using the same Oracle database, so it gives the second Oracle Internet Directory instance a component name of `oid2`.

The instance specific configuration entry for the second Oracle Internet Directory instance is `cn=oid2, cn=osdldapd, cn=subconfigsubentry`. A change of properties in the entry `cn=oid2, cn=osdldapd, cn=subconfigsubentry` does not affect the first instance (`oid1`).

If a third Oracle Internet Directory installation is performed on another computer and that instance uses the same database as the first two instances, the installer gives the third Oracle Internet Directory instance a component name of oid3, and so on for additional instances on different hosts that use the same database.

Note that the shared configuration for all Oracle Internet Directory instances is `cn=dsaconfig, cn=configsets,cn=oracle internet directory`. A change in this entry affects all the instances of Oracle Internet Directory.

This naming scheme helps alleviate confusion when you view your domain using Oracle Enterprise Manager by giving different component names to your Oracle Internet Directory instances.

16.3.2 Monitoring Oracle Virtual Directory

You can use the Oracle Enterprise Manager Fusion Middleware Control to monitor Oracle Virtual Directory, as follows:

1. On the Farm home page for an Identity Management domain, view the Fusion Middleware pie chart. This chart shows the status of Oracle Fusion Middleware components. Green sections of the chart indicate components that are up and running properly, and red sections indicate components that are down.
2. The Identity and Access section below the chart includes the name of each instance of the Oracle Virtual Directory application (for example, ovd1, ovd2), its status, and host name. A green arrow in the Status column indicates that the Oracle

Virtual Directory instance is up and running properly, and a red arrow indicates the instance is down. Click the name of an individual Oracle Virtual Directory instance to view the home page for that instance.

3. The home page for an instance displays metrics and configurations for the instance such as:

   - Oracle Virtual Directory status - A green arrow next to the Oracle Virtual Directory instance name at the top of the page indicates that the instance is up and running properly and a red arrow indicates that the instance is down.
   - Current Load - This indicates the current work load of this Oracle Virtual Directory instance. It includes three metrics: Open Connections, Distinct Connected Users, and Distinct Connected IP Addresses.
   - Average Response Time Metric - This displays the average time (in milliseconds) to complete an LDAP search request.
   - Operations Metric - This displays the average number of LDAP search requests finished per millisecond.
   - Listeners - This table lists the listeners configured for this Oracle Virtual Directory instance to provide services to clients.
   - Adapters - This table lists existing adapters configured with the Oracle Virtual Directory instance. Oracle Virtual Directory uses adapters to connect to different underlying data repositories.
   - Resource Usage - On the right hand side of the page, the CPU and memory utilization metrics are displayed to indicate the system resources consumed by the Oracle Virtual Directory instance.

16.3.3 Monitoring WebLogic Managed Servers

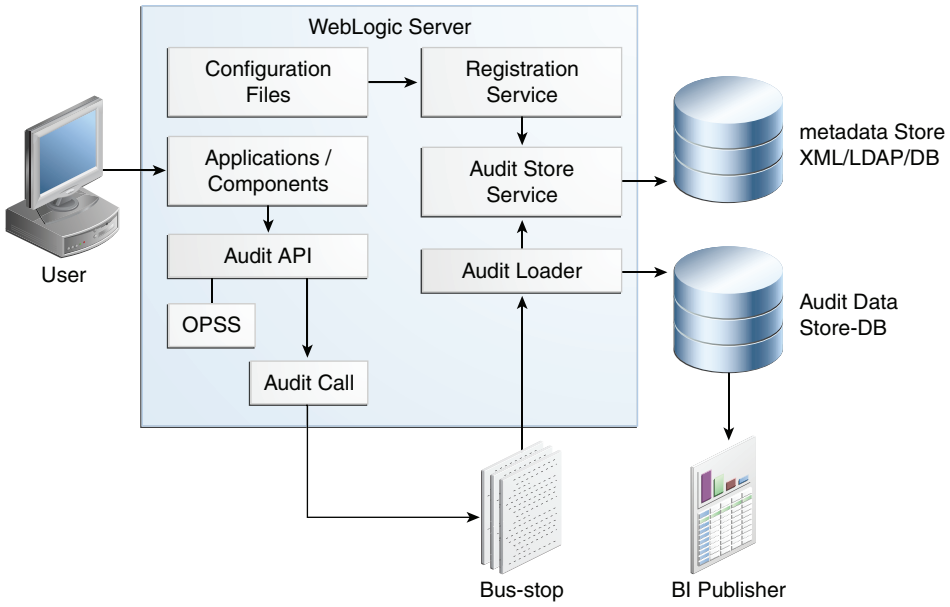
You can use Oracle Enterprise Manager Fusion Middleware Control to monitor Managed Servers and other Fusion Middleware components, such as Oracle Access Manager, Oracle Identity Manager, Oracle Identity Federation, and SOA. For more information, see the administrator guides listed in the Preface under "Related Documents" on page xv.

16.4 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 16–1 is a high-level architectural diagram of the Oracle Fusion Middleware Audit Framework. For more information, see *Oracle Fusion Middleware Application Security Guide*.

Figure 16–1 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.

16.5 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 16.5.1, "Peforming Baseline Backups"

- Section 16.5.2, "Performing Runtime Backups"
- Section 16.5.3, "Performing Backups During Installation and Configuration"

16.5.1 Peforming 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 16–2 Static Artifacts to Back Up in the Identity Management Enterprise Deployment

Type	Host	Location	Tier
Oracle Home (database)	Oracle RAC database hosts: OIDDBHOST1 OIDDBHOST2	User Defined	Directory Tier
Oracle Identity and Access Management Binaries	IDMHOST1 IDMHOST2	Middleware Home: *IAM_MW_HOME*	Application Tier
Oracle Identity Management Binaries	IDMHOST1 IDMHOST2	Middleware Home: *IDM_MW_HOME*	Application Tier
Web Tier Binaries	WEBHOST1 WEBHOST2	Middleware Oracle home, *WEB_ORACLE_HOME*:	Web Tier
Install-Related Files	Each host	OraInventory: *ORACLE_BASE*/orainventory /etc/oratab, /etc/oraInst.loc *user_home*/bea/beahomelist (on hosts where WebLogic Server is installed)	Not applicable.

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*.

16.5.2 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 16–3 Run-Time Artifacts to Back Up in the Identity Management Enterprise Deployments

Type	Host	Location	Tier
Domain Home	IDMHOST1 IDMHOST2	Admin Server and Shared Files: *ASERVER_HOME* Managed Servers: *MSERVER_HOME*	Application Tier
Application Artifacts (ear and war files)	IDMHOST1 IDMHOST2	Look at all the deployments, including Oracle Directory Services Manager, through the WebLogic Server Administration Console to identity all the application artifacts.	Application Tier
OID Instance Home	LDAPHOST1 LDAPHOST2	*OID_ORACLE_INSTANCE*	Directory Tier
OVD Instance Home	LDAPHOST1 LDAPHOST2	*OVD_ORACLE_INSTANCE*	Directory Tier
Oracle HTTP Server	WEBHOST1 WEBHOST2	*WEB_ORACLE_INSTANCE*	Web Tier
Oracle RAC Databases	OIDDBHOST1 OIDDBHOST2	User defined	Directory Tier
OAM	OAMHOST1 OAMHOST2	All the configurations are within the respective home directories described in this table. There are no instance homes.	Application Tier

16.5.3 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 16.5.3.1, "Backing Up Middleware Home"
- Section 16.5.3.2, "Backing Up LDAP Directories"
- Section 16.5.3.3, "Backing Up the Database"
- Section 16.5.3.4, "Backing Up the WebLogic Domain"
- Section 16.5.3.5, "Backing Up the Web Tier"

16.5.3.1 Backing Up Middleware Home

Back up the Middleware homes whenever you create a new one or add components to it. The Middleware homes used in this guide are Oracle Identity Management and Oracle Identity and Access Management, as listed in Table 16–2.

16.5.3.2 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 16.5.3.2.1, "Backing up Oracle Internet Directory"
- Section 16.5.3.2.2, "Backing up Oracle Virtual Directory"
- Section 16.5.3.2.3, "Backing Up Third-Party Directories"

16.5.3.2.1 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
   ```

16.5.3.2.2 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
   ```

16.5.3.2.3 Backing Up Third-Party Directories Refer to your operating system vendor's documentation for information about backing up directories.

16.5.3.3 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 or Oracle Identity Manager.

16.5.3.4 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 16.1, "Starting and Stopping Components."
2. Back up the *`ASERVER_HOME`* directory from shared storage.
3. Back up the *`MSERVER_HOME`* directory from each host.

4. Restart the WebLogic Administration Server and managed servers.

16.5.3.5 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 16.1, "Starting and Stopping Components."
2. Back up the Oracle HTTP Server.
3. Start the Oracle HTTP Server as described in Section 16.1, "Starting and Stopping Components."

16.6 Patching Enterprise Deployments

It is recommended that you patch enterprise deployments by using the IDM Patching Framework provided as part of this release.

The topology data used by the tools is located in the topology store, `topology.xml`. This file is generated by the Oracle Identity Management Provisioning Wizard and contains most of the environment details used by the tools to apply patches. The Patching Framework uses this file to determine how to apply patches.

In `topology.xml`, wo instance groups were created during provisioning. An instance group contains a list of components which are required to keep a system available.

For example, in a typical enterprise deployment, two instance groups are created:

- Instance Group 1, which will contain the components on WEBHOST1, LDAPHOST1 and IDMHOST1
- Instance Group 2, which will contain the components on WEBHOST2, LDAPHOST2 and IDMHOST2

The process of applying patches can be summarized as follows:

1. Create a patch top. A patch top directory contains patches, classified by each product to which patches apply.
2. Run Patch Manager to generate a patch plan. The plan, based on how the Manager is run, holds steps which apply to the entire deployment topology, or to just a specified instance group.
3. Run the Patcher against all hosts which are affected by the plan. You might need to execute the Patcher on a given host multiple times if required by a given plan. As each Patcher invocation completes, it directs you where to run the Patcher next.

When Patcher runs, it stops and starts components as necessary, and ensures that patches are applied in the correct order to satisfy dependencies.

To keep downtime to a minimum, whenever possible, use Patcher to apply patches to each instance group in turn. Doing this ensures that you need not stop the entire deployment during patching.

Full details on how to use the IDM Patching Framework can be found in the chapter "Patching Oracle Identity Management Artifacts" in *Oracle Fusion Applications Patching Guide*.

This section contains the following topics:

- Section 16.6.1, "Patching an Oracle Fusion Middleware Source File"
- Section 16.6.2, "Patching Identity Management Components"

16.6.1 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*.

16.6.2 Patching Identity Management Components

To patch Oracle Identity Management components with minimal down time, it is recommended that you follow these guidelines:

1. Route LDAP traffic to LDAPHOST2 only. Same steps for other servers in instance group 1.
2. Create a patch plan for instance group 1 and execute it.
3. Route all LBR traffic to instance group 1 servers only.
4. Verify applications are working properly.
5. Create a patch plan for instance group 2 and execute it.
6. Route all Load Balancer traffic to instance group 2 servers only.
7. Verify applications are working properly.
8. Route all traffic to servers of both instance groups.

16.7 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.

16.8 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.

This section contains the following topics:

- Section 16.8.1, "Failing Over the Administration Server to IDMHOST2"
- Section 16.8.2, "Starting the Administration Server on IDMHOST2"
- Section 16.8.3, "Validating Access to IDMHOST2 Through Oracle HTTP Server"
- Section 16.8.4, "Failing the Administration Server Back to IDMHOST1"

16.8.1 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.
- 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`
  - ADMINVIP: `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 inIDMHOST2 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.

1. Stop the Administration Server as described in Section 16.1, "Starting and Stopping 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 eth0:1 100.200.140.206 netmask 255.255.255.0
      ```

      Note: Ensure that the netmask and interface to be used match the available network configuration in IDMHOST2.

3. Update routing tables by using `arping`, for example:

   ```
   /sbin/arping -q -U -c 3 -I eth0 100.200.140.206
   ```

16.8.2 Starting the Administration Server on IDMHOST2

Perform the following steps to start Node Manager on IDMHOST2.

1. On IDMHOST2, mount the Administration Server domain directory if it is not already mounted. For example:

   ```
   mount /u01/oracle
   ```

2. Start Node Manager by using the following commands:

   ```
   cd WL_HOME/server/bin
   ./startNodeManager.sh
   ```

3. 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.

4. Run the setNMProps.sh script to set the `StartScriptEnabled` property to `true` before starting Node Manager:

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

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

5. Start the Node Manager as described in Section 16.1, "Starting and Stopping Components."

6. 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','Admin_Password', 'IDMHOST2','5556',
   'IDMDomain','/u1/oracle/config/domains/IDMDomain')
   nmStart('AdminServer')
   ```

7. Test that you can access the Administration Server on IDMHOST2 as follows:

   a. Ensure that you can access the Oracle WebLogic Server Administration Console at:

      `http://ADMINVHN.mycompany.com:7001/console.`

   b. Check that you can access and verify the status of components in the Oracle Enterprise Manager at: `http://ADMINVHN.mycompany.com:7001/em`.

16.8.3 Validating Access to IDMHOST2 Through Oracle HTTP Server

Perform the same steps as in Section 14.1.1, "Verify Connectivity" This is to check that you can access the Administration Server when it is running on IDMHOST2.

16.8.4 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
   ```

   where *x*`:`*y* is the current interface used by `ADMINVHN.mycompany.com`.

5. Run the following command on IDMHOST1:

   ```
   /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 using arping. Run the following command from IDMHOST1.

   ```
   /sbin/arping -q -U -c 3 -I interface 100.200.140.206
   ```

7. If Node Manager is not already started on IDMHOST1, start it, as described in Section 16.1, "Starting and Stopping 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,'Admin_Pasword, IDMHOST1,'5556',
        'IDMDomain','/u01/oracle/config/domains/IDMDomain'
   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 3.7, "Fixed Ports Used by the Provisioning Wizard."

10. Check that you can access and verify the status of components in the Oracle Enterprise Manager at:

```
http://ADMINVHN.mycompany.com:7001/em
```

16.9 Changing Startup Location

When the environment was provisioned, start and stop scripts were generated to start and stop components in the topology. At the time of provisioning, the Admin server was configured to start on IDMHOST1. If you want to permanently change this to start on IDMHOST2, perform the following steps.

Edit the file `serverInstancesInfo.txt`, which is located in the directory: *`SHARED_CONFIG_DIR`*`/scripts`

Locate the line which looks like this:

```
IDMHOST1.mycompany.com AS AdminServer
```

Change `IDMHOST1` to `IDMHOST2` and save the file.

16.10 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 16.10.1, "Troubleshooting Identity Management Provisioning"
- Section 16.10.2, "Troubleshooting Start/Stop Scripts"
- Section 16.10.3, "Troubleshooting Oracle Internet Directory"
- Section 16.10.4, "Troubleshooting Oracle Virtual Directory"
- Section 16.10.5, "Troubleshooting Oracle Directory Services Manager"
- Section 16.10.6, "Troubleshooting Oracle Access Manager 11g"
- Section 16.10.7, "Troubleshooting Oracle Identity Manager"
- Section 16.10.8, "Troubleshooting Oracle SOA Suite"
- Section 16.10.9, "Troubleshooting Oracle Identity Federation"

16.10.1 Troubleshooting Identity Management Provisioning

This section describes some common problems related to provisioning. It contains the following topics:

- Section 16.10.1.1, "Provisioning Fails"
- Section 16.10.1.2, "OID Account is Locked"

16.10.1.1 Provisioning Fails

Problem

Provisioning fails.

Solution

Check the provisioning logs located in the directory:

`SHARED_ROOT``/config/provisioning/logs/`*`hostname`*

where *hostname* is the host where the provisioning step failed.

16.10.1.2 OID Account is Locked

Problem

Investigation into the OID logs shows that the OID account is locked.

This is generally caused by the load balancer. The load balancer is continually polling OID to see if it is available using the given credentials. During setup, this can cause the account to become locked.

Solution

Disable the OID load balancer monitor during preconfiguration. Then enable it when provisioning is complete. Another alternative is to reduce the check frequency.

16.10.2 Troubleshooting Start/Stop Scripts

Problem

Problem: Start/Stop scripts fail to start or stop a managed server.

The start/stop logs in the directory *SHARED_CONFIG_DIR*/scripts/logs contain an error similar to this:

```
weblogic.utils.AssertionError: ***** ASSERTION FAILED *****
        at
weblogic.server.ServerLifeCycleRuntime.getStateRemote(ServerLifeCycleRuntime.java:
734)
        at
weblogic.server.ServerLifeCycleRuntime.getState(ServerLifeCycleRuntime.java:581)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
```

Solution

1. Shut down the failing managed server. You might have to kill the process.
2. Back up the managed server's LDAP data, then remove it. For example:

   ```
   rm -rf LOCAL_CONFIG_DIR/domains/IDMDomain/servers/server_name/data/ldap
   ```

   where *server_name* is the name of the failing managed server.
3. Restart the managed server

16.10.3 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 16.10.3.1, "Oracle Internet Directory Server is Not Responsive."
- Section 16.10.3.2, "SSO/LDAP Application Connection Times Out"
- Section 16.10.3.3, "LDAP Application Receives LDAP Error 53 (DSA Unwilling to Perform)"
- Section 16.10.3.4, "TNSNAMES.ORA, TAF Configuration, and Related Issues"

16.10.3.1 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.

16.10.3.2 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.

16.10.3.3 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

16.10.3.4 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.

16.10.4 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:

16.10.4.1 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` and remove any blank lines at the end of the file. Save the file and run `SSLServerConfig.sh` again.

16.10.4.2 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.

16.10.4.3 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.

16.10.4.4 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.

16.10.4.5 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.

16.10.5 Troubleshooting Oracle Directory Services Manager

This section describes some common problems that can arise with Oracle Directory Services Manager and the actions you can take to resolve the problem. It contains the following topics:

- Section 16.10.5.1, "ODSM Browser Window and Session Issues"
- Section 16.10.5.2, "ODSM Does not Open When Invoked from Fusion Middleware Control"
- Section 16.10.5.3, "ODSM Failover is Not Transparent"
- Section 16.10.5.4, "ODSM Loses Connection and Displays Message that LDAP Server is Down"
- Section 16.10.5.5, "ODSM Loses Connection to Instance Using ORAC Database"
- Section 16.10.5.6, "OHS Must Be Configured to Route ODSM Requests to Multiple Oracle WebLogic Servers"
- Section 16.10.5.7, "ODSM is Not Accessible"

16.10.5.1 ODSM Browser Window and Session Issues

After you have logged into Oracle Directory Services Manager, you can connect to multiple directory instances from the same browser window.

Avoid using multiple windows of the same browser program to connect to different directories at the same time. Doing so can cause a Target unreachable error.

You can log in to the same Oracle Directory Services Manager instance from different browser programs, such as Internet Explorer and Firefox, and connect each to a different directory instance.

If you change the browser language setting, you must update the session to use the new setting. To update the session, either disconnect the current server connection, refresh the browser page (either reenter the Oracle Directory Services Manager URL in the URL filed and press enter or press F5) and reconnect to the same server, or quit and restart the browser.

16.10.5.2 ODSM Does not Open When Invoked from Fusion Middleware Control

Problem

Oracle Directory Services Manager does not open after you attempt to invoke it from Oracle Enterprise Manager Fusion Middleware Control by selecting one of the options from the **Directory Services Manager** entry in the **Oracle Virtual Directory** menu in the Oracle Virtual Directory target or the **Oracle Internet Directory** menu in the Oracle Internet Directory target.

Solution

This is probably an installation problem. See the *Oracle Fusion Middleware Installation Guide for Oracle Identity Management*.

16.10.5.3 ODSM Failover is Not Transparent

Problem

When you perform an Oracle Directory Services Manager failover using Oracle HTTP Server, the failover is not transparent. You see this behavior when you perform the following steps:

1. Oracle Directory Services Manager is deployed in a High Availability active-active configuration using Oracle HTTP Server.
2. Display an Oracle Directory Services Manager page using the Oracle HTTP Server name and port number.
3. Make a connection to an Oracle Internet Directory or Oracle Virtual Directory server.
4. Work with the Oracle Internet Directory or Oracle Virtual Directory server using the current Oracle Directory Services Manager Oracle HTTP Server host and port.
5. Shut down one Managed Server at a time using the WebLogic Server Administration Console.
6. Go back to the Oracle Directory Services Manager page and port, and the connection which was established earlier with Oracle Internet Directory or Oracle Virtual Directory. When you do, a message is displayed advising you to re-establish a new connection to the Oracle Directory Services Manager page.

Solution

If you encounter this problem, perform the following steps:

1. In your web browser, exit the current Oracle Directory Services Manager page.
2. Launch a new web browser page and specify the same Oracle Directory Services Manager Oracle HTTP Server name and port.

3. Re-establish a new connection to the Oracle Internet Directory or Oracle Virtual Directory server you were working with earlier.

16.10.5.4 ODSM Loses Connection and Displays Message that LDAP Server is Down

Problem

Oracle Directory Services Manager temporarily loses its connection to Oracle Internet Directory and displays the message LDAP Server is down.

Solution

In a High Availability configuration where Oracle Directory Services Manager is connected to Oracle Internet Directory through a load balancer, Oracle Directory Services Manager reports that the server is down during failover from one instance of Oracle Internet Directory to another. In other configurations, this message might indicate that Oracle Internet Directory has been shut down and restarted. In either case, the connection is reestablished in less than a minute, and you are able to continue without logging in again.

16.10.5.5 ODSM Loses Connection to Instance Using ORAC Database

Problem

Oracle Directory Services Manager temporarily loses its connection to an Oracle Internet Directory or Oracle Virtual Directory instance that is using an Oracle RAC Database. Oracle Directory Services Manager might display the message `LDAP error code 53 - Function not implemented`.

Solution

This error can occur during failover of the Oracle Database that the Oracle Internet Directory or Oracle Virtual Directory instance is using. The connection is reestablished in less than a minute, and you are able to continue without logging in again.

16.10.5.6 OHS Must Be Configured to Route ODSM Requests to Multiple Oracle WebLogic Servers

Problem

You must perform the following steps to configure Oracle HTTP Server to route Oracle Directory Services Manager requests to multiple Oracle WebLogic Servers in a clustered Oracle WebLogic Server environment.

Solution

Perform these steps:

1. Create a backup copy of the Oracle HTTP Server's `admin.conf` file, which is located in *`ORACLE_INSTANCE`*`/config`. The backup copy provides a source to revert to if you encounter problems after performing this procedure.
2. Add the following text to the end of the Oracle HTTP Server's `admin.conf` file and replace the variable placeholder values with the host names and Managed Server port numbers specific to your environment. Be sure to use the `<Location /odsm/ >` as the first line in the entry. Using `<Location /odsm/faces >` or `<Location /odsm/faces/odsm.jspx >` can distort the appearance of the Oracle Directory Services Manager interface.

```
<Location /odsm/ >
   SetHandler weblogic-handler
   WebLogicCluster host-name-1:managed-server-port,host-name_2:managed_server_
port
</Location>
```

3. Restart the Oracle HTTP Server, as described in Section 16.1, "Starting and Stopping Components," to activate the configuration change.

> **Note:** Oracle Directory Services Manager loses its connection and displays a session time-out message if the Oracle WebLogic Server in the cluster that it is connected to fails. Oracle Directory Services Manager requests is routed to the secondary Oracle WebLogic Server in the cluster that you identified in the httpd.conf file after you log back in to Oracle Directory Services Manager.

16.10.5.7 ODSM is Not Accessible

Problem

Attempting to access Oracle Directory Services Manager using a web browser fails.

Solution

- Verify the Oracle Virtual Directory server is running. The Oracle Virtual Directory server must be running to connect to it from Oracle Directory Services Manager.
- Verify you entered the correct credentials in the Server, Port, User Name and Password fields. You can execute an ldapbind command against the target Oracle Virtual Directory server to verify the server, user name, and password credentials.
- Verify you are using a supported browser. Oracle Directory Services Manager supports the following browsers:
  - Internet Explorer 7
  - Firefox 2.0.0.2 and 3.0
  - Safari 3.1.2 (desktop)
  - Google Chrome 0.2.149.30

> **Note:** While Oracle Directory Services Manager supports all of the preceding browsers, only Internet Explorer 7 and Firefox 2.0.0.2 are certified.

16.10.6 Troubleshooting Oracle Access Manager 11g

This section describes some common problems that can arise with Oracle Access Manager and the actions you can take to resolve the problem. It contains the following topics:

- Section 16.10.6.1, "OAM Fails to Connect to the Identity Store at First Start"
- Section 16.10.6.2, "OAM Runs out of Memory"
- Section 16.10.6.3, "Fusion Applications Preverify Fails to Validate OAM Admin Users"
- Section 16.10.6.4, "User Reaches the Maximum Allowed Number of Sessions"

- Section 16.10.6.5, "Policies Do Not Get Created When Oracle Access Manager is First Installed"
- Section 16.10.6.6, "You Are Not Prompted for Credentials After Accessing a Protected Resource"
- Section 16.10.6.7, "Cannot Log In to OAM Console"

16.10.6.1 OAM Fails to Connect to the Identity Store at First Start

Problem

After IDM Provisioning is complete, attempts to log in to consoles fail. The OAM log contains messages similar to this:

```
could not be initialized due to
oracle.security.am.engines.common.identity.provider.exceptions.IdentityProvide
rException: OAMSSA-20005: Error initializing User/Role API : null..>
```

Solution

The startup script uses `opmnctl startall`, which starts OID and OVD at the same time. In some cases, OVD attempts to connect with OID before OID is ready, resulting in a failed connection. As a workaround, perform the following steps on LDAPHOST1:

1. Stop all `opmnctl` processes.

   ```
   OID_ORACLE_INSTANCE/bin/opmnctl stopall
   ```

2. Start OPMN.

   ```
   OID_ORACLE_INSTANCE/bin/opmnctl start
   ```

3. Start each OID instance. For example:

   ```
   OID_ORACLE_INSTANCE/bin/opmnctl startproc ias-component=oid1
   OID_ORACLE_INSTANCE/bin/opmnctl startproc ias-component=oid2
   ```

4. Wait one minute.
5. Start OVD

   ```
   OID_ORACLE_INSTANCE/bin/opmnctl startproc ias-component=ovd1
   ```

Repeat Steps 1-5 on LDAPHOST2.

Restart all WebLogic servers.

16.10.6.2 OAM Runs out of Memory

Problem

After Oracle Access Manager has been running for a while, you see the following error message in the output:

```
Attempting to allocate 1G bytes
There is insufficient native memory for the Java Runtime Environment to continue.
```

Possible reasons:

- The system is out of physical RAM or swap space.
- In 32 bit mode, the process size limit was reached.

Solutions

- Reduce memory load on the system.
- Increase physical memory or swap space.
- Check if swap backing store is full.
- Use 64 bit Java on a 64 bit OS.
- Decrease Java heap size (-Xmx/-Xms).
- Decrease number of Java threads.
- Decrease Java thread stack sizes (-Xss).
- Disable compressed references (-XXcompressedRefs=false).
- Ensure that command line tool `adrci` can be executed from the command line.
  - at oracle.dfw.impl.incident.ADRHelper.invoke(ADRHelper.java:1309)
  - at oracle.dfw.impl.incident.ADRHelper.createIncident(ADRHelper.java:929
  - at oracle.dfw.impl.incident.DiagnosticsDataExtractorImpl.createADRIncident(DiagnosticsDataExtractorImpl.java:1116)
- On both IDMHOST1 and IDMHOST2, edit the file `setSOADomainEnv.sh`, which is located in *`MSERVER_HOME`*`/bin` and locate the line which begins:

  ```
  PORT_MEM_ARGS=
  ```

  Change this line so that it reads:

  ```
  PORT_MEM_ARGS="-Xms768m -Xmx2560m"
  ```

16.10.6.3 Fusion Applications Preverify Fails to Validate OAM Admin Users

Problem

The Fusion Applications preverify step, described in the task "Run the pre-verify phase" in *Oracle Fusion Applications Enterprise Deployment Guide for Customer Relationship Management*, fails to validate OAM Admin users. In the OAM diagnostic file, you see an error 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]
```

Solution

1. Shut down the administration server and all managed servers in the domain, as described in Section 16.1, "Starting and Stopping Components."
2. Delete the files with names of the form:

   ```
   /tmp/UCP*
   ```

3. Restart the administration server and managed servers.

16.10.6.4 User Reaches the Maximum Allowed Number of Sessions

Problem

The Oracle 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.

16.10.6.5 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.

16.10.6.6 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.

16.10.6.7 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.

16.10.7 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 16.10.7.1, "java.io.FileNotFoundException When Running Oracle Identity Manager Configuration"
- Section 16.10.7.2, "ResourceConnectionValidationxception When Creating User in Oracle Identity Manager"

16.10.7.1 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 (`OH/bin/config.sh`).

16.10.7.2 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.

16.10.8 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:

16.10.8.1 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.

16.10.9 Troubleshooting Oracle Identity Federation

This section describes some common problems that can arise with Oracle Identity Federation and the actions you can take to resolve the problem. It contains the following topics:

- Section 16.10.9.1, "Extending the Domain with Oracle Identity Federation Fails"
- Section 16.10.9.2, "Cannot Change Oracle Identity Federation Parameters by Using Fusion Middleware Control"

16.10.9.1 Extending the Domain with Oracle Identity Federation Fails

Problem

Extending the domain with Oracle Identity Federation fails when Oracle Identity Manager is installed at the Create Managed Server step.

Solution

Copy the file `setDomainEnv.sh` from *`ASERVER_HOME`*`/bin` on IDMHOST1 to OIFHOST1.

Retry the operation.

16.10.9.2 Cannot Change Oracle Identity Federation Parameters by Using Fusion Middleware Control

Problem

You cannot change Oracle Identity Federation parameters by using Oracle Enterprise Manager Fusion Middleware Control. You see the message:

```
Configuration settings are unavailable because ..... OIF ......
is down
```

even though Oracle Identity Federation is up and running.

Solution

Here are the common causes and resolutions:

1. Oracle Identity Federation is up but the EM agent is down.
   a. Check the EM agent status by running:

      `ORACLE_INSTANCE``/EMAGENT/EMAGENT/bin/emctl status agent`

   b. Start the EM agent, if it is down, by running:

      `ORACLE_INSTANCE``/bin/opmnctl startproc ias-component=EMAGENT`

   c. Log in to Fusion Middleware Control again.

2. Oracle Identity Federation and EM agent are up, but the OIF home page and configuration pages in Fusion Middleware Control still show: **OIF is down**.

   a. Check if the EM agent points to the correct Fusion Middleware Control by running:

      ```
      ORACLE_INSTANCE/EMAGENT/EMAGENT/bin/emctl status agent
      ```

      Verify that the `host` and `port` for property `Repository URL` are the same as the Fusion Middleware Control's host and port.

   b. If the `host` and `port` are mismatched, change the `Repository URL` in EM agent to the correct Fusion Middleware Control by running:

      ```
      ORACLE_INSTANCE/EMAGENT/EMAGENT/bin/emctl switchOMS
      http(s)://Host:Port/em/upload'
      ```

   c. Log in to Fusion Middleware Control again.

3. If the issue still exists, once logged in to Fusion Middleware Control, navigate to **Farm**->**Agent-Monitored Targets** (Top Left corner of the page) and click the **Configure** icon of the row that refers to Oracle Identity Federation. On the next page, ensure that all the information is correct and complete. Click **OK** to confirm.

   Check that the WebLogic user name and password are present.

   Check the host value. It might have been specified with an IPv6 address format.

4. If the issue still exists, restart the EM agent.

   a. Stop the EM agent by running:

      ```
      INST_HOME/bin/opmnctl stopproc ias-component=EMAGENT
      ```

   b. Start the EM agent by running:

      ```
      INST_HOME/bin/opmnctl startproc ias-component=EMAGENT
      ```

   c. Log in to Oracle Enterprise Manager Fusion Middleware Control again.

A

Adding Support for Active Directory

This appendix describes how to add support for Active Directory to your enterprise deployment.

This appendix contains the following sections:

- Section A.1, "Creating Adapters in Oracle Virtual Directory"
- Section A.2, "Preparing Active Directory"
- Section A.3, "Modifying Oracle Identity Manager to Support Active Directory"
- Section A.4, "Updating the Username Generation Policy for Active Directory"

A.1 Creating Adapters in Oracle Virtual Directory

Oracle Virtual Directory communicates with other directories through adapters.

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 A.1.1, "Removing Existing Adapters"
- Section A.1.2, "Creating an Oracle Virtual Directory Adapter for Active Directory"
- Section A.1.3, "Validating the Oracle Virtual Directory Adapters"

A.1.1 Removing Existing Adapters

The provisioning process created Oracle Virtual Directory adapters to Oracle Internet Directory. When you switch the identity store to Active Directory, you must remove these adapters.

1. Log in to ODSM at: `http://admin.mycompany.com/odsm`
2. If you have not already done so, create connections to each of your Oracle Virtual Directory instances using the steps in Section 10.3, "Creating ODSM Connections to Oracle Virtual Directory."
3. Select one of the Oracle Virtual Directory instances and connect to it.
4. Click the **Adapter** tab.
5. Click the adapter **User ID**.
6. Click **Delete Selected Adapter**.
7. Repeat for the adapter **CHANGELOG_OID**.
8. Repeat Steps 1- 7 for each Oracle Virtual Directory instance.

A.1.2 Creating an Oracle Virtual Directory Adapter for Active Directory

You can use `idmConfigTool` 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 the environment variable `ORACLE_HOME` to *`IAM_ORACLE_HOME`*.
2. Create a properties file for the Active Directory adapter called `ovd1.props`, with the following content:

```
ovd.host:LDAPHOST1.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_ADMIN_PORT`*) in Section 3.7, "Fixed Ports Used by the Provisioning Wizard").
- `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 (*`OID_LDAP_PORT`* in Section 6.1, "Assembling Information for Identity Management Provisioning").
- `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 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.

A.1.3 Validating the Oracle Virtual Directory Adapters

Perform the following tasks by using ODSM:

1. Access ODSM through the load balancer at: `http://ADMIN.mycompany.com/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 LDAPHOST1.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 LDAPHOST1 -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 LDAPHOST1 -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.

A.2 Preparing Active Directory

Prepare Active Directory as described in the following sections:

- Section A.2.1, "Configuring Active Directory for Use with Oracle Access Manager and Oracle Identity Manager"
- Section A.2.2, "Creating Users and Groups"
- Section A.2.3, "Creating Access Control Lists in Non-Oracle Internet Directory Directories"

A.2.1 Configuring Active Directory for Use with Oracle 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
   ```

A.2.2 Creating Users and Groups

Create users and groups as described in the following sections.

A.2.2.1 Creating Users and Groups by Using the idmConfigTool

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 in which the `idmConfigTool` is run. To ensure that the same file is appended to every time you run the tool, always run the `idmConfigTool` from the directory:

IAM_ORACLE_HOME/idmtools/bin

The syntax of the command on Linux is:

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

For example:

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

When the command runs, it prompts you to enter the password of the account you are connecting to and passwords for the accounts that are being created.

Note: The password 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

Note: This invocation of `idmConfigTool` creates the group `orclFAOAMUserWritePrivilegeGroup`.

A.2.2.2 Creating the Configuration File

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

```
# Common
IDSTORE_HOST: LDAPHOST1.mycompany.com
IDSTORE_PORT: 389
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
# 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
# Required due to bug
IDSTORE_OAAMADMINUSER : oaamadmin
# Fusion Applications
IDSTORE_READONLYUSER: IDROUser
IDSTORE_READWRITEUSER: IDRWUser
IDSTORE_SUPERUSER: weblogic_fa
# Weblogic
IDSTORE_WLSADMINUSER : weblogic_idm
```

Where:

- `IDSTORE_BINDDN` is an administrative user in the Identity Store Directory
- `IDSTORE_GROUPSEARCHBASE` is the location in the directory where Groups are Stored.
- `IDSTORE_HOST` and `IDSTORE_PORT` are, respectively, the host and port of your Identity Store directory. Specify the back end directory here, rather than Oracle Virtual Directory.

  Active Directory: `LDAPHOST1` and `389`

- IDSTORE_LOGINATTRIBUTE is the LDAP attribute which contains the users Login name.
- IDSTORE_OAMADMINUSER is the name of the user you want to create as your Oracle Access Manager Administrator.
- IDSTORE_OAMSOFTWAREUSER is a user that gets created in LDAP that is used when Oracle Access Manager is running to connect to the LDAP server.
- 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_READONLYUSER is the name of a user you want to create which has Read Only permissions on your Identity Store.
- IDSTORE_READWRITEUSER is the name of a user you want to create which has Read/Write permissions on your Identity Store.
- IDSTORE_SUPERUSER is the name of the administration user you want to use to log in to the WebLogic Administration Console in the Oracle Fusion Applications domain.
- 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 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_USERSEARCHBASE is the location in the directory where Users are Stored.
- OAM11G_IDSTORE_ROLE_SECURITY_ADMIN is the name of the group which is used to allow access to the OAM console.
- POLICYSTORE_SHARES_IDSTORE is set to `true` for IDM 11*g*.
- IDSTORE_OAAMADMINUSER is required because of a bug in `idmConfigTool`.

A.2.3 Creating Access Control Lists in Non-Oracle Internet Directory Directories

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 non-Oracle Internet Directory 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.

- Users and groups. ACLs to the users and groups container are provided in Oracle Internet Directory. Set them manually for other directories. The Oracle Identity Manager/Oracle Access Manager integration and Fusion Applications require the following artifacts to be created in the Identity store.
  - Group with read privileges to the users container (`orclFAUserReadPrivilegeGroup`). Configure the local directory ACLs so that this group has privileges to read all the attributes of the users in the Identity Store.
  - Group with read/write privileges to the users container (`orclFAUserWritePrivilegeGroup`)

  - Group with read privileges to the groups container (`orclFAGroupReadPrivilegeGroup`)
  - Group with read privileges to the groups container (`orclFAGroupWritePrivilegeGroup`)
  - Group with write privileges to a partial set of attributes (`orclFAUserWritePrefsPrivilegeGroup`)
- The user specified by the `IDSTORE_READONLYUSER` parameter. When you run the `preconfigIDstore` command, this user is assigned to the groups `orclFAUserReadPrivilegeGroup`, `orclFAWritePrefsPrivilegeGroup`, and `orclFAGroupReadPrivilegeGroup`. The user also needs compare privileges to the `userpassword` attribute of the user entry.
- The user specified by the `IDSTORE_READWRITEUSER` parameter. It is assigned to the groups `orclFAUserWritePrivilegeGroup` and `orclFAGroupWritePrivilegeGroup`.
- 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.
- Oracle Access Manager Admin User. This user is added to the OAM Administrator group, which provides permission for the administration of the Oracle Access Manager Console. No LDAP schema level privileges are required, since this is just an application user.
- Oracle 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.

A.3 Modifying Oracle Identity Manager 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*.

A.4 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 Console at the URL listed in Section 16.2, "About Identity Management Console URLs."
2. Click **Advanced** on the top of the right pane.
3. Click **Search System properties.**
4. On the navigation bar in the left pane, search on **Username Generation**.
5. Click **Default Policy for Username Generation**.
6. In the **Value** field, update the entry from `oracle.iam.identity.usermgmt.impl.plugins.DefaultComboPolicy` to `oracle.iam.identity.usermgmt.impl.plugins.FirstNameLastNamePolicyForAD`.
7. Click **Save**.

Index

P

R

S

T

U

V

W