Oracle® Exalogic Elastic Cloud

Enterprise Deployment Guide for Oracle Identity and Access Management

Release EL X2-2, X3-2, X4-2, and X5-2

E35832-03

February 2015

Documentation for installers that describes how to install and configure Oracle Identity and Access Management on an Exalogic platform in an enterprise deployment.

Oracle Exalogic Elastic Cloud Enterprise Deployment Guide for Oracle Identity and Access Management
Release EL X2-2, X3-2, X4-2, and X5-2

E35832-03

Primary Author: Joe Paul (Writer), Janga Aliminati (Architect), Federico Bardanca, Sriramulu Lakkaraju (Contributing Engineers)

Contributing Authors: Ellen Desmond, Peter LaQuerre, Mike Rhys

Contents

3 Configuring the Network for an Enterprise Deployment

4 Configuring Storage for an Enterprise Deployment

5 Configuring the Compute Nodes for an Enterprise Deployment

6 Configuring a Database for an Enterprise Deployment

7 Installing and Configuring Oracle Traffic Director for an Enterprise Deployment

8 Installing and Configuring Oracle Unified Directory

9 Creating a Domain for an Enterprise Deployment

10 Preparing Identity Stores

11 Extending the Domain to Include Oracle Access Management

12 Extending the Domain to Include Oracle Identity Manager

13 Setting Up Node Manager for an Enterprise Deployment

14 Configuring Server Migration for an Enterprise Deployment

15 Configuring Single Sign-on for Administration Consoles in an Enterprise Deployment

16 Managing the Topology for an Enterprise Deployment

A Worksheet for Identity Management Topology

B Using Multi Data Sources with Oracle RAC

C Enterprise Topology with Oracle HTTP Server

Preface

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

Audience

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

Documentation Accessibility

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

Access to Oracle Support

Oracle customers that have purchased support 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

For more information, see the following documents in the Oracle Other Product One Release 7.0 documentation set or in the Oracle Other Product Two Release 6.1 documentation set:

- *Oracle Other Product One Release Notes*
- *Oracle Other Product One Configuration Guide*
- *Oracle Other Product Two Getting Started Guide*
- *Oracle Other Product Two Reference Guide*
- *Oracle Other Product Two Tuning and Performance Guide*

Conventions

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

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.

1 Overview

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

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

This chapter contains the following sections:

- What Is an Enterprise Deployment?
- About the Reference Topology for Exalogic
- Benefits of Oracle Recommendations

1.1 What Is an Enterprise Deployment?

An enterprise deployment is a carefully designed, reference topology, which demonstrates how you can install, configure, extend, and manage Oracle Fusion Middleware in a typical production environment.

A production environment is an environment where you must take into account high-availability and security considerations, so you can deploy business-critical, custom applications. The people (customers, employees, co-workers) who use your applications can access them from the Internet safely and securely.

In an enterprise deployment, you achieve high availability by deploying the Oracle Fusion Middleware products across multiple hosts. You can then use a hardware load balancer, Oracle WebLogic Server clusters, an Oracle Real Application Clusters database to allow for failover when a host is unavailable.

You build in security by setting up firewalls between the tiers of the topology to restrict access to critical software and hardware components. Security also involves integrating the enterprise deployment with Oracle Identity and Access Management products, which provide authentication, authorization, other important security features.

The enterprise deployment is not the only supported topology for an Oracle Fusion Middleware environment. However, it serves as an example (or reference) you can use to build an environment that meets the needs of your organization and your application users.

1.2 About the Reference Topology for Exalogic

This guide provides a reference topology designed specifically for Exalogic.

Wherever possible, the topology has been modified to take advantage of the unique performance capabilities of the Exalogic Infiniband network fabric. It has also been designed to take advantage of Oracle Traffic Director and ZFS Storage appliance, both of which are available on the Exalogic platform.

Before you start implementing the Oracle Exalogic enterprise deployment topology, you should understand the current state of the Exalogic environment.

For example, it is assumed that you have completed all tasks described in the *Oracle Fusion Middleware Exalogic Machine Owner's Guide*, which discusses your data center site preparation, Oracle Exalogic machine commissioning, initial networking configuration including IP address assignments, and initial setup of the Sun ZFS Storage 7320 appliance.

As with other Enterprise Deployment Guides, you should use the topologies described in this guide as an example (or reference) topology on Exalogic machine, which can be modified to meet the specific needs of your organization.

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 DMZ, 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:

- Even if external communication is received on port 80, it is redirected to port 443
- 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 application or data tier DMZ is allowed.
- 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.
- All communication between components across firewalls 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 without a single point of failure.

2

Introduction and Planning

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

This chapter contains the following topics:

- Planning Your Deployment
- Understanding the Oracle Identity Management Deployment Topology on Exalogic
- Understanding the Topology Components
- Hardware Requirements for the Identity Management on Exalogic
- Software Components for an Enterprise Deployment
- Road Map for the Reference Topology Installation and Configuration

2.1 Planning Your Deployment

This section provides information to help you plan the deployment of Oracle Identity Management on Exalogic:

- Section 2.1.1, "Why the Deployment Topology in This Guide?"
- Section 2.1.2, "Alternative Deployment Topologies"
- Section 2.1.3, "Using a Worksheet to Plan for the Deployment Topology"

2.1.1 Why the Deployment Topology in This Guide?

When planning your deployment, you should be aware that this guide provides detailed instructions for implementing the specific reference topology described in this chapter.

This topology takes advantage of key features of the Exalogic platform, including:

- The high bandwidth and performance of the Exalogic internal Infiniband (IPoIB) network fabric
- The software load balancing capabilities of Oracle Traffic Director.

In this specific topology, Oracle Traffic Director is used as both a Web Listener and as a client-side load balancer for internal communication.

In this configuration, you can take advantage of the Exalogic default IPoIB network for all internal communications between the Traffic Director instances and the Identity Management compute nodes.

Only external traffic between the Traffic Director instances and external users is on the Exalogic Ethernet over IB (EoIB) network.

2.1.2 Alternative Deployment Topologies

Besides the topologies discussed in this guide, you can consider alternative Oracle Identity Manager topologies on Exalogic.

This guide does not provide specific instructions for implementing these alternative topologies, but consider the following when you are preparing your environment for an Oracle Identity Manager deployment on Exalogic:

- Using an External Oracle HTTP Server Web Tier Instead of Oracle Traffic Director
- Using Oracle Exadata Instead of an Oracle RAC Database

2.1.2.1 Using an External Oracle HTTP Server Web Tier Instead of Oracle Traffic Director

As described in Section 2.1.1, the topology in this guide uses Oracle Traffic Director as both a Web server and an internal load balancer. This configuration requires that you dedicate two compute nodes to hosting the Oracle Traffic Director instances.

If you cannot dedicate two compute nodes for Oracle Traffic Directory, or if you would rather use a dedicated Oracle HTTP Server Web Tier, then it is possible to deploy Oracle HTTP Server on an external Web tier, which is located outside the Exalogic machine.

Refer to Appendix C, "Enterprise Topology with Oracle HTTP Server" for a diagram of a typical Oracle Identity Manager topology on Exalogic with an external Oracle HTTP Server Web tier.

2.1.2.2 Using Oracle Exadata Instead of an Oracle RAC Database

The reference topology in this guide provides information on using an external Real Application Clusters (RAC) database as the repository for product schemas and security stores.

The topology assumes that the RAC database is hosted on dedicated servers. These servers can either be independent or as part of an Oracle Exadata database machine.

If an Oracle Exadata machine is used then this should be connected to the Exalogic machine via the infiniband fabric. For more information, see "Connecting Exalogic and Exadata Machines" in the *Oracle Exalogic Elastic Cloud Multi-Rack Cabling Guide*.

2.1.3 Using a Worksheet to Plan for the Deployment Topology

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 on an Exalogic platform.

You can also use Appendix A, "Worksheet for Identity Management Topology" to help you keep track of information, such as host names, IP addresses, and other important information as you procure and identify the machines and resources required for this deployment.

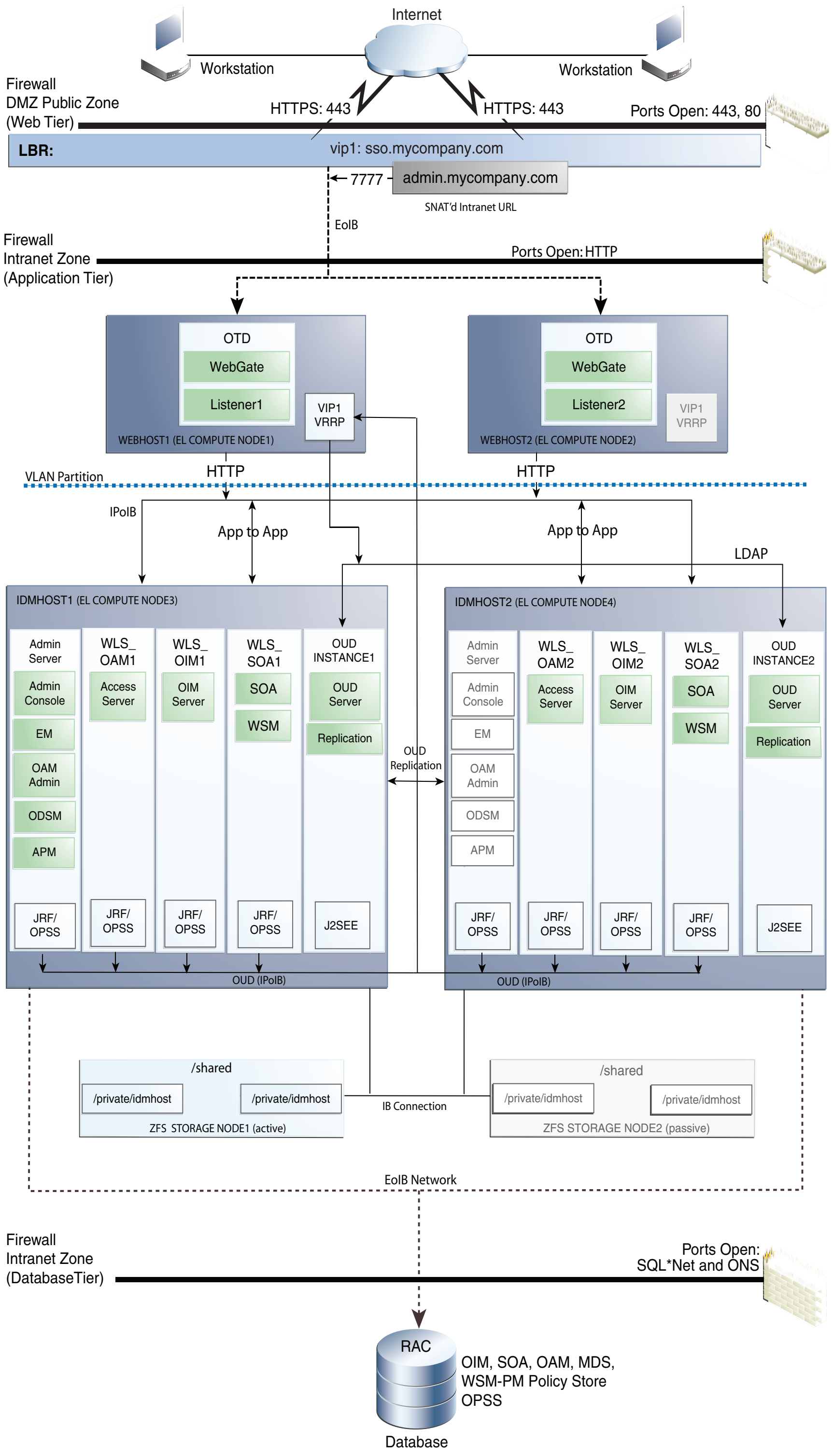
2.2 Understanding the Oracle Identity Management Deployment Topology on Exalogic

Figure 2–1 provides a diagram of a standard, reference topology for Oracle Identity Management on Exalogic.

In this specific topology, the Web tier consists of Oracle Traffic Director instances, and the Exalogic machine is connected to a remote Oracle RAC database over a 10 Gb Ethernet connection.

For a detailed description of the elements of the topology, see Section 2.3, "Understanding the Topology Components".

Figure 2–1 Oracle Identity Management on Exalogic, Deployed with Oracle Traffic Director and an Oracle RAC Database



2.3 Understanding the Topology Components

The topologies consist of three tiers, which are described in the following sections:

- Section 2.3.1, "About EoIB and IPoIB Communication"
- Section 2.3.3, "About the Web Tier"
- Section 2.3.4, "About the DMZ"
- Section 2.3.5, "About the Application Tier"
- Section 2.3.6, "About the Identity Stores"

2.3.1 About EoIB and IPoIB Communication

When you initially set up your Exalogic machine, the default network is running IP over Infiniband (IPoIB). For the different purposes of the topology described in this guide, you must configure Ethernet over Infiniband (EoIB) network access in addition to the IPoIB network. For more information, see "Configuring Ethernet Over InfiniBand" in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*.

In an Exalogic deployment the two different types of network are used as follows:

- IPoIB is used for internal communications for components within the Exalogic machine rack. This network is not visible outside of the Exalogic machine rack itself.
- EoIB is used for components inside the Exalogic machine rack to communicate with components external to the Exalogic machine rack.

The following four types of communication must be configured for the Oracle Identity and Access Management enterprise deployment on Exalogic:

- For the Oracle Traffic Director hosts, the IP addresses must be EoIB addresses accessible from the load balancer. The Oracle Traffic Director IP addresses are the only addresses accessible from the DMZ network.
- For the application tier, the IDMHOST machines IP addresses must be EoIB addresses that can access the Oracle RAC database SCAN and VIP addresses, Additionally, IDM servers use IPoIB address as main listen addresses for internal invocations and for RMI interactions inside the Exalogic rack.
- Communication and routing between Oracle Traffic Director hosts and the application tier must be only over IPoIB.
- For communication between the application tier components, for example, internal JMS destinations routing must be on IPoIB. Any front end address that is exposed ONLY for internal consumption, uses and IPoIB virtual IP on Oracle Traffic Director hosts.
- IDM Servers can also be accessed externally for RMI/JMS/T3 invocations and HTTP invocations. These take place for remote deployments, for external JMS producers and consumers and for other operations that use a listen address of the IDM servers that is available outside the Exalogic rack (EoIB).

For more information about IPoIB and EoIB network configuration, see Chapter 3, "Configuring the Network for an Enterprise Deployment".

2.3.2 About the Load Balancer

In an Exalogic deployment, a hardware load balancer sits outside the Exalogic machine rack. Its function is to receive external requests for the IAM deployment and pass them on to each of the Web hosts. These Web hosts can either be Oracle HTTP Servers or Oracle Traffic Director servers.

The load balancers are configured to receive HTTP and HTTPS requests. If an HTTPS request is received at the load balancer, the SSL is decrypted at the load balancer and passed on to the Web Servers using the HTTP protocol. This is known as SSL Termination at the load balancer.

The communication from the hardware load balancer to the Web tier (WEBHOST1 and WEBHOST2, in this case) is entirely over EoIB.

The load balancer is used to route both application and administrative requests to the Web servers. Administrative requests originate inside the organization's intranet. Application requests may be received through the intranet or the internet.

2.3.3 About the Web Tier

With Exalogic, you can take advantage of Oracle Traffic Director capabilities.

In particular, the architecture of Oracle Traffic Director enables it to handle large volumes of application traffic with low latency. It is optimized for use in Oracle Exalogic Elastic Cloud. It communicates with WebLogic Servers in the back end over Exalogic's InfiniBand fabric (IPoIB).

In this topology, the Oracle Traffic Director instances serve two purposes:

- They receive HTTP requests coming in from the hardware load balancer (over the EoIB network) and then route those requests (over the IPoIB network) to the compute nodes in the application tier.
- They route requests from the application tier components (over the IPoIB network) to other application tier components, such as requests from Oracle Access Manager to the Oracle Unified Directory directory service.

  The internal application to application requests, which are routed only over the internal IPoIB network, are routed via a virtual IP address that is depicted as VIP1 in the topology diagram (Figure 2–1).

  The Oracle Traffic Director instances are configured as part of a failover group. In this configuration, Oracle Traffic Director uses an implementation of the Virtual Routing Redundancy Protocol (VRRP) to provide failover capabilities. If an Oracle Traffic Director instance fails, IP addresses enabled on it are migrated to surviving instances, via VRRP.
- Requests are routed from the Oracle Traffic Director servers to an Oracle WebLogic Server running in the application tier.
- WebGate uses Oracle Access Protocol (OAP) to communicate with Oracle Access Manager to perform operations such as user authentication.
- Oracle Traffic Director performs the following actions.
  - Distributes the requests that it receives from clients to servers in the application tier based on the specific load-balancing method
  - Routes the requests based on specified rules
  - Caches frequently accessed data
  - Prioritizes traffic and controls the quality of service
- Oracle Traffic Director can be used to route HTTP or LDAP requests.
  - Used for internal load balancing (always required).
  - Acts as a Web server

2.3.4 About the DMZ

A DMZ is a means of restricting access to components of your infrastructure to those that actually need it. In the examples in this guide, there is a public DMZ. 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 public zone–This is where the outside world gains access to your systems. You place into this zone only those components that the outside world must access, such as the Load Balancers and Web Tiers. If users from the outside world attempts to access any servers or services below this zone, they are prevented from doing so by firewalls.

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

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

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

2.3.5 About the Application Tier

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

In the application tier, IDMHOST1 and IDMHOST2 include the following components, which are installed on Managed Servers in the Oracle WebLogic Server domain:

- The operational component of the infrastructure. This component is Oracle Access Management Access Manager (OAM). This is a J2EE application which is run within Oracle WebLogic Server.
- The administrative components of Identity management, including Oracle Identity Manager, which is used for user provisioning.
- Oracle SOA Suite (SOA), which is required by Oracle Identity Management for process workflows to manage request approvals.

IDMHOST1 hosts an Oracle WebLogic Administration Server. The Administration Server hosts the Oracle WebLogic Console, Oracle Enterprise Manager Fusion Middleware Control, Oracle Access Management Console, and Oracle Directory Services Manager (ODSM) for OUD.

Note that the Oracle WebLogic Server Administration Server is a singleton process. That is, only one Administration Server can be running at a time within a domain. In the event that the host running the Administration Server fails, the Administration Server can be manually started on a different host.

2.3.5.1 Architecture Notes

- An embedded version of Oracle Entitlement Server is used to control access to Oracle Fusion Middleware components.
- Oracle Entitlements Server uses a centralized policy store that is stored within a database.
- Access Manager uses the OPSS Policy Store to store policy information.
- 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 managed servers WLS_OAM1 and WLS_OAM2 are deployed in a cluster and Access Manager applications deployed to the cluster.
- The managed servers WLS_OIM1 and WLS_OIM2 are deployed in a cluster and Access Manager applications deployed to the cluster.
- The managed servers WLS_SOA1 and WLS_SOA2 are deployed in a cluster and Access Manager applications deployed to the cluster.

2.3.5.2 High Availability Provisions

- Oracle Traffic Director can be configured for high availability in active-passive mode. Virtual Hosts/IP addresses are started on a single OTD instance. A heart beat exists between each OTD instance. Using this heatbeat, a secondary OTD instance will enable the virtual host/IP address in the event of the failure of the primary OTD instance.
- OAM Server, Oracle Identity Manager, and SOA are active-active deployments; these servers communicate with the data tier at run time.
- Oracle Traffic Director directs HTTP and LDAP requests to all WebLogic managed servers or OUD Instances ensuring maximum availability.
- The WebLogic Administration Server and Oracle Enterprise Manager deployment is active-passive (where other components are active-active). There is one Administration Server per domain.
- The WebLogic Administration Server is a singleton component deployed in an active-passive configuration. If the primary fails or the Administration Server on IDMHOST1 does not start, the Administration Server on the secondary host can be started. If a WebLogic managed server fails, the node manager running on that host attempts to restart it.

2.3.5.3 Security Provisions

The adminitration tools for this deployment (for example, Oracle WebLogic Server Console, Oracle Enterprise Manager Fusion Middleware Control console, and Oracle Access Management Console) are accessible only through a virtual host (`admin.mycompany.com`) configured on the hardware load balancer, which is only available inside the firewall.

2.3.6 About the Identity Stores

Identity information is stored in an LDAP compliant directory. In this topology, Oracle supports the Oracle Unified Directory natively.

2.4 Hardware Requirements for the Identity Management on Exalogic

The following sections describe the hardware requirements for the Identity Management enterprise topologies on Exalogic:

- Hardware Load Balancer Requirements
- Exalogic Machine Requirements

2.4.1 Hardware Load Balancer Requirements

The Oracle Fusion Middleware enterprise deployment requires a hardware load balancer to route requests to the Web tier. For information about the minimum set of features required for the load balancer in this topology, see Section 3.9.1, "Load Balancer Requirements."

2.4.2 Exalogic Machine Requirements

Exalogic machines consist of virtual or physical machines, a storage appliance, as well as required InfiniBand and Ethernet networking components. The number of these components in each machine varies based on the hardware configuration.

For complete information about the hardware options available for Exalogic machines, see "Exalogic Hardware Configurations" in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*.

For any of the topologies described in this guide, an Exalogic machine eighth rack can be used. For more information, see Section 2.2, "Understanding the Oracle Identity Management Deployment Topology on Exalogic".

You can assign the Exalogic machines, as follows:

- Assign two machines to the Application Tier. These will be referred to as IDMHOST1 and IDMHOST2.
- If you are using the Oracle Traffic Director topology, assign two additional machines to the Oracle Traffic Director instances. These will be referred to as WEBHOST1 and WEBHOST2.

Note that you can also assign compute nodes for a standard Oracle RAC database, but this guide assumes your database will be hosted on a remote set of hosts.

2.5 Software Components for an Enterprise Deployment

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

This section contains the following topics:

- Section 2.5.1, "Software Required for the Oracle Identity Management Deployment Topology on Exalogic"
- Section 2.5.2, "About Obtaining Software"
- Section 2.5.3, "Applying Patches and Workarounds"

2.5.1 Software Required for the Oracle Identity Management Deployment Topology on Exalogic

Table 2–1 lists the Oracle software you need to obtain before starting the procedures in this guide.

Table 2–1 Software Versions Used

Short Name	Product	Version
OTD	Oracle Traffic Director	11.1.1.7.0
JRockit	Oracle JRockit	jrockit-jdk1.6.0_29-R28.2.0-4.0.1 or newer
WLS	Oracle WebLogic Server	10.3.6.0
IAM	Oracle Identity and Access Management	11.1.2.0.0
SOA	Oracle SOA Suite	11.1.1.6.0
WebGate	WebGate 11*g*	11.1.2.0.0
RCU	Repository Creation Assistant	11.1.2.0.0
OUD	Oracle Unified Directory	11.1.2.0.0

2.5.2 About Obtaining Software

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

2.5.3 Applying Patches and Workarounds

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

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

2.6 Road Map for the Reference Topology Installation and Configuration

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

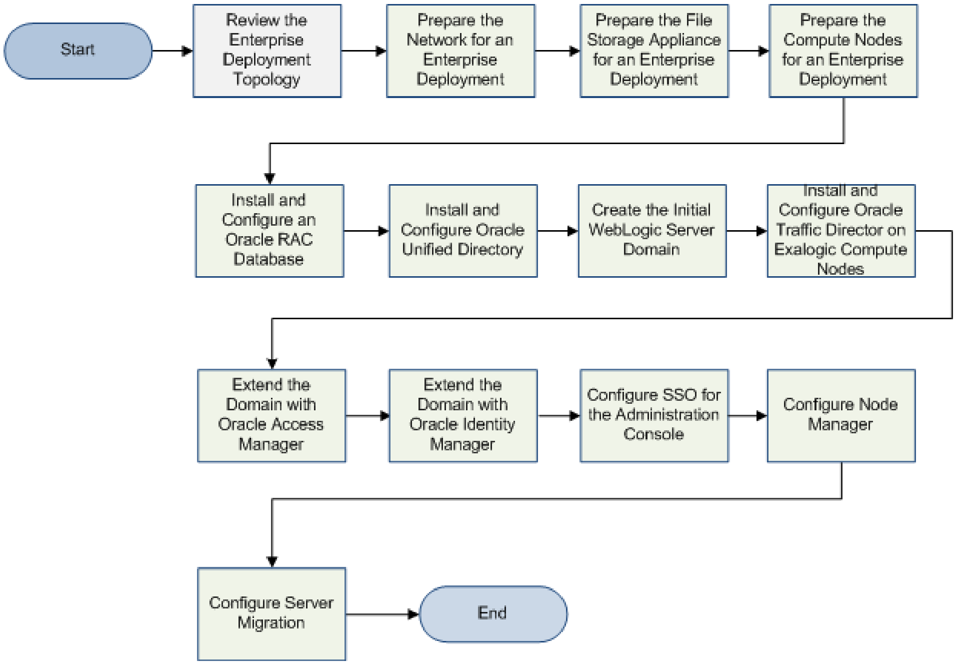
This section covers the following topics:

- Section 2.6.1, "Flow Chart of the Oracle Identity Management Enterprise Deployment Process"
- Section 2.6.2, "Steps in the Oracle Identity Management Enterprise Deployment Process"

2.6.1 Flow Chart of the Oracle Identity Management Enterprise Deployment Process

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

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


Start
Review the Enterprise Deployment Topology
Prepare the Network for an Enterprise Deployment
Prepare the File Storage Appliance for an Enterprise Deployment
Prepare the Compute Nodes for an Enterprise Deployment
Install and Configure an Oracle RAC Database
Install and Configure Oracle Unified Directory
Create the Initial WebLogic Server Domain
Install and Configure Oracle Traffic Director on Exalogic Compute Nodes
Extend the Domain with Oracle Access Manager
Extend the Domain with Oracle Identity Manager
Configure SSO for the Administration Console
Configure Node Manager
Configure Server Migration
End

2.6.2 Steps in the Oracle Identity Management Enterprise Deployment Process

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

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

Step	Description	More Information
Review the Enterprise Deployment Topology	Review the recommended topology and plan the topology best suited for organization and applications.	Section 2.1, "Planning Your Deployment"
Prepare the Network for an 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, "Configuring the Network for an Enterprise Deployment"
Prepare your File Storage Appliance for an 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, "Configuring Storage for an Enterprise Deployment"
Prepare the Compute Nodes for an Enterprise Deployment	To prepare your servers for an enterprise deployment, ensure that your servers meet hardware and software requirements, enable Unicode support and Virtual IP Addresses, mount shared storage, configure users and groups, and, if necessary, install software onto multi-homed systems.	Chapter 5, "Configuring the Compute Nodes for an Enterprise Deployment"

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

Step	Description	More Information
Prepare the Oracle RAC Database for an Enterprise Deployment	To prepare an Oracle RAC database for an enterprise deployment, review database requirements, create database services, load the metadata repository, in the Oracle RAC database, configure Identity Management schemas for transactional recovery privileges, and back up the database.	Chapter 6, "Configuring a Database for an Enterprise Deployment"
Install and Configure Oracle Unified Directory	Install and configure Oracle Unified Directory, which is used as the Identity Store in the recommended topologies. Configure two instances of Oracle Unified Directory by using Oracle Unified Directory configuration assistant.	Chapter 8, "Installing and Configuring Oracle Unified Directory"
Create the Initial WebLogic Server Domain	Run the Configuration Wizard to create the initial WebLogic Server domain.	Chapter 9.4, "Running the Configuration Wizard to Create a Domain"
Install and Configure Oracle Traffic Director on Exalogic Compute Nodes	Install and configure Oracle Traffic Director.	Chapter 7, "Installing and Configuring Oracle Traffic Director for an Enterprise Deployment"
Extend the Domain for Oracle Access Management?	Run the Configuration Wizard again and extend the domain to include Oracle Access Management.	Chapter 11, "Extending the Domain to Include Oracle Access Management"
Extend the Domain for Oracle Identity Manager?	Run the Configuration Wizard again and extend the domain to include Oracle Identity Manager.	Chapter 12, "Extending the Domain to Include Oracle Identity Manager"
Configure SSO for the Administration Console	Configure single sign-on (SSO) for administration consoles in an Identity Management Enterprise deployment.	Chapter 15, "Configuring Single Sign-on for Administration Consoles in an Enterprise Deployment"
Configure Node Manager	Set up Node manager by enabling host name verification, starting Node Manager, and configuring WebLogic Servers to use custom keystores.	Chapter 13, "Setting Up Node Manager for an Enterprise Deployment"
Configure Server Migration	Configure server migration for the WLS_OIM1, WLS_SOA1, WLS_OIM2, and WLS_SOA2 Managed Servers. The WLS_OIM1 and WLS_SOA1 Managed Server are configured to restart on IDMHOST2 should a failure occur. The WLS_OIM2 and WLS_SOA2 Managed Servers are configured to restart on IDMHOST1 should a failure occur.	Chapter 14, "Configuring Server Migration for an Enterprise Deployment"

3

Configuring 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

Table 3–1 summarizes the steps required to set up the network for an Enterprise Deployment on the Exalogic machine.

Table 3–1 Overview of the Network Configuration Process for an Exalogic Enterprise Deployment

Task	Description	More Information
Review network information	Read about the characteristics and goals of IDM Exalogic enterprise deployment network configuration.	Section 3.2, "About the Exalogic Network Configuration for the IDM Enterprise Topology"
Define the required hostname resolution.	It is important that the required DNS (either /etc/hosts or central DNS server) definitions are in place and that WebLogic Servers use hostnames and virtual hostnames instead of using IPs and virtual IPs directly.	Section 3.3, "Hostname and Networking Overview"
Configure virtual IP addresses for IPoIB	Read about and configure the IPoIB network and the required virtual IP addresses.	Section 3.4, "Configuring Virtual IP Addresses for IPoIB on Each Compute Node"

Table 3–1 (Cont.) Overview of the Network Configuration Process for an Exalogic Enterprise Deployment

Task	Description	More Information
Configure virtual IP addresses for IPoIB	Read about and configure the EoIB network and the required virtual IP addresses.	Section 3.5, "Configuring Virtual IP Addresses for EoIB on Each Compute Node"
Define the required virtual server names	The enterprise deployment requires that specific virtual server names be defined on your network. They must resolve to the specific compute nodes and servers in the topology.	Section 3.7, "Defining the Required Virtual Server Names"
Define the required Virtual IP Addresses	The enterprise deployment requires a set of virtual server names for routing requests to the proper server or service within the topology.	Section 3.8, "About IP Addresses and Virtual IP Addresses"
Configure the external hardware load balancer	The external hardware load balancer must be configured to accept requests from both external customers and company administrators and route them to the appropriate URLs in the topology.	Section 3.9, "Configuring the Load Balancer"
Configure the firewalls	When you install and configure the firewalls for your topology, use this information to open only the required ports and set the proper timeouts for each port.	Section 3.10, "Configuring Firewall Ports"

3.2 About the Exalogic Network Configuration for the IDM Enterprise Topology

The following sections provide information about the Exalogic network configuration for the IDM enterprise topology:

- Section 3.2.1, "General Characteristics and Goals of the Exalogic Network Configuration"
- Section 3.2.2, "Map of the Network Interfaces Used by the Components of the IDM Topology on Exalogic"
- Section 3.2.3, "Explanation of the Network Interfaces Map"

3.2.1 General Characteristics and Goals of the Exalogic Network Configuration

Exalogic system consists of three network areas - Management, IP over InfiniBand (IPoIB), and Ethernet over InfiniBand (EoIB).

- **IPoIB Network** - This network is used for inter rack communication. This network is the fastest available, but cannot be accessed from outside of the Exalogic machine rack.
- **Management network** - This EoIB network allows people to connect to the individual compute nodes from the public ethernet. It is used for management and setup only. This network should not be used for regular ethernet communications.
- **EoIB Network** - You can configure this network manually to allow communication between compute nodes and the external public network. This network would be used when:

- You wish the external load balancer to communicate with the Oracle traffic Director instances on compute nodes 1 and 2.
- You wish your compute nodes to communicate with an external database.
- You wish external Web servers (Oracle HTTP servers) to communicate with the WebLogic managed servers running on the compute nodes.

When you initially set up your Exalogic system, Management and IPoIB are configured by default. In addition to Management and IPoIB, you must manually configure EoIB network access for those components that are going to be exposed over ethernet out of the Exalogic machine rack.

An optimized Oracle Fusion Middleware system constrains communication between the various elements of the topology so it is performed over the Exalogic InfiniBand network as much as possible. For example, components should listen in InfiniBand interfaces to eliminate overhead on accessing the appropriate Gateways and to make use of the optimized InfiniBand network.

Additionally, when the same Exalogic machine rack is shared with other Oracle Fusion Middleware systems, such as WebCenter and Fusion Middleware SOA, or even with other type of deployments, such as test or development, then EoIB access might require isolated VLAN-based interfaces for Oracle Identity and Access management. VLANs can be used for this logical division of workload and for enforcing security isolation. However, the definition of such VLANs is outside the scope of this guide.

3.2.2 Map of the Network Interfaces Used by the Components of the IDM Topology on Exalogic

Figure 3–1 describes the components of an Oracle Fusion Middleware Identify and Access Management enterprise deployment on Exalogic, and the type of interfaces and communication protocols they use.

The IP addresses used in Figure 3–1 are examples and are used for consistency throughout this document. Other IPs are valid. It is a good practice to follow an order and separate types of servers in IP ranges. Table 3–2 lists the internal and external IP address used in this guide.

Table 3–2 Internal and External IP Addresses

Purpose	Network	IP Addresses	Netmask
External Compute Node Addresses	EoIB	10.10.10.x	255.255.224.0
External Floating Physical IP Addresses	EoIB	10.10.30.x	255.255.224.0
External Floating Oracle Traffic Director IP Addresses	EoIB	10.10.50.x	255.255.224.0
Internal Compute Node Addresses	IPoB	192.168.10.x	255.255.224.0
Internal Floating Physical IP Addresses	IPoB	192.168.30.x	255.255.240.0
Internal Oracle Traffic Director Addresses	IPoB	192.168.50.x	255.255.224.0

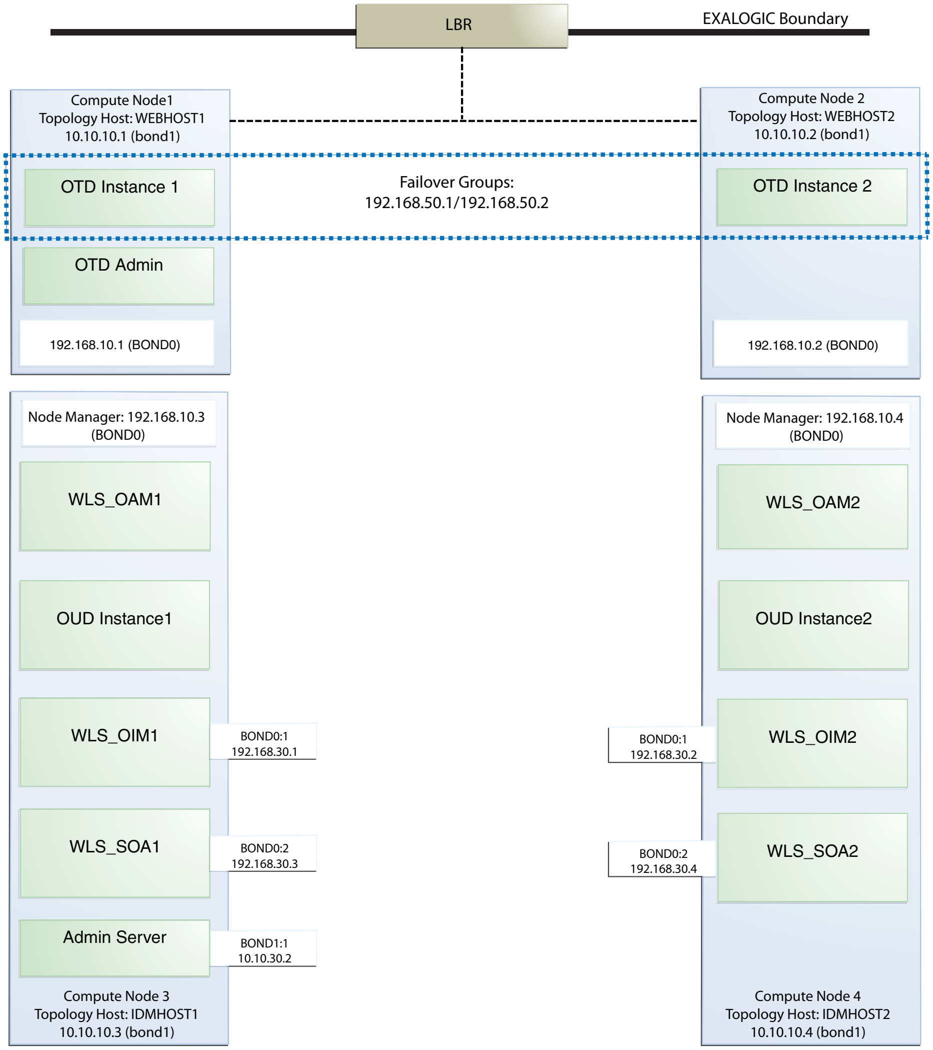
Note: The external IP addresses in Table 3–2 are assumed to be on the front end network.

Note: The subnets used here are examples only. It may be possible to use these subnets, the externally facing subnets follow the standards used in your organization.

For more information about the network map diagram, see the following:

- Table 3–4 lists the IPoIB (bond0) interfaces required for each compute node, as well as suggested IP addresses to assign to each interface.
- Table 3–5 lists the EoIB (bond1) interfaces required for each compute node, as well as the suggested IP addresses to assign to each interface.

Figure 3–1 Oracle IDM Exalogic Network Map



3.2.3 Explanation of the Network Interfaces Map

The Exalogic machine rack used for Oracle Identity and Access Management uses four compute nodes:

- Two compute nodes are used to host Oracle Traffic Director. Oracle Traffic Director acts as both a Web server and an internal load balancer.
- Two compute nodes are used to host the Oracle Identity and Access Management applications.

This section contains the following topics:

- Section 3.2.3.1, "Load Balancer"
- Section 3.2.3.2, "Oracle Traffic Director"
- Section 3.2.3.3, "Compute Node 1"
- Section 3.2.3.4, "Compute Node 2"
- Section 3.2.3.5, "Compute Node 3"
- Section 3.2.3.6, "Compute Node 4"

3.2.3.1 Load Balancer

An external load balancer sits outside of the Exalogic machine rack. Its purpose is to receive requests on the public ethernet network and distribute those requests to the Oracle Traffic Director nodes inside the machine rack using the front end EoIB network.

3.2.3.2 Oracle Traffic Director

Oracle Traffic Director serves two functions: load balancing, and HTTP server.

As a load balancer, Oracle Traffic Director is configured in a way that it can direct requests to the Oracle Unified Directory servers using the internal IPoIB network using TCP and to direct internal call back requests from Oracle Traffic Director to SOA servers using the internal IPoIB network using HTTP.

As an HTTP server, Oracle Traffic Director listens on the front end EoIB network for HTTP requests originating from the external load balancers. If these requests require access to the WebLogic managed servers on the compute nodes, then it directs these requests accordingly using the internal IPoIB network. *HTTP* requests on the front-end EoIB network.

3.2.3.3 Compute Node 1

Compute Node 1 (WEBHOST1) is configured to use the EoIB front end network. It uses this network to communicate with the external load balancer.

Oracle Traffic Director enables an IP address using a failover group to route requests to the Oracle Unified Directory servers using the IPoIB network.

Oracle Traffic Director acts as a failover node in the event that the IP address used for internal callbacks fails.

3.2.3.4 Compute Node 2

Compute Node 2 (WEBHOST2) is configured to use the EoIB front end network. It uses this network to communicate with the external load balancer.

Oracle Traffic Director enables an IP address using a failover group to route internal callback requests to SOA managed servers using the internal IPoIB network.

Oracle Traffic Director acts as a failover node in the event that the IP address used for Oracle Unified Directory fails.

3.2.3.5 Compute Node 3

Compute node 3 hosts the WebLogic and Oracle Unified Directory instances required by Oracle Identity and Access Manager.

Node Manager, which is used to start and stop the WebLogic managed servers is configured to accept requests on the internal IPoIB interface.

The compute node itself is configured for access on the front end EoIB interface as well. This allows virtual IP addresses to be configured on this interface. The virtual IP address is for the Weblogic administration server. This address is configured for external access for the purposes of external monitoring.

In addition, two floating IP addresses are attached to the IPoIB interface, which are used by the OIM and SOA managed servers to facilitate server migration.

Oracle Unified Directory listens for requests on the internal IPoIB network. These requests are received from Oracle Traffic Director.

3.2.3.6 Compute Node 4

Compute Node 4 hosts the WebLogic and Oracle Unified Directory instances required by Oracle and Access Management.

Node Manager, which is used to start and stop the WebLogic managed servers, is configured to accept requests on the internal IPoIB interface.

The compute node itself is configured for access on the front end EoIB interface as well. This allows virtual IP addresses to be configured on this interface. The virtual IP address is for the WebLogic administration server, this is configured for external access for the purposes of external monitoring.

In addition, two floating IP addresses are attached to the IPoIB interface which are used by the OIM and SOA managed servers to facilitate server migration.

Oracle Unified Directory listens for requests on the internal IPoIB network. These requests are received from Oracle Traffic Director.

3.3 Hostname and Networking Overview

Networking is a complicated but critical part of any Exalogic deployment. This guide utilizes the IPoIB network for internal communications and the EoIB network for external communications.

Table 3–3 is a summary of the required networking setup in the Exalogic machine rack. The following sections describe in detail how to set up this networking.

A column has been added to the table to allow you to add your own values for easier cross referencing.

Appropriate hostname resolution is critical to topology designs that can sustain network changes, system relocation and disaster recovery scenarios. It is important that the required DNS (either `/etc/hosts` or central DNS server) definitions are in place and that WebLogic Servers use hostnames and virtual hostnames instead of using IPs and virtual IPs directly. Additionally, the Exalogic enterprise deployment requires a set of virtual server names for routing requests to the proper server or service within the topology through the external load balancer and the Oracle Traffic Director servers.

These virtual server names must be enabled in the corporate network. IPoIB addresses must be resolved only inside the rack's name resolution system. If multiple racks are going to be connected, to elude possible IP conflict, it is good practice to place these also in a central DNS server. Network administrators at the corporate level should enable this. Alternatively hostnames may be resolved through appropriate `/etc/hosts` file propagated through the different nodes. Table 3–3 provides an example of names for the different floating IP addresses used by servers in the SOA system.

Table 3–3 Hostname and Virtual IP Worksheet

Hostname Example for This Guide	Interface	IP Address /Subnet	Customer Value	Type	Host	Bound By	Details
WEBHOST1	bond0	192.168.10.1/255.255.224.0		IPoIB/ Fixed	ComputeNode1/WEBHOST1	NA	Access to ComputeNode1/ WEBHOST1 via the internal IPoIB network.
WEBHOST2	bond0	192.168.10.2/255.255.224.0		IPoIB/ Fixed	ComputeNode2/WEBHOST2	NA	Access to ComputeNode2/ WEBHOST2 via the internal IPoIB network.
IDMHOST1	bond0	192.168.10.3/255.255.224.0		IPoIB/ Fixed	ComputeNode3/IDMHOST1	Node Manager and WLS_OAM1	`BOND0` IP used by Node Manager and OAM running on ComputeNode3.
IDMHOST2	bond0	192.168.10.4/255.255.224.0		IPoIB/ Fixed	ComputeNode4/IDMHOST2	Node Manager and WLS_OAM2	`BOND0` IP used by the Node Manager and OAM running on ComputeNode4.
ADMINVHN	bond1:1	10.10.30.2/255.255.224.0		EoIB /Floating	ComputeNode3/IDMHOST1	Administration Server	A floating IP address for the Administration Server is recommended, if you want to manually migrate the Administration Server from ComputeNode3 to ComputeNode4.
WEBHOST1-VHN1	OTD	10.10.50.1/255.255.224.0		EoIb /Floating	ComputeNode1/WEBHOST1	OTD - Webhost1	A floating IP Address managed by OTD. This is the IP Address to which load balancers will connect.

Table 3–3 (Cont.) Hostname and Virtual IP Worksheet

Hostname Example for This Guide	Interface	IP Address /Subnet	Customer Value	Type	Host	Bound By	Details
WEBHOST2-VHN1	OTD	10.10.50.2/255.255.224.0		EoIb /Floating	ComputeNode2/WEBHOST2	OTD - Webhost2	A floating IP Address managed by OTD. This is the IP Address to which load balancers will connect.
OTDADMIN VHN	bond1:1	10.10.30.1/255.255.224.0		EoIb /Floating	ComputeNode1/WEBHOST1	OTD Administration Server	A floating IP address for the Administration Server is recommended, if you want to manually migrate the OTD Administration Server from ComputeNode1 to ComputeNode2.
SOAHOST1 VHN	bond0:2	192.168.30.3/255.255.240.0		IPoIB/ Floating	ComputeNode3/IDMHOST1	WLS_SOA1 default channel	Initially enabled in ComputeNode3 can be failed over by server migration to ComputeNode4.
SOAHOST2 VHN	bond0:2	192.168.30.4/255.255.240.0		IPoIB/ Floating	ComputeNode4/IDMHOST2	WLS_SOA2 default channel	Initially enabled in ComputeNode4 can be failed over by server migration to ComputeNode3.
OIMHOST1 VHN	bond0:1	192.168.30.1/255.255.240.0		IPoIB/ Floating	ComputeNode3/IDMHOST1	WLS_OIM1 Default Channel	Initially enabled in ComputeNode3 can be failed over by server migration to ComputeNode4.
OIMHOST2 VHN	bond0:1	192.168.30.2/255.255.240.0		IPoIB/ Floating	ComputeNode4/IDMHOST2	WLS_OIM2 Default Channel	Initially enabled in ComputeNode4 can be failed over by server migration to ComputeNode3.
IDMHOST1-EXT	bond1	10.10.10.3/255.255.224.0		EoIB/Fixed	ComputeNode3/IDMHOST1	NA	A fixed IP allowing the compute node to access an external database, or to be accessed via an external Web server.

Table 3–3 (Cont.) Hostname and Virtual IP Worksheet

Hostname Example for This Guide	Interface	IP Address /Subnet	Customer Value	Type	Host	Bound By	Details
IDMHOST2-EXT	bond1	10.10.10.4/255.255.224.0		EoIB/Fixed	ComputeNode 3/IDMHOST2	NA	A fixed IP allowing the compute node to access an external database, or to be accessed via an external Web server.
WEBHOST1-EXT	bond1	10.10.10.1/255.255.240.0		EoIB/Fixed	ComputeNode 1/WEBHOST1	NA	A fixed IP allowing the compute node to access an External Load balancer
WEBHOST2-EXT	bond1	10.10.10.2/255.255.240.0		EoIB/Fixed	ComputeNode 2/WEBHOST2	NA	A fixed IP allowing the compute node to access an External Load balancer
IDMINTERNAL	OTD	192.168.50.1/255.255.224.0		IPoIB/ Floating	ComputeNode 1/WEBHOST1	NA	Oracle Traffic Director failover group for SOA
OUDINTERNAL	OTD	192.168.50.2/255.255.224.0		IPoIB/ Floating	ComputeNode 2/WEBHOST2	NA	Oracle Traffic Director failover group for Oracle Unified Directory

3.4 Configuring Virtual IP Addresses for IPoIB on Each Compute Node

This section provides the following sections:

- Section 3.4.1, "Summary of the Required IPoIB Virtual IP Addresses"
- Section 3.4.2, "Creating the Virtual IP Addresses for the IPoIB Network on IDMHOST1 and IDMHOST2"
- Section 3.4.3, "Verifying the Required Virtual IP Addresses on the IPoIB Network"

3.4.1 Summary of the Required IPoIB Virtual IP Addresses

For all communications over the IPoIB network, the WEBHOST compute nodes and WebLogic Server managed servers use the default `bond0` IP addresses assigned when the Exalogic hardware was commissioned.

Table 3–4 lists the Virtual IPs you must define for the OAM and OIM Managed Servers on IDMHOST1 and IDMHOST2.

For instructions on defining these virtual IP addresses, see Section 3.4.2, "Creating the Virtual IP Addresses for the IPoIB Network on IDMHOST1 and IDMHOST2."

Table 3–4 Virtual IP Addresses Associated with IPoIB Network interfaces

Interface	Address Example	Netmask Example	Used By	Virtual Host Name	Type	Default Host
BOND0:1	192.168.30.1	255.255.240.0	WLS_OIM1	OIMHOST1VHN	Physical	IDMHOST1

Table 3–4 (Cont.) Virtual IP Addresses Associated with IPoIB Network interfaces

Interface	Address Example	Netmask Example	Used By	Virtual Host Name	Type	Default Host
BOND0:1	192.168.30.2	255.255.240.0	WLS_OIM2	OIMHOST2VHN	Physical	IDMHOST2
BOND0:2	192.168.30.3	255.255.240.0	WLS_SOA1	SOAHOST1VHN	Physical	IDMHOST1
BOND0:2	192.168.30.4	255.255.240.0	WLS_SOA2	SOAHOST2VHN	Physical	IDMHOST2
BOND0:1	192.168.50.1	255.255.224.0	OTD Failover group for SOA	IDMINTERNAL	OTD	WEBHOST1
BOND0:1	192.168.50.2	255.255.224.0	OTD Failover group for OUD	OUDINTERNAL	OTD	WEBHOST1

Note: Physical IP addresses are managed manually. Oracle Traffic Director IP Addresses are handled by Oracle Traffic Director.

3.4.2 Creating the Virtual IP Addresses for the IPoIB Network on IDMHOST1 and IDMHOST2

To enable only the physical IP addresses listed in Table 3–4, on IDMHOST1 and IDMHOST2:

1. Use the `ifconfig` command to create the virtual IP address:

   ```
   ifconfig subinterface virtual_ip_address netmask netmask_value
   ```

   For example, on IDMHOST1, enter the following:

   ```
   ifconfig bond0:1 192.168.20.3 netmask 255.255.240.0
   ```

2. For each virtual IP address you define, update the ARP caches using the following command:

   ```
   arping -b -A -c 3 -I bond0 192.168.20.3
   ```

3.4.3 Verifying the Required Virtual IP Addresses on the IPoIB Network

Check that the following commands return a positive result from each of the IDMHOST1, IDMHOST2, WEBHOST1 and WEBHOST2 nodes:

```
ping -I bond0 WEBHOST1 (192.168.10.1)
ping -I bond0 WEBHOST2 (192.168.10.2)
ping -I bond0 IDMHOST1 (192.168.10.3)
ping -I bond0 IDMHOST2 (192.168.10.4)
ping -I bond0 OIMHOST1VHN (192.168.30.1)
ping -I bond0 OIMHOST2VHN (192.168.30.2)
ping -I bond0 SOAHOST1VHN (192.168.30.3)
ping -I bond0 SOAHOST2VHN (192.168.30.4)
```

3.5 Configuring Virtual IP Addresses for EoIB on Each Compute Node

By default, compute nodes are not able to communicate outside of the Exalogic machine rack. In order to do this you must configure the EoIB network for those hosts that are accessed via external hosts or load balancers.

The oracle IAM hosts that require this access are:

- *WEBHOST1* and *WEBHOST2* which interact with an external load balancer.
- *IDMHOST1* and *IDMHOST2* for external database or Oracle HTTP Server access.

  This section contains the following topics:

- Section 3.5.1, "Summary of the IP Addresses for the EoIB Network Interfaces"
- Section 3.5.2, "Step 1 - Gather Information"
- Section 3.5.3, "Step 2 - Create a Virtual LAN"
- Section 3.5.4, "Step 3 - Create Virtual Network Cards"
- Section 3.5.5, "Step 4 - Configure Compute Node Networking and Assign Physical IP Address"
- Section 3.5.6, "Creating the Virtual IP Addresses for the EoIB network"

3.5.1 Summary of the IP Addresses for the EoIB Network Interfaces

Table 3–5 lists the virtual IP addresses you must associate with each EoIB interface on each compute node. Each of these interfaces is shown in Figure 3–1.

Table 3–5 IP Addresses for the EoIB Network and Associated Interfaces

Compute Node	Interface Name	External IP Address	Netmask	Type	Used by
IDMHOST1	BOND1	10.10.10.3	255.255.224.0	Physical	Compute node for external database access
	BOND1:1	10.10.30.2	255.255.224.0	Virtual	Admin Server (ADMINVHN)
IDMHOST2	BOND1	10.10.10.4	255.255.224.0	Physical	Compute node for external database access
WEBHOST1	BOND1	10.10.10.1	255.255.224.0	Physical	Compute node for external load balancer access
	BOND1:1	10.10.30.1	255.255.224.0	Virtual	OTD Admin Server
WEBHOST2	BOND1	10.10.10.2	255.255.224.0	Physical	Compute Node for external load balancer access

Configuring the EoIB network is a multi-stage process:

- Stage 1 - Determine the information required to create the network devices.

- Stage 2 - Create a Virtual LAN (VLAN) on the InfiniBand gateway switches for the compute nodes to communicate.
- Stage 3 - Create Virtual Network Cards on the InfiniBand gateway switches which can be seen by the compute nodes, allowing the compute nodes to utilize the EoIB network.
- Stage 4 - Configure the compute nodes to communicate using the VNICS by assigning IP addresses to them.

3.5.2 Step 1 - Gather Information

The following section describes how to gather the information required to create the VLAN and VNICs. To make things easier, complete the following worksheet as you are progressing:

Table 3–6 VNIC Worksheet

Compute Node	Administrative /External IP Address	Base Lid	GUID	Switch Lid	Switch Name	Connector	Switch GUID	MAC Address
WEBHOST1								
WEBHOST2								
IDMHOST1								
IDMHOST2								

Each compute node is connected to gateway switches, the switches that the compute nodes use must have a VLAN created on them.

> **Note:** Administrative IP is the IP Address of the compute node as configured on the management LAN at the time of commissioning.
>
> The External IP address is the static IP address that you assign to the EoIB interface.

To determine which switches are connected to the compute nodes:

1. Login to the compute node you wish to expose using the root user.

   For example:

   ```
   ssh root@WEBHOST1
   ```

2. Retrieve information about the active links on the InfiniBand framework using the following command:

   ```
   iblinkinfo.pl -R | grep hostname
   ```

   For example:

   ```
   # iblinkinfo.pl -R | grep WEBHOST1

   65   15[  ] ==( 4X 10.0 Gbps Active/  LinkUp)==>    121 2[  ] "el01cn01 EL-C
   192.168.10.3 HCA-1" (Could be 5.0 Gbps)
   64   15[  ] ==( 4X 10.0 Gbps Active/  LinkUp)==>    120 1[  ] "el01cn01 EL-C
   192.168.10.3 HCA-1" (Could be 5.0 Gbps)
   ```

   The first column shows the lid id of each of the gateway switches used. In this example, these are lids `65` and `64`. The number after the `==>` symbol shows the Infiniband Port Base Lid: Make a note of these in Table 3–6, " VNIC Worksheet".

3. Using the `ibswitches` command, determine the names of the gateway switches to which the compute node is connected.

   ```
   #ibswitches

   Switch  : 0x002128548042c0a0 ports 36 "SUN IB QDR GW switch el01gw03" enhanced
   port 0 lid 63 lmc 0
   Switch  : 0x002128547f22c0a0 ports 36 "SUN IB QDR GW switch el01gw02" enhanced
   port 0 lid 6 lmc 0
   Switch  : 0x00212856d0a2c0a0 ports 36 "SUN IB QDR GW switch el01gw04" enhanced
   port 0 lid 65 lmc 0
   Switch  : 0x00212856d162c0a0 ports 36 "SUN IB QDR GW switch el01gw05" enhanced
   port 0 lid 64 lmc 0
   ```

   The example output shows that:

   - `lid 64` is associated with gateway switch `el01gw04`.

   - `lid 65` is associated with gateway switch `el01gw05`.

   The GUID of the switch is the last 16 characters value after the `:`. For example, the GUID of Switch `el101gw04` is `00212856d0a2c0a0`.

   These are the gateway switches that must have a VLAN and VNICs defined. Make a note of these values in the Table 3–6, " VNIC Worksheet".

4. Retrieve information about the InfiniBand configuration using the `ibstat` command.

   ```
   # ibstat

   CA 'mlx4_0'
        CA type: MT26428
        Number of ports: 2
        Firmware version: 2.7.8100
        Hardware version: b0
        Node GUID: 0x0021280001a0a364
        System image GUID: 0x0021280001a0a367
        Port 1:
             State: Active
             Physical state: LinkUp
             Rate: 40
             Base lid: 120
             LMC: 0
   ```

```
          SM lid: 6
          Capability mask: 0x02510868
          Port GUID: 0x0021280001a0a365
          Link layer: IB
     Port 2:
          State: Active
          Physical state: LinkUp
          Rate: 40
          Base lid: 121
          LMC: 0
          SM lid: 6
          Capability mask: 0x02510868
          Port GUID: 0x0021280001a0a366
          Link layer: IB
```

The output shows that the compute node is connected to 2 InfiniBand switches, one for each port. The Base Lid links to the value you obtained in Step 2 above. Make a note of the last 16 characters of each GUID in Table 3–6, " VNIC Worksheet".

You now have the information about the existing networking.

5. Determine the unique MAC address for each of the VNICs you are going to create.

   The MAC address can be derived using the information in the worksheet using the following calculation:

   - The last three octets of the Switch GUID, plus the last three octets of the Internal IP address in hex. For example, the GUID of the switch `el101gw04` is `00212856d162c0a0`. The last three octets are: `a2c0a0`.
   - Separate each octet with a colon (:), for example, `a2:c0:a0`.
   - The internal IP address of the Compute Node WEBHOST1 is: `192.168.10.1`
   - The last three octets are: `168.10.1`. Converted to Hexadecimal and separated by a colon: `a8:0a:01`

   Note: you can determine the last 3 octets of an IP address by issuing the command:

   ```
   IP=<enter-ip-here> && printf '%02X' ${IP//./ }; echo
   ```

   For example:

   ```
   IP=192.168.10.1 && printf '%02X' ${IP//./ }; echo
   ```

   Example output:

   ```
   C0A80A01
   ```

   Therefore, you can derive the MAC address as: `a2:c0:a0:a8:0a:01`

   Make a note of the MAC address in the worksheet.

6. Determine the switch upload connector.

   a. Log in to one of the switches as root.

      For example:

      ```
      ssh root@el101gw04
      ```

b. At the command prompt, run the following:

```
listlinkup | grep Bridge

Bridge-0 Port 0A-ETH-1 (Bridge-0-2) up (Enabled)
Bridge-0 Port 0A-ETH-2 (Bridge-0-2) down (Enabled)
Bridge-0 Port 0A-ETH-3 (Bridge-0-1) down (Enabled)
Bridge-0 Port 0A-ETH-4 (Bridge-0-1) down (Enabled)
Bridge-1 Port 1A-ETH-1 (Bridge-1-2) down (Enabled)
Bridge-1 Port 1A-ETH-2 (Bridge-1-2) down (Enabled)
Bridge-1 Port 1A-ETH-3 (Bridge-1-1) down (Enabled)
Bridge-1 Port 1A-ETH-4 (Bridge-1-1) down (Enabled)
```

Identify the uplinks which can be used in the gateway. Any uplink that has a value of `up` can be used. In the example output, only `OA-ETH-1` is available for use.

Using the examples above, the worksheet entries for WEBHOST1 would look as follows:

Table 3–7 Example Worksheet for WEBHOST1

Compute Node	Administrative /External IP Address	Base Lid	GUID	Switch Lid	Switch Name	Connector	Switch GUID	MAC Address
WEBHOST1	10.168.10.1/10.10.10.1	120	0021280001a0a365	64	el01gw05	0A-ETH-1	00212856d162c0a0	62:C0:A0:A8:0A:01
		121	0021280001a0a366	65	el01gw04	0A-ETH-1	00212856d0a2c0a0	A2:C0:A0:A8:0A:01

3.5.3 Step 2 - Create a Virtual LAN

Create a virtual LAN on each of these switches using the following steps:

1. Log in to the gateway switch that you stored in the worksheet, for example, `el01g04`, as the user `ilom-admin`.

   For example:

   ```
   ssh ilom-admin@el01gw04
   ```

2. Change to the system management framework by entering the following:

   ```
   cd /SYS/Fabric_Mgmt
   ```

   For example:

   ```
   Oracle(R) Integrated Lights Out Manager

   Version ILOM 3.0 r47111

   Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
   -> cd /SYS/Fabric_Mgmt
   ```

3. Launch a restrict shell by entering the `show` command:

   ```
   show
   ```

4. Run the following command to associate a connector with the VLAN that will be used:

   ```
   createvlan connector -vlan 0 -pkey default
   ```

   Where:

   `connector` is the name of the switch interface from the worksheet.

   `vlan` is the number of the Virtual Lan.

   `pkey` is the partition key.

5. Verify the virtual LAN is working using the following command:

   ```
   showvlan
   ```

   Expected output:

   ```
    Connector/LAG  VLN   PKEY
      -----------   ----  ----
      0A-ETH-1      125   ffff
      0A-ETH-1      0     ffff
   ```

6.

7. Repeat once for each switch in the VNIC worksheet.

3.5.4 Step 3 - Create Virtual Network Cards

Create a virtual network card on the switch to allow compute nodes to recognize it as a network card it can use for communication.

You need to create a VNIC for each port on each switch attached to each externally facing compute node. Refer to Table 3–6, " VNIC Worksheet" for details.

To create a VNIC:

1. Login to the gateway switch you stored in the worksheet, for example, `el01g04` as the user `ilom-admin`.

   For example:

   ```
   ssh ilom-admin@el01gw04
   ```

2. Change to the system management framework by entering the following:

   ```
   cd /SYS/Fabric_Mgmt
   ```

   For example:

   ```
   Version ILOM 3.0 r47111

   Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
   -> cd /SYS/Fabric_Mgmt/SYS/Fabric_Mgmt
   ```

3. Launch a restrict shell by entering the `show` command:

   ```
   show
   ```

4. Run the following command to a VNIC:

   ```
   createvnic connector -guid compute_node_port_GUID -mac unique_mac_address -pkey
   default -vlan 0
   ```

Where `connector` is the **Connector** column in the worksheet.

`compute_node_port_GUID` is the **GUID** column in the worksheet.

`unique_mac_address` is the **MAC Address** in the worksheet.

`pkey` and `vlan` are the values you used when you created the VLAN in Section 3.5.3, "Step 2 - Create a Virtual LAN."

For example:

```
createvnic connector -guid 0021280001a0a366 -mac A2:C0:A0:A8:0A:01 -pkey
default -vlan 0
```

5. Verify that the VNIC has been created properly by running the following command:

```
showvnics
```

Example output:

```
ID  STATE    FLG IOA_GUID                NODE                             IID
MAC               VLN PKEY GW
--- -------- --- ----------------------- -------------------------------- ----
----------------- --- ---- --------
94  UP       N   0021280001EFA4BF        el01cn01EL-C 192.168.10.1        0000
A2:C0:A0:A8:0A:01 0   ffff 0A-ETH-1
```

Look for the MAC address of the card created and verify that its status is shown as `up`.

6. Repeat for each interface in the VNIC worksheet

3.5.5 Step 4 - Configure Compute Node Networking and Assign Physical IP Address

Now that Virtual Network cards have been created, configure each compute node so that they can be used. Each compute node will have had two Virtual Network Interface Cards created.

To make configuring the network easier you can use the following worksheet:

Table 3–8 VNIC Worksheet

Compute Node	EoIB IP Address	Netmask	Interface	Network Device	MAC Address	EPORT_ID	IOA_ PORT	Device Name	Interface File
WEBHOST1									
WEBHOST2									
IDMHOST1									
IDMHOST2									

Table 3–8 (Cont.) VNIC Worksheet

Compute Node	EoIB IP Address	Netmask	Interface	Network Device	MAC Address	EPORT_ID	IOA_ PORT	Device Name	Interface File

To configure the network:

1. Log in to the compute node as the root user.

   For example:

   ```
   ssh root@WEBHOST1
   ```

2. On the compute node, run the following command to display the list of VNICs available:

   ```
   mlx4_vnic_info -i
   ```

   This command returns the details of the virtual network cards. Make a note of the following in the worksheet:

   - `Network Device`
   - `MAC Address`
   - `EPORT_ID`
   - The number following the colon (:) of the `IOA_PORT`.

3. Create interface files for the VNICs on the compute node.

   To ensure correct failover behavior, the name of the VNIC interface file and the value of the DEVICE directive in the interface file must not be based on the kernel-assigned `ethX` interface name (`eth4`, `eth5`, and so on). Instead, Oracle recommends that the interface file name and value of the `DEVICE` directive in the interface file be derived from the `EPORT_ID` and `IOA_PORT` values:

   > **Note:** Any other unique naming scheme is also acceptable.

   a. Determine the interface device name using the following convention:

   ```
   ethEPORT_ID_IOA_PORT
   ```

   For example:

   ```
   eth331_1
   ```

   Make a note of the interface device name in the worksheet.

   b. Determine the interface file name using the following convention:

   ```
   ifcfg-DeviceName
   ```

   For example:

   ```
   ifcfg-eth331_1
   ```

   Make a note of the interface file name in the worksheet.

Using the examples above for WEBHOST1, the worksheet entry would look as follows:

Table 3–9 VNIC Worksheet

Cumpute Node	EoIB IP Address	Netmask	Interface	Network Device	MAC Address	EPORT_ID	IOA_PORT	Device Name	Interface File
WEBHOST1	10.10.10.1	255.255.224.0	bond1	eth4	A2:C0:A0:A8:0A:03	331	1	eth331_1	ifcfg-eth331_1
				eth5	62:C0:A0:A8:0A:03	331	2	eth331_2	ifcfg-eth331_2

Note: You can obtain the bond name from the worksheet in table Table 3–4.

Tha MAC address is the value of the MAC address generated in the VNICs worksheet.

---

c. Create the interface file for the first VNIC, `eth4` in the example, by using a text editor, such as VI, and save the file in the following directory:

```
/etc/sysconfig/network-scripts
```

Name the file `ifcfg-eth331_1` (from the worksheet).

This file will have the following contents:

```
DEVICE=eth331_1
BOOTPROTO=none
ONBOOT=yes
HWADDR=a2:c0:a0:a8:0A:03
MASTER=bond1
SLAVE=yes
```

Where:

`DEVICE` the **Derived Name** in the worksheet.

`HWADDR` is the **Mac Address** in the worksheet.

d. Create a second interface file for the remaining network card.

e. Create a bonded Ethernet Card encompassing each of the network devices by creating a file named `ifcfg-`*Interface*, for example:

```
ifcfg-bond1
```

The file will have the following contents:

```
DEVICE=bond1
IPADDR=10.10.10.1
NETMASK=255.255.224.0
BOOTPROTO=none
USERCTL=no
TYPE=Ethernet
ONBOOT=yes
IPV6INIT=no
```

```
BONDING_OPTS="mode=active-backup miimon=100 downdelay=5000 updelay=5000"
GATEWAY=10.10.18.1
MTU=65520
```

Where:

`Device` is the Interface Name.

`IPADDR` is the external IP address being assigned.

`NETMASK` is the netmask of the IP Address.

`GATEWAY` is the IP address of your gateway.

f. Restart networking using the following command:

```
service network restart
```

3.5.6 Creating the Virtual IP Addresses for the EoIB network

Now that the network is created, add virtual IP addresses to the interfaces you created.

To enable each virtual IP address listed in Table 3–4:

1. Use the `ifconfig` command to create the virtual IP address:

   For example, on WEBHOST1, enter the following:

   ```
   ifconfig subinterface virtual_ip_address netmask netmask_value
   ```

   For example, on WEBHOST1, enter the following:

   ```
   ifconfig bond1:1 10.10.30.1 netmask 255.255.224.0
   ```

2. For each virtual IP address you define, update the ARP caches using the following command:

   ```
   arping -b -A -c 3 -I bond1 10.10.30.1
   ```

3.6 Verifying Network Connectivity

Having defined the network connectivity, run the following commands on each node to verify that it is working correctly:

```
ping -I bond1 ADMINVHN
ping -I bond0 SOAHOST1VHN
ping -I bond0 SOAHOST2VHN
ping -I bond0 OIMHOST1VHN
ping -I bond0 OIMHOST2VHN
ping -I bond0 IDMHOST1
ping -I bond0 IDMHOST2
ping -I bond0 WEBHOST1
ping -I bond0 WEBHOST2
ping -I bond1 IDMHOST1-ext
ping -I bond1 IDMHOST2-ext
ping -I bond1 WEBHOST1-ext
ping -I bond1 WEBHOST2-ext
ping -I bond1 OTDADMINVHN
ping -I bond1 DBHOST1
ping -I bond1 DBHOST2
ping -I bond1 IAMDBSCAN
```

3.7 Defining the Required Virtual Server Names

A virtual host is associated with an IP address that is not permanently bound to a server or load balancing appliance. It is always enabled on a server or load balancing appliance but may move between servers/appliances as needed.

The compute nodes on which Oracle Fusion Middleware is running must be able to resolve these virtual server names.

Virtual servers `admin.mycompany.com` and `sso.mycompany.com` should be configured in DNS. Although the others may be configured in DNS, they need not be and can be set up in the local host files of the compute nodes for added security.

3.7.1 Virtual Server Names Required on the Hardware Load Balancer

This section describes the virtual server names required for the load balancer.

This section contains the following topics:

- Section 3.7.1.1, "sso.mycompany.com"
- Section 3.7.1.2, "admin.mycompany.com"

3.7.1.1 sso.mycompany.com

Note the following when defining this virtual server name:

- This virtual server is an EoIB address. It is the virtual name which fronts all Identity Management components, including Oracle Access Management and Oracle Identity Manager.
- This virtual server 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 `https://SSO.mycompany.com:443` and in turn forward these to port 7777 (*OTD_PORT*) on WEBHOST1 and WEBHOST2. All the single sign on enabled protected resources are accessed on this virtual host.
- Configure this virtual server on the hardware load balancer with 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.
- This virtual server is configured on the load balancer and is enabled in DNS.

3.7.1.2 admin.mycompany.com

Note the following when defining this virtual server name:

- This virtual server is an EoIB address. It routes the hardware load balancer requests to Administration console, Enterprise Manager, and the oamconsole servers.
- This virtual server acts as the access point for all internal HTTP traffic that gets directed to the administration services.

  The incoming traffic from clients is non-SSL enabled. Thus, the clients access this service using the address `ADMIN.mycompany.com:80` and in turn forward these to port 7777 (*OTD_PORT*) on WEBHOST1 and WEBHOST2.

The services accessed on this virtual host include the WebLogic Administration Server Console, Oracle Enterprise Manager Fusion Middleware Control, Oracle Authorization Policy Manager, and Oracle Directory Services Manager.

- Configure this virtual server on the hardware load balancer. 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.
- This virtual server is configured on the load balancer and is enabled in DNS.

3.7.2 Virtual Server Names required on Oracle Traffic Director

This section describes the virtual server names required for Oracle Traffic Director.

This section contains the following topics:

- Section 3.7.2.1, "oudinternal.mycompany.com"
- Section 3.7.2.2, "idminternal.mycompany.com"

3.7.2.1 oudinternal.mycompany.com

Note the following about this virtual server name:

- This virtual server is an IPoIB address. It acts as a load balancer, routing requests to the OUD instances.
- This virtual server is defined later in Section 7.7, "Defining the Required Oracle Traffic Director Virtual Servers for an Enterprise Deployment."
- This virtual server acts as the access point for all Identity Store LDAP traffic. The clients access this service using the address `OUDINTERNAL.mycompany.com:1489` for non-SSL.
- Use this virtual server to monitor the heartbeat of the Oracle Unified Directory processes. If an Oracle Unified Directory process stops, the load balancer must continue to route the LDAP traffic to a surviving Oracle Unified Directory instance.
- This virtual server directs traffic to each of the Oracle Unified Directory instances on port 1389 (LDAP_DIR_PORT).
- This virtual server directs traffic received on port 636 (*LDAP_LBR_SSL_PORT*) to each of the Oracle Unified Directory instances on port 1636 *(LDAP_DIR_SSL_PORT*).
- This virtual server is configured using OTD and is resolvable only inside the Exalogic machine rack.

3.7.2.2 idminternal.mycompany.com

Note the following about this virtual server name:

- This virtual server is an IPoIB address. It acts as a load balancer, routing requests to SOA managed servers on IDMHOST1 and IDMHOST2.
- This virtual server is defined later in Section 7.7, "Defining the Required Oracle Traffic Director Virtual Servers for an Enterprise Deployment."
- This virtual server is enabled on Oracle Traffic Director. 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 (*OTD_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.
- Add this virtual server is configured using OTD and is resolvable only inside the Exalogic machine rack

3.8 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.

Configure the Administration Server and the managed servers to listen on different virtual IPs and physical IPs as illustrated in Figure 3–2.

Figure 3–2 IP Addresses and VIP Addresses

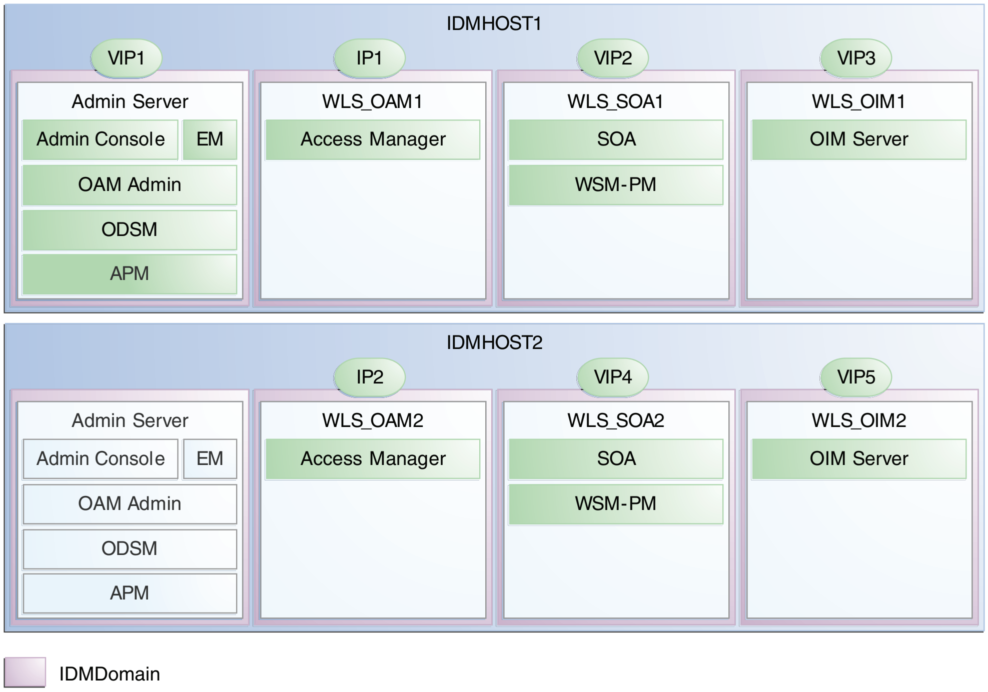


Table 3–10 provides descriptions of the various virtual hosts.

Table 3–10 VIP Addresses and Virtual Hosts

Virtual IP	Virtual Host Name	Network Interface	Description
VIP1	ADMINVHN	EoIB	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).
VIP2	SOAHOST1VHN	IPoIB	SOAHOST1VHN is the virtual host name that maps to the listen address for WLS_SOA1 and fails over with server migration of this managed server. It is enabled on the node where WLS_SOA1 process is running (IDMHOST1 by default).
VIP3	OIMHOST1VHN	IPoIB	OIMHOST1VHN is the virtual host name that maps to the listen address for the WLS_OIM1 server and fails over with server migration of this server. It is enabled in the node where the WLS_OIM1 process us running (IDMHOST1 by default).
VIP4	SOAHOST2VHN	IPoIB	SOAHOST2VHN is the virtual host name that maps to the listen address for WLS_SOA2 and fails over with server migration of this managed server. It is enabled on the node where WLS_SOA2 process is running (IDMHOST2 by default).
VIP5	OIMHOST2VHN	IPoIB	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).

3.9 Configuring the Load Balancer

This enterprise topology uses an external hardware load balancer.

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 physical 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.

Note: Oracle supports most industry-standard load balancers. For a list of load balancers that were supported by previous Oracle middleware software releases, see the following information on the Oracle Technology Network:

`http://www.oracle.com/technetwork/middleware/ias/tested-lbr-fw-sslaccel-100648.html`.

This section contains the following topics:

- Section 3.9.1, "Load Balancer Requirements"

- Section 3.9.2, "Load Balancer Configuration Procedures"
- Section 3.9.3, "Load Balancer Configuration Details"

3.9.1 Load Balancer Requirements

The enterprise topologies use an external load balancer. The 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 URL, 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 fault-tolerant so that if a software or hardware failure occurs in the appliance and alternate failover device can resume operations.
- 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 (this feature is recommended, but not required).
- Configure 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 Management 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

3.9.2 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 (*OTD_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 `https://sso.mycompany.com:443`.
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 Section 3–12, " Ports Used in the Reference Topology". This includes time to detect whether a service is down.

3.9.3 Load Balancer Configuration Details

For an Identity Management deployment, configure your load balancer as shown in Table 3–11.

Table 3–11 Load Balancer Configuration Details

Virtual Host	Server Pool	Protocol	SSL Termination	External	Other Required Configuration/Comments
`SSO.mycompany.com:80`	`WEBHOST1-VHN1.mycompany.com:7777` `WEBHOST2-VHN1.mycompany.com:7777`	HTTP	No	Yes	Identity Management requires that the following be added to the HTTP header: `Header Name: IS_SSL`[1] `Header Value: ssl`

Table 3–11 (Cont.) Load Balancer Configuration Details

Virtual Host	Server Pool	Protocol	SSL Termination	External	Other Required Configuration/Comments

`SSO.mycompany.com:443`	`WEBHOST1-VHN1.mycompany.com:7777`
	`WEBHOST2-VHN1.mycompany.com:7777`
		HTTP	Yes	Yes	Identity Management requires that the following be added to the HTTP header:
					`Header Name: IS_SSL`
					`Header Value: ssl`

`ADMIN.mycompany.com:80`	`WEBHOST1-VHN1.mycompany.com:7777` `WEBHOST2-VHN1.mycompany.com:7777`	HTTP	No	No	

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

3.10 Configuring Firewall 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 during installation.

Table 3–12 lists the ports used in the Oracle Exalogic deployment reference topology, including the ports that you must open on the firewalls in the topology.

Firewall notation:

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

Table 3–12 Ports Used in the Reference Topology

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Browser request	FW0	80	HTTP / Load Balancer	Inbound	Timeout depends on all HTML content and the process models used for the Oracle Fusion Middleware products you are using in the Exalogic environment.
Browser request	FW0	443	HTTPS / Load Balancer	Inbound	Timeout depends on all HTML content and the process models used for the Oracle Fusion Middleware products you are using in the Exalogic environment.
Load balancer to Oracle Traffic Director	n/a	7777 as the example HTTP port for `WEBHOST1` and `WEBHOST2`. 443 as the example HTTPS port for `WEBHOST1` and `WEBHOST2`.	HTTP/HTTPS	n/a	See Section 3.9, "Configuring the Load Balancer." For actual values, see the topic "Port Numbers by Component" in the *Oracle Fusion Middleware Administrator's Guide*.
Administration Console access	FW1	7001	HTTP / Administration Server and Enterprise Manager	Both	You should tune this timeout based on the type of access to the admin console (whether it is planned to use the Oracle WebLogic Server Administration Console from application tier clients or clients external to the application tier).
Administration Console access	FW1	7002	HTTP / Administration Server and Enterprise Manager	Both	You should tune this timeout based on the type of access to the admin console (whether it is planned to use the Oracle WebLogic Server Administration Console from application tier clients or clients external to the application tier). Admin Server SSL Access

Table 3–12 (Cont.) Ports Used in the Reference Topology

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Coherence	n/a	8088 Range: 8080 - 8090		n/a	n/a
Application tier to data tier (Oracle database or RAC outside of Oracle Exalogic machine via Ethernet)	FW2	1521		n/a	n/a
Managed Server Access (WLS_OAM1, WLS_OAM2, WLS_OIM1. WLS_OIM2, WLS_SOA1, WLS_SOA2)	FW1	8001 14000, 14100	HTTP	Inbound	Managed Servers, which use BOND1 floating IP addresses, are accessed via Oracle HTTP Server.

4 Configuring Storage for an Enterprise Deployment

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

The file system 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:

- Overview of Preparing Storage for an Enterprise Deployment
- Terminology for Directories and Directory Variables
- Shared Storage Recommendations for Enterprise Deployments
- Directory Variables for an Oracle Identity Management Enterprise Deployment
- Recommended Directory Locations for an Identity Management Enterprise Deployment
- Configuring Exalogic Storage for Oracle Identity Management
- Allowing Local Root Access to Shares

4.1 Overview of Preparing Storage for an Enterprise Deployment

Before you begin preparing the storage for your enterprise deployment on Exalogic, review the following sections:

- General Information About the Enterprise Deployment File System
- Specific Information About the Exalogic File System

4.1.1 General Information About the Enterprise Deployment File System

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 files system according to information in this chapter. The terminology defined in this chapter is used in diagrams and procedures throughout the guide.

4.1.2 Specific Information About the Exalogic File System

Each Exalogic machine provides an Sun ZFS Storage 7320 appliance that provides extensive storage capabilities for all the compute nodes on the machine. The instructions in this guide assume you will be using the appliance to deploy the enterprise topology on your Exalogic machine.

This guide assumes you have performed the initial hardware setup and configuration steps, and the Sun ZFS Storage 7320 appliance is running and available for use. For more information, see "Configuring the Sun ZFS Storage 7320 appliance" in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*.

4.2 Terminology for Directories and Directory Variables

This section describes the directory variables used throughout this guide for configuring the Exalogic 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 this guide:

- **ORACLE_BASE**: This environment variable and related directory path refers to the base directory under which Oracle products are installed. For example: `/u01/oracle`
- **MW_HOME**: This variable and related directory path refers to the location where Oracle Fusion Middleware resides. A *MW_HOME* has a *WL_HOME*, an *ORACLE_COMMON_HOME* and one or more *ORACLE_HOMEs*. An example of a typical *MW_HOME* is:

  `/u01/oracle/products/access`

  In this guide, this value might be preceded by a product suite abbreviation, for example: *IAM_MW_HOME*, *OIM_MW_HOME*, *WEB_MW_HOME*.
- **WL_HOME**: This variable and related directory path contains installed files necessary to host a WebLogic Server, for example *MW_HOME*`/wlserver_10.3`. The `WL_HOME` directory is a peer of Oracle home directory and resides within the *MW_HOME*.
- **ORACLE_HOME**: This variable points to the location where an Oracle Fusion Middleware product, such as Oracle Traffice Director Server, Oracle SOA Suite, or Oracle Unified Directory is installed and the binaries of that product are being used in a current procedure. In this guide, this value might be preceded by a product suite abbreviation, for example: *IAM_MW_HOME*, *OIM_MW_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 in Section 4.3, "Shared Storage Recommendations for Enterprise Deployments."
- **ORACLE_INSTANCE**: An Oracle instance contains one or more system components, such as Oracle Traffic Director. An Oracle instance directory contains updatable files, such as configuration files, log files, and temporary files. An example is: `/u02/private/oracle/config/instances/web1`

  In this guide, this value might be preceded by a product suite abbreviation, such as `WEB_ORACLE_INSTANCE`.
- **JAVA_HOME**: This is the location where JRockit is installed.

- **ASERVER_HOME**: This is the primary location of the domain configuration. A typical example is: `/u01/oracle/config/domains/`*`domain_name`*
- **MSERVER_HOME**: This is a copy of the domain configuration used to start and stop managed servers. A typical example is: `/u02/private/oracle/config/domains/`*`domain_name`*

4.3 Shared Storage Recommendations for Enterprise Deployments

This section contains the following topics:

- Section 4.3.1, "Shared Storage Recommendations for Binary (Middleware Home) Directories"
- Section 4.3.2, "Shared Storage Recommendations for Domain Configuration Files"
- Section 4.3.3, "Shared Storage Recommendations for JMS File Stores and Transaction Logs"
- Section 4.5, "Recommended Directory Locations for an Identity Management Enterprise Deployment"

4.3.1 Shared Storage Recommendations for Binary (Middleware Home) Directories

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

- Section 4.3.1.1, "About the Binary (Middleware Home) Directories"
- Section 4.3.1.2, "About Using Redundant Binary (Middleware 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 Using Redundant Binary (Middleware Home) Directories

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

In this model, you install two identical Middleware homes for your Oracle Fusion Middleware software on two different shares. 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 shares.

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

4.3.2 Shared Storage 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 4.3.2.1, "About Oracle WebLogic Server Administration and Managed Server Domain Configuration Files"
- Section 4.3.2.2, "Shared Storage Requirements for Administration and Managed Server Domain Configuration Files"

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

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

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

In an enterprise deployment, it is important to understand that the managed servers in a domain can be configured for active-active high availability. However, the Administration Server must be active-passive, meaning that if the active instance fails, the other instance takes over.

4.3.2.2 Shared Storage Requirements for Administration and Managed Server Domain Configuration Files

Oracle recommends creating two copies of the domain configuration files:

- One copy is for the Administration Server configuration files.

  This is known as the ASERVER_HOME directory, and you install this directory on shared storage and mount it exclusively to the host that is running the Administration Server.

  In the event of the failure of that host, you can mount the directory on a different host and the Administration Server started on that host.

- The other copy is for the managed server configuration files.

  This is known as the MSERVER_HOME directory, and it can reside in private or shared storage.

  As a result, the deployment you decide upon should conform to the requirements (if any) of the storage system. Some storage systems offer configuration options to facilitate multiple machines mounting the same shared volume.

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

4.3.3 Shared Storage Recommendations for JMS File Stores and Transaction Logs

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

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

4.4 Directory Variables for an Oracle Identity Management Enterprise Deployment

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. Table 4–1 lists and describes directory variables used to identify the directories installed and configured in the guide.

Note: Figure 4–1, Figure 4–2, and Figure 4–3 also depict the directory variables used to identify the directories installed and configured in this guide.

Table 4–1 Directories and Directory Variables

Variable	Description
ORACLE_BASE	This environment variable and related directory path refers to the base directory under which all Oracle products are installed.
MW_HOME	This variable and related directory path refers to the location where Oracle Fusion Middleware resides. Each MW_HOME has a WL_HOME, an ORACLE_COMMON_HOME and one or more ORACLE_HOME directories. In this guide, this value might be preceded by a product suite abbreviation, for example: IAM_MW_HOME.
WL_HOME	This variable and related directory path contains installed files necessary to host a WebLogic Server.
ORACLE_HOME	This variable points to the location where any Oracle Fusion Middleware product, such as, Oracle SOA Suite, or Oracle Unified Directory is installed and the binaries of that product are being used in a current procedure. In this guide, this value might be preceded by a product suite abbreviation, such as WEB_ORACLE_HOME and IAM_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.
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, "Shared Storage Recommendations for Enterprise Deployments."
ORACLE_INSTANCE	An Oracle instance contains one or more system components, such as Oracle Traffic Director. An Oracle instance directory contains updatable files, such as configuration files, log files, and temporary files. In this guide, this value might be preceded by a product suite abbreviation, such as WEB_ORACLE_INSTANCE.

Table 4–1 (Cont.) Directories and Directory Variables

Variable	Description
JAVA_HOME	This is the location where JDK is installed.
ASERVER_HOME	This is the primary location of the domain configuration where the Administration server is running. It is installed in the *ORACLE_BASE* directory on shared storage.
MSERVER_HOME	This is a copy of the domain configuration used to start and stop managed servers. It is installed in the *ORACLE_BASE* directory on the private storage volume or share.

4.5 Recommended Directory Locations for an Identity Management Enterprise Deployment

This section describes the recommended directory structure for an Identity Management enterprise deployment.

Wherever a shared storage location is directly specified, it is implied that shared storage is required for that directory. When using shared storage is optional, the mount specification is qualified with "if using a shared disk." The shared storage locations are examples and can be changed as long as the provided mount points are used. However, Oracle recommends this structure in the shared storage device for consistency and simplicity.

Note: References to the Web Tier directories and to WEBHOST1 and WEBHOST2 are included here to accommodate the topologies that include installing Oracle Traffic Director on the Exalogic machine.

If you are using remote Oracle HTTP Server instances as your Web tier, then you will be installing the Oracle HTTP Server software and creating the Oracle HTTP Server instances on the private storage for the remote Web Tier host computers, rather than on the Sun ZFS Storage 7320 appliance.

This section includes the following topics:

- Shared Storage for Identity Management Enterprise Deployment on Exalogic
- Private Storage for an Enterprise Deployment

4.5.1 Shared Storage for Identity Management Enterprise Deployment on Exalogic

In an Identity Management Enterprise Deployment on Exalogic, it is recommended that the shares shown in Table 4–2 be created on shared Storage.

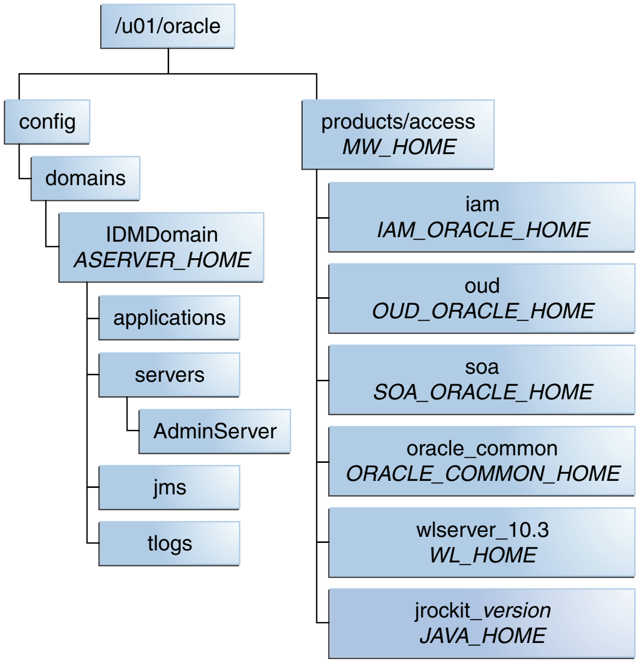
You can mount shared storage either exclusively or shared. If you mount it exclusively, it will be mounted to only one host at a time. (This is typically used for active/passive failover).

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 4–2 Shared Storage Directories

Environment Variable	Mount Point	Mounted on Hosts	Exclusive
MW_HOME	`/u01/oracle/products/access`	IDMHOST1 IDMHOST2	No
ASERVER_HOME	`/u01/oracle/config/`	IDMHOST1 IDMHOST2	Yes

Figure 4–1 Shared Storage for an Identity Management Enterprise Deployment



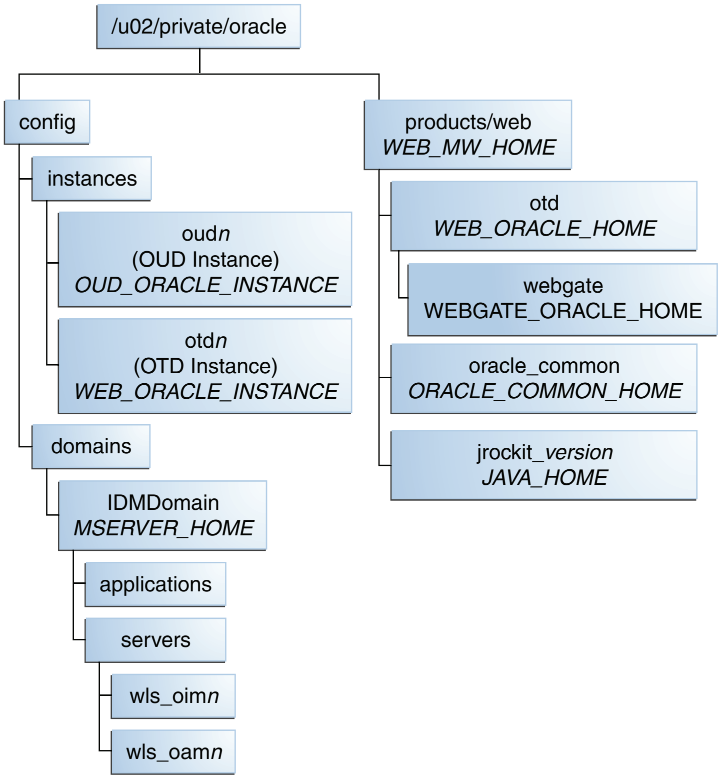
4.5.2 Private Storage for an Enterprise Deployment

Table 4–3 shows the recommended directories to be created on private storage for an enterprise deployment. These directories are not installed on the local disk of the compute node, but instead the mount points are used to point to a specific share on the ZFS file share for each compute node rather than the local physical disk of the compute node.

Table 4–3 Private Storage Directories

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

Figure 4–2 Private Storage for Identity Management Enterprise Deployment



While it is recommended that you put ORACLE_INSTANCE directories onto private storage, you can use shared storage.

4.6 Configuring Exalogic Storage for Oracle Identity Management

The following sections describe how to configure the Sun ZFS Storage 7320 appliance for an enterprise deployment:

- Prerequisite Storage Appliance Configuration Tasks
- Creating the IDM Project Using the Storage Appliance Browser User Interface (BUI)
- Creating the IDM Project Using the Storage Appliance Browser User Interface (BUI)
- Creating the Shares in the IDM Project Using the BUI

4.6.1 Summary of the Storage Appliance Directories and Corresponding Mount Points

For the Oracle Identity Management enterprise topology, you install all software products on the Sun ZFS Storage 7320 appliance, which is a standard hardware storage appliance available with every Exalogic machine. No software is installed on the local storage available for each compute node.

To organize the enterprise deployment software on the appliance, you create a new project, called `IDM`. The shares (`/products` and `/config`) are created within this project on the appliance, so you can later mount the shares to each compute node.

To separate the product binaries from the files specific to each compute node, you create a separate share for each compute node. Sub-directories are for the hostnames are created under `config` and `products` directories. Each private directory is identified by the logical host name; for example, `IDMHOST1` and `IDMHOST2`.

Figure 4–3 shows the recommended physical directory structure on the Sun ZFS Storage 7320 appliance.

Table 4–4 shows how the shares on the appliance map to the mount points you will create on the compute nodes that host the enterprise deployment software.

Figure 4–3 Physical Structure of the Shares on the Sun ZFS Storage Appliance

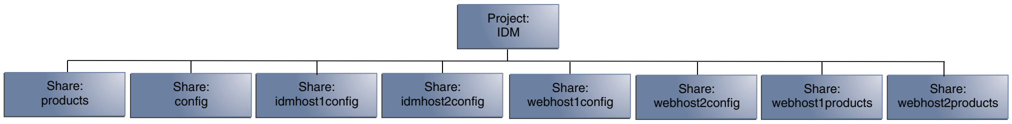


Table 4–4 Mapping the Shares on the Appliance to Mount Points on Each Compute Node

Project	Share	Mount Point	Host	Mounted On
IDM	`products`	`/export/IDM/products`	IDMHOST1/ IDMHOST2	`/u01/oracle/products`
IDM	`config`	`/export/IDM/config`	IDMHOST1/ IDMHOST2	`/u01/oracle/config`
IDM	`idmhost1config`	`/export/IDM/idmhost1config`	IDMHOST1	`/u02/private/oracle/config`
IDM	`idmhost2config`	`/export/IDM/idmhost2config`	IDMHOST2	`/u02/private/oracle/config`
IDM	`webhost1config`	`/export/IDM/webhost1config`	WEBHOST1	`/u02/private/oracle/config`
IDM	`webhost2config`	`/export/IDM/webhost2config`	WEBHOST2	`/u02/private/oracle/config`
IDM	`webhost1products`	`/export/IDM/webhost1products`	WEBHOST1	`/u02/prívate/oracle/products`
IDM	`webhost2products`	`/export/IDM/webhost2products`	WEBHOST2	`/u02/prívate/oracle/products`

4.6.2 Prerequisite Storage Appliance Configuration Tasks

The instructions in this guide assume that the Sun ZFS Storage 7320 appliance is already set up and initially configured. Specifically, it is assumed you have reviewed the following sections in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*:

- "Prerequisites"
- "Getting Started"
- "Sun ZFS Storage 7320 Appliance Overview"
- "Configuration Overview"
- "Naming Service"

4.6.3 Creating the IDM Project Using the Storage Appliance Browser User Interface (BUI)

To configure the appliance for the recommended directory structure, you create a custom project, called `IDM`, using the Sun ZFS Storage 7320 appliance Browser User Interface (BUI).

After you set up and configure the Sun ZFS Storage 7320 appliance, the appliance has a set of default projects and shares. For more information, see "Default Storage Configuration" in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*.

The instructions in this section describe the specific steps for creating a new "IDM" project for the enterprise deployment. For more general information about creating a custom project using the BUI, see "Creating Custom Projects" in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*.

To create a new custom project called IDM on the Sun ZFS Storage 7320 appliance:

1. Direct your browser to the storage system BUI, using either the IP address or host name you assigned to the NET0 port as follows:

   ```
   https://ipaddress:215
   ```

   Or, for example:

   ```
   https://elsn01-priv:215
   ```

2. Log in to the BUI using the storage administrator's user name (root) and password.
3. Navigate to the **Projects** page by clicking on the **Shares** tab, then the **Projects** sub-tab.

   The BUI displays the Project Panel.
4. Click **Add** next to the **Projects** title to display the Create Project window.

   Enter Name: `IDM`

   Click **Apply**.
5. Click **Edit Entry** next to the newly created **IDM** Project.
6. Click the **General** tab on the project page to set project properties.
7. Add `Set Mountpoint` to `/export/IDM`.
8. For the purposes of the enterprise deployment, you can accept the defaults for the remaining project properties.

   For more information about the properties you can set here, see the "Project Settings" table in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*.
9. Click **Apply** on the **General** tab to create the IDM project.

4.6.4 Creating the Shares in the IDM Project Using the BUI

After you have created the IDM project, the next step is to create the required shares within the project.

The instructions in this section describe the specific steps for creating the shares required for an Oracle Identity Management enterprise deployment. For more general information about creating custom shares using the BUI, see "Creating Custom Shares" in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*.

Table 4–5 lists the shares required for all the topologies described in this guide. The table also indicates what privileges are required for each share.

Table 4–5 Shares Required on the Sun ZFS Storage 7320 appliance

Share Name	Privileges to Assign to User, Group, and Other
products	R and W (Read and Write)
config	R and W (Read and Write)
idmhost1config	R and W (Read and Write)
idmhost2config	R and W (Read and Write)
webhost1config	R and W (Read and Write)
webhost2config	R and W (Read and Write)
webhost1products	R and W (Read and Write)
webhost2products	R and W (Read and Write)

> **Note:** The `products` directory can be changed to **read only** after the configuration is complete if desired.

To create each share, use the following instructions, replacing the name and privileges, as described in Table 4–5 :

1. Login to the storage system BUI, using the following URL:

   ```
   https://ipaddress:215
   ```

   For example:

   ```
   https://elsn01-priv:215
   ```

2. Navigate to the Projects page by clicking the **Shares** tab, and then the **Projects** sub-tab.
3. On the Project Panel, click **IDM**.
4. Click the plus (+) button next to **Filesystems** to add a file system.

   The Create Filesystems screen is displayed.
5. In the Create Filesystems screen, choose **IDM** from the **Project** pull-down menu.
6. In the **Name** field, enter the name for the share.

   Refer to Table 4–5 for the name of each share.
7. From the **Data migration source** pull-down menu, choose **None**.
8. Select the **Permissions** option and set the permissions for each share.

   Refer to Table 4–5 for the permissions to assign each share.
9. Select the **Inherit Mountpoint** option.
10. To enforce UTF-8 encoding for all files and directories in the file system, select the **Reject non UTF-8** option.
11. From the **Case sensitivity** pull-down menu, select **Mixed**.
12. From the **Normalization** pull-down menu, select **None**.

13. Click **Apply** to create the share.

Repeat the procedure for each share listed in Table 4–5.

4.7 Allowing Local Root Access to Shares

If you want to run commands or traverse directories on the share as the root user, you must add an NFS exception to allow you to do so. You can create exceptions either at the individual, share, or project level.

To keep things simple, in this example you create the exception at the project level.

To create an exception for NFS at the project level:

1. In the Browser User Interface (BUI), access the Projects user interface by clicking **Configuration**, **STORAGE**, **Shares**, and then **Projects**.

   The Project Panel appears.

2. On the Project Panel, click **Edit** next to the project **IDM**.
3. Select the **Protocols** tab.
4. Click the **+** sign next to NFS exceptions.
5. Select **Type: network**.
6. In the **Entity** field, enter the IP address of the compute node as it appears on the Storage Network (bond0) in CIDR format. For example: 192.168.10.3/19

   ```
   192.168.10.3/19
   ```

7. Set **Access Mode** to **Read/Write** and check **Root Access**.
8. Click **Apply**.
9. Repeat for each compute node that accesses the ZFS appliance.

5 Configuring the Compute Nodes 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 Compute Nodes

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

- The compute nodes 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 Meeting Operating System Requirements

Before starting your operating provisioning you must install a certified operating system.

Note: Be sure to verify you have obtained all required patches. For more info, see Section 2.5.3, "Applying Patches and Workarounds."

5.2.1 Meeting UNIX and Linux Requirements

This section includes the following topics:

5.2.1.1 Configure Kernel Parameters

The kernel parameter and shell limit values shown below are recommended values only. For production database systems, Oracle recommends that you tune these values to optimize the performance of the system. See your operating system documentation for more information about tuning kernel parameters.

Kernel parameters must be set to a minimum of those below on all nodes in the cluster.

The values in Table 5–1 are the current UNIX recommendations. For the latest recommendations for UNIX and other operating systems, see the *Oracle Fusion Middleware System Requirements and Specifications* at the following URL:

`http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-requirements-100147.html`

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
   ```

5.2.1.2 Setting the Open File Limit

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

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

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

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

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

C shell:

```
limit descriptors
```

Bash:

```
ulimit -n
```

5.2.1.3 Setting Shell Limits

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

Add the following lines:

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

After editing the file, reboot the machine.

5.2.1.4 Configuring Local Hosts File

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

```
192.168.30.1 oimhost1vhn.mycompany.com oimhost1vhn

192.168.30.2 oimhost2vhn.mycompany.com oimhost2vhn

192.168.30.3 soahost1vhn.mycompany.com soahost1vhn

192.168.30.4 soahost2vhn.mycompany.com soahost2vhn

192.168.50.1 oudinternal.mycompany.com oudinternal

192.168.50.2 idminternal.mycompany.com idminternal

192.168.10.3 idmhost1vhn.mycompany.com idmhost1vhn

192.168.10.4 idmhost2vhn.mycompany.com idmhost2vhn

192.168.10.1 webhost1vhn.mycompany.com webhost1vhn

192.168.10.2 webhost2vhn.mycompany.com webhost2vhn
```

> **Note:** If `oudinternal.mycompany.com` and `idminternal.mycompany.com` have DNS entries, you do not need to add to the /etc/hosts.

5.2.1.5 Increase Huge Page Allocation

By default huge pages are enabled in Exalogic compute nodes, verify the existing allocation by running.

```
grep Huge /proc/meminfo
```

Set the recommended Huge Page allocation to `25000`.

To set the Huge Page allocation, run the following command as root in the compute node:

```
# echo 25000  > /proc/sys/vm/nr_hugepages
```

5.3 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.

5.4 Configuring an NIS/YP Server

If you are using NFS Version 4, configure a directory service or an NIS (Network Information Server). If your organization does not have one already, use the built-in one on the ZFS storage appliance. See Configuring NFS Version 4 (NFSv4) on Exalogic in the *Oracle Fusion Middleware Exalogic Machine Owner's Guide* for more information.

Once you have configured your NIS server, configure each compute node to use it. If you are using the built-in NIS server on the Exalogic ZFS appliance, use the following steps:

1. Determine the name of the NIS server by logging into the storage BUI using the following URL:

   ```
   https://ipaddress:215
   ```

2. Click **Configuration**, **Services**, and then **NIS**.
3. Make a note of one of the listed NIS servers.
4. Login to the compute node as root.
5. Edit the `/etc/idmapd.conf` configuration file:

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

   Set the domain value, as in the following example:

   ```
   Domain = us.myexample.com
   ```

6. Restart the `rpcidmapd` service:

   ```
   service rpcidmapd restart
   ```

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

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

   Add the following line:

   ```
   domain us.myexample.com server NIS_Server_hostname_or_IP
   ```

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

8. Set NIS domain name on the command line:

```
domainname NIS_DOMAIN_NAME
```

For example:

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

9. Edit the `/etc/nsswitch.conf` configuration file:

   ```
   vi /etc/nsswitch.conf
   ```

   Change the following entries:

   ```
    passwd:     files nis
    shadow:     files nis
    group:      files nis
    automount:  files nis nisplus
    aliases:    files nis nisplus
   ```

10. Restart the `rpcidmapd` service:

    ```
    service rpcidmapd restart
    ```

11. Restart the `ypbind` service by running the following command:

    ```
    service ypbind restart
    ```

12. Check the `yp` service by running this command:

    ```
    ypwhich
    ```

13. Verify if you can access Oracle user accounts:

    ```
    ypcat passwd
    ```

14. Add `ypbind` to your boot sequence, so that it starts automatically after rebooting.

    ```
    chkconfig ypbind on
    ```

5.5 Configuring Users and Groups

Create the following users and groups either locally or in your NIS or LDAP server. This user is the Oracle Software Owner.

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

Groups

You must create the following groups on each node.

- `oinstall`
- `dba`

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

```
groupadd groupname
```

For example

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

Users

You must create the following users on each node.

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

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.
- The user and group should exists at the NIS server due to the NFSv4 mount requirement.

To create users use the following command as root:

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

For example:

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

5.6 Mounting Shares onto the Hosts

Mount the shared storage to the hosts according to the details in Table 5–2.

Table 5–2 Mapping the Shares on the Appliance to Mount Points on Each Compute Node

Volume Mounted	Mounted on Host	Mounted Point	Exclusive
/export/IDM/products	IDMHOST1/ IDMHOST2	/u01/oracle/products	No
/export/IDM/config	IDMHOST1/ IDMHOST2	/u01/oracle/config	No
/export/IDM/configsoahost1	IDMHOST1	/u02/private/oracle/config	Yes
/export/IDM/configsoahost2	IDMHOST2	/u02/private/oracle/config	Yes
/export/IDM/webhost1config	WEBHOST1	/u02/private/oracle/config	Yes
/export/IDM/webhost2config	WEBHOST2	/u02/private/oracle/config	Yes
/export/IDM/webhost1products	WEBHOST1	/u02/prívate/oracle/products	Yes
/export/IDM/webhost2products	WEBHOST2	/u02/prívate/oracle/products	Yes

Note: Each host must have the appropriate privileges set within the SAN.

Mounting the Shares

You must create and mount shared storage locations so that each application tier host can see the same location for the binary installation.

To mount a file system on an Exalogic machine:

1. Create a directory for the mount point for example:

   ```
   mkdir -p /u01/oracle/products
   ```

2. Change the ownership of the directory to the installation user. For example:

   ```
   chown oracle:oinstall /u01/oracle/products
   ```

3. Mount the shared storage onto the host using the following command:

   ```
    mount -t nfs4 -o mount options zfshost:volume_mount_point
   ```

   For example

   ```
   mount -t nfs4 -o rw,bg,hard,nointr,rsize=131072,wsize=131072,proto=tcp
   zfshost:/export/IDM/products /u01/oracle/products
   ```

4. Repeat steps 1 - 3 for each entry in Table 5–2.

   > **Note:** Mounting storage in this way is not persistent. That is the mount will not survive a machine reboot. It is recommended to make the mount persistent that an entry is placed into the file /etc/fstab
   >
   > For example:
   >
   > ```
   > zfshost:/export/IDM/products /u01/oracle/products nfs4
   > auto,rw,bg,hard,nointr,proto=tcp,vers=3,time
   > ```

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
```

6 Configuring a Database for an Enterprise Deployment

This chapter describes how to configure the Identity Management database repositories. The database can exist either on a separate grid infrastructure or on an Exadata server.

This chapter contains the following topics:

- Section 6.1, "Overview of Preparing the Databases for an Identity Management Enterprise Deployment"
- Section 6.2, "Verifying the Database Requirements for an Enterprise Deployment"
- Section 6.3, "Installing the Database for an Enterprise Deployment"
- Section 6.4, "Creating Database Services"
- Section 6.5, "Loading the Identity Management Schemas in the Oracle RAC Database by Using RCU"
- Section 6.6, "Backing up the Database"

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

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

- Verify the database requirements as described in Section 6.2, "Verifying the Database Requirements for an Enterprise Deployment."
- Install and configure the Oracle database repositories. See the installation guides listed in the "Related Documents" section of the Preface and Section 6.3, "Installing the Database for an Enterprise Deployment."
- Create database services, as described in Section 6.4, "Creating Database Services."
- Create the required Oracle schemas in the database using the Repository Creation Utility (RCU). See Section 6.5, "Loading the Identity Management Schemas in the Oracle RAC Database by Using RCU."

6.2 Verifying the Database Requirements for an Enterprise Deployment

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

- Section 6.2.1, "Databases Required"

- Section 6.2.2, "Database Host Requirements"
- Section 6.2.3, "Database Versions Supported"
- Section 6.2.4, "Patching the Oracle Database"
- Section 6.2.5, "About Initialization Parameters"

6.2.1 Databases Required

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

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

Table 6–1 Mapping between Databases and Schemas

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

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

6.2.2 Database Host Requirements

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

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

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

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

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

6.2.3 Database Versions Supported

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

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

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

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

6.2.4 Patching the Oracle Database

Patches are required for some versions of Oracle Database.

6.2.4.1 Patch Requirements for Oracle Database 11g (11.1.0.7)

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

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

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

6.2.4.2 Patch Requirements for Oracle Database 11g (11.2.0.2.0)

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

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

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

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

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

Note:

- Apply this patch in ONLINE mode. Refer to the readme.txt file bundled with the patch for the steps to be followed.
- In some environments, the RDBMS Interim Patch has been unable to resolve the issue, but the published workaround works. Refer to the 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>.

6.2.5 About Initialization Parameters

The databases must have the following minimum initialization parameters defined:

Table 6–4 Minimum Initialization Parameters for Oracle RAC Databases

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

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

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

6.3 Installing the Database for an Enterprise Deployment

Install and configure the database repository as follows.

Oracle Clusterware

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

> **Note:** Be sure to verify you have obtained all required patches. For more info, see Section 2.5.3, "Applying Patches and Workarounds."

6.4 Creating Database Services

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

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

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

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

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

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

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

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

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

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

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

3. Start the service using `srvctl`:

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

6.4.2 Creating Database Services for 11.2.x Databases

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

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

   ```
   srvctl add service -d idmdb -s oamedg.mycompany.com -r idmdb1,idmdb2 -q FALSE
   -m NONE -e NONE -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)

2. Start the Service using `srvctl start service`

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

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

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

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

   ```
   srvctl config service -d idmdb -s oamedg.mycompany.com
   Service name: oamedg.mycompany.com
   Service is enabled
   Server pool: idmdb_oamedg.mycompany.com
   Cardinality: 2
   Disconnect: false
   Service role: PRIMARY
   Management policy: AUTOMATIC
   DTP transaction: false
   AQ HA notifications: false
   Failover type: NONE
   Failover method: NONE
   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*.

6.4.3 Database Tuning

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

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

Run the Repository Creation Utility to create the collection of schemas used by Identity Management and Management Services.

In the Database Connection Details screen, provide the information required to connect to an existing database.

On the Select Components screen, provide the following values:

Create a New Prefix: Enter a prefix to be added to the database schemas. Note that all schemas are required to have a prefix. For example, enter `EDG`. This will allow you to quickly identify the schemas easily when you later configure and extend the Enterprise Deployment domain. In addition, make a note of the password you used for the schemas. You will need this later when you run the Configuration Wizard.

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

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

For more information about the Repository Creation Utility, see *Oracle Fusion Middleware Repository Creation Utility User's Guide*.

For more information about the schemas required for an Identify and Access Management installation, see "Creating Database Schema Using the Oracle Fusion Middleware Repository Creation Utility (RCU)" in the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management*.

6.6 Backing up the Database

After you have prepared your database, back it up. You can back up your database using the appropriate RMAN commands for your environment. See *Oracle Database Backup and Recovery User's Guide*.

7 Installing and Configuring Oracle Traffic Director for an Enterprise Deployment

This chapter describes how to install and configure Oracle Traffic Director for an Exalogic enterprise deployment.

This chapter contains the following sections:

7.1 Overview of Installing and Configuring Oracle Traffic Director for an Enterprise Deployment

Oracle Traffic Director is a software load balancer for load balancing HTTP/S and TCP traffic to servers in the back-end. These back-end servers, which are referred to as origin servers within Oracle Traffic Director, can be application servers, web servers, or LDAP servers.

Installing and configuring Oracle Traffic Director for an enterprise deployment involves performing the steps shown in Table 7–1.

Table 7–1 Overview of Installing and Configuring Oracle Traffic Director for an Enterprise Deployment

Task	Description	More Information
Review Oracle Traffic Director prerequisites.	For example, be sure that you have set up the required virtual IP addresses, that the user account has root permission on the storage appliance, and that you have already created the initial Oracle WebLogic Server domain for the Oracle Identity Management topology.	"Prerequisites" in the *Oracle Traffic Director Installation Guide*
Install the Oracle Traffic Director software on WEBHOST1 and WEBHOST2.	You install the software using the directories and mount points you created in Section 4.6, "Configuring Exalogic Storage for Oracle Identity Management."	Section 7.2, "Installing Oracle Traffic Director on WEBHOST1 and WEBHOST2"
Create and start an Oracle Traffic Director Administration Server.	The Oracle Traffic Director administration server hosts the administration console and command-line interface, through which you can create Oracle Traffic Director configurations, deploy them as instances on administration nodes, and manage the instances.	Section 7.3, "Creating and Starting the Traffic Director Administration Server"
Verify the installation.	Be sure that the installation was successful before you continue configuring the environment.	"Verifying the Installation" in the *Oracle Traffic Director Installation Guide*
Register WEBHOST2 as administration node.	This ensures that Oracle Traffic Director is up and running on both WEBHOST1 and WEBHOST2.	Section 7.4, "Register WEBHOST2 with the Administration Node"
Create a configuration	The configuration should route requests from the Oracle Traffic Director instances to the managed servers in the Oracle WebLogic Server domain you created in Chapter 9, "Creating a Domain for an Enterprise Deployment". The configuration should also define the required origin-server pools to which requests should be routed.	Section 7.5, "Creating a Configuration"
Start the Oracle Traffic Director instances	Start the instances on WEBHOST1 and WEBHOST2, based on the configuration you created earlier in this procedure.	Section 7.6, "Starting the Oracle Traffic Director Instances"
Define the virtual servers.	Define the virtual servers required for accessing the various management tools and login screens for the topology.	Section 7.7, "Defining the Required Oracle Traffic Director Virtual Servers for an Enterprise Deployment"
Create Routes	Adding routes allows a virtual server to direct requests to different server pools depending on what is contained within the URI.	Section 7.8, "Creating Routes"

Table 7–1 (Cont.) Overview of Installing and Configuring Oracle Traffic Director for an Enterprise

Task	Description	More Information
Enable SSL Passthrough for sso.mycompany.com	Perform extra configuration steps to ensure that any application redirects occur correctly.	Section 7.9, "Enabling SSL Passthrough for sso.mycompany.com"
Deploy and test the configuration.	Deploy the configuration and test the virtual server URLs to be sure you have configured the Oracle Traffic Director instances successfully.	Section 7.10, "Deploying the Configuration and Testing the Virtual Server Addresses"
Create an active-passive failover group.	Create a failover group to ensure that requests will continue to be served if WEBHOST1 or WEBHOST2 become unavailable.	Section 7.11, "Creating a Failover Group for Virtual Hosts"

7.2 Installing Oracle Traffic Director on WEBHOST1 and WEBHOST2

This section describes how to install Oracle Traffic Director software.

Note: Be sure that you are not logged in as root user before installing or performing any action on Oracle Traffic Director.

Note: Be sure to verify you have obtained all required patches. For more info, see Section 2.5.3, "Applying Patches and Workarounds."

To install Oracle Traffic Director:

1. Extract the contents of the installer zip file to a directory on WEBHOST1.
2. Change directory to the `Disk1` subdirectory in the directory in which you unzipped the installer.
3. Run the following command:

   ```
   ./runInstaller
   ```

4. On the Welcome Screen click **Next**.
5. On the Software Updates screen, after all the prerequisites have successfully completed, click **Next**.
6. On the Specify Installation Location screen, enter the value of the *WEB_ORACLE_HOME* variable in the **Oracle Home Directory** field

   The recommended directory location for the *WEB_ORACLE_HOME* is listed in Table 4–3, " Private Storage Directories".

   If you need help with any of the other options on the installer screens, click **Help**, or refer to "Installing Oracle Traffic Director in Graphical Mode" in the *Oracle Traffic Director Installation Guide*.

7. On the Installation Summary screen, click **Install**.
8. When the installation is complete, click **Next** on the Installation Progress screen.
9. On the Installation complete screen, click **Finish**.
10. Repeat steps 1 through 9 on WEBHOST2.

7.3 Creating and Starting the Traffic Director Administration Server

After you install Oracle Traffic Director on WEBHOST1 and WEBHOST2, you can then create an Oracle Traffic Director administration server.

For more information, see "Managing the Administration Server" in the *Oracle Traffic Director Administrator's Guide*

To create the Oracle Traffic Director administration server on WEBHOST1 run the **tadm** command from the *WEB_ORACLE_HOME*/bin directory, as follows:

1. On WEBHOST1 enter the following command:

   ```
   WEB_ORACLE_HOME/bin/tadm configure-server --port=8989 --user=otd_admin
   --instance-home=WEB_ORACLE_INSTANCE --host=otdadminvhn
   ```

   Where:

   - *WEB_ORACLE_HOME* the Oracle Home location you entered in the Oracle Traffic Director installer.
   - *WEB_ORACLE_INSTANCE* is the recommended value listed in Table 4–3, " Private Storage Directories".
   - otdadminvhn is the virtual hostname to be used for the Oracle Traffic Director administration server and console.

   For example:

   ```
   WEB_ORACLE_HOME/web/bin/tadm configure-server
   --port=8989 --user=otd_admin
   --instance-home=/u02/private/oracle/config/otdAdm
   --host=otdadminvhn.mycompany.com
   ```

   > **Note:** If you wish to run Oracle Traffic Director as the root user, which is necessary if OTD is to use ports 1024 or lower. Then add the additional parameter to the line above:
   >
   > ```
   > --server-user=root
   > ```
   >
   > Running as root also enables you to start and stop failover groups from within the Oracle Traffic Director administration console.

2. Enter the administrator password.

   You will later use this password to log in to the Oracle Traffic Director administration console.

   A prompt to re-enter the administrator password is displayed, as follows:

   ```
   Please enter admin-user-password again>
   ```

3. Confirm the administrator password by entering it again.

   An Administration Server instance of Oracle Traffic Director is created and deployed on the local host in a directory named admin-server within the WEB_ORACLE_INSTANCE directory that you specified in step 1.

4. Start the Administration Server by running the following command on WEBHOST1:

   ```
   WEB_INSTANCE_HOME/admin-server/bin/startserv
   ```

5. Login to the Administration Server using the following URL:

```
https://OTDADMINVHN:8989
```

   Use the password provided above and verify that you can see the Oracle Traffic Director main page.

7.4 Register WEBHOST2 with the Administration Node

This section assumes you have installed Oracle Traffic Director, started the Administration Server, and verified the installation.

WEBHOST1 and WEBHOST2 have IP over InfiniBand (IPoIB) addresses. For example, 192.168.10.5 and 192.168.10.6.

You can now register WEBHOST2 with the Oracle Traffic Director Administration Server using the **`tadm`** command from the `WEB_ORACLE_HOME/bin` directory, as follows:

1. On the WEBHOST2, run the `configure-server` command to register the host with the remote Administration Server as an administration node.

```
./tadm configure-server --user=otdadmin --port=8989 --host=OTDADMINVHN
--admin-node --node-port=8900 --instance-home=WEB_ORACLE_INSTANCE
--node-host=WEBHOST2
```

   Where:

   - *WEB_ORACLE_HOME* is the path to the Oracle Traffic Director Oracle home on WEBHOST2.
   - *WEB_INSTANCE_HOME* is the recommended directory path listed in Table 4–3, " Private Storage Directories".

   For example:

```
./tadm configure-server --user=admin --port=8989 --host=OTDADMINVHN
--admin-node --node-port=8900
--instance-home=/u02/private/oracle/config/instances/otd2 --node-host=WEBHOST2
```

> **Note:** If you wish to run Oracle Traffic Director as the root user, which is necessary if OTD is to use ports 1024 or lower. Then add the additional parameter to the line above:
>
> `--server-user=root`
>
> Running as root also enables you to start and stop failover groups from within the Oracle Traffic Director administration console.

   For more information, see "configure-server" in the *Oracle Traffic Director Command-Line Reference* or use the `configure-server --help` command to see an explanation of the command line options.

   The following prompt appears after you run `configure-server` command:

```
This command creates an Administration Node and register it with the following
remote Administration Server: https://WEBHOST1.mycompany.com

Enter admin-user password>
```

2. Enter the admin-user password for the Oracle Traffic Director Administration Server.

   The `configure-server` command attempts to connect to the remote administration server by using the specified administration server host, port, user, and password. The Administration Server on WEBHOST1 must be up and running.

   If this is the first time that the host on which you are creating the administration node is attempting to connect to the administration server, the server certificate of the administration server is displayed.

3. Enter `y` to trust the certificate.

   The following message is displayed:

   ```
   OTD-70215 The administration node has been configured successfully.
   The node can be started by executing:
   WEB_ORACLE_INSTANCE/admin-server/bin/startserv
   ```

After you start the administration node, you can create instances of Oracle Traffic Director configurations on the administration node. Note that on each administration node, you can create only one instance of a configuration.

7.5 Creating a Configuration

The next step in installing and configuring Oracle Traffic Director for an enterprise deployment is to create a configuration that will route requests to a server pool that consists of the managed servers in your Oracle WebLogic Server domain.

When creating a new configuration, you are required to provide the host and port information for the origin server, which in turn automatically creates (and names) an origin-server pool called **origin-server-pool-1**. This is the default origin-server pool and this pool can be found when you click the Server Pools option in the administration console. You cannot rename the default origin-server pool.

To create a configuration named IDM by using the administration console:

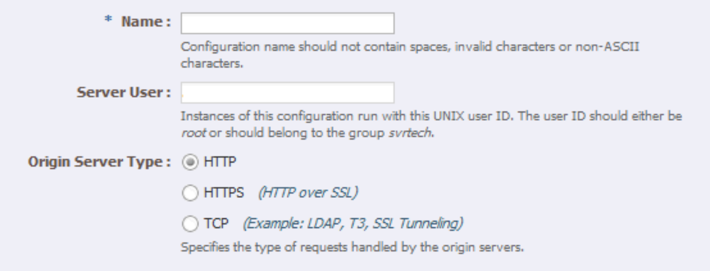
1. Log in to the administration console using the following URL:

   ```
   https://OTDADMINVHN:8989
   ```

2. In the Common Tasks pane, click **New Configuration**.

   The New Configuration wizard starts.

Figure 7–1 New Configuration Wizard



* Name :

Configuration name should not contain spaces, invalid characters or non-ASCII characters.

Server User :

Instances of this configuration run with this UNIX user ID. The user ID should either be *root* or should belong to the group *svrtech*.

Origin Server Type : ● HTTP

○ HTTPS *(HTTP over SSL)*

○ TCP *(Example: LDAP, T3, SSL Tunneling)*

Specifies the type of requests handled by the origin servers.

3. In the Step 1 Configuration Information screen, enter the following information:

   - **Name:** IDM

- **Server User:** oracle (or root if you wish the server instances to run as root)
- **Origin Server Type**: Make sure **HTTP** is selected.

Click **Next**.

4. In the Step 2 Listener Information screen, change the port to 7777. Accept the other default values and click **Next.**
5. In the Step 3 Server Pool Information screen:
   a. In the **Origin Servers: Host:** field, enter `IDMHOST1`, the port `14100`, and click **Add Server**.
   b. Enter `IDMHOST2` and port `14100`, click **Add Server** and click **Next**.
6. In the Step 4 Deployment Information screen, select the **Administration Server** and **WEBHOST2** and click **Next**.

   The Review screen appears.
7. Review the information and click **Create Configuration**.

   The Results screen appears.

   After the configuration is created, the Results screen of the New Configuration wizard displays a message confirming successful creation of the configuration. If you chose to create instances of the configuration, then a message confirming successful creation of the instances is also displayed.
8. Click **Close** on the Results screen.

   In the New Configuration wizard, if you chose not to create an instance of the configuration, the message **Undeployed Configuration** is displayed, indicating that the configuration that you just created is yet to be deployed.

7.6 Starting the Oracle Traffic Director Instances

To start Oracle Traffic Director instances using the administration console:

1. Log in to the administration console using the following URL:

   ```
   https://OTDADMINVHN:8989
   ```
2. Click the **Configurations** button that is situated at the upper left corner of the page.

   A list of the available configurations is displayed.
3. Select the configuration for which you want to start the instance.
4. In the navigation pane, select **Instances**.
5. Click the **Start/Restart** button for the instance that you want to start.

> **Note:** To start or restart *all* instances of the selected configuration, click **Start/Restart Instances** in the Common Tasks pane. To stop all instances of the configuration, click **Stop Instances**.

Starting and Stopping Oracle Traffic Director Administration Instances

In order to access the Oracle Traffic Director Administration Console and the Fusion Middleware Administration Console to be controlled remote OTD instances, start the administration instances.

To start the administration instances:

Run the `startserv` command located in the following directory:

```
WEB_ORACLE_INSTANCE/admin-server/bin
```

To stop the administration services:

Run the `stopserv` command located in the following directory:

```
WEB_ORACLE_INSTANCE/admin-server/bin
```

7.7 Defining the Required Oracle Traffic Director Virtual Servers for an Enterprise Deployment

Create and configure the virtual servers for the Oracle Traffic Director configuration. In this section you create the following Oracle Traffic Director virtual servers for your Oracle Identity and Access Management deployment:

Table 7–2 Defining Virtual Servers

Virtual Server	Purpose	Creating the Virtual Server
sso.mycompany.com	Acts as the access point for all HTTP traffic that gets directed to the single sign on services.	This virtual server is created through administration console in Step 2.
admin.mycompany.com	Acts as the access point for all internal HTTP traffic that gets directed to the administration services.	This virtual server is created through administration console in Step 2.
idminternal.mycompany.com	Acts as the access point for all Identity Store LDAP traffic.	This virtual server is created through administration console in Step 2.
oudinternal.mycompany.com	Acts as a load balancer, routing requests to SOA servers on IDMHOST1 and IDMHOST2.	This virtual server is created when you configure the TCP Proxy for OUD in Step 3.

To create and configure virtual servers using the administration console complete the following steps:

- Step 1, "Creating an Origin-Server Pool"
- Step 2, "Creating Virtual Servers"
- Step 3, "Creating a TCP Proxy and Listener for oudinternal.mycompany.com"

Step 1 Creating an Origin-Server Pool

A server pool is a group of one or more virtualization hosts with the same processor architecture that have access to the same virtual and physical networks, and storage resources. Server pools provide load balancing, high availability capabilities, and sharing of some resources for all members of the pool.

In this section, create the Oracle Traffic Director origin-server pools listed in Table 7–3.

Table 7–3 Origin-Server Pools and Origin Servers

Origin-Server Pool	Origin Server Type	Origin Servers	Port
admin-pool	HTTP	ADMINVHN.mycompany.com	7001

Table 7–3 (Cont.) Origin-Server Pools and Origin Servers

Origin-Server Pool	Origin Server Type	Origin Servers	Port
oud-pool	TCP	IDMHOST1.mycompany.com, IDMHOST2.mycompany.com	1389
oim-pool	HTTP	OIMHOST1VHN.mycompany.com, OIMHOST2VHN.mycompany.com	14000
oam-pool	HTTP	IDMHOST1.mycompany.com, IDMHOST2.mycompany.com	14100
soa-pool	HTTP	SOAHOST1VHN.mycompany.com, SOAHOST2VHN.mycompany.com	8001

To create an origin-server pool:

1. Log in to the Administration Console using the following URL:

   `https://OTDADMINVHN:8989`

2. Click the **Configurations** button that is situated at the upper left corner of the page.

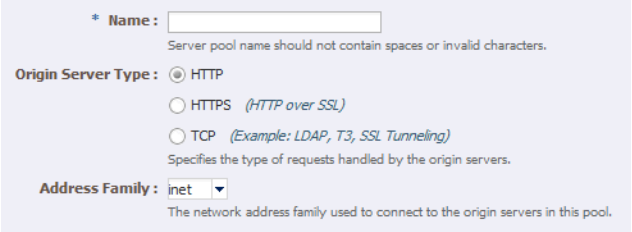
   A list of the available configurations is displayed.

3. Select the configuration for which you want to create a server pool.

4. In the **Common Tasks** pane, click **New Server Pool**.

   The New Origin-Server Pool wizard starts.

Figure 7–2 New Origin-Server Pool Wizard



* Name :

Server pool name should not contain spaces or invalid characters.

Origin Server Type : ● HTTP

○ HTTPS (HTTP over SSL)

○ TCP (Example: LDAP, T3, SSL Tunneling)

Specifies the type of requests handled by the origin servers.

Address Family : inet

The network address family used to connect to the origin servers in this pool.

5. Enter the following information in the Server Pool Information screen:

   - **Name:** Name of the server pool. For example, `oam-pool`
   - **Origin Server Type**: The type of requests the pool handles. For example, `HTTP`.

   Click **Next**.

6. Enter the following information in the Origin Server Information screen:

   - **Origin Server Host**: `IDMHOST1.mycompany.com`
   - **Port**: `14100`

   Click **Add Server**.

7. Enter the information for any other servers. For example:

   - **Origin Server Host**: `IDMHOST2.mycompany.com`
   - **Port**: `14100`

Click **Next**.

Review the information on the Review screen. If the information is correct, click **Create Server Pool**.

8. Repeat steps 4-6 to create each of the server pools listed in table Table 7–3.
9. Click **Close** on the Results screen.
   - The details of the origin-server pool that you just created are displayed on the Origin-Server Pools page.
   - In addition, the **Deployment Pending** message is displayed at the top of the main pane. You can either deploy the updated configuration immediately by clicking **Deploy Changes**, or you can do so later after making further changes as described in Section 7.10, "Deploying the Configuration and Testing the Virtual Server Addresses."

Step 2 Creating Virtual Servers

Create virtual servers using the information in Table 7–4.

Table 7–4 Virtual Server Information

Name	Host	Pool
sso.mycompany.com	sso.mycompany.com	oam-pool
admin.mycompany.com	admin.mycompany.com	admin-pool
idminternal.mycompany.com	idminternal.mycompany.com	oim-pool

To create a virtual server using the administration console:

1. Log in to the administration console using the following URL:

   ```
   https://OTDADMINVHN:8989
   ```

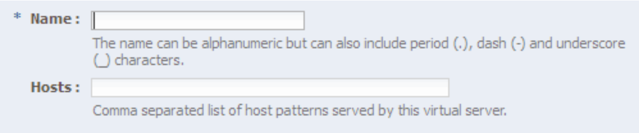
2. Click the **Configurations** button that is situated at the upper left corner of the page.

   A list of the available configurations is displayed.

3. Select the configuration for which you want to create a virtual server.
4. In the Common Tasks pane, click **New Virtual Server**.

   The New Virtual Server wizard starts.

Figure 7–3 New Virtual Server Wizard



* Name :

The name can be alphanumeric but can also include period (.), dash (-) and underscore (_) characters.

Hosts :

Comma separated list of host patterns served by this virtual server.

5. On the Virtual Server Information Page enter the following information:
   - **Name**: The name describing the virtual server. For example, `sso.mycompany.com`
   - **Host**: The name in the DNS/Hosts which is used to access this virtual server. For example, `sso.mycompany.com`

   Click **Next**.

6. Select **HTTP Listener Information**, select listener **7777**, and click **Next**.
7. On the server Pool Information Screen, enter the following information:
   - **Select**: Select a pool of origin servers.
   - **Name**: Select the name of one of the server pools you created in Step 1, "Creating an Origin-Server Pool".

   Click **Next**.
8. Review the supplied information in the Review screen and click **Create Virtual Server**.
9. Repeat steps 4-6 for each virtual server in Table 7–4.

Step 3 Creating a TCP Proxy and Listener for oudinternal.mycompany.com

Create a TCP Proxy using the administration console.

To create a TCP Proxy:

1. Log in to the administration console using the following URL:

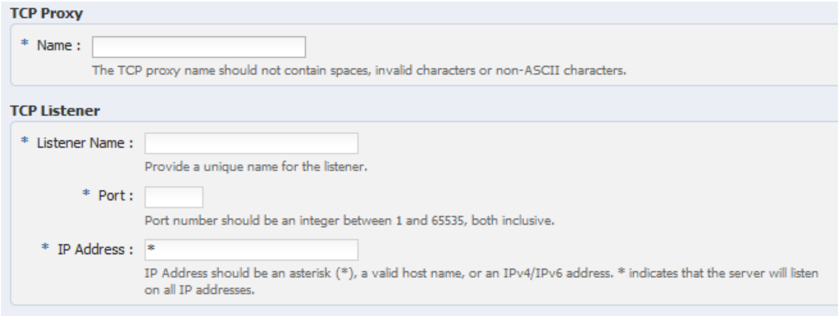
   ```
   https://OTDADMINVHN:8989
   ```

2. Click the **Configurations** button that is situated at the upper left corner of the page.

   A list of the available configurations is displayed.
3. Select the configuration for which you want to create a TCP Proxy.
4. In the Common Tasks pane, click **New TCP Proxy**.

   The New TCP Proxy wizard starts.

Figure 7–4 New TCP Proxy Wizard



TCP Proxy

* Name :

The TCP proxy name should not contain spaces, invalid characters or non-ASCII characters.

TCP Listener

* Listener Name :

Provide a unique name for the listener.

* Port :

Port number should be an integer between 1 and 65535, both inclusive.

* IP Address : *

IP Address should be an asterisk (*), a valid host name, or an IPv4/IPv6 address. * indicates that the server will listen on all IP addresses.

5. In the Step 1: TCP Proxy Information screen, enter the following information and click **Next**:
   - **Name**: `oudinternal.mycompany.com`
   - **Listener Name**: `listener-oud`
   - **Port**: `1489`
   - In the **IP Address** field, enter `*`.
6. In the Step 2: Server Pool Information screen, click **Select a pool of origin servers**.
7. In the drop-down list, select **oud-pool** and click **Next**.

   The Review screen appears.

8. Review the details and click **Create TCP Proxy**.
9. Click **Close** on the Results screen.
   - The details of the TCP Proxies that you just created are displayed on the TCP proxies page.
   - In addition, the **Deployment Pending** message is displayed at the top of the main pane. You can either deploy the updated configuration immediately by clicking **Deploy Changes**, or you can do so later after making further changes, as described in Section 7.10, "Deploying the Configuration and Testing the Virtual Server Addresses."

7.8 Creating Routes

Routes are similar to an Oracle HTTP location directives. Any requests received for a specific URI inside a virtual server are directed to the appropriate server pool. Adding routes allows a virtual server to direct requests to different server pools depending on what is contained within the URI.

Create the routes listed in Table 7–5using the administration console:

Table 7–5 Routes and Conditions

Virtual Host	Route	Origin-Server Pool	Conditions	Cookie Name
admin.mycompany.com	default	admin-pool	N/A	
	oim-admin-route	oim-pool	$uri =~ '/oim' or $uri =~ '/identity' $uri =~ '/sysadmin' or $uri =~ '/xlWebApp' or $uri =~ '/Nexaweb'	oimjsessionid
sso.mycompany.com	default	oam-pool	N/A	OAM_JSESSIONID
	oim-sso-route	oim-pool	$uri =~ '/identity' or $uri =~ '/xlWebApp' or $uri =~ '/HTTPClnt' or $uri =~ '/reqsvc'	oimjsessionid
idminternal.mycompany.com	default	oim-pool	N/A	oimjsessionid
	soa-idminternal-route	soa-pool	$uri =~ '/soa-infra' or $uri =~ '/sodcheck' or $uri =~ '/integration' or $uri =~ '/ucs'	oimjsessionid

To create virtual server routes:

1. Log in to the administration console using the following URL:

   `https://OTDADMINVHN:8989`

2. Click the **Configurations** button that is situated at the upper left corner of the page.

   A list of the available configurations is displayed.

3. Select the configuration for which you want to configure routes.

4. In the navigation pane, expand **Virtual Servers**, expand the **sso.mycompany.com** virtual server, and select **Routes**.

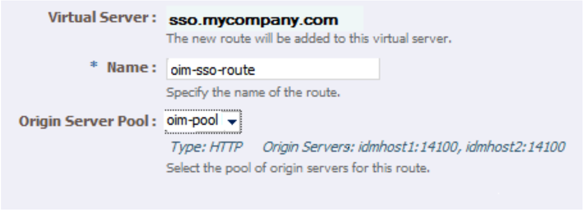
   The Routes page is displayed. It lists the routes that are currently defined for the virtual server.

   Creating a Route

   a. Click **New Route**.

      The New Route dialog box is displayed.

Figure 7–5 New Route Dialog Box



Virtual Server : sso.mycompany.com
The new route will be added to this virtual server.

* Name : oim-sso-route
Specify the name of the route.

Origin Server Pool : oim-pool
Type: HTTP Origin Servers: idmhost1:14100, idmhost2:14100
Select the pool of origin servers for this route.

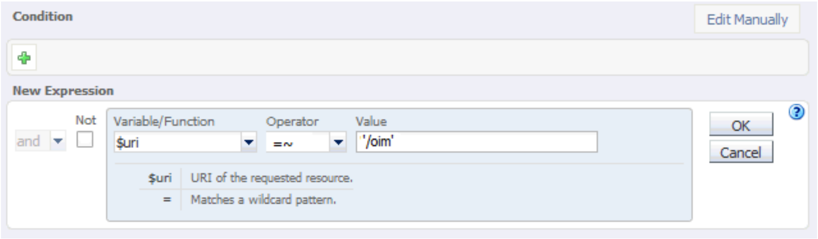
   b. In the Step 1: Route Properties screen, in the **Name** field, enter `oim-sso-route`

   c. In the Origin Server Pool drop-down select `oim-pool`, and click **Next**.

   d. In the Step 2: Condition Information screen, select the **$uri** variable from the **Variable/Function** drop-down list. Select the Operator ('= ~ ' in your example). And enter the value in the **Value** field.

   Note: Joiner, such as `and` or `or`, cannot be used for the first expression in the sequence.

Figure 7–6 New Route Condition Expressions

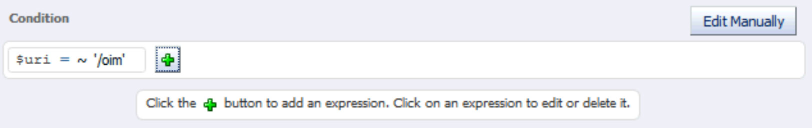


Variable/Function: $uri
Operator: =~
Value: '/oim'

$uri URI of the requested resource.
= Matches a wildcard pattern.

   e. Click **OK** and click the **Plus** button to add the next expression.

Figure 7–7 New Route Condition Information

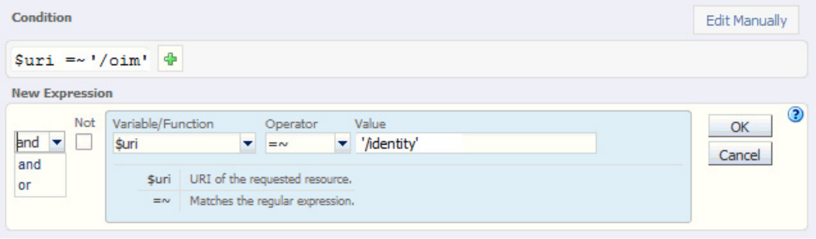


Condition: $uri = ~ '/oim'

Click the ✚ button to add an expression. Click on an expression to edit or delete it.

f. Select the **Variable/Function**, **Operator**, and **Value** and click **OK**.

Figure 7–8 New Route Condition Information



Condition: $uri =~ '/oim'

New Expression: and; Not: □; Variable/Function: $uri; Operator: =~; Value: '/identity'

$uri URI of the requested resource.

=~ Matches the regular expression.

Note the joiner **'or'** can now be selected.

g. Perform steps **d** to **g** until you have added all the required values

You can also click the **Edit Manually** button to edit the expressions in a text field. Note that going into the manual mode, it is not possible to go back to the default edit mode. You must continue in the manual edit mode and save the condition.

5. Click **Next**, and then **Create Route**.

The route that you just created is displayed on the Routes page.

In addition, the **Deployment Pending** message is displayed at the top of the main pane. You can either deploy the updated configuration immediately by clicking **Deploy Changes**, or you can do so later after making further changes as described in Section 7.10, "Deploying the Configuration and Testing the Virtual Server Addresses."

6. Update the cookie name of the newly created route and the default route:

a. Click on the newly created route.

b. Expand the **Advanced Settings**

c. Set **Sticky Cookie** to the cookie name from table Table 7–5.

d. Set the **Sticky URI Parameter** to the cookie name from Table 7–5.

Click **Save**.

7.9 Enabling SSL Passthrough for sso.mycompany.com

In the enterprise deployment, Topology SSL is terminated at the hardware load balancer and passed through to Oracle Traffic Director using the HTTP protocol.

Oracle Traffic Director requires extra configuration steps to ensure that any application redirects occur correctly.

To ensure application redirects occur correctly:

1. Log in to the Administration Console using the following URL:

```
https://OTDADMINVHN:8989
```

2. Click the **Configurations** button at the upper left corner of the page.

   A list of the available configurations is displayed.

3. Select the configuration for which you want to configure routes.
4. In the Navigation Pane, expand **Virtual Servers** and select a virtual server, for example, **sso.mycompany.com**.
5. Click **Routes**.

   The defined routes appear.

6. Click a route, for example, **default-route**.

   The Route Properties screen appears.

7. Expand **Advanced Settings**.
8. In the **Route Properties** section, remove the default value of **Rewrite Headers** (`location,content-location`).
9. In the **Parameters Forwarded to Origin Servers** section, deselect the following:
   - SSL
   - Cipher
   - Key Size
   - Secret Key Size
   - SSL/TLS Session ID
   - Certificate
   - User DN
   - Issuer DN

   Click **Save**.

10. Repeat for each route associated with the virtual server sso.mycompany.com.

7.10 Deploying the Configuration and Testing the Virtual Server Addresses

Deploy the configuration to create an instance of it on an administration node. When you deploy a configuration, the running instances are reconfigured to reflect the configuration changes.

Note: The topology documented in this guide requires the following virtual IP addresses:

- oudinternal.mycompany.com
- idminternal.mycompany.com
- admin.mycompany.com

You can add oudinternal.mycompany.com and idminternal.mycompany.com host entries to resolve them with and internal IP address.

You can register admin.mycompany.com on the DNS.

Deploying a Configuration Using the Administration Console

To deploy a configuration by using the administration console, do the following:

1. Log in to the administration console using the following URL:

   ```
   https://OTDADMINVHN:8989
   ```

2. Click the **Configurations** button at the upper left corner of the page.

   A list of the available configurations is displayed.

3. Select the **IDM** configuration.
4. Click **Deploy**.

   A message is displayed confirming that the updated configuration was successfully deployed.

5. Click **Close**.

7.11 Creating a Failover Group for Virtual Hosts

When a request is sent to one of the virtual hosts `oudinternal.mycompany.com` and `idminternal.mycompany.com` it is directed to the IP address associated with the virtual host name. This IP address is enabled on one of the OTD instances. Move the IP address to an OTD instance that is still available.

Each OTD instance maintains a heart beat with each other OTD instance. If that heartbeat fails then OTD moves active IP addresses on the downed instance to one of the named failover instances. You do this by creating an active-passive failover group for the IP address. This failover group lists a primary and a number of secondary instances.

The enterprise deployment on Exalogic uses the following four failover groups:

- A failover group for distributing internal LDAP requests among the OUD servers.
- A failover group for internal inter-app requests.
- Two failover groups to allow the external load balancer requests among Oracle Traffic Director servers. This failover group is optional, as the load balancer could point to the OTD instances directly. The benefit of using an Oracle Traffic Director failover group is that failures are detected and resolved faster using the failover group resulting in a reduced recovery time from failed servers.

The steps below show you how to create failover groups with the information in Table 7–6.

Table 7–6 Failover Group Details

Virtual IP Address	Router ID	Network Prefix	Primary Node	Primary Network Interface	Secondary Node	Secondary Network Interface
oudinternal.mycomapny.com	50	19	Admin Node	bond0	WEBHOST2	bond0
idminternal.mycompany.com	51	19	WEBHOST2	bond0	Admin Node	bond0
webhost1-vhn1.mycompany.com	52	19	Admin Node	bond1	WEBHOST2	bond1
webhost2-vhn1.mycompany.com	53	19	WEBHOST2	bond1	Admin Node	bond1

Note: The failover groups for the external virtual IP addresses are optional since the load balancer fails over requests between the two Oracle Traffic Director instances. However, they will provide faster failure detection and failover than the typical load balancer monitors.

Note: The router ID is a unique number you assign to the routing. The number must be between 1 and 244.

The Network Prefix is the subnet mask in the CIDR format.

The primary node is the node where the Failover group is initially active.

The Primary Network Interface is the interface on the host where the failover group is bound.

The Secondary Node is the Node on which the failover group can be started if the Primary node is unavailable.

The Secondary Network interface is the Network Interface used on the Secondary node.

To create a failover group by using the administration console, do the following:

1. Log in to the administration console using the following URL:

   ```
   https://OTDADMINVHN:8989
   ```

2. Click the **Configurations** button at the upper left corner of the page.

   A list of the available configurations appears.

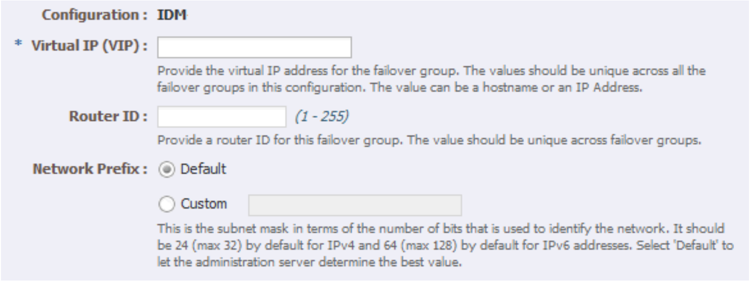
3. Select the configuration for which you want to create a failover group.
4. In the navigation pane, click **Failover Groups**.

   The Failover Groups page is displayed.

5. Click **New Failover Group**.

   The New Failover Group wizard is displayed.

Figure 7–9 New Failover Group Wizard



Configuration : IDM

* Virtual IP (VIP) :

Provide the virtual IP address for the failover group. The values should be unique across all the failover groups in this configuration. The value can be a hostname or an IP Address.

Router ID : (1 - 255)

Provide a router ID for this failover group. The value should be unique across failover groups.

Network Prefix : ● Default

○ Custom

This is the subnet mask in terms of the number of bits that is used to identify the network. It should be 24 (max 32) by default for IPv4 and 64 (max 128) by default for IPv6 addresses. Select 'Default' to let the administration server determine the best value.

6. In the **Virtual IP (VIP)** field, enter the virtual IP address associated with `oudinternal.mycompany.com` (192.168.50.2) and click **Next**.

   To create the failover group for the `idminternal.mycompany.com` use the the VIP associated with the `idminternal.mycompany.com` (192.168.50.1).

7. In the Step 2: Failover Nodes Information screen, select the Primary and Backup nodes, (WEBHOST1, WEBHOST2), and click **Next**.

   The details of the failover group that you just created are displayed on the Failover Groups page.

8. Click **Close** on the Results screen.

   The details of the failover group that you just created are displayed on the Failover Groups page.

Note: A message may be displayed indicating that the failover group could not be started in the involved nodes due to insufficient privileges. To resolve this, log in to each node as root and run the following command:

```
WEB_ORACLE_HOME/bin/tadm start-failover --instance-home=WEB_
INSTANCE_HOME/ --config=IDM
```

7.12 Backing Up the Oracle Traffic Director Configuration

Back up the Oracle Traffic director configuration. For more information, see Section Section 16.6, "Backing Up the Oracle IDM Enterprise Deployment."

8

Installing and Configuring Oracle Unified Directory

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

This chapter includes the following topics:

- Section 8.1, "Overview of Installing and Configuring Oracle Unified Directory"
- Section 8.2, "Prerequisites for Configuring Oracle Unified Directory Instances"
- Section 8.3, "Installing Oracle Unified Directory"
- Section 8.4, "Configuring the Oracle Unified Directory Instances"
- Section 8.5, "Backing Up the Oracle Unified Directory installation"

8.1 Overview of Installing and Configuring Oracle Unified Directory

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

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

8.2 Prerequisites for Configuring Oracle Unified Directory Instances

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

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

Note: Be sure to verify you have obtained all required patches. For more info, see Section 2.5.3, "Applying Patches and Workarounds."

8.3 Installing Oracle Unified Directory

Perform these steps to install Oracle Unified Directory on IDMHOST1 and IDMHOST2.

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

Install the JDK as described in Section 9.2.1.1, "Installing JRockit." To start the Oracle Fusion Middleware 11g Oracle Identity Management Installer, change directory to Disk 1 of the installation media and enter the command:

```
./runInstaller
```

Then proceed as follows:

On the Specify Inventory Directory screen, do the following:

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

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

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

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

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

8.4 Configuring the Oracle Unified Directory Instances

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

This section contains the following topics:

- Section 8.4.1, "Configuring Oracle Unified Directory on IDMHOST1"
- Section 8.4.2, "Validating Oracle Unified Directory on IDMHOST1"
- Section 8.4.3, "Configuring an Additional Oracle Unified Directory Instance on IDMHOST2"
- Section 8.4.4, "Enable Oracle Unified Directory Assured Replication"
- Section 8.4.6, "Validating Oracle Unified Directory on IDMHOST2"

- Section 8.4.7, "Validating the Oracle Unified Directory Virtual IP Address"

8.4.1 Configuring Oracle Unified Directory on IDMHOST1

Use the Oracle Unified Directory Configuration Assistant to configure Oracle Unified Directory.

Ensure that ports 1389 (LDAP_DIR_PORT), 1636 (LDAP_DIR_SSL_PORT), 4444 (LDAP_DIR_ADMIN_PORT), and 8989 (LDAP_DIR_REPL_PORT) are not in use by any service on the computer by issuing these commands for both IDMHOST1 and IDMHOST2. If a port is not in use, no output is returned from the command.

To insure that the ports are open, run the following command:

```
netstat -an | grep "1389"
```

If the ports are in use (that is, if the command returns output identifying either port), free the port.

1. Set the environment variable JAVA_HOME
2. Set the environment variable INSTANCE_NAME to:

   ```
   OUD_ORACLE_INSTANCE
   ```

   For example:

   ```
   ../../../../u02/private/oracle/config/instances/oud2
   ```

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

3. Change Directory to *OUD_ORACLE_HOME*.
4. Start the Oracle Unified Directory configuration assistant by executing the command:

   ```
   ./oud-setup
   ```

5. On the Welcome screen, click **Next**.
6. On the Server Settings screen, enter:
   - **Host Name**: The name of the host where Oracle Unified Directory is running, for example: IDMHOST1.mycompany.com
   - **LDAP Listener Port**: 1389 (*LDAP_DIR_REPL_PORT*)
   - **Administration Connector Port**: 4444 (*LDAP_DIR_ADMIN_PORT*)
   - **LDAP Secure Access**: Click **Configure**
   - In the Security Options page, enter:
     - **SSL Access**: Selected.
     - **Enable SSL on Port**: 1636 (*LDAP_DIR_SSL_PORT*)
     - **Certificate**: Generate Self Signed Certificate OR provide details of your own certificate.
     - Click **OK**

- **Root User DN**: Enter an administrative user for example `cn=oudadmin`
- **Password**: Enter the password you wish to assign to the ouadmin user.
- **Password (Confirm)**: Repeat the password.
- Click **Next**.

7. On the Topology Options screen:

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

   Click **Next**.

8. On the Directory Data screen, enter:

   - **Directory Base DN**: `dc=mycompany,dc=com`
   - **Directory Data**: Only create base entry

   Click **Next**.

9. On the Oracle Components Integration screen, click **Next**.
10. On the Runtime Options screen, click **Next**.
11. On the Review screen, verify that the information displayed is correct and click **Finish**.
12. On the Finished screen, click **Close**.

8.4.2 Validating Oracle Unified Directory on IDMHOST1

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

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

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

8.4.3 Configuring an Additional Oracle Unified Directory Instance on IDMHOST2

Use the Oracle Unified Directory Configuration Assistant to configure Oracle Unified Directory.

Ensure that ports 1389 (LDAP_DIR_PORT), 1636 (LDAP_DIR_SSL_PORT), 4444 (LDAP_DIR_ADMIN_PORT), and 8989 (LDAP_DIR_REPL_PORT) are not in use by any service on the computer by issuing these commands for both IDMHOST1 and IDMHOST2. If a port is not in use, no output is returned from the command.

To insure that the ports are open, run the following command:

```
netstat -an | grep "1389"
```

If the ports are in use (that is, if the command returns output identifying either port), free the port.

1. Set the environment variable JAVA_HOME

2. Set the environment variable INSTANCE_NAME to:

   ```
   OUD_ORACLE_INSTANCE
   ```

   For example:

   ```
   ../../../../u02/private/oracle/config/instances/oud2
   ```

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

3. Change Directory to *`OUD_ORACLE_HOME`*.
4. Start the Oracle Unified Directory configuration assistant by executing the command:

   ```
   ./oud-setup
   ```

5. On the Welcome screen, click **Next**.
6. On the Server Settings screen, enter:
   - **Host Name**: The name of the host where Oracle Unified Directory is running, for example: IDMHOST2
   - **LDAP Listener Port**: 1389 (*LDAP_DIR_PORT*)
   - **Administration Connector Port**: 4444 (*LDAP_DIR_ADMIN_PORT*)
   - LDAP Secure Access
     - Click **Configure**
     - Select **SSL Access**
     - **Enable SSL on Port**: 1636 *(LDAP_DIR_SSL_PORT*)
     - **Certificate**: Generate Self Signed Certificate OR provide details of your own certificate.
     - Click **OK**
   - **Root User DN**: Enter an administrative user for example `cn=oudadmin`
   - **Password**: Enter the password you wish to assign to the ouadmin user.
   - **Password (Confirm)**: Repeat the password.
   - Click **Next**.
7. On the Topology Options screen, enter
   - **This server will be part of a replication topology**
   - **Replication Port:** 8989 (LDAP_DIR_REPL_PORT)
   - Select **Configure As Secure,** if you wish replication traffic to be encrypted.
   - **There is already a server in the topology**: Selected.

     Enter the following:
     - **Host Name**: The name of an existing Oracle Unified Directory server host, for example: IDMHOST1.mycompany.com

  - **Administrator Connector Port**: 4444 (*LDAP_DIR_ADMIN_PORT*)
  - **Admin User**: Name of the Oracle Unified Directory admin user on IDMHOST1, for example: `cn=oudadmin`
  - **Admin Password**: Administrator password.

  Click **Next**.

  If you see a certificate Not Trusted Dialogue, it is because you are using self signed certificates. Click **Accept Permanently.**

  Click **Next**.

8. On The Create Global Administrator screen enter:
   - **Global Administrator ID:** The name of an account you want to use for managing Oracle Unified Directory replication, for example: `oudmanager`
   - **Global Administrator Password / Confirmation**: Enter a password for this account.

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

8.4.4 Enable Oracle Unified Directory Assured Replication

Ensure that data read from every Oracle Unified Directory instance is current. You do this by enabling Oracle Unified Directory Assured Replication in Safe Read Mode, as follows:

1. On IDMHOST1, issue the following command:

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

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

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

Note: `password_file` is a file that contains the OUD administrator password.

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

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

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

8.4.5 Relaxing Oracle Unified Directory Object Creation Restrictions

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

Execute the following command on each Oracle Unified Directory instance:

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

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

8.4.6 Validating Oracle Unified Directory on IDMHOST2

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

```
OUD_ORACLE_INSTANCE/OUD/bin/ldapsearch -h IDMHOST2.mycompany.com -p 1389 -D
cn=oudadmin -b "" -s base "(objectclass=*)" supportedControl
```

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

8.4.7 Validating the Oracle Unified Directory Virtual IP Address

Validate Oracle Unified Directory virtual IP address.

To validate the IP address:

1. On the IDMHOST1, Run the following query:

   ```
   OUD_ORACLE_INSTANCE/OUD/bin/ldapsearch -h oudinternal.mycompany.com -p 1489 -D
   cn=oudadmin -b "" -s base "(objectclass=*)" supportedControl
   ```

2. Stop the OUD Instance on IDMHOST1.

3. Run the same query:

   ```
   OUD_ORACLE_INSTANCE/OUD/bin/ldapsearch -h oudinternal.mycompany.com -p 1489 -D
   cn=oudadmin -b "" -s base "(objectclass=*)" supportedControl
   ```

The query output shows that the OUD instance on IDMHOST2 is serving the request.

4. Stop the OUD instance on IDMHOST2 and start the instance on IDMHOST1.
5. Run the query again to show that OUD is configured correctly on both IDMHOST1 and IDMHOST2.
6. Make sure OUD is started on IDMHOST1 and IDMHOST2.

8.5 Backing Up the Oracle Unified Directory installation

Perform a backup of the Middleware home and of Oracle Unified Directory, as described in Section 16.6, "Backing Up the Oracle IDM Enterprise Deployment."

9 Creating a Domain for an Enterprise Deployment

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

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

This chapter contains the following sections.

- Section 9.1, "Overview of Creating a Domain"
- Section 9.2, "Installing Oracle Fusion Middleware Home"
- Section 9.3, "About Console URLs and Domains"
- Section 9.4, "Running the Configuration Wizard to Create a Domain"
- Section 9.5, "Post-Configuration and Verification Tasks"
- Section 9.6, "Testing Manual Failover the WebLogic Administration Server"
- Section 9.7, "Backing Up the WebLogic Domain"

9.1 Overview of Creating a Domain

Table 9–1 lists the steps for creating a WebLogic domain, including post-configuration tasks.

Table 9–1 Steps for Creating a WebLogic Domain

Step	Description	More Information
Create a WebLogic Domain	Run the Configuration Wizard to create WebLogic domain.	Section 9.4, "Running the Configuration Wizard to Create a Domain"
Post-Configuration and Verification Tasks	Follow the instructions for post-configuration and validation tasks.	Section 9.5, "Post-Configuration and Verification Tasks"
Back Up the Domain	Back up the newly configured WebLogic domain.	Section 9.7, "Backing Up the WebLogic Domain"

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

9.2 Installing Oracle Fusion Middleware Home

As described in Section 4.3, "Shared Storage Recommendations for Enterprise Deployments" you install Oracle Fusion Middleware software in at least two storage locations for redundancy.

> **Note:** Be sure to verify you have obtained all required patches. For more info, see Section 2.5.3, "Applying Patches and Workarounds."

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

1. Oracle WebLogic Server: Section 9.2.1, "Installing Oracle WebLogic Server and Creating the Fusion Middleware Home"
2. One or more of the Oracle Fusion Middleware components
   a. Section 9.2.2, "Installing Oracle Identity and Access Management"
   b. Section 9.2.3, "Installing the Oracle SOA Suite"
3. Oracle Fusion Middleware for Identity Management

9.2.1 Installing Oracle WebLogic Server and Creating the Fusion Middleware Home

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

> **Note:** Be sure to verify you have obtained all required patches. For more info, see Section 2.5.3, "Applying Patches and Workarounds."

9.2.1.1 Installing JRockit

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

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

2. Add execute permissions to JRockit. For example:

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

3. Start the JRockit installer by issuing the command:

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

   For example:

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

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

9.2.1.2 Installing WebLogic Server Using the Generic Installer

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

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

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

   ```
   java -version
   ```

   Ensure that the 64-bit version is displayed if you are using a 64-bit operating system.
4. Start the WebLogic installer using the appropriate command:

   64-Bit Operating System

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

   32-Bit Operating System

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

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

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

   `/u01/oracle/products/access`

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

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

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

   > **Note:** Oracle WebLogic Server and Oracle Coherence are installed.

10. On the JDK Selection screen, select the JRockit JDK that you installed earlier. It should be listed by default.
11. On the Choose Product Installation Directories screen, accept the following:
    - **Middleware Home Directory**: *`IAM_MW_HOME`*
    - **Product Installation Directories for WebLogic Server**: *`IAM_MW_HOME`*`/wlserver_10.3`
    - **Oracle Coherence**: *`IAM_MW_HOME`*`/wlserver_10.3/coherence_3.6`

    Click **Next**.
12. On the Installation Summary screen, click **Next** to start the install process

13. On the Installation complete screen, deselect **Run Quickstart**.
14. Click **Done** to exit the WebLogic Server Installer.

9.2.2 Installing Oracle Identity and Access Management

Oracle Identity and Access Management includes the following products:

- Oracle Access Management Access Manager
- Oracle Identity Manager

> **Note:** Be sure to verify you have obtained all required patches. For more info, see Section 2.5.3, "Applying Patches and Workarounds."

Perform the steps in this section to install Oracle Identity and Access Management on the hosts identified in Section 2.5, "Software Components for an Enterprise Deployment."

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

To start the Oracle Fusion Middleware 11*g* Installer for Oracle Identity and Access Management, change directory to Disk 1 of the installation media and enter the command:

```
./runInstaller
```

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

```
IAM_MW_HOME/jrockit_version
```

Then perform these installation steps:

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

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

   A dialog box appears with the following message:

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

   Log in as `root` and run:

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

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

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

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

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

   Click **Next**.

3. On the Welcome screen click **Next**.
4. On the Prerequisite Checks screen, verify that the checks complete successfully, then click **Next**.
5. On the Specify Installation Location screen, enter the following values:

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

   Click **Next**.

6. On the Application Server Screen select **WebLogic Server** and click **Next**.
7. On the Installation Summary screen, click **Install**.
8. On the Installation Progress screen, click **Next**.
9. On the Installation Complete screen, click **Finish**.

9.2.3 Installing the Oracle SOA Suite

This section describes how to install Oracle SOA Suite.

Note: Be sure to verify you have obtained all required patches. For more info, see Section 2.5.3, "Applying Patches and Workarounds."

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

To start the Oracle Fusion Middleware 11*g* SOA Suite Installer, change directory to Disk 1 of the installation media and enter the appropriate command.

```
./runInstaller
```

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

```
IAM_MW_HOME/jrockit_version
```

Then perform these installation steps:

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

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

   A dialog box appears with the following message:

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

   Log in as `root` and run:

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

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

   > **Note:** The Oracle Inventory screen is not shown if an Oracle product was previously installed on the host. If the Oracle Inventory screen is not displayed for this installation, check the following:
   >
   > 1. The `/etc/oraInst.loc` file exists.
   > 2. The Inventory directory listed is valid.
   > 3. The user performing the installation has write permissions for the Inventory directory.

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

   Click **Next**.
4. On the Prerequisite Checks screen, verify that the checks complete successfully, and then click **Next**.
5. On the Specify Installation Location screen, enter the following values:

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

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

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

   Click **Next**.
8. On the Installation Summary screen, click **Install**.

9. On the Installation Process screen, click **Next**.
10. On the Installation Complete screen, click **Finish**.

9.3 About Console URLs and Domains

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

Table 9–2 URLs Available After Web Tier Integration

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

9.4 Running the Configuration Wizard to Create a Domain

Run the WebLogic Configuration Wizard on IDMHOST1. In later chapters you will extend these domains to include the components of your topology.

To create a domain:

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

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

3. Start the Oracle Fusion Middleware Configuration Wizard:

   ```
   ./config.sh
   ```

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

   Click **Next**.
6. On the Specify Domain Name and Location screen, enter
   - **Domain name**: `IDMDomain`
   - **Domain location**: `ORACLE_BASE/config/domains`
   - **Application location**: `ASERVER_HOME/applications`

   Ensure that the domain directory matches the directory and shared storage mount point recommended in Section 4.3, "Shared Storage Recommendations for Enterprise Deployments."

Click **Next**.

7. On the Configure Administrator Username and Password screen, enter the username (default is `weblogic`) and password to be used for the domain's administrator. For example:

   - **Name**: `weblogic`
   - **User Password**: *`password for weblogic user`*
   - **Confirm User Password**: *`password for weblogic user`*
   - **Description**:`This user is the default administrator.`

   Click **Next**.

8. On the Configure Server Start Mode and JDK screen, do the following:

   - For WebLogic Domain Startup Mode, select **Production Mode**.
   - For JDK Selection, select **JRockit SDK**

   Click **Next**.

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

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

   - **OPSS Schema**

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

   Click **Next**.

10. The Gridlink RAC Component Schema screen appears. In this screen, enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU.

    - **Driver**: Select **Oracle's driver (Thin) for GridLink Connections,Versions:10 and later**.
    - Select **Enable FAN**.

    Do one of the following:

    - If **SSL** is not selected for ONS notifications to be encrypted, deselect **SSL**.
    - Select **SSL** and provide the appropriate wallet and wallet password.
    - **Service Listener**: Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the parameter `remote_listener` in the database:

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

Note:

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

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

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

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

Enter the following RAC component schema information:

Table 9–3 RAC Component Schema Information

Schema Name	Service Name	User Name	Password
OPSS Schema	oesedg.mycompany.com	EDG_OPSS	*password*

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

Click **Next**.

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

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

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

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

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

    Click **Next**.

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

    - **Name**: `AdminServer`

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

Click **Next**.

15. On the Configure Managed Servers screen, click **Next**.
16. On the Configure Clusters screen, click **Next**.
17. On the Configure Machines screen, click the **Unix Machine** tab and then click **Add** to add the following machine. The machine name does not need to be a valid host name or listen address, it is just a unique identifier of a node manager location:
    - **Name**: `ADMINHOST`
    - **Node manager listen address**: `ADMINVHN.mycompany.com`
18. Click **Next**.
19. On the Assign Servers to Machines screen, assign servers to machines as follows:
    - ***ADMINHOST***: **AdminServer**

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

    `ADMINVHN.mycompany.com`

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

9.5 Post-Configuration and Verification Tasks

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

This section includes the following topics:

- Section 9.5.1, "Creating boot.properties for the WebLogic Administration Servers"
- Section 9.5.2, "Associate the Domain with the Existing OPSS Policy Store"
- Section 9.5.3, "Starting Node Manager on IDMHOST1 and IDMHOST2"
- Section 9.5.4, "Updating the Node Manager Credentials"
- Section 9.5.5, "Enabling Exalogic Optimizations"
- Section 9.5.6, "Enabling WebLogic Plug-in"
- Section 9.5.7, "Validating the WebLogic Administration Server"
- Section 9.5.8, "Disabling Host Name Verification for the Oracle WebLogic Administration Server"
- Section 9.5.9, "Stopping and Starting the WebLogic Administration Server"

9.5.1 Creating boot.properties for the WebLogic Administration Servers

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

For each Administration Server:

1. Create the following directory structure.

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

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

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

3. Save the file and close the editor.

> **Note:** The username and password entries in the file are not encrypted until you start the Administration Server, as described in Section 9.5.4, "Updating the Node Manager Credentials." For security reasons, minimize the time the entries in the file are left unencrypted. After you edit the file, start the server as soon as possible so that the entries are encrypted.

9.5.2 Associate the Domain with the Existing OPSS Policy Store

Before starting your domain for the first time, you must associate the domain with the OPSS policy store in the database.

Before re-association, back up the following configuration files:

- *ASERVER_HOME*/config/config.xml
- *ASERVER_HOME*/config/fmwconfig/jps-config.xml
- *ASERVER_HOME*/config/fmwconfig/system-jazn-data.xml

Back up the boot.properties file for the Administration Server in the following directory:

ASERVER_HOME/servers/AdminServer/security

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

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

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

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

9.5.3 Starting Node Manager on IDMHOST1 and IDMHOST2

Perform these steps to start Node Manager on IDMHOST1 and IDMHOST2:

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

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

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

3. Stop the Node Manager by killing the Node Manager process.
4. Start Node Manager by running the `startNodeManager.sh` script located under the *`IAM_MW_HOME`*`/wlserver_10.3/server/bin` directory.

9.5.4 Updating the Node Manager Credentials

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

Setting the memory parameters is required only for the first start operation.

Setting Memory Parameters

To edit the `setDomainEnv.sh` file to change memory allocation setting:

1. Open the `setDomainEnv.sh` file located in the following directory using a text editor:

   ```
   /u01/oracle/config/domains/IDMDomain/bin
   ```

2. Change the following memory allocation:

   ```
   WLS_MEM_ARGS_64BIT="-Xms512m -Xmx512m
   ```

   To

   ```
   1024m and 3072m
   ```

   For example:

   ```
   WLS_MEM_ARGS_64BIT="-Xms1024m -Xmx3072m"
   ```

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

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

Updating Node Manager Credentials

Use the Administration Console to update the Node Manager credentials on the IDM domain.

To update Node Manager's credentials:

1. Log into the administration console.
   a. In a browser, go to the listen address for the domain. For example:

      `http://ADMINVHN.mycompany.com:7001/console` where `7001` is *`WLS_ADMIN_PORT`*, as described in Section A.3, "Port Mapping."
   b. Log in as the administrator.
   c. Click **Lock and Edit**.
   d. Click ***domain_name***.
   e. Select **Security** tab then **General** tab.
   f. Expand **Advanced Options**.
   g. Enter a new username for Node Manager or make a note of the existing one and update the Node Manager password.
   h. Click **Save**.
   i. Click **Activate Changes**.
2. Stop the WebLogic Administration Server by issuing the command `stopWebLogic.sh` located under the *`ASERVER_HOME`*`/bin` directory.
3. Start WLST and connect to the Node Manager with `nmConnect` and the credentials you just updated. Then start the WebLogic Administration Server using `nmStart`.

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

   Once in the WLST shell, execute the following commands:

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

   where *`Port`* is *`NMGR_PORT`* in Section A.3, "Port Mapping.", *`domain_name`* is the name of the domain and *`Admin_User`* and *`Admin_Password`* are the Node Manager username and password. For example:

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

9.5.5 Enabling Exalogic Optimizations

Perform these steps to enable Exalogic optimizations:

1. Log in to the Oracle WebLogic Server Administration Console.
2. Select **IDMDomain** in the left navigation pane.
3. Click **Lock & Edit**.
4. On the Settings page, click the **General** tab.
5. Select **Enable Exalogic Optimizations**, and click **Save and Activate Changes**.
6. Restart the Administration server.

9.5.6 Enabling WebLogic Plug-in

In an enterprise deployment, Oracle WebLogic Server is fronted by a Web server. The Web server is, in turn, fronted by a load balancer, which performs SSL translation. In order for internal loopback URLs to be generated with the `https` prefix, Oracle WebLogic Server must be informed that it receives requests through a Proxy Web Server.

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

To do this perform the following steps:

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

9.5.7 Validating the WebLogic Administration Server

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

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

   `http://ADMINVHN.mycompany.com:7001/console`, where `7001` is *`WLS_ADMIN_PORT`*, as described in Section A.3, "Port Mapping."
2. Log in as the WebLogic administrator that you created in `boot.properties` file, for example: `weblogic`.
3. Check that you can access Oracle Enterprise Manager Fusion Middleware Control at `http://ADMINVHN.mycompany.com:7001/em`.
4. Log in to Oracle Enterprise Manager Fusion Middleware Control as the WebLogic administrator, for example: `weblogic`.

9.5.8 Disabling Host Name Verification for the Oracle WebLogic Administration Server

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

Perform these steps to disable host name verification:

1. Go to the Oracle WebLogic Server Administration Console at: `http://ADMINVHN.mycompany.com:7001/console`

2. Log in as the user `weblogic`, using the password you specified during the installation.
3. Click **Lock** and **Edit**.
4. Expand the Environment node in the Domain Structure window.
5. Click **Servers**. The Summary of Servers page appears.
6. Select **AdminServer(admin)** in the **Name** column of the table. The Settings page for AdminServer(admin) appears.
7. Click the **SSL** tab.
8. Click **Advanced**.
9. Set Hostname Verification to **None**, if it is not already set.
10. Click **Save**.
11. Click **Activate Changes**.

9.5.9 Stopping and Starting the WebLogic Administration Server

Stop the Administration Server as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

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

9.6 Testing Manual Failover the WebLogic Administration Server

Test failover of the Administration Server to IDMHOST2 and then back to IDMHOST1, as described in Section 16.9, "Manually Failing Over the WebLogic Administration Server."

9.7 Backing Up the WebLogic Domain

Back up the Middleware home, the database and the WebLogic domain as described in Section 16.6, "Backing Up the Oracle IDM Enterprise Deployment."

10

Preparing Identity Stores

This chapter describes how to prepare the Identity and Policy Stores in an Oracle Identity Management enterprise deployment.

It contains the following sections:

- Section 10.1, "Overview of Preparing Identity Stores"
- Section 10.2, "Backing up the LDAP Directories"
- Section 10.3, "Prerequisites"
- Section 10.4, "Preparing the Identity Store"

10.1 Overview of Preparing Identity Stores

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

10.2 Backing up the LDAP Directories

The procedures described in this chapter change the configuration of the LDAP directories that host the Identity Store. Before performing any of these tasks, back up your LDAP directories. Refer to *WebLogic Server Managing Server Startup and Shutdown* for detailed LDAP backup procedures.

10.3 Prerequisites

Before proceeding, ensure that Oracle Identity Management 11*g* is installed on IDMHOST1.

Note: Be sure to verify you have obtained all required patches. For more info, see Section 2.5.3, "Applying Patches and Workarounds."

10.4 Preparing the Identity Store

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

- Section 10.4.1, "Overview of Preparing the Identity Store"
- Section 10.4.2, "Creating the Configuration File"

- Section 10.4.3, "Configuring Oracle Unified Directory for Use with Oracle Access Manager and Oracle Identity Manager"
- Section 10.4.4, "Creating Users and Groups"
- Section 10.4.6, "Granting Oracle Unified Directory Change Log Access"
- Section 10.4.5, "Add Missing Oracle Unified Directory Permission"
- Section 10.4.7, "Creating Oracle Unified Directory Indexes"

10.4.1 Overview of Preparing the Identity Store

Before you can use a directory to support Access Manager, you must extend the directory to include Object classes required by these applications.

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

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

10.4.2 Creating the Configuration File

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

Oracle Unified Directory Example

```
# Common
IDSTORE_HOST: IDMHOST1.mycompany.com
IDSTORE_PORT: 1389
IDSTORE_ADMIN_PORT: 4444
IDSTORE_KEYSTORE_FILE: OUD_ORACLE_INSTANCE/OUD/config/admin-keystore
IDSTORE_KEYSTORE_PASSWORD: Password key
IDSTORE_BINDDN: cn=oudadmin
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=mycompany,dc=com
IDSTORE_SEARCHBASE: dc=mycompany,dc=com
IDSTORE_USERNAMEATTRIBUTE: cn
IDSTORE_LOGINATTRIBUTE: uid
IDSTORE_USERSEARCHBASE: cn=Users,dc=mycompany,dc=com
OUDINTERNAL_NEW_SETUP: true
POLICYSTORE_SHARES_oudinternal: true
# OAM
OUDINTERNAL_OAMADMINUSER:oamadmin
OUDINTERNAL_OAMSOFTWAREUSER:oamLDAP
OAM11G_OUDINTERNAL_ROLE_SECURITY_ADMIN:OAMAdministrators
# OAM and OIM
OUDINTERNAL_SYSTEMIDBASE: cn=systemids,dc=mycompany,dc=com
# OIM
OUDINTERNAL_OIMADMINGROUP: OIMAdministrators
OUDINTERNAL_OIMADMINUSER: oimLDAP
# WebLogic
OUDINTERNAL_WLSADMINUSER : weblogic_idm
OUDINTERNAL_WLSADMINGROUP : WLSAdmins
```

Where:

- `OUDINTERNAL_HOST` and `OUDINTERNAL_PORT` are, respectively, the host and port of your Identity Store directory. Specify the back end directory here. In the case of OUD, specify, respectively, Oracle Unified Directory instances, for example:

OUD: IDMHOST1 and 1389

- OUDINTERNAL_ADMIN_PORT is the administration port of your Oracle Unified Directory instance. If you are not using Oracle Unified Directory, you can leave out this parameter.
- OUDINTERNAL_KEYSTORE_FILE is the location of the Oracle Unified Directory Keystore file. It is used to enable communication with Oracle Unified Directory using the Oracle Unified Directory administration port. It is called admin-keystore and is located in OUD_ORACLE_INSTANCE/OUD/config. If you are not using Oracle Unified Directory, you can leave out this parameter.
- OUDINTERNAL_KEYSTORE_PASSWORD is the encrypted password of the Oracle Unified Directory keystore. This value can be found in the file OUD_ORACLE_INSTANCE/OUD/config/admin-keystore.pin. If you are not using Oracle Unified Directory, you can leave out this parameter.
- OUDINTERNAL_BINDDN is an administrative user in the Identity Store Directory
- OUDINTERNAL_GROUPSEARCHBASE is the location in the directory where Groups are Stored.
- OUDINTERNAL_SEARCHBASE is the location in the directory where Users and Groups are stored.
- OUDINTERNAL_USERNAMEATTRIBUTE is the name of the directory attribute containing the user's name. Note that this is different from the login name.
- OUDINTERNAL_LOGINATTRIBUTE is the LDAP attribute which contains the users Login name.
- OUDINTERNAL_USERSEARCHBASE is the location in the directory where Users are Stored.
- OUDINTERNAL_NEW_SETUP is always set to true for Oracle Unified Directory. If you are not using OUD, you do not need to specify this attribute.
- POLICYSTORE_SHARES_IDSTORE is set to true for IDM 11*g*.
- OUDINTERNAL_OAMADMINUSER is the name of the user you want to create as your Access Manager Administrator.
- OUDINTERNAL_OAMSOFTWAREUSER is a user that gets created in LDAP that is used when Access Manager is running to connect to the LDAP server.
- OAM11G_OUDINTERNAL_ROLE_SECURITY_ADMIN is the name of the group which is used to allow access to the OAM console.
- OUDINTERNAL_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.
- OUDINTERNAL_OIMADMINGROUP Is the name of the group you want to create to hold your Oracle Identity Manager administrative users.
- OUDINTERNAL_OIMADMINUSER is the user that Oracle Identity Manager uses to connect to the Identity store.
- OUDINTERNAL_WLSADMINUSER: The username to be used for logging in to the web logic domain once it is enabled by SSO. In the above example, weblogic_idm is used.

- `OUDINTERNAL_WLSADMINGROUP`: is the name of the group to which users who are allowed to log in to the WebLogic system components, such as the WLS Console and EM, belong.

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

10.4.3 Configuring Oracle Unified Directory for Use with Oracle Access Manager and Oracle Identity Manager

This section explains how to configure Oracle Unified Directory for use with Oracle Access Manager and Oracle Identity Manager.

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

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

To do this, perform the following tasks on IDMHOST1:

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

   Set `ORACLE_HOME` to *`IAM_ORACLE_HOME`*.

   Set `JAVA_HOME` to *`JAVA_HOME`*.

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

   `IAM_ORACLE_HOME``/idmtools/bin`

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

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

   For example:

   ```
   idmConfigTool.sh -preConfigIDStore input_file=oudinternal.props
   ```

   When the command runs, you are prompted to enter the password of the account you are connecting to the Identity Store with. This command might take some time to complete.

   Sample command output:

   ```
   Enter ID Store Bind DN password:
   Apr 3, 2013 3:47:37 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
   INFO: -> LOADING: /u01/oracle/products/access/iam/idmtools/templates/oud/oud_
   schema_extn.ldif
   Apr 3, 2013 3:47:38 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
   INFO: -> LOADING:
   /u01/oracle/products/access/iam/oam/server/oim-intg/ldif/ojd/schema/ojd_oam_
   pwd_schema_add.ldif
   ```

```
Apr 3, 2013 3:47:38 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING:
/u01/oracle/products/access/iam/oam/server/oim-intg/ldif/ojd/schema/ojd_user_
schema_add.ldif
Apr 3, 2013 3:47:38 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING:
/u01/oracle/products/access/iam/oam/server/oim-intg/ldif/ojd/schema/ojd_user_
index_generic.ldif
Apr 3, 2013 3:47:39 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/oracle/products/access/iam/idmtools/templates/oud/add_
oraclecontext_container.ldif
Apr 3, 2013 3:47:39 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/oracle/products/access/iam/idmtools/templates/oud/oud_
indexes_extn.ldif
Apr 3, 2013 3:47:39 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/oracle/products/access/iam/idmtools/templates/oud/idm_
idstore_groups_template.ldif
Apr 3, 2013 3:47:39 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/oracle/products/access/iam/idmtools/templates/oud/idm_
idstore_groups_acl_template.ldif
Apr 3, 2013 3:47:39 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING:
/u01/oracle/products/access/iam/idmtools/templates/oud/systemid_pwdpolicy.ldif
Apr 3, 2013 3:47:39 AM oracle.ldap.util.LDIFLoader loadOneLdifFile
INFO: -> LOADING: /u01/oracle/products/access/iam/idmtools/templates/oud/fa_
pwdpolicy.ldif
The tool has completed its operation. Details have been logged to
automation.log
```

3. Check the log file for any errors or warnings and correct them. The file with the name **automation.log** is created in the directory from where you run the tool.

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

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

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

10.4.4 Creating Users and Groups

You must seed the Identity Store with users and groups that are required by the Identity Management components.

To seed the Identity Store, perform the following tasks on IDMHOST1:

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

   Set `ORACLE_HOME` to *`IAM_ORACLE_HOME`*.

   Set `JAVA_HOME` to *`JAVA_HOME`*.

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

   ```
   IAM_ORACLE_HOME/idmtools/bin
   ```

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

   The syntax of the command on Linux is:

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

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

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

   - In all topologies, when you enable single sign-on for your administrative consoles, you must ensure that there is a user in your Identity Store that has the permissions to log in to your WebLogic Administration Console and Oracle Enterprise Manager Fusion Middleware Control. Type:

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

     Run this command first.

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

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

   - If your topology includes Oracle Identity Manager, you must seed the Identity Store with the `xelsysadm` user and assign it to an Oracle Identity Manager administrative group. You must also create a user outside of the standard `cn=Users` location to be able to perform reconciliation. This user is also the user that should be used as the bind DN when connecting to directories with Oracle Virtual Directory. Type:

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

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

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

3. After running each command, check the log file for any errors or warnings and correct them. The file with the name `automation.log` is created in the directory from where you run the tool.

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

10.4.5 Add Missing Oracle Unified Directory Permission

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

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

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

Update Oracle Unified Directory using the command:

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

10.4.6 Granting Oracle Unified Directory Change Log Access

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

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

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

   For example:

   ```
   OUD_ORACLE_INSTANCE/bin/dsconfig set-access-control-handler-prop \
   --remove
   global-aci:"(target=\"ldap:///cn=changelog\")(targetattr=\"*\")(version 3.0;
   acl \"External changelog access\"; deny (all) userdn=\"ldap:///anyone\";)" \
           --hostname IDMHOST1.mycompany.com \
           --port 4444 \
           --trustAll  \
           --bindDN cn=oudadmin \
   ```

```
        --bindPasswordFile mypasswordfile \
        --no-prompt
```

3. Then add the following new ACI:

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

For example:

```
OUD_ORACLE_INSTANCE/bin/dsconfig set-access-control-handler-prop \
--add global-aci:"(target=\"ldap:///cn=changelog\")(targetattr=\"*\")(version
3.0; acl \"External changelog access\"; allow
(read,search,compare,add,write,delete,export)
groupdn=\"ldap:///cn=OIMAdministrators,cn=groups,dc=mycompany,dc=com\";)" \
        --hostname IDMHOST1.mycompany.com \
        --port 4444 \
        --trustAll \
        --bindDN cn=oudadmin \
        --bindPasswordFile mypasswordfile \
        --no-prompt
```

4. Then add the following new ACI:

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

For example:

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

5. Then add the following ACI:

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

For example:

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

10.4.7 Creating Oracle Unified Directory Indexes

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

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

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

ORACLE_INSTANCE/OUD/bin/ldapmodify -h IDMHOST2.mycompany.com -Z -X -p 4444 -a -D
"cn=oudadmin" -j mypasswordfile -c  -f IAM_ORACLE_HOME/idmtools/templates/oud/oud_
indexes_extn.ldif
```

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

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

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

2. Execute the command:

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

3. Restart Oracle Unified Directory by issuing the command:

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

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

10.4.8 Backing Up the Identity Stores

Back up your LDAP directories, as described in Section 16.6, "Backing Up the Oracle IDM Enterprise Deployment."

11 Extending the Domain to Include Oracle Access Management

This chapter describes how to extend the domain to include Oracle Access Management Access Manager in the Oracle Identity Management enterprise deployment.

This chapter includes the following topics:

11.1 Overview of Extending the Domain to Include Oracle Access Management Access Manager

Access Manager enables your users to seamlessly gain access to web applications and other IT resources across your enterprise. It provides a centralized and automated single sign-on (SSO) solution, which includes an extensible set of authentication methods and the ability to define workflows around them. It also contains an authorization engine, which grants or denies access to particular resources based on properties of the user requesting access as well as based on the environment from which the request is made. Comprehensive policy management, auditing, and integration with other components of your IT infrastructure enrich this core functionality.

Access Manager consists of several components, including OAM Server, Oracle Access Management Console, and WebGates. The OAM Server includes all the components necessary to restrict access to enterprise resources. The Oracle Access Management Console is the administrative console to Access Manager. WebGates are web server agents that act as the actual enforcement points for Access Manager. Follow the instructions in this chapter and Section 15, "Configuring Single Sign-on for Administration Consoles in an Enterprise Deployment" to install and configure the Access Manager components necessary for your enterprise deployment.

11.2 About Domain URLs

After you complete this chapter, the following URL will be available:

Table 11–1 OAM URLs After Web Tier Configuration

Component	URLs	User	SSO User
OAM Console	`http://ADMIN.mycompany.com/oamconsole`	`weblogic`	`oamadmin`
Oracle Enterprise Manager Fusion Middleware Control	`http://ADMIN.mycompany.com/em`	`weblogic`	`weblogic_idm`
Oracle Directory Services Manager	`http://ADMIN.mycompany.com/odsm`	`weblogic`	`weblogic_idm`
Oracle Entitlements Server Policy Manager	`http://ADMIN.mycompany.com/apm`	`weblogic`	`weblogic_idm`

11.3 Prerequisites

Before you configure Access Manager, ensure that the following tasks have been performed on IDMHOST1 and IDMHOST2:

1. Prepare the Identity Store as described in Chapter 10, "Preparing Identity Stores."
2. Configure Oracle Web Tier Directory on WEBHOST1 and WEBHOST2 as described in Chapter 7, "Installing and Configuring Oracle Traffic Director for an Enterprise Deployment."
3. Configure the load balancer as described in Section 3.9, "Configuring the Load Balancer."

11.4 Extending Domain with Access Manager

Start the configuration wizard on IDMHOST1 by executing the command:

```
IAM_MW_HOME/oracle_common/common/bin/config.sh
```

Then proceed as follows:

1. On the Welcome screen, select **Extend an Existing WebLogic Domain**. Click **Next**.
2. On the Select a WebLogic Domain screen, using the navigator, select the domain home of the WebLogic Administration Server, for example: *`ASERVER_HOME`*

   Click **Next**
3. On the Select Extension Source screen, select **Oracle Access Management [iam]**.

   Click **Next**
4. On the Configure JDBC Component Schema screen, do the following:

Select **OAM Infrastructure**.

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

Click **Next**.

5. The Gridlink RAC Component Schema screen appears. In this screen, enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU.

   - **Driver**: Select Oracle's driver (Thin) for GridLink Connections,Versions:10 and later.
   - Select **Enable FAN**.
   - Do one of the following:
     - If SSL is not configured for ONS notifications to be encrypted, deselect **SSL**.
     - Select **SSL** and provide the appropriate wallet and wallet password.
   - **Service Listener**: Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the parameter `remote_listener` in the database:

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

     Notes:

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

   - **ONS Host**: Enter the SCAN address for the Oracle RAC database and the ONS remote port as reported by the database:

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

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

     ```
     DBHOST1.mycompany.com (port 6200)
     ```

     and

     ```
     DBHOST2.mycompany.com (port 6200)
     ```

Enter the following RAC component schema information:

Table 11–2 RAC Component Schema Information

Schema Name	Service Name	User name	Password
Access Management	`oamedg.mycompany.com`	`EDG_OAM`	*password*

6. In the Test JDBC Data Sources screen, confirm that all connections were successful.

   The connections are tested automatically. The **Status** column displays the results. If all connections are not successful, click **Previous** to return to the previous screen and correct your entries.

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

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

8. On the Select Optional Configuration screen, select **Managed Servers, Clusters and Machines**.

   Click **Next**

9. When you first enter the Configure Managed Servers screen, a managed server called oam_server1 is created automatically. Rename oam_server1 to WLS_OAM1 and update its attributes as shown in the following table. Then, add a new managed server called WLS_OAM2 with the following attributes.

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_OAM1	IDMHOST1.mycompany.com	`14100`	N/A	No
WLS_OAM2	IDMHOST2.mycompany.com	`14100`	N/A	No

Notes:

- Do not change the configuration of the managed servers that were configured as a part of previous deployments.
- Do not delete the default managed servers that are created. Rename them as described.

   Click **Next**.

10. On the Configure Clusters screen, create a cluster by clicking **Add**. Supply the following information:

Table 11–3 Values for Configure Clusters Screen

Name	Cluster Messaging Mode
`oam_cluster`	Unicast

   Leave all other fields at the default settings and click **Next**.

11. On the Assign Servers to Clusters screen, associate the Managed Servers with the cluster. Click the cluster name in the right pane. Click the Managed Server under Servers, then click the arrow to assign it to the cluster.

    Assign servers to the cluster as follows:

    Table 11–4 Servers to Assign to Cluster

Cluster	Server
oam_cluster	WLS_OAM1
	WLS_OAM2

    > **Note:** Do not change the configuration of any clusters which have already been configured as part of previous application deployments.

    Click **Next**.

12. On the Configure Machines screen, create a machine for each host in the topology. Click the **Unix Machine** tab and then click **Add** to add the following machines:

    > **Note:** "Name" can be any unique string. "Node Manager Listen Address" must be a resolvable host name.

    Table 11–5 Machines

Name	Node Manager Listen Address	Node manager Listen Port	Port Variable
`IDMHOST1.mycompany.com`	`IDMHOST1.mycompany.com`	5556	*NMGR_PORT*
`IDMHOST2.mycompany.com`	`IDMHOST2.mycompany.com`	5556	*NMGR_PORT*

    Leave all other fields to their default values.

    > **Note:** The machine name does not need to be a valid host name or listen address; it is just a unique identifier of a Node Manager location

    Click **Next**.

13. On the Assign Servers to Machines screen, assign servers to machines as follows:

    IDMHOST1: **WLS_OAM1**

    IDMHOST2: **WLS_OAM2**

    Click **Next** to continue.

14. On the Configuration Summary screen, click **Extend** to extend the domain.

Note: If you receive a warning that says:

```
CFGFWK: Server listen ports in your domain configuration conflict
with ports in use by active processes on this host
```

Click **OK**.

This warning appears if Managed Servers have been defined as part of previous installs and can safely be ignored.

15. On the Installation Complete screen, click **Done**.
16. Restart WebLogic Administration Server as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

11.5 Configuring Access Manager

This section contains the following topics:

- Section 11.5.1, "Removing IDM Domain Agent"
- Section 11.5.2, "Setting a Global Passphrase"
- Section 11.5.3, "Configuring Access Manager by Using the IDM Configuration Tool"
- Section 11.5.4, "Validating the Configuration"
- Section 11.5.5, "Updating Newly-Created Agent"
- Section 11.5.6, "Modifying OAM Resources"
- Section 11.5.7, "Updating the Idle Timeout Value"
- Section 11.5.8, "Updating Existing WebGate Agents"
- Section 11.5.9, "Add Condition to the Admin Role as Workaround"

11.5.1 Removing IDM Domain Agent

By default, the IDMDomainAgent provides single sign-on capability for administration consoles. In enterprise deployments, WebGate handles single sign-on, so you must remove the IDMDomainAgent. Remove the IDMDomainAgent as follows:

Log in to the WebLogic console at the URL listed in Section 16.2, "About Identity Management Console URLs."

Then:

1. Select **Security Realms** from the **Domain Structure** Menu
2. Click **myrealm**.
3. Click the **Providers** tab.
4. Click **Lock and Edit** from the Change Center.
5. In the list of authentication providers, select **IAMSuiteAgent**.
6. Click **Delete**.
7. Click **Yes** to confirm the deletion.

8. Click **Activate Changes** from the Change Center.
9. Restart WebLogic Adminisration Server and ALL running Managed Servers, as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

11.5.2 Setting a Global Passphrase

By default, Access Manager is configured to use the Open security model. If you plan to change this mode using `idmConfigTool`, you must set a global passphrase. Although you need not set the global passphrase and the web gate access password to be the same, it is recommended that you do.You do this by performing the following steps.

1. Log in to the OAM console at the URL listed in Section 16.2, "About Identity Management Console URLs."

   as the WebLogic administration user.
2. Click the **System Configuration** tab.
3. Click **Access Manager** located in the Access Manager section.
4. Select **Open** from the **Actions** menu. The access manager settings are displayed.
5. If you plan to use Simple security mode for OAM servers, supply a global passphrase.
6. Click **Apply**.

11.5.3 Configuring Access Manager by Using the IDM Configuration Tool

Now that the initial installation is done, perform the following tasks:

- Configure Access Manager to use an external LDAP Directory, (`oudinternal.mycompany.com`).
- Create Access Manager WebGate Agent.

You perform these tasks by using `idmConfigTool`.

Note: Two parameter settings determine whether you are configuring Access Manager with Oracle Identity Manager integration or Access Manager alone.

- To configure Access Manager with Oracle Identity Manager integration, set `OAM11G_OIM_INTEGRATION_REQ` to `true` and specify a value for `OAM11G_OIM_OHS_URL`.
- To configure Access Manager without Oracle Identity Manager, set `OAM11G_OIM_INTEGRATION_REQ` to `false`.

These parameters are used to add extra links, such as Forgotten Password, to the Access Manager credential collection page

If you configure Access Manager without Oracle Identity Manager, then decide to add Oracle Identity Manager at a later date, you must run this command again to configure Access Manager with Oracle Identity Manager integration.

Perform the following tasks on IDMHOST1:

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

   Set `ORACLE_HOME` to *`IAM_ORACLE_HOME`*.

   Set `JAVA_HOME` to *`JAVA_HOME`*.

2. Create a properties file called `config_oam1.props` with the following contents:

```
WLSHOST: ADMINVHN.mycompany.com
WLSPORT: 7001
WLSADMIN: weblogic
WLSPASSWD: Admin Password
IDSTORE_DIRECTORYTYPE: OUD
IDSTORE_HOST: oudinternal.mycompany.com
IDSTORE_PORT: 1489
IDSTORE_BINDDN: cn=oudadmin
IDSTORE_USERNAMEATTRIBUTE: cn
IDSTORE_LOGINATTRIBUTE: uid
OAM11G_SERVER_LOGIN_ATTRIBUTE: uid
IDSTORE_USERSEARCHBASE: cn=Users,dc=mycompany,dc=com
IDSTORE_SEARCHBASE: dc=mycompany,dc=com
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=mycompany,dc=com
IDSTORE_SYSTEMIDBASE: cn=systemids,dc=mycompany,dc=com
IDSTORE_OAMSOFTWAREUSER: oamLDAP
IDSTORE_OAMADMINUSER: oamadmin
PRIMARY_OAM_SERVERS: IDMHOST1.mycompany.com:5575,IDMHOST2.mycompany.com:5575
WEBGATE_TYPE: ohsWebgate11g
ACCESS_GATE_ID: Webgate_IDM
OAM11G_OIM_WEBGATE_PASSWD: password to be assigned to WebGate
COOKIE_DOMAIN: .mycompany.com
OAM11G_WG_DENY_ON_NOT_PROTECTED: true
OAM11G_IDM_DOMAIN_OHS_HOST: sso.mycompany.com
OAM11G_IDM_DOMAIN_OHS_PORT: 443
OAM11G_IDM_DOMAIN_OHS_PROTOCOL: https
OAM11G_SERVER_LBR_HOST: sso.mycompany.com
OAM11G_SERVER_LBR_PORT: 443
OAM11G_SERVER_LBR_PROTOCOL: https
OAM11G_OAM_SERVER_TRANSFER_MODE: simple
OAM_TRANSFER_MODE: simple
OAM11G_IDM_DOMAIN_LOGOUT_URLS:
/console/jsp/common/logout.jsp,/em/targetauth/emaslogout.jsp
OAM11G_IDSTORE_ROLE_SECURITY_ADMIN: OAMAdministrators
OAM11G_SSO_ONLY_FLAG: false
COOKIE_EXPIRY_INTERVAL: 120
OAM11G_IMPERSONATION_FLAG: false
OAM11G_OIM_INTEGRATION_REQ: false
OAM11G_OIM_OHS_URL:https://SSO.mycompany.com:443
SPLIT_DOMAIN: false
```

   Where:

   - `WLSHOST` (*`ADMINVHN`*) is the host of your administration server. This is the virtual name.
   - `WLSPORT` is the port of your administration server, *`WLS_ADMIN_PORT`* in Section A.3, "Port Mapping".
   - `WLSADMIN` is the WebLogic administrative user you use to log in to the WebLogic console.
   - `WLSPASSWD` is the WebLogic administrator password.
   - `IDSTORE_DIRECTORYTYPE` is `OUD`.

- `IDSTORE_HOST` and `IDSTORE_PORT` are the host and port of the Identity Store directory when accessed through Oracle Traffic Director. These are *`LDAP_LBR_HOST`* and *`LDAP_LBR_PORT`* in the Section A.3, "Port Mapping" worksheet.
- `IDSTORE_BINDDN` is an administrative user in the Identity Store directory.
- `IDSTORE_USERSEARCHBASE` is the location in the directory where Users are stored.
- `IDSTORE_GROUPSEARCHBASE` is the location in the directory where Groups are stored.
- `IDSTORE_SEARCHBASE` is the location in the directory where Users and Groups are stored.
- `IDSTORE_SYSTEMIDBASE` is the location of a container in the directory where the user oamLDAP is stored.
- `IDSTORE_OAMSOFTWAREUSER` is the name of the user you created in Section 10.4, "Preparing the Identity Store" to be used to interact with LDAP.
- `IDSTORE_OAMADMINUSER` is the name of the user you created in Section 10.4, "Preparing the Identity Store" to access your OAM Console.
- `PRIMARY_OAM_SERVERS` is a comma separated list of your OAM Servers and the proxy ports they use, for example: IDMHOST1:*OAM_PROXY_PORT*

Note: To determine the proxy ports your OAM Servers use:

1. Log in to the OAM console at the URL listed in Section 16.2, "About Identity Management Console URLs."
2. Click the **System Configuration** tab.
3. Expand **Server Instances** under the Common Configuration section
4. Click an OAM Server, such as **WLS_OAM1**, and select **Open** from the **Actions** menu.
5. Proxy port is the one shown as **Port**.

- `ACCESS_GATE_ID` is the name you want to assign to the WebGate.
- `OAM11G_OIM_WEBGATE_PASSWD` is the password to be assign to the WebGate.
- `OAM11G_IDM_DOMAIN_OHS_HOST` is the name of the load balancer which is in front of the OTD's.
- `OAM11G_IDM_DOMAIN_OHS_PORT` is the port that the load balancer listens on (*HTTP_SSL_PORT*).
- `OAM11G_IDM_DOMAIN_OHS_PROTOCOL` is the protocol to use when directing requests at the load balancer.
- `OAM11G_WG_DENY_ON_NOT_PROTECTED`, when set to `false`, allows login pages to be displayed. It should be set to `true` when using webgate11g.
- `OAM_TRANSFER_MODE` is the security model that the Oracle Access Manager Servers function in. Valid values are `simple` and `open`. If you use the `simple` mode, you must define a global passphrase, as defined in Section 11.5.2, "Setting a Global Passphrase."

- `OAM11G_OAM_SERVER_TRANSFER_MODE` is the security model that the OAM Servers function in, as defined in Section 11.5.2, "Setting a Global Passphrase."
- `OAM11G_IDM_DOMAIN_LOGOUT_URLS` is set to the various logout URLs.
- `OAM11G_SSO_ONLY_FLAG` confgures Access Manager as authentication only mode or normal mode, which supports authentication and authorization.

  If `OAM11G_SSO_ONLY_FLAG` is `true`, the OAM Server operates in authentication only mode, where all authorizations return true by default without any policy validations. In this mode, the server does not have the overhead of authorization handling. This is recommended for applications which do not depend on authorization policies and need only the authentication feature of the OAM Server.

  If the value is `false`, the server runs in default mode, where each authentication is followed by one or more authorization requests to the OAM Server. WebGate allows the access to the requested resources or not, based on the responses from the OAM Server.
- `OAM11G_IMPERSONATION_FLAG` is set to `true` if you are configuring OAM Impersonation.
- `OAM11G_SERVER_LBR_HOST` is the name of the load balancer fronting your site. This and the following two parameters are used to construct your login URL.
- `OAM11G_SERVER_LBR_PORT` is the port that the load balancer is listening on (*HTTP_SSL_PORT*).
- `OAM11G_SERVER_LBR_PROTOCOL` is the URL prefix to use.
- `OAM11G_OIM_INTEGRATION_REQ` should be set to `true` if you are building a topology which contains both OAM and OIM. Otherwise set to `false` at this point. This value is only set to true when performing Access Manager/Oracle Identity Manager integration and is set during the integration phase.
- `OAM11G_OIM_OHS_URL` should be set to the URL of your load balancer. This parameter is only required if your topology contains OAM and OIM.
- `COOKIE_DOMAIN` is the domain in which the WebGate functions.
- `WEBGATE_TYPE` is the type of WebGate agent you want to create.
- `OAM11G_IDSTORE_NAME` is the Identity Store name. If you already have an Identity Store in place which you wish to reuse (rather than allowing the tool to create a new one for you), then set the value of this parameter to the name of the Identity Store you wish to reuse.
- `OAM11G_SERVER_LOGIN_ATTRIBUTE` when set to `uid`, ensures that when users log in, their username is validated against the `uid` attribute in LDAP.
- `SPLIT_DOMAIN` set to `true` if you are building an OAM only topology. Otherwise set to `false`.

3. Configure Access Manager using the command `idmConfigTool` which is located at:

   IAM_ORACLE_HOME`/idmtools/bin`

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

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

For example:

```
idmConfigTool.sh -configOAM input_file=config_oam1.props
```

When the command runs you are prompted to enter the password of the account you are connecting to the Identity Store with. You are also asked to specify the passwords you want to assign to these accounts:

- IDSTORE_PWD_OAMSOFTWAREUSER
- IDSTORE_PWD_OAMADMINUSER

4. Check the log file for any errors or warnings and correct them. A file named automation.log is created in the directory where you run the tool.
5. Restart WebLogic Administration Server as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

> **Note:** After you run idmConfigTool, several files are created that you need for subsequent tasks. Keep these in a safe location.
>
> Two 11*g* WebGate profiles are created: Webgate_IDM, which is used for intercomponent communication and Webgate_IDM_11g, which is used by 11*g* Webgates.
>
> The following files exist in the directory *ASERVER_HOME*/output/Webgate_IDM_11g. You need these when you install the WebGate software.
>
> - cwallet.sso
> - ObAccessClient.xml
> - password.xml
>
> Additionally, you need the files aaa_cert.pem and aaa_key.pem, which are located in the directory *ASERVER_HOME*/output/Webgate_IDM.

11.5.4 Validating the Configuration

To Validate that this has completed correctly.

1. Access the OAM console at: http://admin.mycompany.com/oamconsole
2. Log in as the Access Manager administration user you created in Section 10.4, "Preparing the Identity Store," for example, oamadmin.
3. Click the **System Configuration** tab
4. Expand **Access Manager** - **SSO Agents** - **OAM Agents**.

5. Click the open folder icon, then click **Search**.
6. You should see the WebGate agents `Webgate_IDM` and `Webgate_IDM_11g`, which you created in Section 11.5.3, "Configuring Access Manager by Using the IDM Configuration Tool."

11.5.5 Updating Newly-Created Agent

After generating the initial configuration, you must edit the configuration and add advanced configuration entries.

1. Select **System Configuration** Tab
2. Select **Access Manager - SSO Agents - OAM Agent** from the directory tree. Double-click or select the open folder icon.
3. On the displayed search page click **Search** to perform an empty search.
4. Click the Agent `Webgate_IDM`.
5. Select **Open** from the Actions menu.
6. Set **Maximum Number of Connections** to `4` for all of the OAM Servers listed in the primary servers list.
7. If the following **Logout URLs** are not listed, add them:
   - `/oamsso/logout.html`
   - `/console/jsp/common/logout.jsp`
   - `/em/targetauth/emaslogout.jsp`
8. Click **Apply**.
9. Repeat Steps 4 through 7 for the WebGate agent Webgate_IDM_11g.
10. Click **Policy Configuration** tab.
11. Click **Host Identifiers.**
12. Click **Open**.
13. Click **Search**.
14. Click **IAMSuiteAgent**.
15. Click **+** in the **Host Name Variations** box.
16. Enter the following information:
    - **Host Name**: `ADMIN.mycompany.com`
    - **Port**: `80(`*HTTP_PORT*`)`
17. Click **Apply**.

11.5.6 Modifying OAM Resources

When Oracle Access Management is installed, a number of resources are created with protection levels set. In order for Oracle Identity Management to function correctly, one of these resources must be modified, and one created.

To modify one resource and create another:

1. Create a resource in Access Manager by logging in to the OAM console at the URL listed in Section 16.2, "About Identity Management Console URLs."

2. Click **Application Domains**, and then click **Open**.
3. Click SearchClick **IAM Suite**, and then click the **Resource** tab.
4. Click **New Resource**, and enter the following information:
   - **Type**: `http`
   - **Description**: `provisioning-callback`
   - **Host Identifier**: `IAMSuiteAgent`
   - **Resource URL**: `/provisioning-callback/**`
   - **Protection Level**: `Excluded`
   - **Authentication Policy**: `n/a`
   - **Authorization Policy**: `n/a`

   Click **Apply**.
5. In the **Search Results** window, click the resource `/identity/**`.
6. Click **Edit**.
7. Change the **Protection Level** to `Excluded`.
8. Click **Apply**.

11.5.7 Updating the Idle Timeout Value

By default the OAM idle timeout is set to two hours. This can cause issues with users not being logged out after a session has timed out. Update this value to fifteen minutes.

To update the value:

1. Login to the OAM console at the following URL:

   ```
   http://admin.mycompany.com/oamconsole
   ```

2. Log in as the Access Manager administration user you created in Section 10.4, "Preparing the Identity Store," for example, `oamadmin`.
3. Select the **System Configuration** tab.
4. Click on **Common Settings** under **Common configuration**.
5. Click **Open**.
6. Change **Idle Time Out (minutes)** to `15`.
7. Click **Apply**.

11.5.8 Updating Existing WebGate Agents

If you have changed the OAM security model using the idmConfigTool you must change the security model used by any existing Webgates to reflect this change.

To do this, perform the following steps:

1. Log in to the Oracle Access Management Console as the Access Manager administration user you created in Section 10.4, "Preparing the Identity Store," at the URL listed in Section 16.2, "About Identity Management Console URLs."

2. Click the **System Configuration** tab.
3. Expand **Access Manager - SSO Agents**.
4. Click **OAM Agents** and select **Open** from the **Actions** menu.
5. In the Search window, click **Search**.
6. Click each Agent that was not created by `idmconfigTool` in Section 11.5.3, "Configuring Access Manager by Using the IDM Configuration Tool", for example: **IAMSuiteAgent**.
7. Set the Security value to the new security model. Add any missing Access Manager servers to the displayed list.

   Click **Apply**.

11.5.9 Add Condition to the Admin Role as Workaround

To work around a know issue, add a condition to the Admin role using the WebLogic Administration Server Console.

> **Note:** If you configured OAM using `SPLIT_DOMAIN:true`, perform the procedure in this section. However, if you configured OAM with `SPLIT_DOMAIN: false` then perform the steps in this section AFTER you have integrated Oracle Identity Management with Oracle Access Manager in Section 12.21.4, "Integrating Oracle Identity Manager with Oracle Access Manager Using the idmConfigTool."

To add conditions to the Admin role in the Security Realm:

1. Log in to the WebLogic Administration Server Console at the URL listed in Section 16.2, "About Identity Management Console URLs."
2. In the left pane of the console, click **Security Realms**.
3. On the Summary of Security Realms page, click **myrealm** under the Realms table.
4. On the Settings page for myrealm, click the **Roles & Policies** tab.
5. On the Realm Roles page, expand the **Global Roles** entry under the Roles table. This brings up the entry for Roles.
6. Click the **Roles** link to go to the Global Roles page.
7. On the Global Roles page, click the **Admin** role to go to the Edit Global Role page:
8. On the Edit Global Roles page, under the Role Conditions table, click **Add Conditions**.
9. On the Choose a Predicate page, select **Group** from the predicates list and click **Next**.
10. On the Edit Arguments Page, specify `OAMAdministrators` in the **Group Argument** field and click **Add**.
11. Click **Finish** to return to the Edit Global Rule page.

    The Role Conditions table now shows the `OAMAdministrators` Group as an entry.
12. Click **Save** to finish adding the Admin role to the `OAMAdministrators` Group.

11.6 Deploying Managed Server Configuration to Local Storage

Once the configuration is complete, you must propagate the Oracle Identity Manager configuration to the managed server directory on IDMHOST1 and IDMHOST2.

You do this by packing and unpacking the domain, you pack the domain first on IDMDomain on IDMHOST1 then unpack it on IDMHOST1 and IDMHOST2.

Follow these steps to propagate the domain to the managed server domain directory.

1. Invoke the `pack` utility from *`ORACLE_COMMON_HOME`*`/common/bin/` on IDMHOST1.

   ```
   ./pack.sh -domain=ASERVER_HOME -template=iam_domain.jar  -template_name="IAM
   Domain" -managed=true
   ```

   This creates a file called `iam_domain.jar`. Copy this file to IDMHOST2.

2. On IDMHOST1 and IDMHOST2, invoke the utility `unpack`, which is also located in the directory: ORACLE_COMMON_HOME/common/bin/

   ```
   ./unpack.sh -domain=MSERVER_HOME -template=iam_domain.jar -overwrite_
   domain=true -app_dir=MSERVER_HOME/applications
   ```

11.7 Starting Managed Servers WLS_OAM1 and WLS_OAM2

Start the managed servers WLS_OAM1 and WLS_OAM2 as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

11.8 Validating Access Manager

You can validate Access Manager using the `oamtest` tool.

> **Note:** If you have not applied the latest OAM Bundle Patch, you may see the OAM Test tool throwing Null Pointer Exceptions.
>
> If this is the case ignore this test. This issue is related to the OAM test tool itself and not the underlying configuration.

To validate Access Manager:

1. Ensure that *`JAVA_HOME`* is set in your environment.
2. Add *`JAVA_HOME`*`/bin` to your *`PATH`*, for example:

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

3. Change directory to:

   `IAM_ORACLE_HOME``/oam/server/tester`

4. Start the test tool in a terminal window using the command:

   ```
   java -jar oamtest.jar
   ```

5. When the OAM test tool starts, enter the following information in the **Server Connection** section of the page:

   - **Primary IP Address**: `IDMHOST1.mycompany.com`
   - **Port**: `5575` (*OAM_PROXY_PORT*)

- **Agent ID**: `Webgate_IDM_11g`
- **Agent Password**: *`webgate password`*

> **Note:** if you configured simple mode, you must select **Simple** and provide the global passphrase.

Click **Connect**.

In the status window you see:

```
[reponse] Connected to primary access server
```

6. In the **Protected Resource URI** section enter:

   - **Scheme**: `http`
   - **Host**: `ADMIN.mycompany.com`
   - **Port**: `80` (*HTTP_PORT*)
   - **Resource**: `/oamconsole`

   Click **Validate**.

   In the status window you see:

   ```
   [request][validate] yes
   ```

7. In the **User Identity** window, enter:

   - **Username**: `oamadmin`
   - **Password**: *`oamadmin password`*

   Click **Authenticate**.

   In the status window, you see:

   ```
   [request] [authenticate] yes
   ```

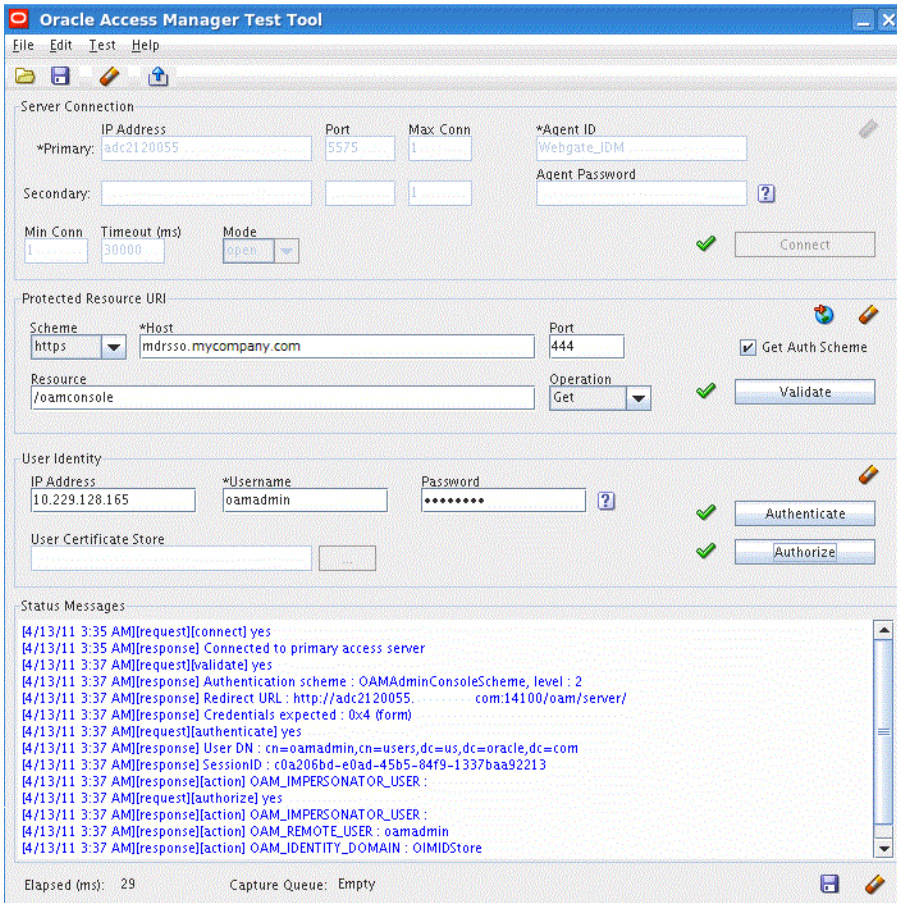
   Click **Authorize**.

   In the status window you see.

   ```
   [request] [authorize] yes
   ```

The following is an example of a test:

Figure 11–1 Oracle Access Manager Test Tool



Server Connection

IP Address · *Primary: adc2120055 · Port: 5575 · Max Conn: 1 · *Agent ID: Webgate_IDM

Secondary: · Agent Password:

Min Conn: 1 · Timeout (ms): 30000 · Mode: open

Protected Resource URI

Scheme: https · *Host: mdrsso.mycompany.com · Port: 444 · ☑ Get Auth Scheme

Resource: /oamconsole · Operation: Get

User Identity

IP Address: 10.229.128.165 · *Username: oamadmin · Password: ••••••••

User Certificate Store:

Status Messages

```
[4/13/11 3:35 AM][request][connect] yes
[4/13/11 3:35 AM][response] Connected to primary access server
[4/13/11 3:37 AM][request][validate] yes
[4/13/11 3:37 AM][response] Authentication scheme : OAMAdminConsoleScheme, level : 2
[4/13/11 3:37 AM][response] Redirect URL : http://adc2120055.          com:14100/oam/server/
[4/13/11 3:37 AM][response] Credentials expected : 0x4 (form)
[4/13/11 3:37 AM][request][authenticate] yes
[4/13/11 3:37 AM][response] User DN : cn=oamadmin,cn=users,dc=us,dc=oracle,dc=com
[4/13/11 3:37 AM][response] SessionID : c0a206bd-e0ad-45b5-84f9-1337baa92213
[4/13/11 3:37 AM][response][action] OAM_IMPERSONATOR_USER :
[4/13/11 3:37 AM][request][authorize] yes
[4/13/11 3:37 AM][response][action] OAM_IMPERSONATOR_USER :
[4/13/11 3:37 AM][response][action] OAM_REMOTE_USER : oamadmin
[4/13/11 3:37 AM][response][action] OAM_IDENTITY_DOMAIN : OIMIDStore
```

Elapsed (ms): 29 · Capture Queue: Empty

Repeat this test for each access server in the topology, remembering to change the connection details for each server.

11.9 Creating a Single Keystore for Integrating Access Manager with Other Components

When you configure Access Manager to work using the simple transport protocol, all traffic to Access Manager is encrypted. When you integrate Access Manager with other components, such as Oracle Identity Manager, you must enable the product being integrated to understand this encryption (This is not necessary when the transport model is open.). You do this by using a keystore.

When you change Access Manager to use the simple protocol, keystores are created automatically in the directory `*ASERVER_HOME*/output/webgate-ssl`. This directory contains the following files:

- `oamclient-keystore.jks`–contains the private key.
- `oamclient-truststore.jks`–contains the Access Manager simple mode CA certificate

These files are accessed using the Global Passphrase defined at the time of enabling Access Manager in simple mode.

Some products require configuring with both of the files above and some products, such as Oracle Identity Manager require a single consolidated keystore.

To create a keystore suitable for use by Oracle Identity Manager, perform the following steps.

1. Change directory to *ASERVER_HOME*/output/webgate-ssl, for example:

   ```
   cd ASERVER_HOME/output/webgate-ssl
   ```

2. Copy the file oamclient-keystore.jks to ssoKeystore.jks, for example

   ```
   cp oamclient-keystore.jks ssoKeystore.jks
   ```

3. Import the trust store into the new keystore ssoKeystore.jks using the command:

   ```
   keytool -importcert -file IAM_ORACLE_HOME/oam/server/config/cacert.der
   -trustcacerts -keystore PathName_to_keystore -storetype JKS
   ```

   Enter the keystore password when prompted. For example:

   ```
   keytool -importcert -file IAM_ORACLE_HOME/oam/server/config/cacert.der
   -trustcacerts -keystore ssoKeystore.jks -storetype JKS
   ```

> **Note:** The files ssoKeystore.jks and oamclient-truststore.jks are required when you integrate Access Manager running in Simple mode with Oracle Identity Manager. When you integrate these components, you are asked to copy these files to the *ASERVER_HOME*/config/fmwconfig directory. If you subsequently extend the domain on machines where these files have been placed using pack/unpack, you must recopy ssoKeystore.jks and oamclient-truststore.jks after unpacking.

11.10 Backing Up the Application Tier Configuration

Back up the database, the WebLogic domain, and the LDAP directories, as described in Section 16.6, "Backing Up the Oracle IDM Enterprise Deployment."

12 Extending the Domain to Include Oracle Identity Manager

This chapter describes how to install and configure Oracle Identity Manager for use in the Oracle Identity Management Enterprise Deployment Topology.

This chapter contains the following topics:

12.1 Overview of Extending the Domain to Include Oracle Identity Manager

Oracle Identity Manager is a user provisioning and administration solution that automates the process of adding, updating, and deleting user accounts from applications and directories. It also improves regulatory compliance by providing granular reports that attest to who has access to what. Oracle Identity Manager is available as a standalone product or as part of Oracle Identity Management.

Automating user identity provisioning can reduce Information Technology (IT) administration costs and improve security. Provisioning also plays an important role in regulatory compliance. Key features of Oracle Identity Manager include password management, workflow and policy management, identity reconciliation, reporting and auditing, and extensibility through adapters.

Oracle Identity Manager provides the following key functionalities:

- User Administration
- Workflow and Policy
- Password Management
- Audit and Compliance Management
- Integration Solutions
- User Provisioning
- Organization and Role Management

For details about Oracle Identity Manager, see the *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Manager*.

12.2 About Domain URLs

After you complete this chapter, the following URL will be available:

Table 12–1 OIM URLs

Component	URLs	SSO User
Self-service Console	`https://SSO.mycompany.com/identity`	xelsysadm
OIM Administration Console	`http://ADMIN.mycompany.com/sysadmin`	xelsysadm

12.3 Prerequisites

Before extending the domain with Oracle Identity Manager, ensure that the following tasks have been performed:

1. Ensure that the virtual IP addresses for the Oracle Identity Manager and SOA managed servers have been provisioned and enabled. See Section 3, "Configuring the Network for an Enterprise Deployment" for details
2. Ensure that you have created the wlfullclient.jar file, as described in Section 12.5, "Creating the wlfullclient.jar File."
3. Ensure the Identity Store is installed and configured.
4. Provision the Oracle Identity Management users as described in Section 10.4, "Preparing the Identity Store."
5. Stop all the managed servers running in your domain, as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components." before extending the domain with Oracle Identity Manager.

> **Note:** Oracle SOA deployed along with Oracle Identity Manager is used exclusively for Oracle Identity Manager work flow. It cannot be used for other purposes.

> **Note:** Be sure to verify you have obtained all required patches. For more info, see Section 2.5.3, "Applying Patches and Workarounds."

12.4 Provisioning the OIM Login Modules Under the WebLogic Server Library Directory

Due to issues with versions of the configuration wizard, some environmental variables are not added to the *ASERVER_HOME*/bin/setDomainenv.sh script. This causes certain install sequences to fail. This section is a temporary workaround for that problem. The steps in this section must be performed on all *MW_HOME*s that are associated with the domain hosting Oracle Identity Manager, that is, *IAM_MW_HOME*.

Apply the following steps across all the WebLogic Server homes in the domain.

1. Copy the `OIMAuthenticator.jar`, `oimmbean.jar`, `oimsigmbean.jar` and `oimsignaturembean.jar` files located under the *IAM_ORACLE_HOME*/server/loginmodule/wls directory to the *IAM_MW_HOME*/wlserver_10.3/server/lib/mbeantypes directory.

   ```
   cp IAM_ORACLE_HOME/server/loginmodule/wls/* IAM_MW_HOME/wlserver_
   10.3/server/lib/mbeantypes
   ```

2. Change directory to *MW_HOME*/wlserver_10.3/server/lib/mbeantypes/

   ```
   cd IAM_MW_HOME/wlserver_10.3/server/lib/mbeantypes
   ```

3. Change the permissions on these files to 750 by using the `chmod` command.

   ```
   chmod 750 *
   ```

12.5 Creating the wlfullclient.jar File

Oracle Identity Manager uses the `wlfullclient.jar` library for certain operations. Oracle does not ship this library, so you must create this library manually. Oracle recommends creating this library under the *IAM_MW_HOME*/wlserver_10.3/server/lib directory on all the machines hosting Oracle Identity Manager in the application tier of your environment, such as *IAM_MW_HOME* and *OIM_MW_HOME*.

Follow these steps to create the `wlfullclient.jar` file:

1. Navigate to the `IAM_MW_HOME/wlserver_10.3/server/lib` directory
2. Set your *JAVA_HOME* environment variable and ensure that the `JAVA_HOME/bin` directory is in your path.
3. Create the `wlfullclient.jar` file by running:

   ```
   java -jar wljarbuilder.jar
   ```

12.6 Synchronize System Clocks

Oracle SOA uses Quartz to maintain its jobs and schedules in the database. Synchronize the system clocks for the SOA WebLogic cluster to enable proper functioning of jobs, adapters, and Oracle B2B.

12.7 Extending the Domain to Configure Oracle Identity Manager and Oracle SOA Suite

You must extend your domain to include Oracle Identity Manager. When extending the domain, you must do so from the host that is running the domain's Administration Server. This is the domain IDMDomain on IDMHOST1.

To extend the domain with Oracle Identity Manager, start the configuration wizard on IDMHOST1 by executing the command:

```
ORACLE_COMMON_HOME/common/bin/config.sh
```

Proceed as follows

1. On the Welcome screen, select **Extend an existing WebLogic Domain**.

   Click **Next**.

2. On the Select WebLogic Domain Directory screen, select the location of the domain directory for IDMDomain, for example: `/u01/oracle/config/domains/IDMDomain`

   Click **Next**.

3. On the Select Extension Source screen, select **Extend my domain automatically to support the following added products**. From the list below, select: **Oracle Identity Manager**.

   > **Note:** **Oracle SOA Suite**, **Oracle JRF Webservices Asynchronous Services**, and **Oracle WSM Policy Manager** are selected automatically. If Oracle WSM Policy Manager has already been installed, the choice is not available.

   Select **Next**.

4. On the Configure JDBC Component Schemas screen, do the following.

   Select all the data sources listed on the page:

   - **SOA Infrastructure**

- **User Messaging Service**
- **OIM MDS Schema**
- **OWSM MDS Schema**
- **SOA MDS Schema**
- **OIM Schema**

Select **Convert to GridLink**.

Click **Next**.

5. The Gridlink RAC Component Schema screen appears. In this screen, enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU.

   Select all the schemas for your component. Do not select schemas listed for previously configured components.

   For each entry provide the following common information.

   - **Driver**: Select Oracle's driver (Thin) for GridLink Connections,Versions:10 and later.
   - Select **Enable FAN**.
   - Do one of the following:
     - If SSL is not configured for ONS notifications to be encrypted, deselect **SSL**.
     - Select **SSL** and provide the appropriate wallet and wallet password.
   - **Service Listener**: Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the parameter `remote_listener` in the database:

     ```
     SQL>show parameter remote_listener;

     NAME            TYPE   VALUE
     -------------------------------------------------------------
     remote_listener string DB-SCAN.mycompany.com:1521
     ```

     Note: For Oracle Database 11*g* Release 1 (11.1), use the virtual IP and port of each database instance listener, for example:

     ```
     DBHOST1-vip.mycompany.com (port 1521)
     ```

     and

     ```
     DBHOST2-vip.mycompany.com (port 1521)
     ```

     For Oracle Database 10*g*, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources see "Verifying Adapters for Multiple Directory Identity Stores by Using ODSM" in *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.

   - **ONS Host**: Enter the SCAN address for the Oracle RAC database and the ONS remote port as reported by the database:

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

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

```
DBHOST1.mycompany.com (port 6200)
```

and

```
DBHOST2.mycompany.com (port 6200)
```

Enter the following RAC component schema information:

Table 12–2 RAC Component Schema Information

Schema Name	Service Name	User Name	Password
`OIM Schema`	`oimedg.mycompany.com`	EDG_OIM	*password*
`SOA Infrastructure`	`oimedg.mycompany.com`	EDG_SOAINFRA	*password*
`User Messaging Service`	`oimedg.mycompany.com`	EDG_ORASDPM	*password*
`OIM MDS Schema`	`oimedg.mycompany.com`	EDG_MDS	*password*
`SOA MDS Schema`	`oimedg.mycompany.com`	EDG_MDS	*password*
`OPSS Schema`	`oimedg.mycompany.com`	EDG_OPSS	*password*

If you prefer to use RAC multi datasources, see "Verifying Adapters for Multiple Directory Identity Stores by Using ODSM" in *Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite*.

Click **Next**.

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

   Click **Next**.

7. On the Select Optional Configuration screen, Select:

   - **JMS Distributed Destination**
   - **Managed Servers, Clusters and Machines**
   - **JMS File Store**

Click **Next**.

8. On the JMS Distributed Destination screen, ensure that all the JMS system resources listed on the screen are uniform distributed destinations. If they are not, select **UDD** form the drop down box. Ensure that the entries look like this:

JMS System Resource	Uniform/Weighted Distributed Destination
UMSJMSSystemResource	`UDD`
SOAJMSModule	`UDD`
OIMJMSModule	`UDD`
BPMJMSModule	`UDD`

Click **Next**.

An Override Warning box with the following message is displayed:

```
CFGFWK-40915: At least one JMS system resource has been selected for conversion
to a Uniform Distributed Destination (UDD). This  conversion will take place
only if the JMS System resource is assigned to a cluster
```

Click **OK** on the Override Warning box.

9. When you first enter the Configure Managed Servers screen, two managed servers called oim_server1 and soa_server1 are created automatically. Rename soa_server1 to WLS_SOA1 and oim_server1 to WLS_OIM1 and update their attributes as shown in the following table. Then, add two new managed servers called WLS_OIM2 and WLS_SOA2 with the following attributes.

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_SOA1	SOAHOST1VHN	`8001`	N/A	No
WLS_SOA2	SOAHOST2VHN	`8001`	N/A	No
WLS_OIM1	OIMHOST1VHN	`14000`	N/A	No
WLS_OIM2	OIMHOST2VHN	`14000`	N/A	No

To keep track of ports, host names, and other details for your enterprise deployment, see Appendix A, "Worksheet for Identity Management Topology.".

Notes:

- Do not change the configuration of the managed servers that were configured as a part of previous deployments.
- Do not delete the default managed servers that are created. Rename them as described.

10. On the Configure Clusters screen, create each cluster by clicking **Add**. Supply the following information:

Table 12–3 Cluster Configurations

Name	Messaging Mode	Multicast Address	Multicast Port	Cluster Address
oim_cluster	unicast	n/a	n/a	OIMHOST1VHN:14000,OIMHOST2VHN:14000
soa_cluster	unicast	n/a	n/a	SOAHOST1VHN:8001,SOAHOST2VHN:8001

Leave all other fields at the default settings and click **Next**.

> **Note:** Do not change the configuration of the clusters that were configured as a part of previous deployments.

11. On the Assign Servers to Clusters screen, associate the managed servers with the cluster. Click the cluster name in the right pane. Click the managed server under **Servers**, then click the arrow to assign it to the cluster. Assign the following values:

Table 12–4 Servers to Assign to Clusters

Cluster	Server
oim_cluster	WLS_OIM1
	WLS_OIM2
soa_cluster	WLS_SOA1
	WLS_SOA2

> **Note:** Do not make any changes to clusters that already have entries defined.

Click **Next**.

12. On the Configure Machines screen, create a machine for each host in the topology.
    a. Click the **Unix Machine** tab.
    b. **Name**: Name of the host. Best practice is to use the DNS name.
    c. **Node Manager Listen Address**: DNS name of the machine.
    d. **Node Manager Port**: Port for Node Manager

    Provide the information shown in the following table.

Name	Node Manager Listen Address	Node Manager Listen Port
`IDMHOST1`	`IDMHOST1`	`5556`
`IDMHOST2`	`IDMHOST2`	`5556`

Leave the default values for all other fields.

Delete the default local machine entry under the **Machines** tab.

Click **Next**.

13. On the Assign Servers to Machines screen, assign servers to machines as follows:
    - **IDMHOST1**: `WLS_OIM1` and `WLS_SOA1`
    - **IDMHOST2**: `WLS_OIM2` and `WLS_SOA2`

    Click **Next** to continue.

14. On the Configure JMS File Stores screen, update the directory locations for the JMS file stores. Provide the information shown in the following table.

Name	Directory
UMSJMSFileStore_auto_1	*ASERVER_HOME*/jms/UMSJMSFileStore_auto_1
UMSJMSFileStore_auto_2	*ASERVER_HOME*/jms/UMSJMSFileStore_auto_2
BPMJMSServer_auto_1	*ASERVER_HOME*/jms/BPMJMSServer_auto_1
BPMJMSServer_auto_2	*ASERVER_HOME*/jms/BPMJMSServer_auto_2
SOAJMSFileStore_auto_1	*ASERVER_HOME*/jms/SOAJMSFileStore_auto_1
SOAJMSFileStore_auto_2	*ASERVER_HOME*/jms/SOAJMSFileStore_auto_2
OIMJMSFileStore_auto_1	*ASERVER_HOME*/jms/OIMJMSFileStore_auto_1
OIMJMSFileStore_auto_2	*ASERVER_HOME*/jms/OIMJMSFileStore_auto_2

Click **Next**.

15. On the Configuration Summary screen, click **Extend** to extend the domain.
16. On the Installation Complete screen, click **Done**.
17. Restart WebLogic Administration Server, as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

12.8 Deploying Oracle Identity Manager and Oracle SOA to Managed Server Domain Directory on IDMHOST1 and IDMHOST2

Once the configuration is complete, you must propagate the Oracle Identity Manager configuration to the managed server directory on IDMHOST1 and IDMHOST2.

You do this by packing and unpacking the domain. You pack the domain first on IDMDomain on IDMHOST1, then unpack it on IDMHOST1 and IDMHOST2.

Follow these steps to propagate the domain to the managed server domain directory.

1. Invoke the pack utility from *ORACLE_COMMON_HOME*`/common/bin/` on IDMHOST1.

```
./pack.sh -domain=ASERVER_HOME -template=oim_domain.jar -template_name="OIM
Domain" -managed=true
```

2. This creates a file called `oim_domain.jar`. Copy this file to IDMHOST2.

3. On IDMHOST1 and IDMHOST2, invoke the utility unpack, which is also located in the directory: `ORACLE_COMMON_HOME/common/bin/`

   ```
   ./unpack.sh -domain=MSERVER_HOME -template=oim_domain.jar -overwrite_
   domain=true -app_dir=MSERVER_HOME/applications
   ```

12.9 Configuring Oracle Coherence for Deploying Composites

Although deploying composites uses multicast communication by default, Oracle recommends using unicast communication in SOA enterprise deployments. Use unicast if you disable multicast communication for security reasons.

Unicast communication does not enable nodes to discover other cluster members in this way. Consequently, you must specify the nodes that belong to the cluster. You do not need to specify all of the nodes of a cluster, however. You need only specify enough nodes so that a new node added to the cluster can discover one of the existing nodes. As a result, when a new node has joined the cluster, it is able to discover all of the other nodes in the cluster. Additionally, in configurations such as SOA enterprise deployments where multiple IPs are available in the same system, you must configure Oracle Coherence to use a specific host name to create the Oracle Coherence cluster.

> **Note:** An incorrect configuration of the Oracle Coherence framework used for deployment may prevent the SOA system from starting. The deployment framework must be properly customized for the network environment on which the SOA system runs. Oracle recommends the configuration described in this section.

This section contains the following topics:

- Section 12.9.1, "Enabling Communication for Deployment Using Unicast Communication"
- Section 12.9.2, "Specifying the Host Name Used by Oracle Coherence"

12.9.1 Enabling Communication for Deployment Using Unicast Communication

Specify the nodes using the `tangosol.coherence.wka<n>` system property, where *<n>* is a number between 1 and 9. You can specify up to 9 nodes. Start the numbering at 1. This numbering must be sequential and must not contain gaps. In addition, specify the host name used by Oracle Coherence to create a cluster through the `tangosol.coherence.localhost` system property. This local host name should be the virtual host name used by the SOA server as the listener addresses (SOAHOST1VHN and SOAHOST2VHN). Set this property by adding the `-Dtangosol.coherence.localhost` parameters to the Arguments field of the Oracle WebLogic Server Administration Console's Server Start tab.

> **Tip:** To guarantee high availability during deployments of SOA composites, specify enough nodes so that at least one of them is running at any given time.

> **Note:** SOAHOST1VHN is the virtual host name that maps to the virtual IP where WLS_SOA1 listening (in SOAHOST1). SOAHOST2VHN is the virtual host name that maps to the virtual IP where WLS_SOA2 is listening (in SOAHOST2).

12.9.2 Specifying the Host Name Used by Oracle Coherence

Use the Administration Console to specify a host name used by Oracle Coherence.

To add the host name used by Oracle Coherence:

1. Log into the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Environment** node.
3. Click **Servers**. The Summary of Servers page appears.
4. Click the name of the server (**WLS_SOA1** or **WLS_SOA2**, which are represented as hyperlinks) in Name column of the table. The settings page for the selected server appears.
5. Click **Lock & Edit**.
6. Click the **Server Start** tab.
7. Enter the following for WLS_SOA1 and WLS_SOA2 into the Arguments field.

   For WLS_SOA1, enter the following:

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

   For WLS_SOA2, enter the following:

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

Note: There should be no breaks in lines between the different `-D` parameters. Do not copy or paste the text to your Administration Console's arguments text field. It may result in HTML tags being inserted in the Java arguments. The text should not contain other text characters than those included the example above.

Note: The Coherence cluster used for deployment uses port 8088 by default. This port can be changed by specifying a different port (for example, 8089) with the -Dtangosol.coherence.wkan.port and -Dtangosol.coherence.localport startup parameters. For example:

WLS_SOA1 (enter the following into the Arguments field on a single line, without a carriage return):

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

WLS_SOA2 (enter the following into the Arguments field on a single line, without a carriage return):

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

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

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

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.

9. Stop the WebLogic Administration Server on IDMHOST1. by using the WebLogic Administration Console as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."
10. Start the Administration Server on IDMHOST1 using the Node Manager, as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."
11. Start SOA server `WLS_SOA1`.
12. If desired, start other servers that you shut down in Section 12.3, "Prerequisites."

12.10 Configuring Oracle Identity Manager

You must configure the Oracle Identity Manager server instance before you can start the Oracle Identity Manager and SOA Managed Servers. This is performed on IDMHOST1. The Oracle Identity Management Configuration Wizard loads the Oracle Identity Manager metadata into the database and configures the instance.

Before proceeding, ensure that the following are true:

- The Administration Server is up and running.
- The environment variables *MSERVER_HOME* and *WL_HOME* are ***not*** set in the current shell.

The Oracle Identity Management Configuration Wizard is located under the Identity Management Oracle home. To start the Configuration Wizard, type:

```
IAM_ORACLE_HOME/bin/config.sh
```

Proceed as follows:

1. On the Welcome screen, click **Next**
2. On the Components to Configure screen, Select **OIM Server**.

   Click **Next**.
3. On the Database screen, provide the following values:

   - **Connect String**: The connect string for the Oracle Identity Manager database:

     ```
     IDMDB1-VIP.mycompany.com:1521:OIMEDG1^IDMDB2-VIP.mycompany.com:1521:OIMEDG2
     @OIMEDG.mycompany.com
     ```

     Where 1521 is the `DB_LSNR_PORT` port from Section A.3.

     If you are using Oracle Database 11.2, replace the `vip` address and port with the 11.2 SCAN address and port.
   - **OIM Schema User Name**: `EDG_OIM`
   - **OIM Schema password**: *password*
   - **MDS Schema User Name**: `EDG_MDS`
   - **MDS Schema Password**: *password*

   Click **Next**.
4. On the WebLogic Administration Server screen, provide the following details for the WebLogic Administration Server:

   - **URL**: The URL to connect to the WebLogic Administration Server. For example:

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

     Where Port 7001 is `WLS_ADMIN_PORT`
   - **UserName**: `weblogic`
   - **Password**: Password for the `weblogic` user

   Click **Next**.
5. On the OIM Server screen, provide the following values:

   - **OIM Administrator Password**: Password for the Oracle Identity Manager Administrator. This is the password for the `xelsysadm` user. The password must contain an uppercase letter and a number. Best practice is to use the same password that you assigned to the user `xelsysadm` in Section 10.4, "Preparing the Identity Store."

- **Confirm Password**: Confirm the password·
- **OIM HTTP URL**: Proxy URL for the Oracle Identity Manager Server. For example: `http://IDMINTERNAL.mycompany.com:7777`.
- **Enable LDAP Sync**: Selected.

Click **Next**.

6. On the LDAP Server Screen, the information you enter is dependent on your implementation. Provide the following details:

   - **Directory Server Type**: `OUD`, if your Identity Store is Oracle Unified Directory.
   - **Directory Server ID**: A name for your directory server. For example: `IdStore`. This is only required if the directory type is `OUD`.
   - **Server URL**: The LDAP server URL. For example: `ldap://OUDINTERNAL.mycompany.com:1489`
   - **Server User**: The user name for connecting to the LDAP Server. For example: `cn=oimLDAP,cn=systemids,dc=mycompany,dc=com`
   - **Server Password**: The password for connecting to the LDAP Server.
   - **Server Search DN**: The Search DN, if you are accessing your IDStore using Oracle Unified Directory Server. For example: `dc=mycompany,dc=com`.

   Click **Next**.

7. On the LDAP Server Continued screen, provide the following LDAP server details:

   - **LDAP Role Container**: The DN for the Role Container. This is the container where the Oracle Identity Manager roles are stored. For example: `cn=Groups,dc=mycompany,dc=com`
   - **LDAP User Container**: The DN for the User Container. This is the container where the Oracle Identity Manager users are stored. For example: `cn=Users,dc=mycompany,dc=com`
   - **User Reservation Container**: The DN for the User Reservation Container. For example: `cn=Reserve,dc=mycompany,dc=com`.

   Click **Next**.

8. On the Configuration Summary screen, verify the summary information.

   Click **Configure** to configure the Oracle Identity Manager instance

9. On the Configuration Progress screen, once the configuration completes successfully, click **Next**.

10. On the Configuration Complete screen, view the details of the Oracle Identity Manager Instance configured.

    Click **Finish** to exit the Configuration Wizard.

11. Restart WebLogic Administration Server, as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

12.11 Copy SOA Directory

Copy the `soa` directory located under *`ASERVER_HOME`* on IDMHOST1 to *`MSERVER_HOME`* directory on IDMHOST1 and IDMHOST2.

For example:

```
scp -rp ASERVER_HOME/soa user@IDMHOST2:MSERVER_HOME
```

12.12 Starting SOA and Oracle Identity Manager Managed Servers on IDMHOST1 and IDMHOST2

Follow this sequence of steps to start the WLS_OIM1 and WLS_SOA1 Managed Servers on IDMHOST1:

1. Validate that the Administration Server started up successfully by bringing up the Oracle WebLogic Administration Console.
2. If it is not already started, start the WLS_SOA1 Managed Server, using the WebLogic Administration Console as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."
3. Start the WLS_OIM1 Managed Server using the WebLogic Administration Console as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

Follow this sequence of steps to start the WLS_OIM2 and WLS_SOA2 Managed Servers on IDMHOST2:

1. Validate that the Administration Server started up successfully by bringing up the Oracle WebLogic Administration Console.
2. Start the WLS_SOA2 Managed Server, using the WebLogic Administration Console as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."
3. Start the WLS_OIM2 Managed Server using the WebLogic Administration Console as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

12.13 Validating Oracle Identity Manager Instance on IDMHOST1 and IDMHOST2

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

```
http://OIMHOST1VHN.mycompany.com:14000/identity
```

```
http://OIMHOST1VHN.mycompany.com:14000/sysadmin
```

```
http://OIMHOST2VHN.mycompany.com:14000/identity
```

```
http://OIMHOST2VHN.mycompany.com:14000/sysadmin
```

Log in using the `xelsysadm` username and password.

> **Note:** When you log in for the first time, you are prompted to setup Challenge Questions. Please do so before proceeding further.

Validate Oracle SOA Suite using the URLs:

`http://SOAHOST1VHN.mycompany.com:8001/soa-infra`

`http://SOAHOST2VHN.mycompany.com:8001/soa-infra`

Log in as the `weblogic` user.

12.14 Configuring Oracle Identity Manager to Reconcile from OUDINTERNAL

In the current release, the `LDAPConfigPostSetup` script enables all the LDAPSync-related incremental Reconciliation Scheduler jobs, which are disabled by default. The LDAP configuration post-setup script is located under the *`IAM_ORACLE_HOME`*`/server/ldap_config_util` directory. Run the Script on IDMHOST1, as follows:

1. Edit the `ldapconfig.props` file located under the *`IAM_ORACLE_HOME`*`/server/ldap_config_util` directory and provide the following values:

Parameter	Value	Description
`OIMProviderURL`	`t3://OIMHOST1VHN.mycompany.com:14000,OIMHOST2VHN.mycompany.com:14000`[1]	List of Oracle Identity Manager managed servers.
`LIBOVD_PATH_PARAM`	*`MSERVER_HOME`*`/config/fmwconfig/ovd/oim`	Required unless you access your identity store using Oracle Virtual Directory.

[1] Where `14000` is the *`OIM_PORT`* from Section A.3 .

> **Note:** `usercontainerName`, `rolecontainername`, and `reservationcontainername` are not used in this step.

2. Save the file.
3. Set `MW_HOME` to *`IAM_MW_HOME`*.

   Set `ORACLE_HOME` to *`IAM_ORACLE_HOME`*.

   Set `JAVA_HOME` to *`JAVA_HOME`*.

   Set `WL_HOME` to *`MW_HOME`*`/wlserver_10.3`.

   Set `APP_SERVER` to `weblogic`.

   Set `OIM_ORACLE_HOME` to *`IAM_ORACLE_HOME`*.

   Set `DOMAIN_HOME` set *`MSERVER_HOME`*.
4. Run LDAPConfigPostSetup.sh. The script prompts for the LDAP admin password and the Oracle Identity Manager admin password. For example:

   ```
   IAM_ORACLE_HOME/server/ldap_config_util/LDAPConfigPostSetup.sh path_to_
   property_file
   ```

   For example:

   ```
   IAM_ORACLE_HOME/server/ldap_config_util/LDAPConfigPostSetup.sh IAM_ORACLE_
   ```

```
HOME/server/ldap_config_util
```

Example output:

```
Successfully Enabled Changelog based Reconciliation schedule jobs.
```

12.15 Configuring Oracle Identity Manager to Work with the Oracle Web Tier

This section describes how to configure Oracle Identity Manager to work with the Oracle Web Tier.

This section contains the following topics:

- Section 12.15.1, "Configuring Oracle Traffic Director to Front End the Oracle Identity Manager and SOA Managed Servers"
- Section 12.15.2, "Changing Host Assertion in WebLogic"

12.15.1 Configuring Oracle Traffic Director to Front End the Oracle Identity Manager and SOA Managed Servers

If you are adding OIM to an existing domain you must include OIM in the Web Tier configuration. For more information see Section 7.7, "Defining the Required Oracle Traffic Director Virtual Servers for an Enterprise Deployment."

12.15.2 Changing Host Assertion in WebLogic

Because the Oracle HTTP Server acts as a proxy for WebLogic, by default certain CGI environment variables are not passed through to WebLogic. These include the host and port. You must tell WebLogic that it is using a virtual site name and port so that it can generate internal URLs appropriately.

To do this, log in to the WebLogic administration console at the URL listed in Section 16.2, "About Identity Management Console URLs." Proceed as follows:

1. Select **Clusters** from the home page or, alternatively, select **Environment** -> **Clusters** from the **Domain** structure menu.
2. Click **Lock and Edit** in the Change Center Window to enable editing.
3. Click the **Cluster Name** (**soa_cluster**).
4. In the **Configuration** tab, select the **HTTP** subtab.

   Enter:

   - **Frontend Host**: `IDMINTERNAL.mycompany.com`
   - **Frontend HTTP Port**: 7777 (*HTTP_PORT*)
5. Click **Save**.
6. Click **Activate Changes** in the Change Center window to enable editing.
7. Restart WLS_SOA1 and WLS_SOA2 as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

12.15.3 Updating SOA Endpoints

Update SOA endpoints, as follows:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control at the address listed in Section 16.2, "About Identity Management Console URLs."
2. Expand the **SOA** folder in the Navigation pane and right click **soa-infra**
3. Select **SOA Administration -> Common Properties**
4. Click on the link **More SOA Infra Advanced Configuration Properties**.
5. Edit the following properties and apply the changes:
   - **ServerURL**: `http://idminternal.mycompany.com:7777`
   - **CallbackServerURL**: `http://idminternal.mycompany.com:7777`
   - **HttpServerURL**: `http://idminternal.mycompany.com:7777`
6. Click **Apply**.
7. Restart WLS_SOA1 and WLS_SOA2 as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

12.15.4 Validating Web Tier Integration

Validate web tier integration as follows:

12.15.4.1 Validating Oracle Identity Manager Instance from the Web Tier

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

```
https://sso.mycompany.com/identity
```

and

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

Log in using the `xelsysadm` username and password.

12.15.4.2 Validating Accessing SOA from the Web Tier

Validate SOA by accessing the URL:

```
http://IDMINTERNAL.mycompany.com:7777/soa-infra
```

and logging in as the WebLogic administration user.

Note: After WebGate is enabled, **soa-infra** is not available.

12.16 Configuring a Default Persistence Store for Transaction Recovery

The WLS_OIM and WLS_SOA Managed Servers have a transaction log that stores information about committed transactions that are coordinated by the server that might not have been completed. The WebLogic Server uses this transaction log for recovery from system crashes or network failures. To leverage the migration capability of the Transaction Recovery Service for the servers within a cluster, store the transaction log in a location accessible to a server and its backup servers.

Note: Preferably, this location should be on a dual-ported SCSI disk or on a Storage Area Network (SAN).

Perform these steps to set the location for the default persistence stores for the Oracle Identity Manager and SOA Servers:

1. Create the following directory on the shared storage:

   ASERVER_HOME/tlogs

2. Log in to the Oracle WebLogic Server Administration Console.
3. Click **Lock and Edit**.
4. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node.

   The Summary of Servers page is displayed.

5. Click the name of either the Oracle Identity Manager or the SOA server (represented as a hyperlink) in the **Name** column of the table.
6. The Settings page for the selected server is displayed, and defaults to the **Configuration** tab.
7. Open the **Services** sub tab.
8. Under the **Default Store** section of the page, provide the path to the default persistent store on shared storage. The directory structure of the path is as follows:

   - For Oracle Identity Manager Servers: *ASERVER_HOME*/tlogs
   - For SOA Servers: *ASERVER_HOME*/tlogs

   > **Note:** To enable migration of the Transaction Recovery Service, specify a location on a persistent storage solution that is available to other servers in the cluster. All the servers that are a part of the cluster must be able to access this directory.

9. Click **Save and Activate**.
10. Repeat these steps, selecting the other SOA server on the Summary of Servers page.
11. Restart the Oracle Identity Manager and SOA Managed Servers, as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components." to make the changes take effect.

12.17 Configuring UMS Email Notification

This section describes how to configure UMS email notification. This is optional. The following steps assume that an email server has been set up and that Oracle Identity Management can use it to send the email notifications.

1. Log in to the Oracle Enterprise Manager Fusion Middleware Control instance that is associated with Oracle Identity Manager, at the URL listed in Section 16.2, "About Identity Management Console URLs.".
2. Expand **User Messaging Service**.
3. Right click **usermessagingdriver-email (wls_soa1)** and select **email driver properties**.
4. Enter the following information:

- **OutgoingMailServer**: name of the SMTP server, for example: SMTP.mycompany.com
- **OutgoingMailServerPort**: port of the SMTP server, for example: 465 for SSL outgoing mail server and 25 for non-SSL
- **OutgoingMailServerSecurity**: The security setting used by the SMTP server Possible values can be None/TLS/SSL. If the mail server is configured to accept SSL requests, perform these additional steps to remove DemoTrust store references from the SOA environment:

  a. Modify the `ASERVER_HOME/bin/setDomainEnv.sh` file to remove the DemoTrust references `-Djavax.net.ssl.trustStore=WL_HOME/server/lib/DemoTrust.jks` from `EXTRA_JAVA_PROPERTIES`.

  b. Modify the `startManagedWeblogic.sh` file on IDMHOST1 and IDMHOST2. Remove the `weblogic.security.SSL.trustedCAKeyStore` property set in `JAVA_OPTIONS` from this file. That is, remove the line that looks like this:

  ```
  JAVA_OPTIONS="-Dweblogic.security.SSL.trustedCAKeyStore="{MW_
  HOME}/server/server/lib/cacerts" ${JAVA_OPTIONS}"
  ```

  c. Restart Oracle Identity Manager and the OIM and SOA managed servers.

- **OutgoingUsername**: Any valid username
- **OutgoingPassword:**

  a. Choose **Indirect Password**, **Create New User**

  b. Provide a unique string for **Indirect Username/Key**, for example: `OIMEmailConfig`. This will mask the password and not expose it in clear text in the configuration file.

  c. Provide valid password for this account.

Click **Apply**.

Repeat Steps 3 and 4 for each SOA server.

5. From the Navigator Select **WebLogic Domain** -> **DomainName**.
6. From the menu, select **System Mean Browser**.
7. Expand **Application Defined MBeans** -> **oracle.iam** -> **Server: wls_oim1** -> **Application: oim** -> **IAMAppRuntimeMBean**.
8. Click **UMSEmailNotificationProviderMBean**.
9. Enter:
   - **WSUrl**: `http://IDMINTERNAL.mycompany.com:7777/ucs/messaging/webservice`
   - **Policies**: Leave blank.
   - **CSFKey**: `Notification.Provider.Key`
10. Click **Apply**.

12.18 Add Load Balancer Certificate to SOA Keystore

Using a browser, obtain the certificate for SSO.mycompany.com. (Refer to your browser documentation to determine how to do this.) Save the file to IDMHOST1 in the `.pem` format, for example: `/tmp/sso.pem`.

Then import the certificate into the SOA keystore using the `keytool` command, which is provided as part of the JDK (Java Development Kit). Proceed as follows:

1. Set the environment variables.
   - Set `JAVA_HOME` to *`JAVA_HOME`*.
   - Set `PATH` to *`JAVA_HOME/bin:$PATH`*.
2. Change directory to *`WL_HOME`*`/server/lib`.

   ```
   cd WL_HOME/server/lib
   ```

3. Add the certificate to the SOA keystore using the following command:

   ```
   keytool -import -file /tmp/sso.pem -alias SSOAlias -keystore DemoTrust.jks
   -storepass DemoTrustKeyStorePassPhrase
   ```

To add this ceritifcate using CLI commands, run the following:

```
openssl x509 -in <(openssl s_client -connect SSO.mycompany.com:443 -prexit
2>/dev/null) > /tmp/sso.pem
```

12.19 Excluding Users from Oracle Identity Manager Reconciliation

By default Oracle Identity Management reconciles all users that are located in LDAP. Once reconciled, these users are subject to the usual password ageing policies defined in Oracle Identity Manager. This is not desirable for system accounts. It is recommended that you exclude the following accounts from this reconciliation:

In the container `cn=Users`:

- `xelsysadm`

In the container `cn=systemids`:

- `oimLDAP`
- `oamLDAP`

To exclude these users from reconciliation and discard failed reconciliation events, perform the following steps, using ODSM and the OIM Console:

12.19.1 Adding the orclAppIDUser Object Class to the User by Using ODSM

Users can be excluded from OIM reconciliation by attaching the object class `orclAppIDUser` to each of the users.

The example below is for Oracle Unified Directory using ODSM for Oracle Unified Directory. For directories other than Oracle Unified Directory refer to your system documentation for information on how to do this.

1. Log in to ODSM at:

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

2. Connect to one of the LDAP instances that hosts the user to be excluded.

- **Server**: One of the Oracle Unified Directory hosts, for example: `IDMHOST1.mycompany.com`
- **Administration Port**: The Oracle Unified Directory administration port, for example: `4444`
- **User Name**: Directory Administrator, for example: `cn=oudadmin`

If prompted, trust the server certificate.

3. Select **Data Browser**.
4. Navigate to the user you wish to exclude in the data tree. For example:

   Root -> **dc=mycompany,dc=com** -> **cn=systemids** -> **cn=UserId**
5. Click on the user to bring up the Edit window.
6. Click **Attributes**.
7. Click **+** in the Object Classes box to add a new class.
8. Click **Advanced Search**, enter `orclAppIDUser` in the search box, and click **Search**.
9. Click on the attribute **orclAppIDUser** and click **OK**.
10. Click **Apply**.

Repeat Steps 2-10 for each user to be excluded.

12.19.2 Closing Failed Reconciliation Events by Using the OIM Console

This step is required to clear out failed reconciliation events. Failed reconcilation events are repeatedly retried, which puts an unecessary load on the system.

1. Log in to the OIM Administration Console as the `xelsysadm` user, using the URL: `http://admin.mycompany.com/sysadmin`
2. Click **Reconciliation** under **Event Management**.
3. Click **Advanced Search**.
4. In the **Current Status** field, select **Equals**. In the **Search** box, select **Creation Failed** from the list.
5. Click **Search**.
6. Select each of the events.
7. From the Actions menu, select **Close Event**.
8. In the Confirmation window enter a justification, such as `Close Failed Reconciliation Events`.
9. Click **Closed**.
10. Click **OK** to acknowledge the confirmation message.

12.20 Backing Up Oracle Identity Manager

Perform a backup of the Oracle Identity Manager configuration at this point. Back up the database, the WebLogic domain, and the LDAP directories, as described in Section 16.6, "Backing Up the Oracle IDM Enterprise Deployment."

12.21 Integrating Oracle Identity Manager and Oracle Access Management Access Manager

This section describes how to integrate Oracle Identity Manager and Oracle Access Management Access Manager.

> **Note:** If you are adding Oracle Identity Manager to an existing domain that already has Access Manager, then if you have not already done so run the command as described in Section 11.5.3, "Configuring Access Manager by Using the IDM Configuration Tool" with the Oracle Identity Manager integration parameters

This section contains the following topics:

- Section 12.21.1, "Prerequisites"
- Section 12.21.2, "Adding Forgotten Password Links to the OAM Login Page"
- Section 12.21.3, "Copying OAM Keystore Files to IDMHOST1 and IDMHOST2"
- Section 12.21.4, "Integrating Oracle Identity Manager with Oracle Access Manager Using the idmConfigTool"
- Section 12.21.5, "Updating Existing LDAP Users with Required Object Classes."
- Section 12.21.6, "Update TAP Authentication Scheme"
- Section 12.21.7, "Managing the Password of the xelsysadm User."
- Section 12.21.8, "Enabling Cluster-Level Session Replication Enhancements for OIM and SOA."
- Section 12.21.9, "Validating Integration."

12.21.1 Prerequisites

1. Ensure that OIM11g has been installed and configured as described in Chapter 12, "Extending the Domain to Include Oracle Identity Manager."
2. Ensure that Oracle Access Management has been installed and configured as described in Chapter 11, "Extending the Domain to Include Oracle Access Management."
3. Ensure that Oracle Traffic Director has been installed and configured as described in Chapter 7, "Installing and Configuring Oracle Traffic Director for an Enterprise Deployment." Or, Ensure that Oracle Traffic Director has been installed and configured as described in Section 7.2, "Installing Oracle Traffic Director on WEBHOST1 and WEBHOST2."

12.21.2 Adding Forgotten Password Links to the OAM Login Page

If you ran `idmConfigTool` in Section 11.5.3, "Configuring Access Manager by Using the IDM Configuration Tool" with the parameter `OAM11G_OIM_INTEGRATION_REQ` is set to `true`, you can skip this step.

If you ran the command with `OAM11G_INTEGRATION_FLAG` set to **false**, you must now rerun the command, this time setting `OAM11G_OIM_INTEGRATION_REQ` to **true** and specifying a value for `OAM11G_OIM_OHS_URL`.

12.21.3 Copying OAM Keystore Files to IDMHOST1 and IDMHOST2

If you are using Access Manager with the Simple Security Transport model, you must copy the OAM keystore files that were generated in Section 11.9, "Creating a Single Keystore for Integrating Access Manager with Other Components" to IDMHOST1 and IDMHOST2. Copy the keystore files `ssoKeystore.jks` and `oamclient-truststore.jks` from the directory *ASERVER_HOME*`/output/webgate-ssl`to the directory *MSERVER_HOME*`/config/fmwconfig` on IDMHOST1 and IDMHOST2.

12.21.4 Integrating Oracle Identity Manager with Oracle Access Manager Using the idmConfigTool

Integrating Oracle Identity Manager with Access Manager using a WebGate profile employs an Access Manager Trusted Authentication Protocol (TAP) scheme. This is different from previous releases which used Network Assertion Protocol (NAP).

To integrate Access Manager with Oracle Identity Manager, perform the following steps on IDMHOST1:

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

   Set `ORACLE_HOME` to *`IAM_ORACLE_HOME`*.

   Set `JAVA_HOME` to *`JAVA_HOME`*.

2. Create a properties file for the integration called `oimitg.props`, with the following contents.

   ```
   LOGINURI: /${app.context}/adfAuthentication
   LOGOUTURI: /oamsso/logout.html
   AUTOLOGINURI: None
   ACCESS_SERVER_HOST: IDMHOST1.mycompany.com
   ACCESS_SERVER_PORT: 5575
   ACCESS_GATE_ID: Webgate_IDM
   COOKIE_DOMAIN: .mycompany.com
   COOKIE_EXPIRY_INTERVAL: 120
   OAM_TRANSFER_MODE: simple
   WEBGATE_TYPE: ohsWebgate11g
   SSO_ENABLED_FLAG: true
   IDSTORE_PORT: 1489
   IDSTORE_HOST: oudinternal.mycompany.com
   IDSTORE_DIRECTORYTYPE: OUD
   IDSTORE_ADMIN_USER: cn=oamLDAP,cn=systemids,dc=mycompany,dc=com
   IDSTORE_USERSEARCHBASE: cn=Users,dc=mycompany,dc=com
   IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=mycompany,dc=com
   IDSTORE_LOGINATTRIBUTE: uid
   MDS_DB_URL: jdbc:oracle:thin:@(DESCRIPTION=(LOAD_
   BALANCE=on)(FAILOVER=on)(ADDRESS_
   LIST=(ADDRESS=(protocol=tcp)(host=IDMDBHOST1-VIP.mycompany.com)(port=1521))(ADD
   RESS=(protocol=tcp)(host=IDMDBHOST2-VIP.mycompany.com)(port=1521)))(CONNECT_
   DATA=(SERVER=DEDICATED)(SERVICE_NAME=OIMEDG.mycompany.com)))
   MDS_DB_SCHEMA_USERNAME: EDG_MDS
   OIM_MANAGED_SERVER_NAME: WLS_OIM1
   WLSADMIN: weblogic
   WLSPORT: 7001
   WLSHOST: ADMINVHN.mycompany.com
   DOMAIN_NAME: IDMDomain
   DOMAIN_LOCATION: ASERVER_HOME
   ```

   where:

- `ACCESS_SERVER_PORT` is the Access Server Proxy port. This is *`OAM_PROXY_PORT`* in Section A.3.
- `OAM_TRANSFER_MODE` is set to `simple` if your access manager servers are configured to accept requests using the simple mode. Otherwise set `OAM_TRANSFER_MODE` to `open`
- `SSO_ENABLED_FLAG` always set to `true`.
- `WEBGATE_TYPE` is the type of WebGate agent you want to create. Valid values are `otdWebgate11g` and `otdWebgate10`.
- `IDSTORE_HOST` is the load balancer virtual host fronting your Identity store (*`LDAP_LBR_HOST`*)
- `IDSTORE_PORT` is the load balancer virtual port fronting your Identity store (*`LDAP_LBR_PORT`*).
- `IDSTORE_DIRECTORYTYPE` Set it to `OUD`.
- `IDSTORE_USERSEARCHBASE` is the location in the directory where Users are Stored.
- `IDSTORE_GROUPSEARCHBASE` is the location in the directory where Groups are Stored.
- `IDSTORE_LOGINATTRIBUTE` is the LDAP attribute which contains the users Login name.
- `MDS_DB_URL` contains the JDBC connection information for your database in the form: `jdbc:oracle:thin:@(DESCRIPTION=(LOAD_BALANCE=on)(FAILOVER=on)(ADDRESS_LIST=(ADDRESS=(protocol=tcp)(host=IDMDBHOST1-VIP.mycompany.com)(port=1521))(ADDRESS=(protocol=tcp)(host=IDMDBHOST2-VIP.mycompany.com)(port=1521)))(CONNECT_DATA=(SERVER=DEDICATED)(SERVICE_NAME=OIMEDG.mycompany.com)))` where `1521` is the *`DB_LSNR_PORT`* in Section A.3.
- `MDS_DB_SCHEMA_USERNAME` is the name of the schema in the Identity Management Database that holds MDS data. See Section 6.5, "Loading the Identity Management Schemas in the Oracle RAC Database by Using RCU."
- `OIM_MANAGED_SERVER_NAME` is the name of one of the OIM Managed Servers. It does not matter which one you use.
- `WLSHOST` (*ADMINVHN*) is the host of your administration server, *`WLS_ADMIN_HOST`* in Section A.3. This is the virtual name.
- `WLSPORT` is the port of your administration server, *`WLS_ADMIN_PORT`* in Section A.3.
- `WLSADMIN` is the WebLogic administrative user you use to log in to the WebLogic console.
- `DOMAIN_NAME` is the name of the domain that hosts Oracle Identity Manager.
- `DOMAIN_LOCATION` is the path to the domain on disk, that is, *`ASERVER_HOME`*.

3. Integrate Access Manager with Oracle Identity Manager using the command `idmConfigTool`, which is located at:

   `IAM_ORACLE_HOME``/idmtools/bin`

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

IAM_ORACLE_HOME/idmtools/bin

The syntax of the command is

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

For example:

```
IAM_ORACLE_HOME/idmtools/bin/idmConfigTool.sh -configOIM input_
file=oimitg.props
```

When the script runs you are prompted for the following information:

- Access Gate Password
- SSO Keystore Password
- Global Passphrase
- Idstore Admin Password
- MDS Database schema password
- Admin Server User Password

Sample output:

```
Enter sso access gate password :
Enter sso keystore jks password :
Enter sso global passphrase :
Enter mds db schema password :
Enter idstore admin password :
Enter admin server user password :


********* Seeding OAM Passwds in OIM *********


Completed loading user inputs for - CSF Config


Completed loading user inputs for - Dogwood Admin WLS

Connecting to t3://ADMINVHN.mycompany.com:7001

Connection to domain runtime mbean server established

Seeding credential :SSOAccessKey

Seeding credential :SSOGlobalPP

Seeding credential :SSOKeystoreKey


********* ********* *********
```

```
********* Activating OAM Notifications *********


Completed loading user inputs for - MDS DB Config

Apr 3, 2012 11:56:09 PM oracle.mds
NOTIFICATION: PManager instance is created without multitenancy support as JVM
flag "oracle.multitenant.enabled" is not set to enable multitenancy support.
Initialized MDS resources

Apr 3, 2012 11:56:09 PM oracle.mds
NOTIFICATION: PManager instance is created without multitenancy support as JVM
flag "oracle.multitenant.enabled" is not set to enable multitenancy support.
Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: transfer operation started.
Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: transfer is completed. Total number of documents successfully
processed : 1, total number of documents failed : 0.
Upload to DB completed


Releasing all resources

Notifications activated.


********* ********* *********


********* Seeding OAM Config in OIM *********


Completed loading user inputs for - OAM Access Config

Validated input values

Initialized MDS resources

Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: PManager instance is created without multitenancy support as JVM
flag "oracle.multitenant.enabled" is not set to enable multitenancy support.
Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: transfer operation started.
Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: transfer is completed. Total number of documents successfully
processed : 1, total number of documents failed : 0.
Download from DB completed

Releasing all resources

Updated /u01/oracle/products/access/iam/server/oamMetadata/db/oim-config.xml

Initialized MDS resources

Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: PManager instance is created without multitenancy support as JVM
flag "oracle.multitenant.enabled" is not set to enable multitenancy support.
Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: transfer operation started.
```

```
Apr 3, 2012 11:56:10 PM oracle.mds
NOTIFICATION: transfer is completed. Total number of documents successfully
processed : 1, total number of documents failed : 0.
Upload to DB completed


Releasing all resources

OAM configuration seeded. Please restart oim server.


********* ********* *********


********* Configuring Authenticators in OIM WLS *********


Completed loading user inputs for - LDAP connection info

Connecting to t3://ADMINVHN.mycompany.com:7001

Connection to domain runtime mbean server established

Starting edit session

Edit session started

Connected to security realm.

Validating provider configuration

Validated desired authentication providers

Created OAMIDAsserter successfuly

OAMIDAsserter is already configured to support 11g webgate

Created OIMSignatureAuthenticator successfuly

Control flags for authenticators set sucessfully

Reordering of authenticators done sucessfully

Saving the transaction

Transaction saved

Activating the changes

Changes Activated. Edit session ended.

Connection closed sucessfully


********* ********* *********

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

Note: If you have already enabled single sign-on for your WebLogic Administration Consoles as described in Section 13.3, "Enabling Host Name Verification Certificates for Node Manager" when this script is run, you might see the following errors when this script is run:

```
ERROR: Desired authenticators already present.
[Ljava.lang.String;@7fdb492]
ERROR: Error occurred while configuration. Authentication providers
to be configured already present.
ERROR: Rolling back the operation..
```

These errors can be ignored.

4. Check the log file for errors and correct them if necessary.
5. Restart the Administration Servers as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

12.21.5 Updating Existing LDAP Users with Required Object Classes

You must update existing LDAP users with the object classes OblixPersonPwdPolicy, OIMPersonPwdPolicy, and OblixOrgPerson.

Note: This is not required in the case of a fresh setup where you do not have any existing users.

1. On IDMHOST1, create a properties file for the integration called user.props, with the following contents:

```
IDSTORE_HOST: oudinternal.mycompany.com
IDSTORE_PORT: 1489
IDSTORE_ADMIN_USER: cn=oudadmin
IDSTORE_DIRECTORYTYPE: OUD
IDSTORE_USERSEARCHBASE: cn=Users,dc=mycompany,dc=com
IDSTORE_GROUPSEARCHBASE: cn=Groups,dc=mycompany,dc=com
PASSWORD_EXPIRY_PERIOD: 7300
IDSTORE_LOGINATTRIBUTE: uid
```

Where:

- OUDINTERNAL_HOST is the name of LDAP server. For example:

  oudinternal.mycompany.com

- IDSTORE_PORT is the port of the LDAP server.
- IDSTORE_ADMIN_USER is the bind DN of an administrative user. For example:

  cn=oudadmin

- IDSTORE_DIRECTORYTYPE is the type of directory, valid value is OUD.
- IDSTORE_USERSEARCHBASE is the location of users in the directory. For example:

  cn=Users,dc=mycompany,dc=com

- IDSTORE_GROUPSEARCHBASE is the location of groups in the directory. For example:

  cn=Groups,dc=mycompany,dc=com

- IDSTORE_LOGINATTRIBUTE this is the directory login attribute name. For example:

  uid.

- PASSWORD_EXPIRY_PERIOD is the password expiry period.

2. Set ORACLE_HOME to *IAM_ORACLE_HOME*.

   Set MW_HOME to *MW_HOME*.

   Set JAVA_HOME to *JAVA_HOME*.

3. Upgrade existing LDAP, using the command idmConfigTool, which is located at: *IAM_ORACLE_HOME*/idmtools/bin

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

   The syntax of the command is:

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

   For example:

   ```
   idmConfigTool.sh -upgradeLDAPUsersForSSO input_file=user.props
   ```

   When prompted, enter the password of the user you are using to connect to your Identity Store.

   Sample output:

   ```
   Enter IDSTORE_ADMIN_PASSWD :
   ********* Upgrading LDAP Users With OAM ObjectClasses *********


   Completed loading user inputs for - LDAP connection info


   Completed loading user inputs for - LDAP Upgrade

   Upgrading ldap users at - cn=Users,dc=mycompany,dc=com

   Parsing - cn=weblogic_idm,cn=Users,dc=mycompany,dc=com

   objectclass OIMPersonPwdPolicy not present in cn=weblogic_
   idm,cn=Users,dc=mycompany,dc=com. Seeding it

   obpasswordexpirydate added in cn=weblogic_idm,cn=Users,dc=mycompany,dc=com


   Parsing - cn=oamadmin,cn=Users,dc=mycompany,dc=com
   ```

```
objectclass OIMPersonPwdPolicy not present in
cn=oamadmin,cn=Users,dc=mycompany,dc=com. Seeding it

obpasswordexpirydate added in cn=oamadmin,cn=Users,dc=mycompany,dc=com


Finished parsing LDAP


LDAP Users Upgraded.
```

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

12.21.6 Update TAP Authentication Scheme

After integrating Oracle Access Management Access Manager with Oracle Identity Manager, you must update the TAP authentication scheme to perform user validation using the LDAP attribute `uid`.

Proceed as follows:

1. Log in to the OAM console at: `http://ADMIN.mycompany.com/oamconsole`
2. Click **Policy Configuration**.
3. Click **TAPResponseOnlyScheme** under **Authentication Schemes**.
4. Click **Open**.
5. Add `MatchLDAPAttribute=uid` to the **Challenge Parameters** field.
6. Click **Apply**.
7. Restart the Administration Server and the Access Manager managed servers as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

12.21.7 Managing the Password of the xelsysadm User

After you integrate Oracle Identity Manager with Access Manager, two `xelsysadm` accounts exist. One is the internal account created by Oracle Identity Manager. The other is the account you created in the Identity Store in Section 10.4, "Preparing the Identity Store."

The `xelsysadm` account located in the LDAP store is the one used to access the OIM console. If you want to change the password of this account, change it in LDAP. You can use ODSM to do this. Do not change it through the OIM console.

12.21.8 Enabling Cluster-Level Session Replication Enhancements for OIM and SOA

You can enable session replication enhancements for Managed Servers in a WebLogic cluster to which you will deploy a web application at a later time.

To enable session replication enhancements for `oim_cluster`:

1. Log in to the Oracle WebLogic console at:
   `http://ADMIN.mycompany.com/oamconsole`
2. Ensure that Managed Servers in the `oim_cluster` cluster are up and running, as described in Section 12.12, "Starting SOA and Oracle Identity Manager Managed

Servers on IDMHOST1 and IDMHOST2.".

3. To set replication ports for a Managed Server, such as WLS_OIM1, complete the following steps:

   a. Under **Domain Structure**, click **Environment** and **Servers**. The Summary of Servers page is displayed.

   b. Click **Lock & Edit**.

   c. Click WLS_OIM1 on the list of servers. The Settings for WLS_OIM1 are displayed.

   d. Click the **Cluster** tab.

   e. In the **Replication Ports** field, enter a range of ports for configuring multiple replication channels. For example, replication channels for Managed Servers in oim_cluster can listen on ports starting from 7005 to 7015. To specify this range of ports, enter 7005-7015.

   f. Click **Save**.

   g. Select **Protocols**, and then **Channels**.

   h. Click **New**.

   i. Enter **ReplicationChannel** as the name of the new network channel and select **t3** as the protocol, then click **Next**.

   j. Enter the following information:

      Listen address: **OIMHOST1VHN.mycompany.com**

      Note: This is the WLS_OIM1 floating IP assigned to WebLogic Server.

      Listen port: **7005**

   k. Click **Next**, and in the Network Channel Properties page, select **Enabled** and **Outbound Enabled**.

   l. Click **Finish**.

   m. Click **Save**.

   You must repeat the above steps to create a network channel each for the remaining Managed Servers in the cluster. Enter the required properties, as described in Table 12–5.

Table 12–5 Network Channels Properties

Managed Server	Name	Protocol	Listen Address	Listen Port	Additional Channel Ports
WLS_OIM2	ReplicationChannel	t3	OIMHOST2VHN.mycompany.com	7005	7006 to 7014
WLS_SOA1	ReplicationChannel	t3	SOAHOST1VHN.mycompany.com	7005	7006 to 7014

Table 12–5 (Cont.) Network Channels Properties

Managed Server	Name	Protocol	Listen Address	Listen Port	Additional Channel Ports
WLS_SOA2	`ReplicationChannel`	t3	SOAHOST2VHN.mycompany.com	7005	7006 to 7014

4. After creating the network channel for each of the Managed Servers in your cluster, click **Environment** > **Clusters**. The Summary of Clusters page is displayed.
5. Click **oim_cluster**.

   The Settings for oim_cluster page is displayed.
6. Click the **Replication** tab.
7. In the **Replication Channel** field, ensure that `ReplicationChannel` is set as the name of the channel to be used for replication traffic.
8. In the **Advanced** section, select the **Enable One Way RMI for Replication** option.
9. Click **Save**.
10. Repeat the steps above for the **soa_cluster**.
11. To activate these changes, in the Change Center of the Administration Console, click **Activate Changes**.
12. Manually add the system property `-Djava.net.preferIPv4Stack=true` to the **startWebLogic.sh** script, which is located in the `bin` directory of `ASERVER_HOME`, using a text editor as follows:
    1. Locate the following line in the startWebLogic.sh script:

       ```
       . ${DOMAIN_HOME/bin/setDomainEnv.sh $*
       ```
    2. Add the following property immediately after the above entry:

       ```
       JAVA_OPTIONS="${JAVA_OPTIONS}
       -Djava.net.preferIPv4Stack=true"
       ```
    3. Save the file and close.
13. Restart the Administration Server and the Access Manager managed servers as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

12.21.9 Validating Integration

To validate integration, you must assign Identity Management administrators to WebLogic security groups and install WebGate as described in Chapter 15, "Configuring Single Sign-on for Administration Consoles in an Enterprise Deployment."

To validate that the wiring of Access Manager with Oracle Identity Manager 11g was successful, attempt to log in to the Oracle Identity Manager Self Service Console, as follows:

1. Using a browser, navigate to:

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

   This redirects you to the OAM11*g* single sign-on page.

2. Log in using the `xelsysadm` user account created in Section 10.4, "Preparing the Identity Store."
3. If you see the OIM Self Service Console Page, the integration was successful.

You can perform additional validation as follows:

1. Log in to the OIM Console as the `xelsysadm` user.
2. Create a new user.
3. Log out as the `xelsysadm` user.
4. Log in as the new user you just created. As the new user, you are redirected to the Password Management page.
5. Enter the credentials and click **Submit**. If integration has been performed correctly, you arrive at the page you are trying to access.

12.22 Enabling Oracle Identity Manager to Connect to SOA Using the Administrative Users Provisioned in LDAP

Oracle Identity Manager connects to SOA as SOA administrator, with the username `weblogic` by default. As mentioned in the previous sections, a new administrator user is provisioned in the central LDAP store to manage Identity Management Weblogic Domain.

Perform the following postinstallation steps to enable Oracle Identity Manager to work with the Oracle WebLogic Server administrator user provisioned in the central LDAP store. This enables Oracle Identity Manager to connect to SOA without any problem:

1. Log in to Enterprise Manager at the URL listed in Section 16.2, "About Identity Management Console URLs."
2. Select **Farm_IDMDomain** –> **Identity and Access** –> **OIM** –> **oim(11.1.2.0.0).**
3. Select **System MBean Browser** from the menu or right click to select it.
4. Select **Application defined Mbeans** –> **oracle.iam** –> **Server: wls_oim1** –> **Application: oim** –> **XML Config** –> **Config** –> **XMLConfig.SOAConfig** –> **SOAConfig**
5. Change the **username** attribute to the Oracle WebLogic Server administrator username provisioned in Section 10.4, "Preparing the Identity Store" for example: `weblogic_idm`.

   Change **SOA Config RMI URL** to:

   ```
   cluster:t3://soa_cluster
   ```

   Change **SOA Config SOAP URL** to:

   ```
   http://IDMINTERNAL.mycompany.com:7777
   ```

6. Click **Apply**.
7. Select **Weblogic Domain** –> **IDMDomain** from the Navigator.
8. Select **Security** –> **Credentials** from the down menu.
9. Expand the key **oim**.
10. Click **SOAAdminPassword**.
11. Click **Edit**.

12. Change the username to `weblogic_idm` and set the password to the accounts password.
13. Click **OK**.
14. Add the `WLSAdmins` group as a member of SOAAdmin application role using the following WLST command:

    ```
    ORACLE_COMMON_HOME/wlst.sh
    MW_HOME/oracle_common/modules/oracle.jps_
    11.1.1/common/wlstscripts/grantAppRole.py -principalClass
    weblogic.security.principal.WLSGroupImpl -appStripe soa-infra -appRoleName
    SOAAdmin -principalName "WLSAdmins"
    ```

    Where `WLSADMINS` is the group created in Section 10.4, "Preparing the Identity Store" (IDSTORE_WLSADMINGROUP).
15. Run the reconciliation process to enable the Oracle WebLogic Server administrator, `weblogic_idm`, to be visible in the OIM Identity Console. Follow these steps:
    a. Log in to the OIM Administration Console at the URL `http://ADMIN.mycompany.com/sysadmin` as the user `xelsysadm`.
    b. Click **Scheduler** under System Management.
    c. Enter **LDAP*** in the search box.
    d. Click the arrow for the **Search Scheduled Jobs** to list all the schedulers.
    e. Select **LDAP User Create and Update Full Reconciliation**.
    f. Click **Run Now** to run the job.
    g. Repeat for the job **Append and LDAP Role Membership Full Reconciliation**.
    h. Log in to the OIM Identity Console at the URL listed in Section 16.2, "About Identity Management Console URLs." Perform a search to verify that the user `weblogic_idm` is visible.
16. Restart WLS_SOA1 and WLS_SOA2 as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."
17. Log in to the WebLogic Console.
18. Click **Lock & Edit** in the Change Center.
19. Navigate to `IDMDomain -> Services -> Foreign JNDI Providers`
20. Click on `ForeignJNDIProvider-SOA`
21. Under the **Configuration -> General** tab, change the username to `weblogic_idm` and specify the corresponding password.
22. Click **Save** and **Ativate Changes**.

13

Setting Up Node Manager for an Enterprise Deployment

This chapter describes how to configure Node Manager in accordance with Oracle best practice recommendations.

This chapter contains the following sections:

- Section 13.1, "Overview of the Node Manager"
- Section 13.2, "Setting Up Node Manager"
- Section 13.3, "Enabling Host Name Verification Certificates for Node Manager"
- Section 13.4, "Starting Node Manager"

13.1 Overview of the Node Manager

Node Manager enables you to start and stop the Administration Server and the Managed Servers.

Process

The procedures described in this chapter must be performed on IDMHOST1 and IDMHOST2 for various components of the enterprise deployment topologies outlined in Chapter 2.

Note that the procedures in this chapter must be performed multiple times for each VIP-and-IP pair using the information provided in the component-specific chapters.

Recommendations

Oracle provides two main recommendations for Node Manager configuration in enterprise deployment topologies:

1. Oracle recommends placing the Node Manager log file in a location different from the default one (which is inside the Middleware Home where Node Manager resides). See Section 13.2, "Setting Up Node Manager" for further details.
2. Oracle also recommends using host name verification for the communication between Node Manager and the servers in the domain. This requires the use of certificates for the different addresses used in the domain. This chapter explains the steps for configuring certificates in the hosts for host name verification. See Section 13.3, "Enabling Host Name Verification Certificates for Node Manager" for further details.

> **Note:** The passwords used in this guide are used only as examples. Use secure passwords in a production environment. For example, use passwords that consist of random sequences of both uppercase and lowercase characters as well as numbers.

13.2 Setting Up Node Manager

This section describes how to set up Node Manager for an enterprise deployment.

This section contains the following topics:

- Section 13.2.1, "Changing the Location of Node Manager Configuration Files"
- Section 13.2.2, "Editing the Node Manager Property File"
- Section 13.2.3, "Starting Node Manager"

13.2.1 Changing the Location of Node Manager Configuration Files

Create a new directory for Node Manager configuration and log files outside the *MW_HOME* directory, and perform all Node Manager configuration tasks from this directory.

To create the new directory:

1. Stop the NodeManagers running on the IDMHOST1 and IDMHOST2 by running the following command:

   ```
   ps -ef | grep NodeManager
   ```

2. Run the following commands on IDMHOST1 and IDMHOST2:

   ```
   mkdir -p /u02/private/oracle/config/nodemanager
   ```

3. Copy the `nodemanager.properties` file in the following directory:

   ```
   /u01/oracle/products/access/wlserver_10.3/common/nodemanager
   ```

   To the new nodemanager folders you created on IDMHOST1 and IDMHOST2.

4. Copy the `startNodeManager.sh` file in the following directory:

   ```
   /u01/oracle/products/access/wlserver_10.3/server/bin
   ```

   And the `nodemanager.domains` files located in the following folder:

   ```
   /u01/oracle/products/access/wlserver_10.3/common/nodemanager
   ```

   To the new nodemanager folders you created on IDMHOST1 and IDMHOST2.

5. Open `startNodeManager.sh` for IDMHOST1 and IDMHOST2 located in the new `nodemanager` folder in IDMHOST1 and IDMHOST2) using a text editor, and make the following change:

   On IDMHOST1 and IDMHOST2:

   ```
   NODEMGR_HOME="/u02/private/oracle/config/nodemanager"
   ```

13.2.2 Editing the Node Manager Property File

Update the `nodemanager.properties` file located in the following directory on IDMHOST1 and IDMHOST2:

```
/u02/private/oracle/config/nodemanager
```

On IDMHOST1 edit the file as follows:

```
NodeManagerHome=/u02/private/oracle/config/nodemanager
 ListenAddress=192.168.10.200
 LogFile= /u02/private/oracle/config/nodemanager/nodemanager.log
Properties Value
SecureListener=false
StartScriptEnabled=true
StopScriptEnabled=true
StopScriptName=stopWebLogic.sh
Specify a name for the stop script, for example stopWebLogic.sh.
DomainsFile=/u02/private/oracle/config/nodemanager/nodemanager.domains
```

On IDMHOST2:

```
NodeManagerHome=/u02/private/oracle/config/nodemanager
 ListenAddress= 192.168.10.101
 LogFile= /u02/private/oracle/config/nodemanager/nodemanager.log
Properties Value
SecureListener=false
StartScriptEnabled=true
StopScriptEnabled=true
StopScriptName=stopWebLogic.sh
Specify a name for the stop script, for example stopWebLogic.sh.
DomainsFile=/u02/private/oracle/config/nodemanager/nodemanager.domains
```

13.2.3 Starting Node Manager

Start Node Manager on IDMHOST1 and IDMHOST 2 using `startNodeManager.sh` located in the following directory:

```
/u02/private/oracle/config/nodemanager
```

For example run the following command on IDMHOST1 and IDMHOST2:

```
./startNodeManager.sh
```

13.3 Enabling Host Name Verification Certificates for Node Manager

This section describes how to set up host name verification certificates for communication between Node Manager and the Administration Server. It consists of the following steps:

- Section 13.3.1, "Generating Self-Signed Certificates Using the utils.CertGen Utility"
- Section 13.3.2, "Creating an Identity Keystore Using the utils.ImportPrivateKey Utility"
- Section 13.3.3, "Creating a Trust Keystore Using the Keytool Utility"
- Section 13.3.4, "Configuring Node Manager to Use the Custom Keystores"
- Section 13.3.5, "Using a Common or Shared Storage Installation"

- Section 13.3.6, "Configuring Managed WebLogic Servers to Use the Custom Keystores"
- Section 13.3.7, "Changing the Host Name Verification Setting for the Managed Servers"

13.3.1 Generating Self-Signed Certificates Using the utils.CertGen Utility

The certificates added in this chapter (as an example) address a configuration where Node Manager listens on a physical host name (*HOST*.mycompany.com) and a WebLogic Managed Server listens on a virtual host name (*VIP*.mycompany.com). Whenever a server is using a virtual host name, it is implied that the server can be migrated from one node to another. Consequently, the directory where keystores and trust keystores are maintained ideally must reside on a shared storage that is accessible from the failover. If additional host names are used in the same or different nodes, the steps in this example must be extended to:

1. Add the required host names to the certificate stores (if they are different from *HOST*.mycompany.com and *VIP*.mycompany.com).
2. Change the identity and trust store location information for Node Manager (if the additional host names are used by Node Manager) or for the servers (if the additional host names are used by Managed Servers).

Follow these steps to create self-signed certificates on *HOST*. These certificates should be created using the network name or alias. For information on using trust CA certificates instead, see "Configuring Identity and Trust" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*. The following examples configure certificates for *HOST*.mycompany.com and *VIP*.mycompany.com; that is, it is assumed that both a physical host name (*HOST*) and a virtual host name (*VIP*) are used in *HOST*. It is also assumed that *HOST*.mycompany.com is the address used by Node Manager and *VIP*.mycompany.com is the address used by a Managed Server or the Administration Server. This is the common situation for nodes hosting an Administration Server and a Fusion Middleware component, or for nodes where two Managed Servers coexist with one server listening on the physical host name and one server using a virtual host name (which is the case for servers that use migration servers).

1. Set up your environment by running the *WL_HOME*/server/bin/setWLSEnv.sh script. In the Bourne shell, run the following commands:

   ```
   cd WL_HOME/server/bin
   . ./setWLSEnv.sh
   ```

   Verify that the *CLASSPATH* environment variable is set:

   ```
   echo $CLASSPATH
   ```

2. Create a user-defined directory for the certificates. For example, create a directory called 'certs' under the *ASERVER_HOME*/*domain_name* directory. Note that certificates can be shared across WebLogic domains.

   ```
   cd ASERVER_HOME/domain_name
   mkdir certs
   ```

> **Note:** The directory where keystores and trust keystores are maintained must be on shared storage that is accessible from all nodes so that when the servers fail over (manually or with server migration), the appropriate certificates can be accessed from the failover node. Oracle recommends using central or shared stores for the certificates used for different purposes (like SSL set up for HTTP invocations, for example).

3. Change directory to the directory that you just created:

   ```
   cd certs
   ```

4. Run the `utils.CertGen` tool from the user-defined directory to create the certificates for both *HOST*.mycompany.com and *VIP*.mycompany.com.

   Syntax (all on a single line):

   ```
   java utils.CertGen Key_Passphrase Cert_File_Name Key_File_Name
   [export | domestic] [Host_Name]
   ```

   Examples:

   ```
   java utils.CertGen Key_Passphrase IDMHOST1.mycompany.com_cert
   IDMHOST1.mycompany.com_key domestic IDMHOST1.mycompany.com

   java utils.CertGen Key_Passphrase IDMHOST2.mycompany.com_cert
   IDMHOST2.mycompany.com_key domestic IDMHOST2.mycompany.com

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

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

13.3.2 Creating an Identity Keystore Using the utils.ImportPrivateKey Utility

Follow these steps to create an identity keystore on IDMHOST1:

1. Create a new identity keystore called `appIdentityKeyStore` using the `utils.ImportPrivateKey` utility. Create this keystore under the same directory as the certificates, that is, *ASERVER_HOME*`/certs`.

   > **Note:** The Identity Store is created (if none exists) when you import a certificate and the corresponding key into the Identity Store using the `utils.ImportPrivateKey` utility.

2. Import the certificate and private key for `IDMHOST1.mycompany.com`, `IDMHOST2.mycompany.com` and `ADMINVHN.mycompany.com` into the Identity Store. Ensure that you use a different alias for each of the certificate/key pairs imported.

   Syntax (all on a single line):

   ```
   java utils.ImportPrivateKey Keystore_File Keystore_Password
   Certificate_Alias_to_Use Private_Key_Passphrase
   Certificate_File
   Private_Key_File
   ```

```
[Keystore_Type]
```

Examples:

```
java utils.ImportPrivateKey appIdentityKeyStore.jks Key_Passphrase
appIdentityIDMHOST1 Key_Passphrase ASERVER_HOME/certs/IDMHOST1.mycompany.com_
cert.pem ASERVER_HOME/certs/IDMHOST1.mycompany.com_key.pem

java utils.ImportPrivateKey appIdentityKeyStore.jks Key_Passphrase
appIdentityIDMHOST2 Key_Passphrase ASERVER_HOME/certs/IDMHOST2.mycompany.com_
cert.pem ASERVER_HOME/certs/IDMHOST2.mycompany.com_key.pem

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

13.3.3 Creating a Trust Keystore Using the Keytool Utility

Follow these steps to create the trust keystore on each host, IDMHOST1 and IDMHOST2:

1. Copy the standard Java keystore to create the new trust keystore since it already contains most of the root CA certificates needed. Oracle does not recommend modifying the standard Java trust keystore directly. Copy the standard Java keystore CA certificates located under the *WL_HOME*/server/lib directory to the same directory as the certificates. For example:

   ```
   cp WL_HOME/server/lib/cacerts ASERVER_HOME/certs/appTrustKeyStoreIDMHOST1.jks
   ```

2. The default password for the standard Java keystore is `changeit`. Oracle recommends always changing the default password. Use the `keytool` utility to do this. The syntax is:

   ```
   keytool -storepasswd -new New_Password -keystore Trust_Keystore -storepass
   Original_Password
   ```

   For example:

   ```
   keytool -storepasswd -new Key_Passphrase -keystore appTrustKeyStoreIDMHOST1.jks
   -storepass changeit
   ```

3. The CA certificate CertGenCA.der is used to sign all certificates generated by the utils.CertGen tool. It is located in the *WL_HOME*`/server/lib` directory. This CA certificate must be imported into the `appTrustKeyStore` using the `keytool` utility. The syntax is:

   ```
   keytool -import -v -noprompt -trustcacerts -alias Alias_Name
   -file CA_File_Location -keystore Keystore_Location -storepass Keystore_Password
   ```

   For example:

   ```
   keytool -import -v -noprompt -trustcacerts -alias clientCACert -file WL_
   HOME/server/lib/CertGenCA.der -keystore appTrustKeyStoreIDMHOST1.jks -storepass
   Key_Passphrase
   ```

13.3.4 Configuring Node Manager to Use the Custom Keystores

Configure Node Manager to use the custom keystores by editing the `nodemanager.properties` file located in the following directory on IDMHOST1 and IDMHOST2:

```
/u02/private/oracle/config/nodemanager_directory
```

Add the following lines to the `nodemanager.properties` file:

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=Identity_Keystore
CustomIdentityKeyStorePassPhrase=Identity_Keystore_Password
CustomIdentityAlias=Identity_Keystore_Alias
CustomIdentityPrivateKeyPassPhrase=Private_Key_Used_When_Creating_Certificate
```

For example:

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=ASERVER_HOME/certs/appIdentityKeyStore.jks
CustomIdentityKeyStorePassPhrase=Key_Passphrase
CustomIdentityAlias=appIdentityIDMHOST1
CustomIdentityPrivateKeyPassPhrase=Key_Passphrase
```

The passphrase entries in the `nodemanager.properties` file get encrypted when you start Node Manager as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components." For security reasons, minimize the time the entries in the `nodemanager.properties` file are left unencrypted. After you edit the file, start Node Manager as soon as possible so that the entries get encrypted.

13.3.5 Using a Common or Shared Storage Installation

When using a common or shared storage installation for MW_HOME, Node Manager is started from different nodes using the same base configuration (nodemanager.properties). Add the certificate for all the nodes that share the binaries to the appIdentityKeyStore.jks identity store.by creating the certificate for the new node and import it to appIdentityKeyStore.jks, as described in Section 13.3.1, "Generating Self-Signed Certificates Using the utils.CertGen Utility.". Once the certificates are available in the store, each node manager must point to a different identity alias to send the correct certificate to the Administration Server.

To set different environment variables before starting Node Manager in the different nodes:

```
cd WL_HOME/server/bin
export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentityIDMHOST1

cd WL_HOME/server/bin
export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentityIDMHOST2
```

Note: Make sure to specify the custom identity alias specifically assigned to each host, for example `appIdentity1` for ...HOST1 and `appIdentity2` for ...HOST2.

13.3.6 Configuring Managed WebLogic Servers to Use the Custom Keystores

Follow these steps to configure the identity and trust keystores for *WLS_SERVER*:

1. Log in to Oracle WebLogic Server Administration Console at the URL listed in Section 16.2, "About Identity Management Console URLs."
2. Click **Lock and Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Click the name of the server for which you want to configure the identity and trust keystores (*WLS_SERVER*). The settings page for the selected server is displayed.
6. Select **Configuration**, then **Keystores**.
7. In the Keystores field, select the **Custom Identity and Custom Trust** method for storing and managing private keys/digital certificate pairs and trusted CA certificates.
8. In the Identity section, define attributes for the identity keystore:
   - **Custom Identity Keystore:** The fully qualified path to the identity keystore:

     ```
     ASERVER_HOME/certs/appIdentityKeyStore.jks
     ```

   - **Custom Identity Keystore Type:** Leave blank; it defaults to JKS.
   - **Custom Identity Keystore Passphrase:** The password (`Keystore_Password`) you provided in Section 13.3.3, "Creating a Trust Keystore Using the Keytool Utility." This attribute is optional or required depending on the type of keystore. All keystores require the passphrase to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. WebLogic Server only reads from the keystore, so whether you define this property depends on the requirements of the keystore.
9. In the Trust section, define properties for the trust keystore:
   - **Custom Trust Keystore:** The fully qualified path to the trust keystore:

     ```
     ASERVER_HOME/certs/appTrustKeyStoreIDMHOST1.jks
     ```

   - **Custom Trust Keystore Type:** Leave blank; it defaults to JKS.
   - **Custom Trust Keystore Passphrase:** The password you provided as *New_Password* in Section 13.3.3, "Creating a Trust Keystore Using the Keytool Utility." This attribute is optional or required depending on the type of keystore. All keystores require the passphrase to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. WebLogic Server only reads from the keystore, so whether you define this property depends on the requirements of the keystore.
10. Click **Save**.
11. Click **Activate Changes** in the Administration Console's Change Center to make the changes take effect.
12. Select **Configuration**, then **SSL**.
13. Click **Lock and Edit**.
14. In the **Private Key Alias** field, enter the alias you used for the host name the Managed Server listens on, for example:
    - For `WLS_OAM1`, use `appIdentityIDMHOST1`.
    - For `WLS_OAM2` use `appIdentityIDMHOST2`.
    - For `ADMINSERVER` use `appIdentityADMINVHN`.

In the **Private Key Passphrase** and the **Confirm Private Key Passphrase** fields, enter the password for the keystore that you created in Section 13.3.2, "Creating an Identity Keystore Using the utils.ImportPrivateKey Utility."

15. Click **Save**.
16. Click **Activate Changes** in the Administration Console's Change Center to make the changes take effect.
17. Restart the server for which the changes have been applied, as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

13.3.7 Changing the Host Name Verification Setting for the Managed Servers

Once the previous steps have been performed, set host name verification for the affected Managed Servers to `Bea Hostname Verifier`. To do this, perform the following steps:

1. Log in to Oracle WebLogic Server Administration Console.
2. Select **Lock and Edit** from the change center.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Select the Managed Server in the Names column of the table. The settings page for the server is displayed.
6. Open the SSL tab.
7. Expand the **Advanced** section of the page.
8. Set host name verification to `BEA Hostname Verifier`.
9. Click **Save**.
10. Click **Activate Changes**.

13.4 Starting Node Manager

Start Node Manager on IDMHOST1 and IDMHOST2 by running `startNodeManager.sh` located in the following directory:

```
/u02/private/oracle/config/nodemanager
```

To start Node manager, run the following command on IDMHOST1 and IDMHOST2:

```
./startNodeManager.sh
```

Note: If you have not configured and started Node Manager for the first time yet, run the `setNMProps.sh` script as specified in section Section 9.5.3, "Starting Node Manager on IDMHOST1 and IDMHOST2." This enables the use of the start script that is required for Identity Management Components.

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/certs/appIdentityKeyStore.jks, Type=jks,
PassPhraseUsed=true>
```

Host name verification works if you apply a test configuration change to the servers and it succeeds without Node Manager reporting any SSL errors.

14 Configuring Server Migration for an Enterprise Deployment

Configuring server migration allows SOA 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:

- Section 14.1, "Overview of Server Migration for an Enterprise Deployment"
- Section 14.2, "Setting Up a User and Tablespace for the Server Migration Leasing Table"
- Section 14.3, "Creating a GridLink Data Source for Leasing Using the Oracle WebLogic Administration Console"
- Section 14.4, "Editing Node Manager's Properties File"
- Section 14.5, "Setting Environment and Superuser Privileges for the wlsifconfig.sh Script"
- Section 14.6, "Configuring Server Migration Targets"
- Section 14.7, "Testing the Server Migration"
- Section 14.8, "Backing Up the Server Migration Configuration"

14.1 Overview of Server Migration for an Enterprise Deployment

Configure server migration for the WLS_OIM1, WLS_SOA1, WLS_OIM2, and WLS_SOA2 Managed Servers. The WLS_OIM1 and WLS_SOA1 Managed Server are configured to restart on IDMHOST2 should a failure occur. The WLS_OIM2 and WLS_SOA2 Managed Servers are configured to restart on IDMHOST1 should a failure occur. The WLS_OIM1, WLS_SOA1, WLS_OIM2 and WLS_SOA2 servers listen on specific floating IPs that are failed over by WebLogic Server Migration.

Perform the steps in the following sections configure server migration for the WLS_OIM1, WLS_SOA1, WLS_OIM2, and WLS_SOA2 Managed Servers.

14.2 Setting Up a User and Tablespace for the Server Migration Leasing Table

In this section, you set up a user and tablespace for the server migration leasing table:

> **Note:** If other servers in the same domain have already been configured with server migration, the same tablespace and data sources can be used. In that case, the data sources and multi data source for database leasing do not need to be re-created, but they must be retargeted to the clusters being configured with server migration.

1. Create a tablespace called `leasing`. For example, log on to SQL*Plus as the sysdba user and run the following command:

   ```
   create tablespace leasing
   logging datafile 'DB_HOME/oradata/orcl/leasing.dbf' size 32m autoextend on next
   32m maxsize 2048m extent management local;
   ```

2. Create a user named `leasing` and assign to it the `leasing` tablespace:

   ```
   create user leasing identified by password;
   grant create table to leasing;
   grant create session to leasing;
   alter user leasing default tablespace leasing;
   alter user leasing quota unlimited on leasing;
   ```

3. Create the `leasing` table using the `leasing.ddl` script:

   a. Copy the `leasing.ddl` file located in either of the following directories to your database node:

      ```
      WL_HOME/server/db/oracle/817
      WL_HOME/server/db/oracle/920
      ```

   b. Connect to the database as the `leasing` user.

   c. Run the leasing.ddl script in SQL*Plus:

      ```
      @Copy_Location/leasing.ddl;
      ```

   d. After the tool completes, enter the following at the SQL*Plus prompt:

      ```
      commit;
      ```

14.3 Creating a GridLink Data Source for Leasing Using the Oracle WebLogic Administration Console

In this section, you create a GridLink data source for the Leasing table from the Oracle WebLogic Server Administration Console.

To create a GridLink data source:

1. Log in to the Oracle WebLogic Server Administration Console at the URL listed in Section 16.2, "About Identity Management Console URLs."
2. If you have not already done so, in the **Change Center**, click **Lock & Edit**.
3. In the **Domain Structure** tree, expand **Services**, then select **Data Sources**.
4. On the Summary of Data Sources page, click **New** and select **GridLink Data Source**, and enter the following:
   - **Name**: Enter a logical name for the data source. For example, **Leasing**.
   - **JNDI**: Enter a name for JNDI. For example, **jdbc/leasing**.

- **Database Driver**: Select **For the Database Driver, select Oracle's Driver (Thin) for GridLink Connections Versions: 11 and later**.
- Click **Next**.

5. In the Transaction Options page, de-select **Supports Global Transactions**, and click **Next**.
6. In the GridLink Data Source Connection Properties Options screen, select **Enter individual listener information** and click **Next**.
7. Enter the following connection properties:

   - **Service Name**: Enter the service name of the database with lowercase characters. For a GridLink data source, you must enter the Oracle RAC service name. For example:

     ```
     oamedg.mycompany.com
     ```

   - **Host Name and Port**: Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the appropriate parameter in the database using the TCP Protocol:

     ```
     show parameter remote_listener;

     NAME                 TYPE        VALUE

     -------------------------------------------------

     remote_listener     string      DB-SCAN.mycompany.com:1521
     ```

     Note: For Oracle Database 11g Release 1 (11.1), use the virtual IP and port of each database instance listener, for example: `CUSTDBHOST1-VIP.mycompany.com` (port `1521`) and `CUSTDBHOST2-VIP.mycompany.com` (port 1521), where 1521 is *`DB_LSNR_PORT`*

     For Oracle Database 10*g*, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources see Appendix B, "Using Multi Data Sources with Oracle RAC."

   - **Database User Name**: Leasing
   - **Password**: For example: welcome1
   - **Confirm Password**: Enter the password again and click **Next**.

8. On the Test GridLink Database Connection page, review the connection parameters and click **Test All Listeners**. Here is an example of a successful connection notification:

   ```
   Connection test for jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_
   LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=DB-SCAN.mycompany.com)
   (PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=PS5oamedg.mycompany.com))) succeeded.
   ```

   where port 1521 is *DB_LSNR_PORT*.

   Click **Next**.

9. In the ONS Client Configuration page, do the following:

   - Select **FAN Enabled** to subscribe to and process Oracle FAN events.

- Enter here also the SCAN address for the RAC database and the ONS remote port as reported by the database (example below) and click **ADD**:

  ```
  srvctl config nodeapps -s

  ONS exists: Local port 6100, remote port 6200, EM port 2016
  ```

- Click **Next**.

> **Note:** For Oracle Database 11g Release 1 (11.1), use the hostname and port of each database's ONS service, for example:
>
> ```
> CUSTDBHOST1.mycompany.com (port 6200)
> ```
>
> and
>
> ```
> CUSTDBHOST2.mycompany.com (6200)
> ```

10. On the Test ONS Client Configuration page, review the connection parameters and click **Test All ONS Nodes**.

    Here is an example of a successful connection notification:

    ```
    Connection test for DB-SCAN.mycompany.com:6200 succeeded.
    ```

    Click **Next**.

11. In the Select Targets page, select **oim_cluster** and **soa_cluster** as the targets, and **All Servers in the cluster**.
12. Click **Finish**.
13. Click **Activate Changes**.

14.4 Editing Node Manager's Properties File

In this section, you edit Node Manager's properties file. This must be done for the Node Managers on the nodes where the servers are running, IDMHOST1 and IDMHOST2.

The `nodemanager.properties` file is located in the following directory:

```
/u02/private/oracle/config/nodemanager
```

Add the following properties to enable server migration to work properly:

- `Interface:`

  ```
  Interface=bond0
  ```

  This property specifies the interface name for the floating IP (for example, bond0).

> **Note:** Do not specify the sub-interface, such as `bond0:1` or `bond0:2`. 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 `bond0`, `bond1`, `bond2`, `bond3`, `bond`*n*, depending on the number of interfaces configured.

- `NetMask:`

  ```
  NetMask=255.255.248.0
  ```

  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
bond0=*,NetMask=255.255.248.0
UseMACBroadcast=true
```

> **Note:** The following steps are not required if the server properties (start properties) have been properly set and Node Manager can start the servers remotely.

1. If not done already, set the `StartScriptEnabled` property in the `nodemanager.properties` file to true. This is required to enable Node Manager to start the managed servers.
2. Start Node Manager on IDMHOST1 and IDMHOST2 by running the `startNodeManager.sh` script, which is located in the `/u02/private/oracle/config/nodemanager/server/bin` directory.

> **Note:** When running Node Manager from a shared storage installation, multiple nodes are started using the same `nodemanager.properties` file. However, each node may require different NetMask or `Interface` properties. In this case, specify individual parameters on a per-node basis using environment variables. For example, to use a different interface (`bond3`) in HOST*n*, use the `Interface` environment variable as follows:
>
> ```
> export JAVA_OPTIONS=-DInterface=bond3
> ```
>
> and start Node Manager after the variable has been set in the shell.

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

Set environment and superuser privileges for the `wlsifconfig.sh` script:

Ensure that your PATH environment variable includes the files listed inTable 14–1.

Table 14–1 Files Required for the PATH Environment Variable

File	Located in this directory
wlsifconfig.sh	*`MSERVER_HOME`*`/bin/server_migration`

Table 14–1 (Cont.) Files Required for the PATH Environment Variable

File	Located in this directory
wlscontrol.sh	*WL_HOME*/common/bin
nodemanager.domains	*WL_HOME*/common/nodemanager

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

For security reasons, sudo should be restricted to the subset of commands required to run the wlsifconfig.sh script. For example, perform the following steps to set the environment and superuser privileges for the wlsifconfig.sh script.

> **Note:** Ask the system administrator for the appropriate sudo and system rights to perform this step.

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
```

14.6 Configuring Server Migration Targets

In this section, you configure server migration targets. Configuring Cluster Migration sets the DataSourceForAutomaticMigration property to true.

To configure migration in a cluster:

1. Log in to the Oracle WebLogic Server Administration Console at the URL listed in Section 16.2, "About Identity Management Console URLs."
2. In the Domain Structure window, expand **Environment** and select **Clusters**. The Summary of Clusters page is displayed.
3. Click the cluster for which you want to configure migration (**oim_cluster**) in the Name column of the table.
4. Click the **Migration** tab.
5. Click **Lock and Edit**.
6. In the **Available** field, select the machines to which to allow migration, **IDMHOST1** and **IDMHOST2**, and click the right arrow.
7. Select the data source to be used for automatic migration. In this case, select the leasing data source.
8. Click **Save**.
9. Click **Activate Changes**.
10. Repeat steps 2 through 9 for the SOA cluster.

11. Set the candidate machines for server migration. You must perform this task for all of the Managed Servers as follows:

    a. Click **Lock and Edit**.

    b. In the Domain Structure window of the Oracle WebLogic Server Administration Console, expand **Environment** and select **Servers**.

    c. Select the server for which you want to configure migration.

    d. Click the **Migration** tab.

    e. In the **Available** field, located in the **Migration Configuration** section, select the machines to which to allow migration and click the right arrow. For **WLS_OIM1**, select **IDMHOST2**. For **WLS_OIM2**, select **IDMHOST1**.

    f. Select **Automatic Server Migration Enabled** and click **Save**.

       This enables Node Manager to start a failed server on the target node automatically.

    g. Click **Activate Changes**.

    h. Repeat the previous steps for the WLS_SOA1 and WLS_SOA2 Managed Servers.

    i. Restart the managed servers for which server migration has been configured as described in Section 16.1, "Starting and Stopping Oracle Identity Management 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.

14.7 Testing the Server Migration

In this section, you test the server migration. Perform these steps to verify that server migration is working properly:

To test from IDMHOST1:

1. Stop the WLS_OIM1 Managed Server. To do this, run this command:

   ```
   kill -9 pid
   ```

   where *pid* specifies the process ID of the Managed Server. You can identify the pid in the node by running this command:

   ```
   ps -ef | grep WLS_OIM1
   ```

2. Watch the Node Manager console. You should see a message indicating that WLS_OIM1's floating IP has been disabled.

3. Wait for Node Manager to try a second restart of WLS_OIM1. It waits for a fence period of 30 seconds before trying this restart.

4. Once Node Manager restarts the server, stop it again. Node Manager should now log a message indicating that the server will not be restarted again locally.

To test from IDMHOST2:

1. Watch the local Node Manager console. After 30 seconds since the last try to restart WLS_OIM1 on IDMHOST1, Node Manager on IDMHOST2 should prompt

that the floating IP for WLS_OIM1 is being brought up and that the server is being restarted in this node.

2. Access the OIM Console using the Virtual Host Name, for example:

```
http://OIMHOST1VHN.mycompany.com:14000/identity
```

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

Table 14–2 shows the Managed Servers and the hosts they migrate to in case of a failure.

Table 14–2 Managed Server Migration

Managed Server	Migrated From	Migrated To
WLS_OIM1	IDMHOST1	IDMHOST2
WLS_OIM2	IDMHOST2	IDMHOST1
WLS_SOA1	IDMHOST1	IDMHOST2
WLS_SOA2	IDMHOST2	IDMHOST1

Verification From the Administration Console

Migration can also be verified in the Administration Console:

1. Log in to the Administration Console.
2. Click **Domain** 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.

14.8 Backing Up the Server Migration Configuration

Back up the database and the WebLogic domain, as described in Section 16.6, "Backing Up the Oracle IDM Enterprise Deployment."

15

Configuring Single Sign-on for Administration Consoles in an Enterprise Deployment

This chapter describes how to configure single sign-on (SSO) for administration consoles in an Identity Management Enterprise deployment.

This chapter includes the following topics:

15.1 Overview of Configuring Single Sign-on for Administration Consoles in an Enterprise Deployment

If you have not integrated Oracle Access Management Access Manager with Oracle Identity Manager, you must first create WebLogic Security Providers. Then proceed as follows.

You assign WebLogic Administration groups, update boot.properties, and restart the servers. Then you install and configure WebGate and validate the setup. After WebGate is installed and configured, the Oracle Traffic Director intercepts requests for the consoles and forwards them to Access Manager for validation

The administration consoles referred to in the chapter title are:

- Oracle Enterprise Manager Fusion Middleware Control
- Oracle WebLogic Server Administration Console
- Oracle Access Management Console

- Oracle Identity Manager Console

15.2 Prerequisites

Before you attempt to integrate administration consoles with single sign-on, ensure that the following tasks have been performed in the IDMDomain:

1. Configuring Oracle Traffic Director, as described in Chapter 7, "Installing and Configuring Oracle Traffic Director for an Enterprise Deployment."
2. Configuring Access Manager, as described in Chapter 11, "Extending the Domain to Include Oracle Access Management."
3. Provisioning Weblogic Administrators in LDAP as described in Section 10.4, "Preparing the Identity Store."

15.3 Configuring WebLogic Security Providers

When you run `idmConfigTool` with the `configOAM` or `configOIM` option, the tool creates security providers in the domain IDMDomain. These security providers restrict access to the consoles in those domains based on the security policies of Access Manager. If you have other domains, you must create security providers in those domains manually and then update them as described in the following sections.

> **Note:** Once you have enabled single sign-on for the administration consoles, ensure that at least one OAM Server is running to enable console access.
>
> If you have used the Oracle Weblogic console to shut down all of the Access Manager Managed Servers, then restart one of those Managed Servers manually before using the console again.
>
> To start WLS_OAM1 manually, use the command:
>
> ```
> MSERVER_HOME/bin/startManagedWeblogic.sh WLS_OAM1
> t3://ADMINVHN:7001
> ```

This section contains the following topics:

- Section 15.3.1, "Updating Oracle Unified Directory Authenticator"
- Section 15.3.2, "Reordering the Security Providers"

15.3.1 Updating Oracle Unified Directory Authenticator

When the OUD authenticator is created, it is created with some missing information, which must be added. If you are using OUD as your identity store, you must add this information by performing the following steps.

1. Log in to the WebLogic Administration Console.
2. Click **Security Realms** from the Domain structure menu.
3. Click **Lock and Edit** in the Change Center.
4. Click **myrealm**.
5. Click on **Providers**.
6. Click on **OUDAuthenticator**.

7. Click on **Provider Specific** tab.
8. On the Provider Specific screen update the following values:
   - **All Users Filter**: `(&(uid=*)(objectclass=person))`
   - **User From Name Filter**: `(&(uid=%u)(objectclass=person))`
   - **User Name Attribute**: `uid`
   - **Static Group Object Class**: `groupofuniquenames`
   - **Static Member DN Attribute**: `uniquemember`
   - **Static Group DNs from Member DN Filter**: `(&(uniquemember=%M)(objectclass=groupofuniquenames))`
   - **Dynamic Group Name Attribute**: `cn`
   - **Dynamic Group Object Class**: `groupOfURLs`
   - **Dynamic Member URL Attribute**: `memberURL`
9. Click **Save**.
10. Click **Activate Changes**.

15.3.2 Reordering the Security Providers

This section sets up an Access Manager asserter to enable you to delegate responsibility for credential collection to Access Manager.

1. Log in to the WebLogic Administration Console at the URL listed in Section 16.2, "About Identity Management Console URLs."
2. Click **Security Realms** from the Domain structure menu.
3. Click **Lock and Edit** in the **Change Center**.
4. Click **myrealm**.
5. Select the **Providers** tab.
6. Click **Reorder**.
7. Using the arrows on the right hand side order the providers such that the order is:
   - **OAMIDAsserter**
   - **OIM Signature Authenticator**, if present
   - **OIMAuthenticationProvider**, if present
   - **OUD Authenticator**
   - **Default Authenticator**
   - **Default Identity Asserter**

   Note: Oracle Identity Manager providers only exist if Oracle Identity Manager has been configured.

8. Click **OK**.
9. Click **Activate Changes**.

10. Restart WebLogic Administration Server and all the Managed Servers, as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

15.4 Assigning WLSAdmins Group to WebLogic Administration Groups

In an enterprise, it is typical to have a centralized Identity Management domain where all users, groups and roles are provisioned and multiple application domains (such as a SOA domain and WebCenter Portal domain). The application domains are configured to authenticate using the central Identity Management domain.

In Section 10.4, "Preparing the Identity Store" you created a user called `weblogic_idm` and assigned it to the group WLSAdmins. To be able to manage WebLogic using this account you must add the WLSAdmins group to the list of Weblogic Administration groups. This section describes how to add the WLSAdmins Group to the list of WebLogic Administrators.

Perform this step for each domain in the topology.

1. Log in to the WebLogic Administration Server Console at the URL listed in Section 16.2, "About Identity Management Console URLs."
2. In the left pane of the console, click **Security Realms**.
3. On the Summary of Security Realms page, click **myrealm** under the **Realms** table.
4. On the Settings page for `myrealm`, click the **Roles & Policies** tab.
5. On the Realm Roles page, expand the **Global Roles** entry under the **Roles** table. This brings up the entry for Roles. Click the **Roles** link to go to the Global Roles page.
6. On the Global Roles page, click the **Admin** role to go to the Edit Global Role page:
   a. On the Edit Global Roles page, under the **Role Conditions** table, click the **Add Conditions** button.
   b. On the Choose a Predicate page, select **Group** from the list for predicates and click **Next**.
   c. On the Edit Arguments Page, Specify **WLSAdmins** in the **Group Argument** field and click **Add**.
7. Click **Finish** to return to the Edit Global Rule page.
8. The **Role Conditions** table now shows the WLSAdmins Group as an entry.
9. Click **Save** to finish adding the Admin role to the WLSAdmins Group.
10. Validate that the changes were successful by bringing up the WebLogic Administration Server Console using a web browser. Log in using the credentials for the `weblogic_idm` user.

15.5 Authorize Access Manager Administrators to Access APM Console

By default, only users in the WebLogic administrators group can access the APM console. After SSO is enabled, you will login as an Access Manager Administrator.

To enable this functionality perform the following steps:

1. Log in to the APM console at `http://ADMIN.mycompany.com/apm` as WebLogic administrator.
2. Click the **System Configuration** tab.

3. Click **Add** in the External Role Mapping box.
4. Click **Search**.
5. Select **OAMAdministrators** from the returned search results.
6. Click **Add Selected**.
7. Click **Add Principals**.

15.6 Updating the boot.properties File

Update the `boot.properties` file for the Administration Server with the WebLogic `admin` user created in LDAP.

You must update `boot.properties` on each Administration Server node. Follow the steps in the following sections to update the file.

This section contains the following topics:

- Section 15.6.1, "Update the Administration Servers on All Domains"
- Section 15.6.2, "Restarting the Servers"

15.6.1 Update the Administration Servers on All Domains

1. On each of the servers in the topology, go the directory:

   ```
   ASERVER_HOME/servers/serverName/security
   ```

   For example:

   ```
   cd ASERVER_HOME/servers/AdminServer/security
   ```

2. Rename the existing `boot.properties` file.
3. Use a text editor to create a file called boot.properties under the security directory. Enter the following lines in the file:

   ```
   username=adminUser
   password=adminUserPassword
   ```

   For example:

   ```
   username=weblogic_idm
   password=Password for weblogic_idm user
   ```

> **Note:** When you start the Administration Server, the username and password entries in the file get encrypted.
>
> For security reasons, minimize the time the entries in the file are left unencrypted. After you edit the file, you should start the server as soon as possible so that the entries get encrypted.

15.6.2 Restarting the Servers

Restart the WebLogic Administration Server and all managed servers, as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

15.7 Installing and Configuring WebGate 11*g*

This section describes how to install and configure WebGate.

This section contains the following topics:

- Section 15.7.1, "Prerequisites"
- Section 15.7.2, "Installing Oracle WebGate on WEBHOST1 and WEBHOST2"

15.7.1 Prerequisites

Ensure that the following tasks have been performed before installing the Oracle Web Gate:

1. Install and configure the Oracle Traffic Director as described in Chapter 7.
2. Ensure Oracle Access Management Access Manager has been configured as described in Chapter 11.

15.7.2 Installing Oracle WebGate on WEBHOST1 and WEBHOST2

Before starting the installer ensure that Java is installed on your machine. To install Oracle WebGate, run complete the following steps on WEBHOST1 and WEBHOST2.

1. Start the WebGate installer by issuing the command:

   ```
   ./runInstaller
   ```

   You are asked to specify the location of the Java Development Kit for example:

   WEB_MW_HOME/jrockit_*version*

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

   Click **Next**.

4. If the prerequisites fail because of missing 32-bit libraries, you can safely ignore this failure.
5. Click **Next**.
6. On the Installation Location Screen, enter the following information:

   Oracle Home Directory: *WEBGATE_ORACLE_HOME*

   Click **Next**.

7. On the installation summary screen, click **Install**.
8. Click **Next**.
9. Click **Finish**.
10. Execute the `deployWebGateInstance.sh` command from the following directory:

    ```
    WEBGATE_ORACLE_HOME/webgate/iplanet/tools/deployWebGate
    ```

    Make sure this tool has executable permission.

    For example:

    ```
    ./deployWebGateInstance.sh -w WEB_ORACLE_INSTANCE/webgate/ -oh WEBGATE_ORACLE_
    HOME -ws otd
    ```

Expected output:

```
Copying files from WebGate Oracle Home to WebGate Instancedir
```

11. Set the environment variable *LD_LIBRARY_PATH* to:

    ```
    WEBGATE_ORACLE_HOME/lib
    ```

    For example:

    ```
    export LD_LIBRARY_PATH=/u02/private/oracle/config/webgate/lib
    ```

    > **Note:** The deployed location of webgate must be the same on every host.

12. Edit the properties in the `sso.mycompany.com-obj.conf` and `admin.mycompany.com-obj.conf` files using the `EditObjConf` tool located in the following directory:

    ```
    WEBGATE_ORACLE_HOME/webgate/iplanet/tools/setup/InstallTools
    ```

    For example, on WEBHOST1, run the following:

    ```
    ./EditObjConf -f WEB_ORACLE_INSTANCE/net-IDM/config/sso.mycompany.com-obj.conf
    -oh WEBGATE_ORACLE_HOME -w /u02/private/oracle/config/webgate -ws otd

    ./EditObjConf -f WEB_ORACLE_
    INSTANCE/net-IDM/config/admin.mycompany.com-obj.conf -oh WEBGATE_ORACLE_HOME -w
    /u02/private/oracle/config/webgate/webgate -ws otd

    ./EditObjConf -f WEB_ORACLE_
    INSTANCE/net-IDM/config/idminternal.mycompany.com-obj.conf -oh WEBGATE_ORACLE_
    HOME -w /u02/private/oracle/config/webgate/webgate -ws otd
    ```

    Expected output:

    ```
    WEB_ORACLE_INSTANCE/config/magnus.conf has been backed up as WEB_ORACLE_
    INSTANCE/config/magnus.conf.ORIG
    WEB_ORACLE_INSTANCE/config/instance_config_name-obj.conf has been backed up as
    WEB_ORACLE_INSTANCE/instance_config_name-obj.conf.ORIG
    ```

13. Register WebGate to the OAM 11g Server by copying the WebGate artifacts Located in the following directory:

    ```
    ASERVER_HOME/output/Webgate_IDM_11g
    ```

    to the following directories:

    Copy `aaa_cert.pem` and `aaa_key.pem` to:

    ```
    WEB_ORACLE_INSTANCE/webgate/config/simple
    ```

    and

    Copy `cwallet.sso`, `ObAccessClient.xml` and `password.xml` to:

    ```
    WEB_ORACLE_INSTANCE/webgate/config
    ```

    To copy the artifacts run the following commands:

```
cp ASERVER_HOME/output/Webgate_IDM_11g/aaa* to
/u02/private/oracle/config/webgate/webgate/config/simple

cp ASERVER_HOME/output/Webgate_IDM_11g/password.xml to
/u02/private/oracle/config/webgate/webgate/config/

cp ASERVER_HOME/output/Webgate_IDM_11g/ObAccessClient.xml to
/u02/private/oracle/config/webgate/webgate/webgate/config/

cp ASERVER_HOME/output/Webgate_IDM_11g/cwallet.sso to
/u02/private/oracle/config/webgate/webgate/config/
```

14. Add LD_LIBRARY_PATH to Oracle Traffic Director Start Scripts.

    To prevent you having to enter the LD_LIBRARY_PATH each time you start Oracle traffic Director, add it to the OTD start script:

    a. Edit the startserv file located in the following directory

       ```
       WEB_ORACLE_INSTANCE/net-IDM/bin
       ```

    b. Locate the following line:

       ```
       # Set LD_LIBRARY_PATH for Solaris and Linux
       LD_LIBRARY_PATH="${SERVER_LIB_PATH}:${LD_LIBRARY_PATH}"; export LD_LIBRARY_
       PATH
       ```

    c. Add the following line immediately after:

       ```
       LD_LIBRARY_PATH=$LD_LIBRARY_PATH:WEBGATE_ORACLE_HOME/lib; export LD_
       LIBRARY_PATH
       ```

       After editing, the file appears as follows:

       ```
       # Set LD_LIBRARY_PATH for Solaris and Linux

       LD_LIBRARY_PATH="${SERVER_LIB_PATH}:${LD_LIBRARY_PATH}"; export LD_LIBRARY_
       PATH
       LD_LIBRARY_PATH=$LD_LIBRARY_PATH:WEBGATE_ORACLE_HOME/lib; export LD_
       LIBRARY_PATH
       ```

    d. Save this file.

> **Note:** Configuring webgate in this way directly modifies the Oracle Traffic Director (OTD) configuration files. These changes are not reflected in the OTD configuration store. When you go back into and modify the OTD configuration, you are notified that there is a discrepancy between that config store and the values on disk. It will ask you what you want to do. YOU MUST inform OTD that you wish to pull the configuration from the files, and NOT push the configuration back to the files. Selecting the wrong option removes the webgate configuration you just performed.

15.8 Restarting the Oracle Traffic Director Instance

Use the startserv command to start, or the stopserv command to stop your Oracle Traffic Director instance.

To stop the server, run the following command:

```
WEB_ORACLE_INSTANCE/net-IDM/bin/stopserv
```

To start the server, run the following command:

```
export LD_LIBRARY_PATH=/WEBGATE_ORACLE_HOME/lib

WEB_ORACLE_INSTANCE/net-IDM/bin/startserv
```

To restart the Oracle Traffic Director instance, stop all running instances, and then run the start command.

15.9 Validating WebGate and the Access Manager Single Sign-On Setup

To validate that WebGate is functioning correctly, open a web browser and go the OAM console URL listed in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

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

Note: After logging into the Oracle Access Management Console, and before trying to log in to the WebLogic Console, ensure that you log out of the OAM Console, as the user `oamadmin` does not have the access rights to access the WebLogic Console.

To validate the single sign-on setup, open a web browser and go the WebLogic Administration Console and to Oracle Enterprise Manager Fusion Middleware Control at the URLs listed in Section 16.2, "About Identity Management Console URLs."

The Oracle Access Management Single Sign-On page displays. Provide the credentials for the `weblogic_idm` user to log in. Once logged in, you can move back and forth between the WebLogic Console and Fusion Middleware Control without being prompted for a password.

15.10 Backing Up Single Sign-on

Back up the Web Tier and WebLogic domain, as described in Section 16.6, "Backing Up the Oracle IDM Enterprise Deployment."

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:

- Section 16.1, "Starting and Stopping Oracle Identity Management Components"
- Section 16.2, "About Identity Management Console URLs"
- Section 16.3, "Monitoring Enterprise Deployments"
- Section 16.4, "Scaling Enterprise Deployments"
- Section 16.5, "Auditing Identity Management"
- Section 16.6, "Backing Up the Oracle IDM Enterprise Deployment"
- Section 16.7, "Patching Enterprise Deployments"
- Section 16.8, "Preventing Timeouts for SQL"
- Section 16.9, "Manually Failing Over the WebLogic Administration Server"
- Section 16.10, "Troubleshooting"

16.1 Starting and Stopping Oracle Identity Management Components

This section describes how to start, stop and restart the various components of the Oracle Enterprise Deployment for Identity Management.

This section contains the following topics:

- Section 16.1.1, "Startup Order."
- Section 16.1.2, "Starting and Stopping Oracle Unified Directory."
- Section 16.1.3, "Starting, Stopping, and Restarting Access Manager Managed Servers."
- Section 16.1.4, "Starting, Stopping, and Restarting WebLogic Administration Server."
- Section 16.1.5, "Starting and Stopping Node Manager."
- Section 16.1.6, "Starting, Stopping, and Restarting Oracle Traffic Director."
- Section 16.1.7, "Starting, Stopping, and Restarting Oracle Identity Manager."

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 Unified Directory
4. Node Manager
5. Oracle Access Manager Server(s)
6. WebLogic Administration Server
7. Oracle Traffic Director
8. SOA Server(s)
9. Oracle Identity Manager Server(s)

16.1.2 Starting and Stopping Oracle Unified Directory

Start and stop Oracle Unified Directory as follows:

16.1.2.1 Starting Oracle Unified Directory

To start Oracle Unified Directory issue the following command:

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

16.1.2.2 Stopping Oracle Unified Directory

To stop Oracle Unified Directory issue the command:

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

16.1.3 Starting, Stopping, and Restarting Access Manager Managed Servers

Start and stop Oracle Access Manager Managed Servers as follows:

16.1.3.1 Starting an Access Manager Managed Server When None is Running

Normally, you start Access Manager managed servers by using the WebLogic console. After you have enabled Single Sign-On for the administration consoles, however, you must have at least one Access Manager Server running in order to access a console. If no Access Manager server is running, you can start one by using WLST.

To invoke WLST on Linux or UNIX, type:

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

Once you are in the WLST shell, execute the following commands:

```
nmConnect('Admin_User','Admin_Password', 'OAMHOST','Port',  'domain_
name','MSERVER_HOME')
nmStart('OAMServer')
```

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

```
nmConnect('weblogic','password', 'IDMHOST1','5556',  'IDMDomain','MSERVER_HOME')
nmStart('WLS_OAM1')
```

16.1.3.2 Starting an Access Manager Managed Server When Another is Running

To start an Oracle Access Manager managed server when you already have another one running, log in to the WebLogic console using the URL listed in Section 16.2, "About Identity Management Console URLs."

Then proceed as follows:

1. Select **Environment - Servers** from the Domain Structure menu.
2. Click the **Control** tab.
3. Select **OAM Servers (WLS_OAM1 and/or WLS_OAM2)**.
4. Click the **Start** button.
5. Click **Yes** when asked to confirm that you want to start the server(s).

16.1.3.3 Stopping Access Manager Managed Servers

To stop the Oracle Access Manager Managed Server(s), log in to the WebLogic console using the URL listed in Section 16.2, "About Identity Management Console URLs." Then proceed as follows:

1. Select **Environment - Servers** from the Domain Structure menu.
2. Click the **Control** tab.
3. Select **OAM Servers (WLS_OAM1 and/or WLS_OAM2)**.
4. Click the **Shutdown** button and select **Force Shutdown now**.
5. Click **Yes** when asked to confirm that you want to shut down the server(s).

16.1.3.4 Restarting Access Manager Managed Servers

Restart the server by following the `Stop` and `Start` procedures in the previous sections.

16.1.4 Starting, Stopping, and Restarting WebLogic Administration Server

Start and stop the WebLogic Administration Server as described in the following sections.

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

16.1.4.1 Starting WebLogic Administration Server

The recommended way to start the Administration server is to use WLST and connect to Node Manager:

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

```
./wlst.sh
```

Once in WLST shell, execute

```
nmConnect('Admin_User','Admin_Password','ADMINVHN','5556', 'IDMDomain','ASERVER_
HOME')
nmStart('AdminServer')
```

Alternatively, you can start the Administration server by using the command:

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

16.1.4.2 Stopping WebLogic Administration Server

To stop the Administration Server, log in to the WebLogic console using the URL listed in Section 16.2, "About Identity Management Console URLs."

Then proceed as follows:

1. Select **Environment - Servers** from the Domain Structure menu.
2. Click the **Control** tab.
3. Select **AdminServer(admin)**.
4. Click **Shutdown** and select **Force Shutdown now**.
5. Click **Yes** when asked to confirm that you want to shut down the Administration Server.

16.1.4.3 Restarting WebLogic Administration Server

Restart the server by following the `Stop` and `Start` procedures in the previous sections.

16.1.5 Starting and Stopping Node Manager

Start and stop the Node Manager as follows:

16.1.5.1 Starting Node Manager

If the Node Manager being started is the one that controls the Administration Server (IDMHOST1 or IDMHOST2), then prior to starting the Node Manager issue the command:

```
export JAVA_OPTIONS=-DDomainRegistrationEnabled=true
```

If you are using shared storage for Node Manager, set `JAVA_OPTIONS` as described in Section 13.3.5, "Using a Common or Shared Storage Installation."

To start Node Manager, issue the commands:

```
cd /u02/private/oracle/config/nodemanager
./startNodeManager.sh
```

16.1.5.2 Stopping Node Manager

To stop Node Manager, kill the process started in the previous section.

16.1.5.3 Starting Node Manager for an Administration Server

```
cd /u02/private/oracle/config/nodemanager
export JAVA_OPTIONS=-DDomainRegistrationEnabled=true
./startNodeManager.sh
```

Note: It is important to set `-DDomainRegistrationEnabled=true` whenever you start a Node Manager that manages the Administration Server.

16.1.6 Starting, Stopping, and Restarting Oracle Traffic Director

To start and stop Oracle Traffic Director instances see Section 7.6, "Starting the Oracle Traffic Director Instances."

16.1.7 Starting, Stopping, and Restarting Oracle Identity Manager

Start and stop Oracle Identity Manager and Oracle SOA Suite servers as follows:

16.1.7.1 Starting Oracle Identity Manager

To start the Oracle Identity Manager Managed Server(s), log in to the WebLogic console using the URL listed in Section 16.2, "About Identity Management Console URLs."

Then proceed as follows:

1. Select **Environment - Servers** from the Domain Structure menu.
2. Click the **Control** tab.
3. Select **SOA Servers (WLS_SOA1 and/or WLS_SOA2)**.

Note: You can start the Oracle Identity Manager and Oracle SOA Suite servers independently of each other. There is no dependency in their start order. However, the SOA server must be up and running for all of the Oracle Identity Manager functionality to be available.

4. Click the **Start** button.
5. Click **Yes** when asked to confirm that you want to start the server(s).
6. After WLS_SOA1 and/or WLS_SOA2 have started, select WLS_OIM1 and/or WLS_OIM2
7. Click **Start**.
8. Click **Yes** when asked to confirm that you want to start the server(s).

16.1.7.2 Stopping Oracle Identity Manager

To stop the Oracle Identity Manager Managed Server(s), log in to the WebLogic console using the URL listed in Section 16.2, "About Identity Management Console URLs." Then proceed as follows:

1. Select **Environment - Servers** from the Domain Structure menu.
2. Click the **Control** tab.
3. Select **OIM Servers (WLS_OIM1 and/or WLS_OIM2)** and **(WLS_SOA1 and/or WLS_SOA2)**.
4. Click the **Shutdown** button and select **Force Shutdown now**.
5. Click **Yes** when asked to confirm that you want to shutdown the server(s).

16.1.7.3 Restarting Oracle Identity Manager

Restart the server by following the `Stop` and `Start` procedures in the previous sections.

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`
	Enterprise Manager FMW Control	`http://admin.mycompany.com/em`
	OAM Console	`http://admin.mycompany.com/oamconsole`
	OIM Console	`https:sso.mycompany.com/identity`
	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 WebLogic Managed Servers"

16.3.1 Monitoring WebLogic Managed Servers

You can use Oracle Enterprise Manager Fusion Middleware Control to monitor Managed Servers and other Fusion Middleware components, such as Access Manager, Oracle Identity Manager, nd SOA. For more information, see the administrator guides listed in the Preface under "Related Documents" on page -xiii.

16.4 Scaling Enterprise Deployments

The reference enterprise topology discussed in this manual is highly scalable. It can be scaled up and or scaled out. When the topology is scaled up, a new server instance is added to a node already running one or more server instances. When the topology is scaled out, new servers are added to new nodes.

This section contains the following topics:

- Section 16.4.1, "Scaling Up the Topology"
- Section 16.4.2, "Scaling Out the Topology"

16.4.1 Scaling Up the Topology

The Oracle Identity Management topology described in the guide has two tiers: application tier and the Web tier. The components in the application tier can be scaled up by adding a new server instance to a node that already has one or more server instances running.

You cannot scale up the Web tier because you cannot have two Oracle Traffic Director instances on same compute node with the same configuration name.

- Section 16.4.1.1, "Scaling Up Oracle Unified Directory"
- Section 16.4.1.2, "Scaling Up the Application Tier"
- Section 16.4.1.3, "Scaling Up Oracle Traffic Director"

16.4.1.1 Scaling Up Oracle Unified Directory

The directory tier has two Oracle Unified Directory nodes, IDMHOST1 and IDMHOST2, each running an Oracle Unified Directory instance. The Oracle Unified Directory binaries on either node can be used for creating the new Oracle Unified Directory instance.

To add a new Oracle Unified Directory instance to either Oracle Unified Directory host, follow the steps in Section 8.4.3, "Configuring an Additional Oracle Unified Directory Instance on IDMHOST2." with the following variations:

- In Step 2 and Step 4, choose ports other than 1389, 1636, or 4444, as those ports are being used by the existing Oracle Unified Directory instance on the node.
- Use the location for the new Oracle Unified Directory instance as the value for *ORACLE_INSTANCE*.
- Reconfigure the load balancer with the host and port information of the new Oracle Unified Directory instance.

16.4.1.2 Scaling Up the Application Tier

The application tier consists of several nodes in pairs, depending on the products installed. These application servers run WebLogic Managed servers.

If you add a new managed server, after adding the managed server you must update your Oracle Traffic Director configuration.

To update Oracle Traffic Director for a new managed server:

1. Log into the Oracle Traffic Director Administration Console.
2. Click **Server Pools** on the left panel.
3. Click the **oam-pool**.
4. On the left panel, click **New Origin Server**.
5. Add the **IDMHOST3, 14100** of the Origin Server and click **Next**.
6. Click **New Origin Server**, and then **Close**.
7. Click **Deploy Changes** on the top of the panel.

16.4.1.2.1 Scaling Up Oracle Access Manager 11g Scale up Oracle Access Manager as follows:

Log in to the Oracle WebLogic Server Administration Console at the URL listed in Section 16.2, "About Identity Management Console URLs."

1. From the Domain Structure window of the Oracle WebLogic Server Administration Console, expand the **Environment** node and then **Servers**. The Summary of Servers page appears.
2. Click **Lock & Edit** from the Change Center menu.
3. Select an existing server on the host you want to extend, for example: `WLS_OAM1`.

4. Click **Clone**.
5. Enter the following information:
   - **Server Name**: A new name for the server, for example: `WLS_OAM3`.
   - **Server Listen Address**: The name of the host on which the Managed Server runs.
   - **Server Listen Port**: The port the new Managed Server uses. This port must be unique within the host.
6. Click **OK**.
7. Click the newly created server **WLS_OAM3**
8. Click **Save**.
9. Disable host name verification for the new Managed Server. Before starting and verifying the `WLS_OAM3` Managed Server, you must disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Oracle WebLogic Administration Server and the Node Manager in `IDMHOST`*`n`*.

   If the source server from which the new one was cloned had already disabled host name verification, these steps are not required, as the host name verification settings were propagated to the cloned server. To disable host name verification:

   a. 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**.
10. Click **Activate configuration** from the Change Center menu.

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 as the `oamadmin` user. Use the URL listed in Section 16.2, "About Identity Management Console URLs."
2. Click the **System Configuration** tab.
3. Click **Server Instances**.
4. Select **Create** from the Actions menu.
5. Enter the following information:
   - **Server Name**: `WLS_OAM3`
   - **Host**: Host that the server runs on
   - **Port**: Listen port that was assigned when the Managed Server was created

- **OAM Proxy Port**: Port you want the Oracle Access Manager proxy to run on. This is unique for the host
- **Proxy Server ID**: `AccessServerConfigProxy`
- **Mode**: Set to `Open` or `Simple`, depending on the mode your existing Oracle Access Manager servers are operating in.

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 Oracle Identity Management Components."

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

For example, to add the Oracle Access Manager server to `Webgate_IDM`, access the OAM console at the URL listed in Section 16.2, "About Identity Management Console URLs." Then proceed as follows:

1. Log in as the Oracle Access Manager Admin User you created in Section 10.4, "Preparing the Identity Store."
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** drop down list.
9. Set **Max Connections** to `4`.
10. Click **Apply**.

Repeat Steps 5 through 10 for **IAMSuiteAgent** and all other WebGates that might be in use.

The procedures described in this section show you how to create a new managed server or directory instance. Add a third Instance in the Oracle Traffic Director OAM server pool:

To add a third instance to the Oracle Traffic Director OAM server pool:

1. Log into the Oracle Traffic Director Administration Console.
2. Click **Server Pools** on the left panel.
3. Click the **oam-pool**.
4. On the left panel, click **New Origin Server**.
5. Add the **IDMHOST3, 14100** of the Origin Server and click **Next**.
6. Click **New Origin Server**, and then **Close**.
7. Click **Deploy Changes** on the top of the panel.

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

16.4.1.2.2 Scaling Up Oracle Identity Manager (Adding Managed Servers to Existing Nodes) In this case, you already have a node that runs a Managed Server configured with Oracle SOA Suite and Oracle Identity Manager components. The node contains a Middleware home, a SOA Oracle home, an Oracle Identity Manager Oracle home, and a domain directory for existing Managed Servers.

You can use the existing installations (the Middleware home, and domain directories) for creating new WLS_OIM and WLS_SOA servers. There is no need to install the Oracle Identity and Access Management or Oracle SOA Suite binaries in a new location, or to run pack and unpack.

Follow these steps for scaling up the topology:

1. Log in to the Administration Console at the URL listed in Section 16.2, "About Identity Management Console URLs." Clone either the **WLS_OIM1** or the **WLS_SOA1** into a new Managed Server. The source Managed Server to clone should be one that already exists on the node where you want to run the new Managed Server.

   To clone a Managed Server:

   a. Select **Environment** -> **Servers** from the Administration Console.

   b. From the Change Center menu, click **Lock and Edit**.

   c. Select the Managed Server that you want to clone (for example, **WLS_OIM1** or **WLS_SOA1**).

   d. Select **Clone**.

   Name the new Managed Server WLS_OIM*n* or WLS_SOA*n*, where *n* is a number to identify the new Managed Server.

   The rest of the steps assume that you are adding a new server to IDMHOST1, which is already running WLS_SOA1 and WLS_OIM1.

2. For the listen address, assign the host name or IP address to use for this new Managed Server. If you are planning to use server migration as recommended for this server, this should be the VIP (also called a floating IP) to enable it to move to another node. The VIP should be different from the one used by the Managed Server that is already running.

3. Create JMS Servers for SOA, Oracle Identity Manager, UMS, and BPM on the new Managed Server.

   a. Use the Oracle WebLogic Server Administration Console to create a new persistent store for the new SOAJMSServer and name it, for example, `SOAJMSFileStore_`*N*. Specify the path for the store. This should be a directory on shared storage, as recommended in Section 4.3, "Shared Storage Recommendations for Enterprise Deployments."

   Note: This directory must exist before the Managed Server is started or the start operation fails.

   ```
   ASERVER_HOME/jms/SOAJMSFileStore_N
   ```

b. Create a new JMS server for SOA, for example, `SOAJMSServer_auto_`*N*. Use the `SOAJMSFileStore_`*N* for this JMSServer. Target the `SOAJMSServer_auto_`*N* server to the recently created Managed Server (`WLS_SOA`*n*).

c. Create a new JMS server for BPM, for example, `BPMJMSServer_auto_`*N*. Use the `BPMJMSServer_auto_`*N* for this JMSServer. Target the `BPMJMSServer_auto_`*N* server to the recently created Managed Server `WLS_SOA`*n*.

d. Create a new persistence store for the new BPMJMSServer for example, `BPMJMSFileStore_`*N*. Specify the path for the store. This should be a directory on shared storage, as recommended in Section 4.3, "Shared Storage Recommendations for Enterprise Deployments."

e. Create a new persistence store for the new `UMSJMSServer`, for example, `UMSJMSFileStore_`*N* Specify the path for the store. This should be a directory on shared storage as recommended in Section 4.3, "Shared Storage Recommendations for Enterprise Deployments."

   ASERVER_HOME`/jms/UMSJMSFileStore_`*N*.

   Note: This directory must exist before the Managed Server is started or the start operation fails. You can also assign `SOAJMSFileStore_`*N* as store for the new UMS JMS servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

   ---

f. Create a new JMS Server for UMS, for example, `UMSJMSServer_`*N*. Use the `UMSJMSFileStore_`*N* for this JMSServer. Target the `UMSJMSServer_`*N* server to the recently created Managed Server (`WLS_SOA`*n*).

g. Create a new persistence store for the new OIMJMSServer, for example, `OIMJMSFileStore_`*N* Specify the path for the store. This should be a directory on shared storage as recommended in Section 4.3, "Shared Storage Recommendations for Enterprise Deployments."

   ASERVER_HOME`/jms/OIMJMSFileStore_`*N*

   Note: This directory must exist before the Managed Server is started or the start operation fails. You can also assign `SOAJMSFileStore_`*N* as store for the new Oracle Identity Manager JMS Servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

   ---

h. Create a new JMS Server for Oracle Identity Manager, for example, `OIMJMSServer_`*N*. Use the `OIMJMSFileStore_`*N* for this JMSServer. Target the `OIMJMSServer_`*N* server to the recently created Managed Server (`WLS_OIM`*n*).

i. Update the SubDeployment targets for the SOA JMS Module to include the recently created SOA JMS Server. To do this, expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click **SOAJMSModule** (represented as a hyperlink in the **Names** column of the table). The Settings page for SOAJMSModule appears. Click the **SubDeployments** tab. The subdeployment module for **SOAJMS** appears.

Note: This subdeployment module name is a random name in the form of `SOAJMSServer`*`XXXXXX`* resulting from the Configuration Wizard JMS configuration for the first two servers (`WLS_SOA1` and `WLS_SOA2`).

Click the `SOAJMSServer`*`XXXXXX`* subdeployment. Add the new JMS Server for SOA called `SOAJMSServer_`*`N`* to this subdeployment. Click `Save`.

j. Update the SubDeployment targets for the UMSJMSSystemResource to include the recently created UMS JMS Server. To do this, expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click **UMSJMSSystemResource** (represented as a hyperlink in the Names column of the table). The Settings page for UMSJMSSystemResource appears. Click the **SubDeployments** tab. The subdeployment module for UMSJMS appears.

Note: This subdeployment module name is a random name in the form of `UCMJMSServer`*`XXXXXX`* resulting from the Configuration Wizard JMS configuration for the first two servers (`WLS_SOA1` and `WLS_SOA2`).

Click the `UMSJMSServer`*`XXXXXX`* subdeployment. Add the new JMS Server for UMS called `UMSJMSServer_`*`N`* to this subdeployment. Click **Save**.

k. Update the SubDeployment targets for the BPMJMSSystemResource to include the recently created BPM JMS Server. To do this, expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click **BPMJMSSystemResource** (represented as a hyperlink in the Names column of the table). The Settings page for BPMJMSSystemResource appears. Click the **SubDeployments** tab. The subdeployment module for BPMJMS appears.

Note: This subdeployment module name is a random name in the form of `BPMJMSServer`*`XXXXXX`* resulting from the Configuration Wizard JMS configuration for the first two servers (`WLS_SOA1` and `WLS_SOA2`).

Click the **BPMJMSServer*XXXXXX*** subdeployment. Add the new JMS Server for BPM called BPMJMSServer_N to this subdeployment. Click Save.

l. Update the SubDeployment targets for OIMJMSModule to include the recently created Oracle Identity Manager JMS Server. To do this, expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click **OIMJMSModule** (represented as a hyperlink in the Names column of the table). The Settings page for OIMJMSSystemResource appears. Click the **SubDeployments** tab. The subdeployment module for OIMJMS appears.

> **Note:** This subdeployment module name is a random name in the form of `OIMJMSServer`*`XXXXXX`* resulting from the Configuration Wizard JMS configuration for the first two servers (`WLS_OIM1` and `WLS_OIM2`).

Click the `OIMJMSServer`*`XXXXXX`* subdeployment. Add the new JMS Server for Oracle Identity Manager called `OIMJMSServer_`*`N`* to this subdeployment. Click **Save**.

4. Configure Oracle Coherence, as described in Section 12.9, "Configuring Oracle Coherence for Deploying Composites."

5. Configure TX persistent store for the new server. This should be a location visible from other nodes as indicated in the recommendations about shared storage.

   From the Administration Console, select 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.

6. 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**.

7. 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.

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

9. Update the SOA host and port using Oracle Enterprise Manager Fusion Middleware Control. Follow these steps:

   a. Open a browser and go to Oracle Enterprise Manager Fusion Middleware Control at the URL listed in Section 16.2, "About Identity Management Console URLs."

   b. Log in to Oracle Enterprise Manager Fusion Middleware Control using the Admin user credentials.

> **Note:** At least one of the Oracle Identity Manager Managed Servers must be running for when these steps are executed.

c. Navigate to **Identity and Access**, and then **oim**.

d. Right-click **oim** and navigate to **System MBean Browser**.

e. Under **Application Defined MBeans**, navigate to **oracle.iam**, **Application:oim**, **XMLConfig**, **Config**, **XMLConfig.SOAConfig**, and then **SOAConfig**.

f. Update the value for the **Rmiurl** attribute with the host and port of the new SOA server. Click **Apply** to save the changes.

g. The **Rmiurl** attribute is used for accessing SOA EJBs deployed on SOA Managed Servers. This is the application server URL. The following is an example value for this attribute:

```
cluster:t3://soa_cluster
```

10. Restart the WebLogic Administration Server as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

11. 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.

12. Configure the newly created managed server for server migration. Follow the steps in Section 14.6, "Configuring Server Migration Targets" to configure server migration.

13. 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.

16.4.1.3 Scaling Up Oracle Traffic Director

To scale up Oracle traffic director:

1. Install Oracle Traffic Director on the new host as described in Section 7.2, "Installing Oracle Traffic Director on WEBHOST1 and WEBHOST2."
2. Create a new instance of Oracle Traffic Director on the new host as described in Section 7.4, "Register WEBHOST2 with the Administration Node."
3. Deploy the configuration to the new node by following the instructions in Section 7.10, "Deploying the Configuration and Testing the Virtual Server Addresses."
4. Create a new failover group for the new Oracle Traffic Director instance as described in Section 7.11, "Creating a Failover Group for Virtual Hosts."
5. Add the new Oracle Traffic Director failover group to the hardware load balancer pool.

16.4.2 Scaling Out the Topology

In scaling out a topology, new servers are added to new nodes. The components in all three tiers of the Oracle Identity Management topology described in this manual can be scaled out by adding a new server instance to a new node.

This section contains the following topics:

- Section 16.4.2.1, "Scaling Out the Web Tier"
- Section 16.4.2.2, "Scaling Out the Application Tier"

16.4.2.1 Scaling Out the Web Tier

The procedures described in this section show you how to create a new managed server or directory instance. Add a third Instance in the Oracle Traffic Director OAM server pool:

To add a third instance to the Oracle Traffic Director OAM server pool:

1. Log into the Oracle Traffic Director Administration Console.
2. Click **Server Pools** on the left panel.
3. Click the **oam-pool**.
4. On the left panel, click **New Origin Server**.
5. Add the **IDMHOST3, 14100** of the Origin Server and click **Next**.
6. Click **New Orign Server**, and then **Close**.
7. Click **Deploy Changes** on the top of the panel.

16.4.2.2 Scaling Out the Application Tier

The application tier has two nodes (IDMHOST1 and IDMHOST2) running Managed Servers for Oracle Access Manager and Oracle Identity Manager.

Some of the procedures described in this section show you how to create a new WebLogic managed server on a third node. If you add a new managed server to your topology, after adding the managed server you must update your Oracle Traffic Director configuration files (on all nodes) and add the new server to the existing WebLogic cluster directives.

The procedures described in this section show you how to create a new managed server on a third node.

16.4.2.2.1 Scaling Out Oracle Access Manager 11g

Scale out is very similar to scale up but first requires the software to be installed on the new node.

Use the existing installations in shared storage for creating the new Managed Servers. You do not need to install WebLogic Server or Identity Management binaries in a new location but you do need to run pack and unpack to bootstrap the domain configuration in the new node.

> **Note:** If you are using shared storage, allow the new host access to that shared storage area.

1. On the new node, mount the existing Middleware home, which should include the SOA installation and the domain directory, and ensure that the new node has access to this directory, just like the rest of the nodes in the domain.
2. To update the Middleware home list, create (or edit, if another WebLogic installation exists in the node) the *`IAM_MW_HOME`*`/bea/beahomelist` file and add *`IAM_MW_HOME`*`/product/fmw` to it.
3. Log in to the Oracle WebLogic Server Administration Console at the URL listed in Section 16.2, "About Identity Management Console URLs."
4. From the Domain Structure window of the Oracle WebLogic Server Administration Console, expand the **Environment** node and then **Servers**. The Summary of Servers page appears.
5. Click **Lock & Edit** from the Change Center menu.
6. Select an existing server on the host you want to extend, for example: **WLS_OAM1**.
7. Click **Clone**.
8. Enter the following information:
   - **Server Name**: A new name for the server, for example: `WLS_OAM3`.
   - **Server Listen Address**: The name of the host on which the Managed Server runs.
   - **Server Listen Port**: The port the new Managed Server uses. This port must be unique within the host.
9. Click **OK**.
10. Click the newly created server **WLS_OAM3**.
11. Set the SSL listen port. This should be unique on the host that the Managed Server runs on.
12. Click **Save**.
13. Disable host name verification for the new Managed Server. Before starting and verifying the WLS_OAM3 Managed Server, you must disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Oracle WebLogic Administration Server and the Node Manager in IDMHOST*n*.

    If the source server from which the new one was cloned had already disabled host name verification, these steps are not required, as the host name verification settings was propagated to the cloned server. To disable host name verification, proceed as follows:

   a. In Oracle Enterprise Manager Fusion Middleware Control, select Oracle WebLogic Server Administration Console.
   b. Expand the **Environment** node in the Domain Structure pane.
   c. Click **Servers**. The Summary of Servers page appears.
   d. Select **WLS_OAM3** in the Names column of the table. The Settings page for server appears.
   e. Click the **SSL** tab.
   f. Click **Advanced**.
   g. Set **Hostname Verification** to **None**.
   h. Click **Save**.

14. Click **Activate Configuration** from the Change Center menu.
15. Restart the WebLogic Administration Server as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."
16. Pack the domain on IDMHOST1 using the command:

    ```
    pack.sh -domain=ORACLE_BASE/config/domains/IDMDomain -template
    =/tmp/IDMDomain.jar -template_name="OAM Domain" -managed=true
    ```

    The `pack.sh` script is located in *`ORACLE_COMMON_HOME`*`/common/bin`.
17. Unpack the domain on the new host using the command:

    ```
    unpack.sh -domain=MSERVER_HOME/IDMDomain -template=/tmp/IDMDomain.jar -app_
    dir=MSERVER_HOME/applications
    ```

    The `unpack.sh` script is located in *`ORACLE_COMMON_HOME`*`/common/bin`.
18. Start Node Manager and update the property file.
    a. Start and stop Node Manager as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."
    b. Run the script `setNMProps.sh`, which is located in *`ORACLE_COMMON_HOME`*`/common/bin`, to update the node manager properties file, for example:

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

    c. Start Node Manager once again as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

Register the new Managed Server with Oracle Access Manager. The new Managed Server now must be configured as an Oracle Access Manager server. You do this from the Oracle OAM console, as follows:

1. Log in to the OAM console as the `oamadmin` user. Use the URL listed in Section 16.2, "About Identity Management Console URLs."
2. Click the **System Configuration** tab.
3. Click **Server Instances**.
4. Select **Create** from the Actions menu.
5. Enter the following information:
   - **Server Name**: `WLS_OAM3`

- **Host**: Host that the server is running on, IDMHOST3.
- **Port**: Listen port that was assigned when the Managed Server was created.
- **OAM Proxy Port**: Port you want the Oracle Access Manager proxy to run on. This is unique for the host.
- **Proxy Server ID:** AccessServerConfigProxy
- **Mode**: Set to Open or Simple, depending on the mode your existing Oracle Access Manager servers are operating in.

6. Click **Apply**.

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

For example, to add the Oracle Access Manager server to Webgate_IDM, access the OAM console at the URL listed in Section 16.2, "About Identity Management Console URLs." Then proceed as follows:

1. Log in as the Oracle Access Manager admin user you created in Section 10.4.3, "Configuring Oracle Unified Directory for Use with Oracle Access Manager and Oracle Identity Manager."
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** drop down list.
9. Set **Max Connections** to 4.
10. Click **Apply**

Repeat Steps 5 through 10 for IAMSuiteAgent and other WebGates that are in use.

Update the Web Tier. Now that the new Managed Server has been created and started, the web tier starts to direct requests to it. Best practice, however, is to inform the web server that the new Managed Server has been created.

The procedures described in this section show you how to create a new managed server or directory instance. To configure the Web tier, add a third Instance in the Oracle Traffic Director OAM server pool:

To add a third instance to the Oracle Traffic Director OAM server pool:

1. Log into the Oracle Traffic Director Administration Console.
2. Click **Server Pools** on the left panel.
3. Click the **oam-pool**.
4. On the left panel, click **New Origin Server**.
5. Add the **IDMHOST3, 14100** of the Origin Server and click **Next**.
6. Click **New Origin Server**, and then **Close**.

7. Click **Deploy Changes** on the top of the panel.

16.4.2.2.2 Scaling Out Oracle Identity Manager (Adding Managed Servers to New Nodes)

When you scale out the topology, you add new Managed Servers configured with OIM and SOA to new nodes.

Before performing the steps in this section, check that you meet these requirements:

- There must be existing nodes running Managed Servers configured with OIM and SOA within the topology.
- The new node can access the existing home directories for WebLogic Server, OIM, and SOA.

  Use the existing installations in shared storage for creating a new WLS_SOA or WLS_OIM Managed Server. You do not need to install WebLogic Server, OIM, or SOA binaries in a new location but you do need to run pack and unpack to bootstrap the domain configuration in the new node.

Notes:

- If there is no existing installation in shared storage, installing WebLogic Server, IAM, and SOA in the new nodes is required as described in Section 12.9, "Configuring Oracle Coherence for Deploying Composites."
- When an *ORACLE_HOME* or *WL_HOME* is shared by multiple servers in different nodes, Oracle recommends keeping the Oracle Inventory and Middleware home list in those nodes updated for consistency in the installations and application of patches. To update the oraInventory in a node and attach an installation in a shared storage to it, use:

  `OIM_ORACLE_HOME/oui/bin/attachHome.sh`

- To update the Middleware home list to add or remove a *WL_HOME*, edit the `user_home/bea/beahomelist` file. See the following steps.

Follow these steps for scaling out the topology:

1. On the new node, mount the existing Middleware home, which should include the Oracle Fusion Middleware installation and the domain directory, and ensure that the new node has access to this directory, just like the rest of the nodes in the domain.
2. To attach *ORACLE_BASE* in shared storage to the local Oracle Inventory, execute the following command:

   ```
   cd IAM_MW_HOME/product/fmw/iam/oui/bin
   /attachHome.sh -jreLoc JAVA_HOME
   ```

3. To update the Middleware home list, create (or edit, if another WebLogic installation exists in the node) the `IAM_MW_HOME/bea/beahomelist` file and add `IAM_MW_HOME/product/fmw` to it.
4. Log in to the Oracle WebLogic Administration Console at the URL listed in Section 16.2, "About Identity Management Console URLs."
5. Create a new machine for the new node to be used, and add the machine to the domain.

6. Update the machine's Node Manager's address to map the IP address of the node that is being used for scale out.

7. Use the Oracle WebLogic Server Administration Console to clone the managed servers `WLS_OIM` and `WLS_SOA1` into new Managed Servers. Name them `WLS_SOA`*n* and `WLS_OIM`*n*, respectively, where *n* is a number.

> **Note:** These steps assume that you are adding a new server to node *n*, where no Managed Server was running previously.

8. Assign the host names or IP addresses to the listen addresses of the new Managed Servers.

9. If you are planning to use server migration for this server (which Oracle recommends) this should be the VIP address (also called a floating IP address) for the server. This VIP address should be different from the one used for the existing Managed Server.

10. Create JMS servers for SOA, Oracle Identity Manager (if applicable), and UMS on the new Managed Server.

    a. Use the Oracle WebLogic Server Administration Console to create a new persistent store for the new SOAJMSServer and name it, for example, `SOAJMSFileStore_`*N*. Specify the path for the store. This should be a directory on shared storage as recommended in Section 4.3, "Shared Storage Recommendations for Enterprise Deployments." For example:

    ```
    ASERVER_HOME/jms/SOAJMSFileStore_N
    ```

    > **Note:** This directory must exist before the Managed Server is started or the start operation fails.

    b. Create a new JMS Server for SOA, for example, `SOAJMSServer_auto_`*N*. Use the `SOAJMSFileStore_`*N* for this JMSServer. Target the `SOAJMSServer_auto_`*N* Server to the recently created Managed Server (`WLS_SOA`*n*).

    c. Create a new persistence store for the new UMSJMSServer, and name it, for example, `UMSJMSFileStore_`*N*. Specify the path for the store. This should be a directory on shared storage as recommended in Section 4.3, "Shared Storage Recommendations for Enterprise Deployments."

    ```
    ASERVER_HOME/jms/UMSJMSFileStore_N
    ```

    > **Notes:**
    >
    > - This directory must exist before the Managed Server is started or the start operation fails.
    > - It is also possible to assign `SOAJMSFileStore_`*N* as the store for the new UMS JMS Servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

    d. Create a new JMS server for UMS: for example, `UMSJMSServer_`*N*. Use the `UMSJMSFileStore_`*N* for this JMS server. Target the `UMSJMSServer_`*N* server to the recently created Managed Server (`WLS_SOA`*n*).

e. Create a new persistence store for the new BPMJMSServer, and name it, for example, `BPMJMSFileStore_N`. Specify the path for the store. This should be a directory on shared storage as recommended in Section 4.3, "Shared Storage Recommendations for Enterprise Deployments."

```
ASERVER_HOME/jms/BPMJMSFileStore_N
```

Notes:

- This directory must exist before the Managed Server is started. Otherwise, the start operation fails.
- It is also possible to assign `SOAJMSFileStore_N` as the store for the new BPM JMS Servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

f. Create a new JMS server for BPM, for example, `BPMJMSServer_N`. Use the `BPMJMSFileStore_N` for this JMS server. Target the `BPMJMSServer_N` server to the recently created Managed Server (`WLS_SOAn`).

g. Create a new persistence store for the new `OIMJMSServer`, and name it, for example, `OIMJMSFileStore_N`. Specify the path for the store. This should be a directory on shared storage as recommended in Section 4.3, "Shared Storage Recommendations for Enterprise Deployments."

```
ASERVER_HOME/jms/OIMJMSFileStore_N
```

Notes:

- This directory must exist before the Managed Server is started or the start operation fails.
- It is also possible to assign `SOAJMSFileStore_N` as the store for the new Oracle Identity Manager JMS Servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

h. Create a new JMS Server for Oracle Identity Manager: for example, `OIMJMSServer_N`. Use the `OIMJMSFileStore_N` for this JMS Server. Target the `OIMJMSServer_N` Server to the recently created Managed Server (`WLS_OIMn`).

i. Update the SubDeployment targets for the BPMJMSSystemResource to include the recently created BPM JMS Server. To do this, expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click **BPMJMSSystemResource** (represented as a hyperlink in the Names column of the table). The Settings page for BPMJMSSystemResource appears. Click the **SubDeployments** tab. The subdeployment module for BPMJMS appears.

Note: This subdeployment module name is a random name in the form of `BPMJMSServer`*`XXXXXX`* resulting from the Configuration Wizard JMS configuration for the first two servers (WLS_SOA1 and WLS_SOA2).

Click the `BPMJMSServer`*`XXXXXX`* subdeployment. Add the new JMS Server for BPM called `BPMJMSServer_`*`N`* to this subdeployment. Click **Save**.

j. Update the SubDeployment targets for the SOA JMS Module to include the recently created SOA JMS Server. To do this, expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click **SOAJMSModule** (represented as a hyperlink in the **Names** column of the table). The Settings page for **SOAJMSModule** appears. Open the SubDeployments tab. The subdeployment module for **SOAJMS** appears.

Note: This subdeployment module name is a random name in the form of `SOAJMSServer` resulting from the Configuration Wizard JMS configuration for the first two servers (WLS_SOA1 and WLS_SOA2).

Click the `SOAJMSServer`*`XXXXXX`* subdeployment. Add the new JMS Server for SOA called `SOAJMSServer_`*`N`* to this subdeployment. Click **Save**.

k. Update the SubDeployment targets for UMSJMSSystemResource to include the recently created UMS JMS Server. To do this, expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click **UMSJMSSystemResource** (represented as a hyperlink in the **Names** column of the table). The Settings page for `UMSJMSSystemResource` appears. Open the **SubDeployments** tab. The subdeployment module for `UMSJMS` appears

---

Note: This subdeployment module is a random name in the form of `UMSJMSServerXXXXXX` resulting from the Config Wizard JMS configuration for the first two servers (WLS_SOA1 and WLS_SOA2).

Click the `UMSJMSServer`*`XXXXXX`* subdeployment. Add the new JMS Server for UMS called `UMSJMSServer_`*`N`* to this subdeployment. Click **Save**.

l. Update the SubDeployment Targets for `OIMJMSModule` to include the recently created Oracle Identity Manager JMS Server. To do this, expand the **Services** node and then expand the **Messaging** node. Choose JMS Modules from the Domain Structure window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click **OIMJMSModule** (represented as a hyperlink in the Names column of the table). The Settings page for `OIMJMSModule` appears. Click the **SubDeployments** tab. The subdeployment module for `OIMJMS` appears.

Note: This subdeployment module is a random name in the form of `OIMJMSServer`*`XXXXXX`* resulting from the Config Wizard JMS configuration for the first two servers (WLS_SOA1 and WLS_SOA2).

Click the `OIMJMSXXXXXX` subdeployment. Add the new JMS Server for Oracle Identity Manager called `OIMJMSServer_N` to this subdeployment. Click **Save**.

11. Click **Activate Configuration** from the Change Center menu.
12. Run the `pack` command to create a template pack. Run it on IDMHOST1 if you are using a single domain or on IDMHOST1. Proceed as follows:

    ```
    cd ORACLE_COMMON_HOME/common/bin
    ./pack.sh -managed=true -domain=ASERVER_HOME -template=/templates/oim_
    domain.jar -template_name="OIM Domain"
    ```

    Run the `scp` command on IDMHOST1 to copy the template file created to IDMHOST*n*. For example:

    ```
    scp /templates/oim_domain.jar IDMHOSTN:/templates/oim_domain.jar
    ```

    Run the `unpack` command on IDMHOST*n* to unpack the template in the Managed Server domain directory as follows:

    ```
    cd ORACLE_COMMON_HOME/oracle_common/bin
    ./unpack.sh -domain=MSERVER_HOME -template=/templates/oim_domain.jar -app_
    dir=MSERVER_HOME/applications
    ```

13. Configure Oracle Coherence, as described in Section 12.9, "Configuring Oracle Coherence for Deploying Composites."
14. Update the SOA host and port using Oracle Enterprise Manager Fusion Middleware Control. Follow these steps:
    a. Open a browser and go to Oracle Enterprise Manager Fusion Middleware Control at the URL listed in Section 16.2, "About Identity Management Console URLs."
    b. Log in to Oracle Enterprise Manager Fusion Middleware Control using the `admin` user credentials.

    > **Note:** At least one of the Oracle Identity Manager Managed Servers must be running for when these steps are executed.

    c. Navigate to **Identity and Access**, and then **oim**.
    d. Right-click **oim** and navigate to **System MBean Browser**.
    e. Under **Application Defined MBeans**, navigate to **oracle.iam**, **Application:oim**, **XMLConfig**, **Config**, **XMLConfig.SOAConfig**, and then **SOAConfig**.
    f. Update the value for the **Rmiurl** attribute with the host and port of the new SOA server. Click **Apply** to save the changes.
    g. The **Rmiurl** attribute is used for accessing SOA EJBs deployed on SOA Managed Servers. This is the application server URL. The following is an example value for this attribute:

    ```
    cluster:t3://soa_cluster
    ```

15. Configure TX persistent store for the new server. This should be a location visible from other nodes as indicated in the recommendations about shared storage.

From the Administration Console, select **Server_name** > **Services** tab. Under **Default Store**, in **Directory**, enter the path to the folder where you want the default persistent store to store its data files.

16. Disable host name verification for the new Managed Server. Before starting and verifying the WLS_SOA*n* and WLS_OIM*n* Managed Server, you must disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Oracle WebLogic Administration Server and the Node Manager in IDMHOST*n*. If the source server from which the new one has been cloned had already disabled host name verification, these steps are not required (the host name verification setting is propagated to the cloned server).

    To disable host name verification for WLS_SOA*n*:

    a. Expand the **Environment** node in the **Domain Structure** window.
    b. In the Oracle Enterprise Manager Console, select **Oracle WebLogic Server Administration Console**.
    c. Click **Servers**. The Summary of Servers page appears.
    d. Select **WLS_SOA*n*** in the **Names** column of the table.

       The Settings page for server appears.
    e. Click the **SSL** tab.
    f. Click **Advanced**.
    g. Set **Hostname Verification** to `None`.
    h. Click **Save**.

    To disable host name verification for WLS_OIM*n*, repeat the same steps, but select **WLS_OIM*n*** in the **Names** column in Step d.

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

18. Start the Node Manager on the new node. To start the Node Manager, use the installation in shared storage from the existing nodes, and start Node Manager by passing the host name of the new node as a parameter as follows:

    ```
    WL_HOME/server/bin/startNodeManager new_node_ip
    ```

19. Start and test the new Managed Server from the Oracle WebLogic Server Administration Console:

    a. Shut down all the existing Managed Servers in the cluster.
    b. Ensure that the newly created Managed Servers, WLS_SOA*n* and WLS_SOA*n*, are running.
    c. Access the applications on the newly created Managed Servers (`http://vip:port/soa-infra` and `http://vip:port/oim`). The applications should be functional.

20. Configure server migration for the new Managed Server.

    > **Note:** Since this new node is using an existing shared storage installation, the node is already using a Node Manager and an environment configured for server migration that includes netmask, interface, `wlsifconfig` script superuser privileges. The floating IP addresses for the new Managed Servers are already present in the new node.

Configure server migration following these steps:

a. Log in to the Administration Console.

b. In the left pane, expand **Environment** and select **Servers**.

c. Select the server (represented as hyperlink) for which you want to configure migration from the **Names** column of the table. The Setting page for that server appears.

d. Click the **Migration** tab.

e. In the **Available** field, in the **Migration Configuration** section, select the machines to which to enable migration and click the right arrow.

> **Note:** Specify the least-loaded machine as the migration target for the new server. The required capacity planning must be completed so that this node has enough available resources to sustain an additional Managed Server.

f. Select the **Automatic Server Migration Enabled** option. This enables the Node Manager to start a failed server on the target node automatically.

g. Click **Save**.

h. Restart the Administration Server, Managed Servers, and Node Manager.

i. Test server migration for the new servers WLS_SOA*n* and WLS_OIM*n*, as follows.

1. Determine the PID of the WLS_SOA*n* Managed Server by typing

```
ps -ef | grep WLS_SOAn
```

2. From the node where you added the new server, abruptly stop the WLS_SOA*n* Managed Server by typing:

```
kill -9 pid
```

3. Watch the Node Manager Console. You should see a message indicating that floating IP address for WLS_SOA1 has been disabled.

4. Wait for the Node Manager to try a second restart of WLS_SOA*n*. Node Manager waits for a fence period of 30 seconds before trying this restart.

5. Once Node Manager restarts the server, stop it again. Now Node Manager should log a message indicating that the server will not be restarted again locally.

6. Repeat Steps 1-5 for WLS_OIM*n*.

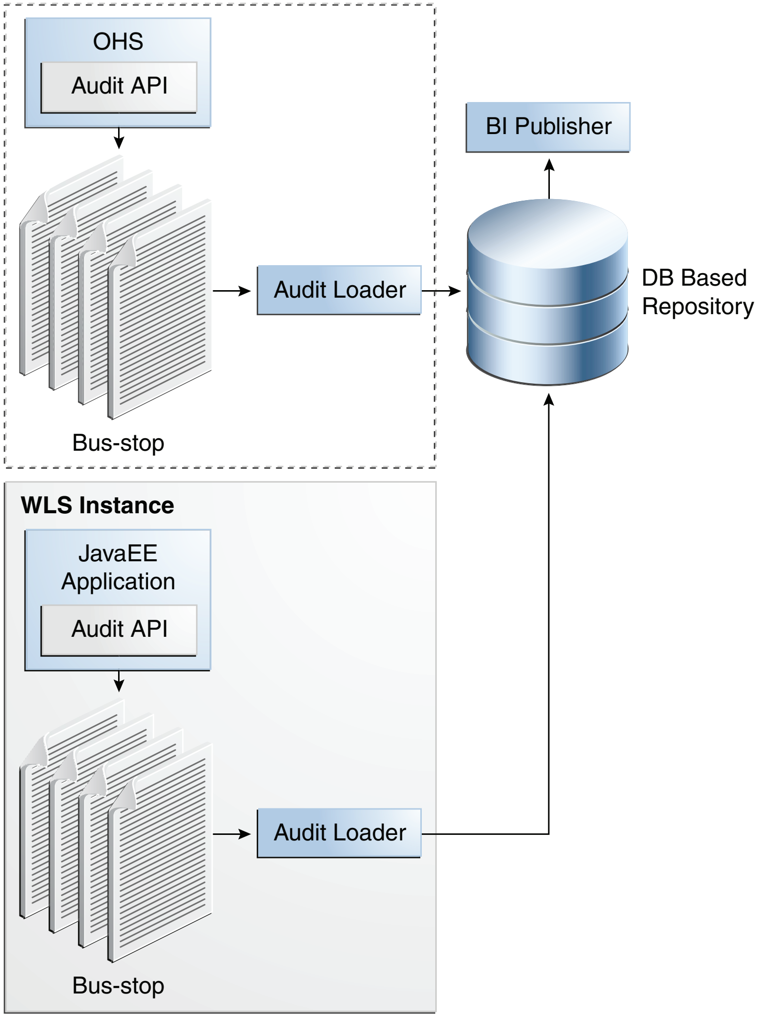
16.5 Auditing Identity Management

Oracle Fusion Middleware Audit Framework is a new service in Oracle Fusion Middleware 11*g*, designed to provide a centralized audit framework for the middleware family of products. The framework provides audit service for platform components such as Oracle Platform Security Services (OPSS) and Oracle Web Services. It also provides a framework for JavaEE applications, starting with Oracle's own JavaEE components. JavaEE applications are able to create application-specific audit events. For non-JavaEE Oracle components in the middleware such as C or

JavaSE components, the audit framework also provides an end-to-end structure similar to that for JavaEE applications.

Figure 16–1 is a high-level architectural diagram of the Oracle Fusion Middleware Audit Framework.

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 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 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.

16.6 Backing Up the Oracle IDM Enterprise Deployment

Back up the topology before and after any configuration changes.

16.6.1 Backing Up the Database

Back up the database. This is a full database backup (either hot or cold) using Oracle Recovery Manager (recommended) or OS tools, such as tar for cold backups if possible.

16.6.2 Backing Up the Administration Server Domain Directory

Back up the Administration Server domain directory to save your domain configuration. The configuration files are located in the following directory:

```
ASERVER_HOME
```

To back up the Administration Server run the following command on IDMHOST1:

```
tar -cvpf edgdomainback.tar ASERVER_HOME
```

16.6.3 Backing Up the Web Tier

Backup the Web tier. The configuration files are located in the following directories:

```
WEB_ORACLE_ADMININSTANCE
```

To back up the Oracle Traffic Director Administration Server, run the following command on WEBHOST1:

```
tar -cvpf webasback.tar WEB_ORACLE_ADMININSTANCE
```

16.6.4 Backing up the Middleware Home

If a new install has modified the *MW_HOME*, back it up using the following command:

```
tar -cvpf mw_home.tar MW_HOME
```

16.7 Patching Enterprise Deployments

This section describes how to apply an Oracle Fusion Middleware patch file and how to patch Oracle Identity Management components with minimal down time.

This section contains the following topics:

- Section 16.7.1, "Patching an Oracle Fusion Middleware Source File"
- Section 16.7.2, "Patching Identity and Access Management"
- Section 16.7.3, "Patching Identity Management Components"

16.7.1 Patching an Oracle Fusion Middleware Source File

For information on patching an Oracle Fusion Middleware source file, see the *Oracle Fusion Middleware Administrator's Guide*.

16.7.2 Patching Identity and Access Management

In a single domain topology, apply patches as follows:

IDMDomain MW_HOME

- Common patches
- Oracle Access Manager Patches
- Oracle Identity Manager Patches
- IDM Tool Patches

16.7.3 Patching Identity Management Components

To patch Oracle Identity Management components with minimal down time, it is recommended that you follow these guidelines:

1. Route the LDAP traffic from IDMHOST1 to IDMHOST2.
2. Bring down the Oracle Unified Directory server on the host on which you are applying the patch (IDMHOST1).
3. Apply the Oracle Unified Directory patch on the host.
4. Start the Oracle Unified Directory server on the host.
5. Test the patch.
6. Route the traffic to IDMHOST1 again.
7. Verify the applications are working properly.
8. Route the LDAP traffic on IDMHOST2 to IDMHOST1.
9. Bring down the Oracle Unified Directory server on the host on which you are applying the patch (IDMHOST2).
10. Apply the patch or Oracle Unified Directory patch on the host.
11. Start the Oracle Unified Directory server on the host.
12. Test the patch.
13. Route the traffic to both hosts on which the patch has been applied (IDMHOST1 and IDMHOST2).

16.8 Preventing Timeouts for SQL

Most of the production deployment involves firewalls. Because database connections are made across firewalls, Oracle recommends that the firewall be configured so that the database connection is not timed out. For Oracle Real Application Clusters (Oracle RAC), the database connections are made on Oracle RAC VIPs and the database listener port. You must configure the firewall so it does not time out these connections. If such a configuration is not possible, set the SQLNET.EXPIRE_TIME=n parameter in the `*ORACLE_HOME*/network/admin/sqlnet.ora` file on the database server, where `n` is the time in minutes. Set this value to less than the known value of the timeout for the network device (that is, a firewall). For Oracle RAC, set this parameter in all of the Oracle home directories.

16.9 Manually Failing Over the WebLogic Administration Server

This section discusses how to fail over the Administration Server to IDMHOST2 and how to fail it back to IDMHOST1.

The same procedure can be applied to each domain you have created.

This section contains the following topics:

- Section 16.9.1, "Failing over the Administration Server to IDMHOST2"
- Section 16.9.2, "Starting the Administration Server on IDMHOST2"
- Section 16.9.3, "Validating Access to IDMHOST2"
- Section 16.9.4, "Failing the Administration Server Back to IDMHOST1"

16.9.1 Failing over the Administration Server to IDMHOST2

If a node fails, you can fail over the Administration Server to another node. This section describes how to fail over the Administration Server from IDMHOST1 to IDMHOST2.

Assumptions:

- The Administration Server is configured to listen on `ADMINVHN.mycompany.com`, and not on `ANY` address. See Section 9.4, "Running the Configuration Wizard to Create a Domain."
- The Administration Server is failed over from IDMHOST1 to IDMHOST2, and the two nodes have these IP addresses:
  - IDMHOST1: `192.168.20.3`
  - IDMHOST2: `192.168.20.4`
  - ADMINVHN: `10.10.30.1`

    This is the Virtual IP address where the Administration Server is running, assigned to *interface*:*index* (for example, bond1:1), available in IDMHOST1 or IDMHOST2.
- The domain directory where the Administration Server is running in IDMHOST1 is on a shared storage and is mounted also from IDMHOST2.

  > **Note:** NM in IDMHOST2 does not control the domain at this point, since `unpack/nmEnroll` has not been run yet on IDMHOST2. But for the purpose of AdminServer failover and control of the AdminServer itself, Node Manager is fully functional

- Oracle WebLogic Server and Oracle Fusion Middleware Components have been installed in IDMHOST2 as described in previous chapters. That is, the same path for *`IDM_ORACLE_HOME`* and *`MW_HOME`* that exists in IDMHOST1 is available in IDMHOST2.

The following procedure shows how to fail over the Administration Server to a different node, IDMHOST2.

1. Stop the Administration Server as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."
2. Migrate the IP address to the second node.
   a. Run the following command as root on IDMHOST1 (where *x*:*y* is the current interface used by `ADMINVHN.mycompany.com`):

      ```
      /sbin/ifconfig x:y down
      ```

      For example:

      ```
      /sbin/ifconfig bond1:1 down
      ```

   b. Run the following command on IDMHOST2:

      ```
      /sbin/ifconfig interface:index ADMINVHN netmask netmask
      ```

      For example:

      ```
      /sbin/ifconfig bond1:1 10.10.30.1 netmask 255.255.255.0
      ```

      Note: Ensure that the netmask and interface to be used match the available network configuration in IDMHOST2.

3. Update routing tables by using `arping`, for example:

   ```
   /sbin/arping -b -A -c 3 -I bond0 10.10.30.1
   ```

16.9.2 Starting the Administration Server on IDMHOST2

Perform the following steps to start Node Manager on IDMHOST2.

1. On IDMHOST1, un-mount the Administration Server domain directory. For example:

   ```
   umount /u01/oracle/config
   ```

2. On IDMHOST2, mount the Administration Server domain directory. For example:

   ```
   mount /u01/oracle/config
   ```

3. Start Node Manager by using the following commands:

   ```
   cd /u02/private/oracle/config/nodemanager
   ./startNodeManager.sh
   ```

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

   Note: Starting and stopping Node Manager at this point is only necessary the first time you run Node Manager. Starting and stopping it creates a property file from a predefined template. The next step adds properties to that property file.

5. Run the setNMProps.sh script to set the `StartScriptEnabled` property to `true` before starting Node Manager:

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

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

6. Start the Node Manager as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."
7. Start the Administration Server on IDMHOST2.

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

   Once in the WLST shell, execute the following commands:

   ```
   nmConnect('Admin_User','Admin_Password', 'IDMHOST2','5556',
   'IDMDomain','ASERVER_HOME')
   nmStart('AdminServer')
   ```

8. 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`.

16.9.3 Validating Access to IDMHOST2

1. Test that you can access the Oracle WebLogic Server Administration Console at:

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

2. Check that you can access and verify the status of components in the Oracle Enterprise Manager at:

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

16.9.4 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, un-mount the Administration Server domain directory. For example:

   ```
   umount /u01/oracle/config
   ```

3. On IDMHOST1, mount the Administration Server domain directory. For example:

   ```
   mount /u01/oracle/config/IDMDomain/aserver/
   ```

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 10.10.30.1 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 -b -A -c 3 -I interface 10.10.30.1
   ```

7. If Node Manager is not already started on IDMHOST1, start it, as described in Section 16.1, "Starting and Stopping Oracle Identity Management Components."

8. Start the Administration Server again on IDMHOST1.

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

   Once in the WLST shell, execute

   ```
   nmConnect(Admin_User,'Admin_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`

10. Check that you can access and verify the status of components in the Oracle Enterprise Manager at:

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

16.10 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 Access Manager 11g"
- Section 16.10.2, "Troubleshooting Oracle Identity Manager"
- Section 16.10.3, "Troubleshooting Oracle SOA Suite"
- Section 16.10.4, "Using My Oracle Support for Additional Troubleshooting Information"
- Section 16.10.5, "OIM Reconciliation Jobs Fail"

16.10.1 Troubleshooting Access Manager 11g

This section describes some common problems that can arise with Access Manager and the actions you can take to resolve the problem. It contains the following topics:

16.10.1.1 User Reaches the Maximum Allowed Number of Sessions

Problem

The Access Manager server displays an error message similar to this:

```
The user has already reached the maximum allowed number of sessions. Please close
one of the existing sessions before trying to login again.
```

Solution

If users log in multiple times without logging out, they might overshoot the maximum number of configured sessions. You can modify the maximum number of configured sessions by using the OAM Administration Console.

To modify the configuration by using the OAM Administration Console, proceed as follows:

1. Go to **System Configuration** -> **Common Settings** -> **Session**
2. Increase the value in the **Maximum Number of Sessions per User** field to cover all concurrent login sessions expected for any user. The range of values for this field is from 1 to any number.

16.10.1.2 Policies Do Not Get Created When Oracle Access Manager is First Installed

Problem

The Administration Server takes a long time to start after configuring Oracle Access Manager.

Solution

Tune the OAM database. When the Administration Server first starts after configuring Oracle Access Manager, it creates a number of default policies in the database. If the database is distant or in need of tuning, this can take a significant amount of time.

```
Resources
Authentication Policies
   Protected Higher Level Policy
   Protected Lower Level Policy
   Publicl Policy
Authorization Policies
   Authorization Policies
```

If you do not see these items, the initial population has failed. Check the Administration Server log file for details.

16.10.1.3 You Are Not Prompted for Credentials After Accessing a Protected Resource

Problem

When you access a protected resource, Oracle Access Manager should prompt you for your user name and password. For example, after creating a simple HTML page and adding it as a resource, you should see credential entry screen.

Solution

If you do not see the credential entry screen, perform the following steps:

1. Verify that Host Aliases for IDMDomain have been set. You should have aliases for IDMDomain:80, IDMDomain:Null:, admin.mycompany.com:80, and sso.mycompany.com:443.
2. Verify that WebGate is installed.
3. Verify that `OBAccessClient.xml` was copied from *`ASERVER_HOME`*`/output` to the WebGate Lib directory and that Oracle Traffic Director was restarted.
4. When OBAccessClient.xml was first created, the file was not formatted. When the Oracle Traffic Director is restarted, reexamine the file to ensure that it is now formatted. Oracle Traffic Director 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.

16.10.1.4 Cannot Log In to OAM Console

Problem

You cannot log in to the OAM Console. The Administration Server diagnostic log might contain an error message similar to this:

```
Caused by: oracle.security.idm.OperationFailureException:
oracle.security.am.common.jndi.ldap.PoolingException [Root exception is
oracle.ucp.UniversalConnectionPoolException:
Invalid life cycle state.
 Check the status of the Universal Connection Pool]
        at
oracle.security.idm.providers.stdldap.UCPool.acquireConnection(UCPool.java:112)
```

Solution

Remove the /tmp/UCP* files and restart the Administration Server.

16.10.2 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 16.10.2.1, "java.io.FileNotFoundException When Running Oracle Identity Manager Configuration"
- Section 16.10.2.2, "ResourceConnectionValidationxception When Creating User in Oracle Identity Manager"

- Section 16.10.3.1, "Transaction Timeout Error"

16.10.2.1 java.io.FileNotFoundException When Running Oracle Identity Manager Configuration

Problem

When you run Oracle Identity Manager configuration, the error `java.io.FileNotFoundException: soaconfigplan.xml (Permission denied)` may appear and Oracle Identity Manager configuration might fail.

Solution

To workaround this issue:

1. Delete the file `/tmp/oaconfigplan.xml`.
2. Start the configuration again (`IAM_ORACLE_HOME/bin/config.sh`).

16.10.2.2 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.

16.10.3 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:

16.10.3.1 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.

16.10.4 Using My Oracle Support for Additional Troubleshooting Information

You can use My Oracle Support (formerly MetaLink) to help resolve Oracle Fusion Middleware problems. My Oracle Support contains several useful troubleshooting resources, such as:

- Knowledge base articles
- Community forums and discussions
- Patches and upgrades
- Certification information

> **Note:** You can also use My Oracle Support to log a service request.

You can access My Oracle Support at `https://support.oracle.com`.

16.10.5 OIM Reconciliation Jobs Fail

OIM reconciliation jobs fail, or the following message is seen in the log files:

```
LDAP Error 53 : [LDAP: error code 53 - Full resync required. Reason: The provided
cookie is older than the start of historical in the server for the replicated
domain : dc=mycompany,dc=com
]]
```

This error is caused by Oracle Unified Directory not been written to for a certain amount of time, and the data in the OUD changelog cookie has expired.

Solution:

1. Open a browser and go to the following location:

   ```
   https://admin.mycompany.com
   ```

2. Log in a as `xelsysadm` using the `XelsysadmUserPswd`.
3. Under **System Management**, click **Scheduler**.

4. Under **Search Scheduled Jobs**, enter `LDAP  *` (there is a space before *) and hit **Enter**.

5. For each job in the search results, click on the job name on the left, then click **Disable** on the right.

   Do this for all jobs. If the job is already disabled do nothing.

6. Run the following commands on OUDHOST1:

   ```
   cd OUD_ORACLE_INSTANCE/OUD/bin
   ./ldapsearch -h idmhost1 -p 1389 -D "cn=oudadmin" -b "" -s base "objectclass=*"
   lastExternalChangelogCookie

   Password for user 'cn=oudadmin': <OudAdminPwd>
   dn:
   lastExternalChangelogCookie: dc=mycompany,dc=com:00000140c682473c263600000862;
   ```

   Copy the output string that follows `lastExternalChangelogCookie:`. This value is required in the next step. For example,

   ```
   dc=mycompany,dc=com:00000140c682473c263600000862;
   ```

   The Hex portion must be 28 chars long. If this value has more than one Hex portion then separate each 28char portion with a space. For example:

   ```
   dc=mycompany,dc=com:00000140c4ceb0c07a8d00000043 00000140c52bd0b9104200000042
   00000140c52bd0ba17b9000002ac 00000140c3b290b076040000012c;
   ```

7. Run each of the following LDAP reconciliation jobs once to reset the last change number.:

   - LDAP Role Delete Full Reconciliation
   - LDAP User Delete Full Reconciliation
   - LDAP Role Create and Update Full Reconciliation
   - LDAP User Create and Update Full Reconciliation
   - LDAP Role Hierarchy Full Reconciliation
   - LDAP Role Membership Full Reconciliation

   To run the jobs:

   a. Login to the OIM System Administration Console as the user `xelsysadm`.

   b. Under **System Management**, click **Scheduler**.

   c. Under **Search Scheduled Jobs**, enter `LDAP  *` (there is a space before *) and hit **Enter**.

   d. Click on the job to be run.

   e. Set the parameter **Last Change Number** to the value obtained in step 6.

      For example:

      ```
      dc=mycompany,dc=com:00000140c4ceb0c07a8d00000043
      00000140c52bd0b9104200000042 00000140c52bd0ba17b9000002ac
      00000140c3b290b076040000012c;
      ```

   f. Click **Run Now**.

   g. Repeat for each of the jobs in the list at the beginning of this step.

8. For each incremental recon job whose last changelog number has been reset, execute the job and check that the job now completes successfully.
9. After the job runs successfully, re-enable periodic running of the jobs according to your requirements.

If the issue continues to occur, increase the cookie retention time to two months by running the following command on each OUD instance:

If, the error appears again after the incremental jobs have been re-enabled and run successfully ("Full resync required. Reason: The provided cookie is older..."), then increase the OUD cookie retention time. Although there is no hard and fast rule as to what this value should be, it should be long enough to avoid the issue, but small enough to avoid unnecessary resource consumption on OUD. One or two weeks should suffice; two week is given in the following example:

```
./dsconfig set-replication-server-prop --provider-name "Multimaster
Synchronization" --set replication-purge-delay:2w -D cn=oudadmin --trustAll -p
4444 -h IDMHOST1

Password for user 'cn=oudadmin':  <OudAdminPswd>
Enter choice [f]: f
```

16.10.6 LDAP Reconciliation Jobs Fail with LDAP 32 - USER SEARCH BASE WAS CORRECT ON FILES

Problem: The system management console under **IT resources**, **Directory**, will contain a search base level above what it needs to be.

For example:

```
dc=com
```

Instead of:

```
dc=mycompany, dc=com
```

Solution:

Edit the `adapters.os_xml` file located in the following directory:

```
DOMAIN_HOME/config/fmwconfig/ovd/oim
```

Add the searchbase into `<remoteBase>` and `<root>`

```
***Changed IT Resource and applied the changes on sysadmin console to :
dc=mycompany,dc=com
```

Go to the System Administration Console at the following URL:

```
http://ssomycompany.com/sysadmin
```

When prompted, enter your administration ID and password.

Got to **Configuration**, and then **IT Resource**.

In the **Search** field enter **Directory Services** and click **Search**.

Edit : `dc=com` to `dc=mycompany,dc=com`, and click **Save**.

If after you change the search base you see the following error:

```
INSUFFICIENT PRIVILEGES LDAP 50
```

Add the following ACI to the `config.ldif` file of both OUD servers.

```
ds-cfg-global-aci: (targetcontrol="1.2.840.113556.1.4.319") (version 3.0; acl
"page read control access"; allow(read)
userdn="ldap:///cn=oimLDAP,cn=systemids,dc=mycompany,dc=com";)
```

A

Worksheet for Identity Management Topology

This appendix contains worksheets to help you keep track of machine names, IP addresses, directories, and other important data.

We recommend that you open the PDF version if this Guide in a PDF reader and print out this appendix. Update these worksheet as you set up your enterprise deployment.

This chapter contains the following worksheets:

- Section A.1, "Hosts, Virtual Hosts, and Virtual IP Addresses for Identity Management"
- Section A.2, "Directory Mapping"
- Section A.3, "Port Mapping"
- Section A.4, "LDAP Directory Details"
- Section A.5, "Database Details"
- Section A.6, "Web Tier Details"
- Section A.7, "Application Tier Details"
- Section A.8, "Account Mapping"

A.1 Hosts, Virtual Hosts, and Virtual IP Addresses for Identity Management

Use this worksheet to record information about hosts and IP addresses.

Table A–1 Hosts, Virtual Hosts, and Virtual IP Addresses for topologyName Worksheet Table

Documented Alias	Type	Your Host Name	IP Address
WEBHOST1	Host		
WEBHOST2	Host		
IDMHOST1	Host		
IDMHOST2	Host		
IDMDBHOST1	Database Host		
IDMDBHOST2	Database Host		
ADMINVHN	Virtual Host		
SOAHOST1VHN	Virtual Host		

Table A–1 (Cont.) Hosts, Virtual Hosts, and Virtual IP Addresses for topologyName Worksheet Table

Documented Alias	Type	Your Host Name	IP Address
SOAHOST2VHN	Virtual Host		
OIMHOST1VHN	Virtual Host		
OIMHOST2VHN	Virtual Host		
idminternal.mycompany.com	OTD Virtual Name		
oudinternal.mycompany.com	OTD Virtual Host Name for load balancing of OUD instances		
sso.mycompany.com	Load Balancer Virtual Name		
ADMIN.mycompany.com	Load Balancer Virtual Name		
IDMDomain	Domain Name		n/a

A.2 Directory Mapping

Use this worksheet to keep track of directories.

Table A–2 Directory Mapping Table

Documented Variable	Sample Directory Path	Shared	Your Directory Path
IAM_MW_HOME	/u01/oracle/products/access	Yes	
IAM_ORACLE_HOME	/u01/oracle/products/access/iam	Yes	
WEB_MW_HOME	/u02/private/oracle/products/web		
SOA_ORACLE_HOME	/u01/oracle/products/access/soa	Yes	
OUD_ORACLE_HOME	/u01/oracle/products/access/oud	Yes	
WEB_ORACLE_HOME	/u02/private/oracle/products/web/web		
ORACLE_COMMON_HOME	/u01/oracle/products/access/oracle_common	Yes	
WL_HOME	/u01/oracle/products/access/wlserver_10.3	Yes	
JAVA_HOME	/u01/oracle/products/access/jrockit_version	Yes	
OUD_ORACLE_INSTANCE	/u02/private/oracle/config/instances/oud*n*	No	
WEB_ORACLE_INSTANCE	/u02/private/oracle/config/instances/webn	No	
ASERVER_HOME	/u01/oracle/config/domains/IDMDomain	Yes	
MSERVER_HOME	/u02/private/oracle/config/domains/IDMDomain	No	

A.3 Port Mapping

Use this worksheet to keep track of ports.

Table A–3 Port Mapping Table

Documented Port	Description	Your Port
443	SSL Port for accessing the site externally	
80	Non SSL Port used for accessing admin functions internally	
389	LDAP Access Port on Load Balancer	
636	LDAPS Access Port from Load Balancer	
1389	OUD Access port	
1636	OUD SSL Access port	
4444	OUD Admin Port	
8899	OUD Replication Port	
7777	Oracle HTTP Server Listen Port	
5575	OAM Listen Port	

A.4 LDAP Directory Details

Use this worksheet to keep track of LDAP information.

Table A–4 LDAP Directory Details Table

Description	Documented Value	Customer Value
LDAP Directory Hosts	IDMHOST1 IDMHOST2	
LDAP Directory SSL Port	1636	
LDAP Directory Non SSL Port	1389	
LDAP Administration Port	4444	
Back end Directory Type	OUD	
LDAP Virtual host	oudinternal.mycompany.com	
LDAP Load Balanced Non-SSL Port	636	
LDAP Administration User	cn=oudadmin	
OUD_ORACLE_INSTANCE	/u02/private/oracle/config/instances/oud1 /u02/private/oracle/config/instances/oud2	
LDAP Directory Tree	dc=mycompany,dc=com	
LDAP Group Search Base	cn=Groups,dc=mycompany,dc=com	
LDAP User Search Base	cn=Users,dc=mycompany,dc=com	

Table A–4 (Cont.) LDAP Directory Details Table

Description	Documented Value	Customer Value
LDAP Reserve Location	cn=Reserve,dc=mycompany,dc=com	
LDAP System ID Location	cn=systemids,dc=mycompany,dc=com	

A.5 Database Details

Use this worksheet to keep track of database information.

Table A–5 Database Details Table

Description	Documented Value	Customer Value
Database Hosts	IDMDBHOST1 IDMDBHOST2	
Scan Address Name	db-scan	
Database Name	idmdb.mycompany.com	
Database Service Names defined	oamedg.mycompany.com oimedg.mycompany.com oesedg.mycompany.com	

A.6 Web Tier Details

Use this worksheet to keep track of Web Tier information.

Table A–6 Web Tier Details Table

Description	Documented Value	Customer Value
Web Tier Hosts	WEBHOST1 WEBHOST2	
Oracle HTTP Server Listen Port	7777	
WEB_ORACLE_HOME	/u02/private/oracle/products/web/web	
WEB_ORACLE_INSTANCE	/u02/private/oracle/config/instances/web1 /u02/private/oracle/config/instances/web2	
Virtual Hosts	admin.mycompany.com sso.mycompany.com internal.mycompany.com	
System Account Name and Password	system/xxxxx	
RCU Schema Prefix	EDG	

Table A–6 (Cont.) Web Tier Details Table

Description	Documented Value	Customer Value
ONS Port	6200	
Listener Port	1521	

A.7 Application Tier Details

Use this worksheet to keep track of Application Tier information

Table A–7 Application Tier Details Table

Description	Documented Value	Customer Value
Host (Virtual Hosts)	IDMHOST1 (ADMINVHN, OIMHOST1VHN, SOAHOST1VHN) IDMHOST2 (OIMHOST2VHN, SOAHOST2VHN	
Domain Name	IDMDomain	
ASERVER_HOME	/u01/oracle/config/domains/IDMDomain	
MSERVER_HOME	/u02/private/oracle/config/domains/IDMDomain	
Components Installed	OAM Console, OES Console, OIN, OAM, OIM	
OAM Managed Server Names	WLS_OAM1 WLS_OAM2	
OIM Managed Server Names	WLS_OIM1 WLS_OIM2	
OAM Managed Server Port	14100	
OIM Managed Server Port	14000	

A.8 Account Mapping

Use this worksheet to keep track of administrative accounts.

Table A–8 User Mapping Table

configTool Parameter	Documented Value	Customer Value
IDSTORE_OAMADMINUSER	oamadmin	
IDSTORE_OAMSOFTWAREUSER	oamLDAP	
OAM11G_IDSTORE_ROLE_SECURITY_ADMIN	OAMAdministrators	
IDSTORE_OIMADMINGROUP	OIMAdministrators	
IDSTORE_OIMADMINUSER	weblogic_idm	

Table A–8 (Cont.) User Mapping Table

configTool Parameter	Documented Value	Customer Value
IDSTORE_WLSADMINGROUP	WLSAdmins	

B Using Multi Data Sources with Oracle RAC

Oracle recommends using GridLink data sources when developing new Oracle RAC applications. However, if you are using legacy applications and databases that do not support GridLink data sources, refer to the information in this appendix.

This appendix provides the following topics:

- Section B.1, "About Multi Data Sources and Oracle RAC"
- Section B.2, "Typical Procedure for Configuring Multi Data Sources for an EDG Topology"

B.1 About Multi Data Sources and Oracle RAC

A multi data source provides an ordered list of data sources to use to satisfy connection requests. Normally, every connection request to this kind of multi data source is served by the first data source in the list. If a database connection test fails and the connection cannot be replaced, or if the data source is suspended, a connection is sought sequentially from the next data source on the list.

For more information about configuring Multi Data Sources with Oracle RAC, see "Using Multi Data Sources with Oracle RAC" in the *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server*.

B.2 Typical Procedure for Configuring Multi Data Sources for an EDG Topology

You configure data sources when you configure a domain. For example, when you are configuring the initial Administration domain for an Enterprise Deployment reference topology, you use the configuration wizard to define the characteristics of the domain, as well as the data sources.

The procedures for configuring the topologies in this Enterprise Deployment Guide include specific instructions for defining GridLink data sources with Oracle RAC. If you want to use Multi Data Sources instead of GridLink data sources, replace the GridLink instructions with the following:

1. In the Configure JDBC Component Schema screen:

   a. Select the appropriate schemas.

   b. For the RAC configuration for component schemas, **Convert to RAC multi data source**.

   c. Ensure that the following data source appears on the screen with the schema prefix when you ran the Repository Creation Utility.

   d. Click **Next**.

2. The Configure RAC Multi Data Sources Component Schema screen appears In this screen, do the following:

   a. Enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU.

      – **Driver:** Select **Oracle driver (Thin) for RAC Service-Instance connections, Versions:10, 11**.

      – **Service Name:** Enter the service name of the database.

      – **Username:** Enter the complete user name (including the prefix) for the schemas.

      – **Password:** Enter the password to use to access the schemas.

   b. Enter the host name, instance name, and port.

   c. Click **Add**.

   d. Repeat this for each Oracle RAC instance.

   e. Click **Next**.

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

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

C

Enterprise Topology with Oracle HTTP Server

This chapter describes an Oracle Identity Manager enterprise deployment on Exalogic with an external Oracle HTTP Server Web tier. It is one of the alternative topologies, discussed in Section 2.1.2, "Alternative Deployment Topologies."

This appendix contains the following topics:

- Viewing the Oracle Identity Management Deployment Topology with Oracle HTTP Server on Exalogic
- Understanding the Oracle Identity Manager with Oracle HTTP Server Topology Components

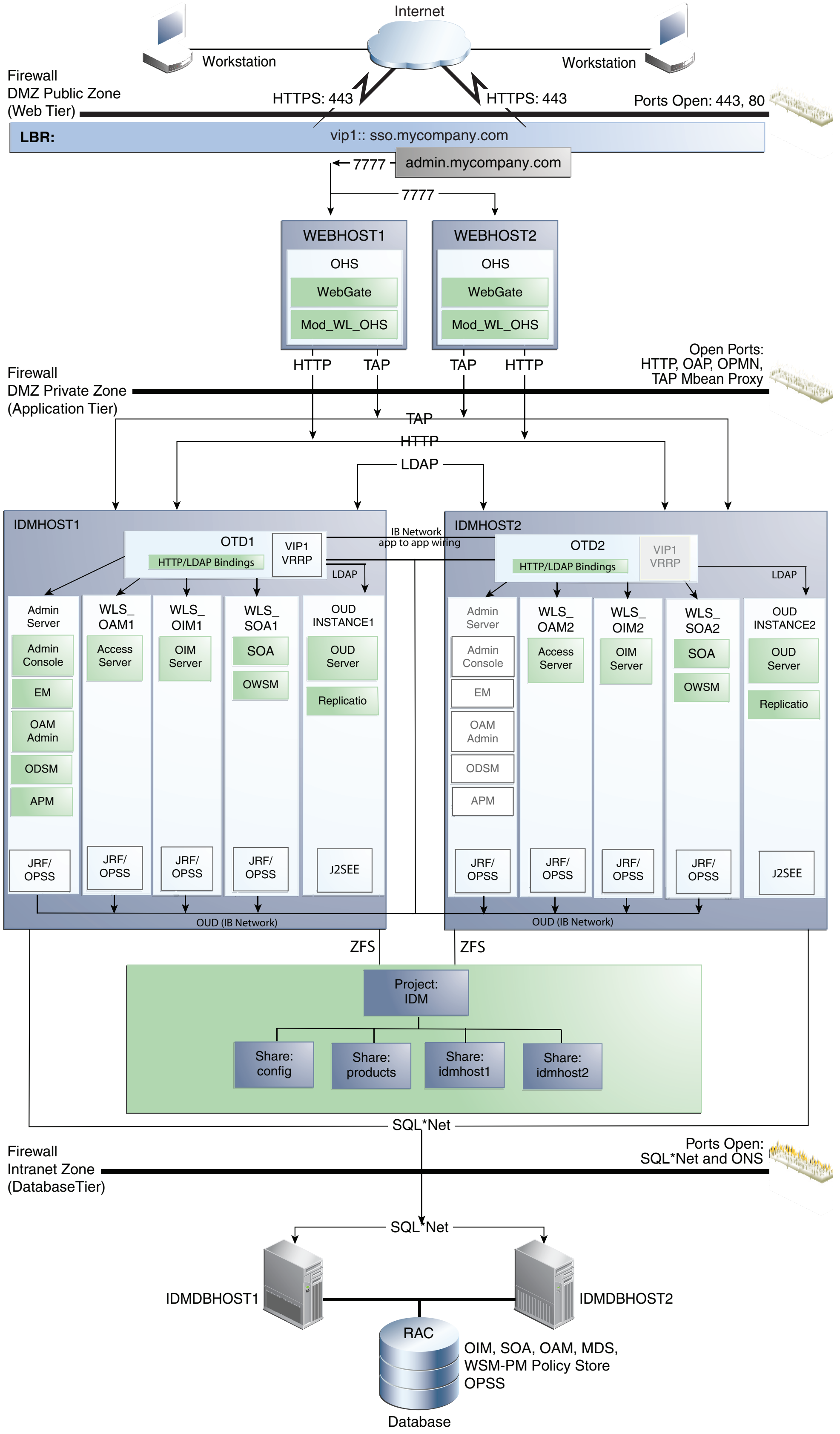
C.1 Viewing the Oracle Identity Management Deployment Topology with Oracle HTTP Server on Exalogic

In this alterntative Oracle Identity Manager topology on Exalogic topology, user requests are being routed by an Oracle HTTP Server Web tier, rather than the Oracle Traffic Director Web listeners.

Compare this topology with the one shown in Chapter 2, "Introduction and Planning.".

The Oracle HTTP Server topology shown here is similar to the one documented in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management* available in the Oracle Identity Management 11g Release 2 (11.1.2.1) documentation library, which is available on the Oracle Technology Network (OTN).

Figure C–1 Identity Management with Oracle HTTP Server and an Oracle RAC Database



C.2 Understanding the Oracle Identity Manager with Oracle HTTP Server Topology Components

The components of the alternative Oracle Identity Manager with Oracle HTTP Server topology are identical to those described in Chapter 2, except for the following:

- Section C.2.1, "About the Oracle HTTP Server Instances in the Web Tier"
- Section C.2.2, "About the Oracle Traffic Director Instances on the Application Tier"

C.2.1 About the Oracle HTTP Server Instances in the Web Tier

The Web tier in the Oracle HTTP Server topology consists of two Oracle HTTP Server instances on separate WEHOST1 and WEBHOST2 host computers. These computers are outside of the Exalogic machine, and a firewall separates them from the application tier.

Most of the Identity Management components can function without the Web tier, but for most enterprise deployments, the Web tier is desirable.

In the Web tier:

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

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

For information about configuring the Web tier for an IDM enterprise deployment, see "Installing and Configuring Oracle Web Tier for an Enterprise Deployment" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*

C.2.2 About the Oracle Traffic Director Instances on the Application Tier

Similar to the topology in Section 2.3, "Understanding the Topology Components," Oracle Traffic Director is used as a load balancer for internal communications within the Exalogic rack. By using Oracle Traffic Director rather than routing requests through the load balancer, you can utilise the internal IPoIB network which is both more secure and faster.

In this topology, the Oracle Traffic Director instances are in an active-passive configuration and the required virtual IP addresses used for internal communication (such as oudinternal.mycompany.com) are defined in the Oracle Traffic Director configuration.

For more information on configuring Oracle Traffic Director failover groups for active-passive mode, see "Creating Failover Groups" in the *Oracle Traffic Director Administrator's Guide*.