

Oracle® Access Manager

Installation Guide

10g (10.1.4.0.1)

B25353-01

July 2006

This guide describes everything you need to know to successfully install Oracle Access Manager in your environment.

Oracle Access Manager Installation Guide 10g (10.1.4.0.1)

B25353-01

Copyright © 2000, 2006, Oracle. All rights reserved.

Primary Author: Gail Tiberi

Contributing Author: Nina Wishbow, Norman Bock

Contributor: Paresh Borkar, Pradnyesh Rane, Ramakrishna Narla, Chetan Barhate, Steven Frehe.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Contents

Preface	xvii
Audience	xvii
Documentation Accessibility	xvii
Related Documents	xviii
Conventions	xix
What's New in Oracle Access Manager	xxi
Product and Component Name Changes	xxi
New Features in Oracle Access Manager 10g (10.1.4.0.1)	xxii
Updates to Specific Chapters	xxiii
Part I Installation Planning and Prerequisites	
1 About the Installation Task, Options, and Methods	
About the Installation Task	1-1
Installation Options	1-5
Updating the Schema and Attributes Automatically Versus Manually	1-5
Replicating an Installed Oracle Access Manager Component	1-7
Silent Mode	1-8
Cloning and Synchronizing Installed Components	1-8
Upgrading from a Earlier Release of Oracle Access Manager	1-8
Installation Methods	1-8
GUI Method	1-8
Console Method	1-9
2 Preparing for Installation	
About Installation Prerequisites	2-1
Setting Secure Passwords and Permissions	2-2
Synchronizing System Clocks	2-2
About the Network Time Protocol	2-3
On Unix Systems	2-3
On Windows Systems	2-3
Meeting Oracle Access Manager Requirements	2-4
General Guidelines	2-4

Preparing Linux Host Machines	2-5
Identity System Guidelines.....	2-6
Access System Guidelines.....	2-6
Policy Manager Guidelines	2-6
Access Server Guidelines.....	2-7
WebGate Guidelines.....	2-8
Assessing Disk Space Requirements	2-10
Choosing an Installation Directory.....	2-10
Securing Oracle Access Manager Component Communications	2-11
Transport Security Guidelines	2-11
Open Mode	2-12
Simple Mode.....	2-12
Cert Mode.....	2-13
Meeting Web Server Requirements	2-14
Web Server-Specific Installation Packages	2-15
General Considerations for Web Servers.....	2-16
Meeting Directory Server Requirements.....	2-17
Assigning a Bind DN.....	2-18
Assessing Directory Server Space.....	2-18
Securing Directory Server Communications.....	2-19
Guidelines	2-19
Caveats	2-19
Data Storage Requirements	2-21
User Data and the Searchbase	2-24
Configuration Data and the Configuration DN	2-25
Policy Data and the Policy base	2-26
About Person and Group Object Classes.....	2-26
Confirming Platform Requirements.....	2-26
Preparing a Temporary Directory for Installers	2-27
Uninstalling Oracle Access Manager Components.....	2-28
Installation Preparation Checklists	2-28

3 About Multi-Language Environments

About Installing in Multi-Language Environments	3-1
Setting Environment Variables for Command-Line Tools (Optional).....	3-2
Setting NLS_LANG and COREID_NLS_LANG on Windows Systems	3-4
Setting NLS_LANG and COREID_NLS_LANG on Unix Systems.....	3-4
Installing with Language Packs	3-5
Directory Structure.....	3-6
Language Directories.....	3-7
Removing Language Packs.....	3-7

Part II Identity System Installation and Setup

4 Installing the Identity Server

About the Identity Server and Installation	4-1
---	------------

The Identity Server and the Software Developer Kit.....	4-2
About Installing Multiple Identity Servers	4-2
Adding a New Identity Server to an Upgraded Environment	4-3
Identity Server Prerequisites Checklist	4-4
Installing the Identity Server	4-4
Starting the Installation	4-5
Installing the Identity Server	4-6
Specifying a Transport Security Mode.....	4-7
Specifying Identity Server Configuration Details	4-7
Defining Communication Details	4-8
Defining Directory Server Details.....	4-9
Installing the First Identity Server.....	4-10
Installing Additional Identity Servers on Windows.....	4-12
Finishing the Identity Server Installation	4-13
Tuning Oracle Internet Directory	4-13

5 Installing WebPass

About WebPass and Installation	5-1
About Installing Multiple WebPass Instances	5-2
WebPass Prerequisites Checklist	5-2
Installing the WebPass	5-2
Starting the Installation	5-3
Specifying a Transport Security Mode.....	5-4
Specifying WebPass Configuration Details.....	5-4
Updating the WebPass Web Server Configuration.....	5-5
Finishing the WebPass Installation.....	5-6
Manually Configuring Your Web Server	5-6
Verifying WebPass Permissions on IIS	5-7
Establishing Communication with the Identity Server.....	5-8
Confirming WebPass Installation	5-8

6 Setting Up the Identity System

About Setting Up the Identity System	6-1
Identity System Setup Considerations	6-2
Identity System Setup Prerequisites Checklist.....	6-4
Setting up the Identity System	6-4
Starting the Setup Process.....	6-4
Specifying Directory Server and Data Location Details	6-5
Specifying Object Class Details	6-6
About Oracle Access Manager Object Classes.....	6-7
Specifying Person and Group Object Classes	6-7
Confirming Object Class Changes	6-8
Configuring Master Administrators.....	6-8
Completing Identity System Setup.....	6-9
Configuring Attributes Manually.....	6-10
Novell Directory Server Considerations.....	6-10

Configuring or Refining Attributes	6-11
Setting Up Other Identity Server Instances	6-12

Part III Access System Installation and Setup

7 Installing the Policy Manager

About Policy Manager Installation and Setup	7-1
About Installing Multiple Policy Managers	7-2
Policy Manager Prerequisites Checklist	7-2
Installing the Policy Manager	7-2
Starting the Installation	7-3
Defining a Directory Server Type and Policy Data Location	7-4
Continuing on Solaris Without Updating the Schema	7-4
Continuing on Windows Without Updating the Schema	7-5
Storing Policy Data Separately and Updating the Schema	7-5
Specifying a Transport Security Mode	7-6
Updating Your Policy Manager Web Server Configuration	7-7
Finishing the Policy Manager Installation	7-8
Manually Configuring Your Web Server	7-8
Verifying Policy Manager Permissions on IIS	7-9
Setting Up the Policy Manager	7-9
Starting the Setup Process	7-9
Specifying Directory Server Details and Data Locations	7-10
Configuring Authentication Schemes	7-13
Completing Policy Manager Setup	7-14
Confirming Policy Manager Setup	7-14

8 Installing the Access Server

About the Access Server and Installation	8-1
About Installing Multiple Access Servers	8-2
Adding New Access Servers to an Upgraded Environment	8-2
Access Server Prerequisites Checklist	8-3
Creating an Access Server Instance in the System Console	8-4
Installing the Access Server	8-5
Starting the Installation	8-5
Specifying a Transport Security Mode	8-6
Specifying Directory Server and Communication Details	8-6
Finishing the Access Server Installation	8-7

9 Installing the WebGate

About WebGate Installation	9-1
About Installing Multiple WebGates	9-2
WebGate Prerequisites Checklist	9-2
Creating a WebGate Instance	9-2
Associating a WebGate and Access Server	9-4
Installing the WebGate	9-5

Starting the Installation	9-5
Specifying a Transport Security Mode.....	9-6
Specifying WebGate Configuration Details	9-6
Updating the WebGate Web Server Configuration	9-7
Finishing the WebGate Installation	9-8
Manually Configuring Your Web Server	9-8
Completing IIS WebGate Installations	9-9
Enabling SSL on the IIS Web Server	9-9
Ordering the ISAPI Filters	9-10
Installing postgate.dll on IIS Web Servers.....	9-10
Setting Up IIS Web Server Isolation Mode.....	9-11
Installing the Postgate ISAPI Filter	9-11
Protecting a Web Site When the Default Site is Not Setup	9-12
Completing httpd.conf Updates	9-12
Confirming WebGate Installation	9-13

Part IV Installing Optional Components

10 Setting Up Oracle Access Manager with Oracle Virtual Directory

About Oracle Access Manager Implementations with Oracle Virtual Directory	10-2
Key Terms and Features.....	10-2
Federated Data Stores.....	10-4
About Searchbase Options	10-5
Split Profiles.....	10-7
Aggregated Namespaces	10-8
Aggregated Schema Mapping.....	10-8
Implementation Limitations	10-9
About Limitations on Multi-Value Attributes	10-10
About Limitations on Embedded Virtual Data Sources.....	10-11
Implementation Architecture	10-11
About Oracle Virtual Directory Drivers and Adapters.....	10-14
About Oracle Access Manager-Specific Data.....	10-14
About Schema Extension	10-15
Virtual Directory Schema.....	10-17
Target Directory Schemas	10-17
About Adding Attributes to Target Database Tables	10-17
Customer Schemas.....	10-18
Implementation Scenarios and Limitations	10-19
Heterogeneous LDAP Directories	10-19
Multiple RDBMS Databases	10-20
About Joining Database Tables in an Embedded Virtual Data Source.....	10-21
Split-Profiles.....	10-22
Join View Adapter Requirements and Limitations.....	10-23
Implementation Requirements	10-24
Security Connection Support	10-25
Authentication Support.....	10-26

About Pass Credential Authentication	10-26
Access Control Support.....	10-26
Failover Support.....	10-26
About the Implementation Process	10-28
Implementing Oracle Virtual Directory when Installing Oracle Access Manager.....	10-28
Implementing Oracle Virtual Directory with Existing Oracle Access Manager Installations	10-28
Preparing Your Environment	10-30
Identifying Factors for Designing Your Implementation.....	10-30
Preparing Directory Servers for Implementation.....	10-33
Preparing Relational Databases for Implementation.....	10-33
Installing and Configuring Oracle Virtual Directory and Virtual Directory Manager	10-34
Installing Oracle Virtual Directory	10-35
Installing Virtual Directory Manager.....	10-35
Creating a Project Space and Server	10-35
Obtaining/Updating Sample Adapter and Mapping Templates	10-36
Deploying JDBC Driver Libraries for Your RDBMS.....	10-37
Configuring the Oracle Virtual Directory SSL Listener (Optional).....	10-37
Installing the First Identity Server.....	10-39
Extending Directory Schemas.....	10-40
Creating Mapping Files for Adapters.....	10-43
Creating Data Store Adapters	10-44
Creating Adapters for LDAP Directories	10-44
Configuring a Database Adapter	10-47
Creating a Split-Profile Adapter	10-49
Creating a Multiple-Directories Adapter.....	10-51
Creating an Local Data Store Adapter	10-51
Creating a Physical Node for the Virtual Root.....	10-52
Customizing Adapters and Mapping Files	10-52
Customization Examples	10-52
Customized Mapping Script for Active Directory	10-53
Customized Mapping Script for Oracle Database	10-58
Customized Adapter for Oracle Database	10-60
Customizing General Settings for Oracle Access Manager	10-63
Customizing Routing Settings.....	10-64
Editing an Adapter Plug-in to Refer to Your Mapping File	10-64
Completing Identity System Installation and Setup	10-66
Testing Your Implementation	10-67
Reference Information	10-67
Oracle Access Manager Auxiliary Attributes	10-67
About DN Conversion Toolkit.....	10-70
Conditions.....	10-72
Requirements.....	10-73
Details	10-73
Oracle Access Manager-Oracle Virtual Directory Implementation Templates.....	10-73
Templates for Active Directory	10-74
OblixADAdapterUsingMapper for Active Directory.....	10-74
OblixADAdapterUsingScript for Active Directory	10-76

OblixADSSLAdapterUsingMapper for Active Directory	10-76
Templates for ADAM	10-76
OblixADAMAdapterUsingMapper for ADAM	10-77
OblixADAMAdapterUsingScript for ADAM	10-78
OblixADAMSSLAdapterUsingMapper for ADAM	10-79
Templates for Sun Directory Server	10-79
OblixSunOneAdapterUsingMapper for SunOne	10-79
OblixSunOneAdapterUsingScript for SunOne	10-79
Templates for eDirectory	10-80
OblixeDirectoryAdapterUsingMapper for eDirectory	10-80
OblixeDirectoryAdapterUsingScript for eDirectory	10-80
Database Template: OblixDBAdapterUsingScript	10-80
Schema Mapping Script Templates	10-80
Tips	10-81
Database Connectivity Tips	10-82
Troubleshooting Implementations with Oracle Virtual Directory	10-84

11 Installing the SNMP Agent

About the SNMP Agent and Installation	11-1
SNMP Agent Installation Considerations	11-1
SNMP Installation Prerequisites Checklist	11-2
Installing the Oracle Access Manager SNMP Agent	11-2
Starting the Installation	11-2
Specifying SNMP Agent Configuration Details	11-3
Finishing the Installation	11-4

12 Installing Language Packs Independently

About Language Packs and Installation	12-1
Language Pack Installation Considerations	12-3
Language Pack Prerequisites Checklist	12-4
Installing the Language Pack Independently	12-4
Installed Files	12-5
Confirming Language Status	12-5

13 About Installing Audit-to-Database Components

14 About the Software Developer Kit

Part V Replication

15 Replicating Components

About the Silent Mode Options File	15-1
Additional Uses of the Silent Mode Options File	15-2
Running the Silent Mode Options File	15-2
Selecting an Installation Directory on HP-UX and AIX	15-2

Inputting Installation Passwords	15-3
Editing the Silent Mode Options File	15-3
Sample Options Files	15-3
Sample Access Server Options Files.....	15-3
Silent Mode Parameters	15-6
Identity Server Parameters	15-6
WebPass Parameters.....	15-11
Policy Manager Parameters.....	15-13
Access Server Parameters	15-15
WebGate Parameters	15-19
Access Manager SDK Parameters.....	15-22
BEA WebLogic SSPI Parameters.....	15-22
WAS Registry Parameters.....	15-26
Uninstalling a Component Installed With Silent Mode	15-29
Cloning and Synchronizing Installed Components.....	15-29
An Example of Using np_sync.....	15-30
Syntax and Options for np_sync	15-30
UNIX-Specific Notes.....	15-31
Windows-Specific Notes	15-31
Uninstalling a Cloned Component	15-32
Uninstalling a Cloned Component on UNIX.....	15-32
Uninstalling a Cloned Component on Windows	15-32
Uninstalling Oracle Access Manager System	15-33

Part VI Web Server Configuration

16 Configuring the Apache v1.3 and Oracle HTTP Server Web Servers

About OHS and Oracle Access Manager	16-1
OHS Web Component Caveats on Linux	16-2
OHS Web Component Caveats on Linux and Windows Platforms	16-2
About Apache v1.3 and Oracle Access Manager	16-2
Identity Server Accessed through WebPass.....	16-2
Policy Manager.....	16-3
WebGate	16-3
Example: Apache v1.3 Configuration for Unix Systems.....	16-3
Apache v1.3, Oracle HTTP Server (OHS), and Stronghold Requirements	16-4
Apache v1.3, OHS, and IHS Web Server Support	16-5
Downloading and Compiling the Base Apache Web Server	16-6
Apache Release Notes	16-6
Other Useful Links.....	16-6
Platform-Specific Compilation Options	16-7
Platform Specific Run-Time Settings for AIX	16-7
Installation Order for Oracle Access Manager Web Components	16-8
Updating Web Server Configuration for Oracle Access Manager Web Components	16-8
Tuning Apache 1.3 for Oracle Access Manager Web Components	16-8
Policy Manager Tuning Factors	16-10
Setting OHS Client Certificates	16-10

Tuning OHS for Oracle Access Manager Web Components	16-11
Starting and Stopping the Web Server	16-11
Starting and Stopping OHS Web Servers	16-12
Starting and Stopping Apache on Unix	16-12
Stopping Apache Web Server on Unix	16-12
Starting and stopping the Apache Web Server on Unix	16-12
Starting the Server in SSL Mode	16-12
Starting and Stopping Apache on Windows.....	16-13

17 Configuring Apache v2, IHS, and OHS Web Servers for Oracle Access Manager

About OHS and Oracle Access Manager	17-1
About Oracle Access Manager with Apache and IHS v2 Web Components.....	17-2
About the Apache HTTP Server	17-3
About the IBM HTTP Server	17-3
About the Apache and IBM HTTP Reverse Proxy Server.....	17-3
About Apache v2 Architecture and Oracle Access Manager	17-4
Compatibility and Platform Support	17-6
Requirements for OHS/IHS/Apache v2 Web Servers	17-6
Requirements for IHS2 Web Servers.....	17-6
Requirements for Apache and IHS v2 Reverse Proxy Servers	17-7
Requirements for Apache v2 Web Servers.....	17-7
Preparing Your Web Server	17-8
Preparing the IHS v2 Web Server	17-9
Preparing the Host for IHS v2 Installation.....	17-9
Installing the IBM HTTP Server v2	17-10
Setting Up SSL-Capability	17-11
Starting a Secure Virtual Host.....	17-11
Preparing Apache and OHS Web Servers on Linux	17-12
Preparing OHS Web Servers on Linux	17-12
Preparing OHS Web Servers on Linux and Windows Platforms	17-12
Setting OHS Client Certificates	17-13
Preparing the Apache v2 Web Server on Unix	17-13
Preparing the Apache v2 SSL Web Server on AIX.....	17-17
Preparing the Apache v2 Web Server on Windows	17-18
Activating Reverse Proxy.....	17-20
Activating Reverse Proxy For Apache v2 Web Servers.....	17-20
Activating Reverse Proxy For IHS v2 Web Servers	17-21
Installing Oracle Access Manager Web components	17-23
Manually Updating a Web Server Configuration for Oracle Access Manager	17-24
Verifying httpd.conf Updates for Oracle Access Manager Web Components	17-25
Verifying WebPass Details.....	17-25
Verifying Policy Manager Details.....	17-27
Verifying WebGate Details	17-28
Verifying Language Encoding	17-30
Tuning OHS for Oracle Access Manager Web Components	17-30
Starting and Stopping OHS Web Servers.....	17-31

Tuning Apache/IHS v2 for Oracle Access Manager Web Components	17-31
Tips and Troubleshooting	17-34
Apache v2 on HP-UX.....	17-34
Apache v2 Bundled with Red Hat Enterprise Linux 4	17-34
Apache v2 on Unix with the mpm_worker_module	17-34
Helpful Information	17-35

18 Setting Up Lotus Domino Web Servers for WebGates

Installing the Domino Web Server	18-1
Setting Up the First Domino Web Server	18-2
Starting the Domino Web Server	18-3
Enabling SSL (Optional).....	18-3
Installing a Domino Security (DSAPI) Filter	18-4
Completing the WebGate Installation	18-5
Tips	18-6

Part VII Product Removal, Tips, and Troubleshooting

19 Important Notes

Enabling Java and JavaScript On The Client.....	19-1
Changing MIME Type Settings	19-1
Choosing a Unique ID for Each User	19-2
Contacting Oracle.....	19-2

20 Removing Oracle Access Manager

Uninstalling Oracle Access Manager Components.....	20-1
Recycling an Identity Server Instance Name.....	20-4

Part VIII Appendixes

A Installing Oracle Access Manager with Active Directory

About Active Directory	A-1
Domain Controllers and Partitions.....	A-2
About Oracle Access Manager and Active Directory	A-2
About Statically-Linked Auxiliary Classes.....	A-3
About Dynamically-Linked Auxiliary Classes	A-3
About Oracle Access Manager and Active Directory Forests	A-4
Oracle Access Manager and the Searchbase in a Parent-Child Domain.....	A-6
Installation and Setup Considerations for Active Directory	A-7
Active Directory Schema Choices.....	A-7
Determining which Schema to Load.....	A-8
All Configurations.....	A-9
ADSI Option Considerations.....	A-10
LDAP Open Bind Considerations.....	A-13
LDAP Over SSL Considerations	A-13

Installing Oracle Access Manager with Active Directory	A-14
Setting Up Your Environment	A-14
Setting Up Domain Controllers	A-14
Installing the Certificate Server	A-15
Retrieving the Certificate	A-15
Installing the Identity System.....	A-16
Installing the Identity System	A-16
Setting Up ADSI (Optional).....	A-17
Setting Up the Identity System	A-17
Enabling Active Directory Attributes	A-18
Enabling Change-Password Permissions	A-18
Setting Up the Identity System	A-18
Validating Your Identity System Setup	A-19
Installing and Setting Up the Access System	A-19
Preparing for Access System Installation	A-20
Installing and Setting Up the Access System.....	A-20
Setting Up ADSI on the Access Server (Optional)	A-21
Active Directory Tips and Troubleshooting.....	A-22

B Installing Oracle Access Manager with ADAM

About Oracle Access Manager and ADAM	B-1
ADAM Instances and Partitions	B-3
The ADAM Schema	B-4
The Oracle Access Manager Schema Extension for ADAM	B-5
Windows Users and Security Principals.....	B-6
Oracle Access Manager Directory Profiles	B-7
Replication of an ADAM Instance	B-7
ADSI with Oracle Access Manager and ADAM.....	B-7
ADAM and APIs	B-8
Authentication, Authorization, and Password Changes.....	B-8
ADAM and Active Directory Differences	B-8
Support Requirements	B-9
Installing Oracle Access Manager with ADAM.....	B-9
Preparing ADAM for Oracle Access Manager	B-9
Installing and Setting the Identity System with ADAM	B-11
Installing the Access System with ADAM	B-14
Oracle Access Manager Silent Mode Installation Parameters.....	B-16
Identity Server Silent Mode Installer for ADAM	B-16
Policy Manager Silent Mode Installer for ADAM.....	B-17
Access Server Silent Mode Installer for ADAM	B-17
Troubleshooting ADAM Issues.....	B-17

C Adding Directory Certificates After Oracle Access Manager Installation

About Directory Certificates	C-1
Prerequisites	C-2
Creating a New Certificate Store.....	C-2

Adding Certificates	C-3
Changing the Directory Server Configuration	C-4

D Changing Directory Server Hosts

About Changing Directory Server Hosts	D-1
Minimizing Down Time	D-1
Configuring Failover between an Identity Server and WebPass	D-2
Configuring Failover between an Access Server and WebGate	D-3
Preparing the New Directory Server Instance	D-4
Reconfiguring the Primary Identity Server.....	D-5
Reconfiguring the Policy Manager	D-6
Reconfiguring the Access Server.....	D-8

E Troubleshooting Installation Issues

Browser Issues	E-1
Character Display Issues.....	E-1
Microsoft Internet Explorer 6 with Sun VM v1.4.2_04	E-2
Unable to Authenticate Resource on Internet Explorer.....	E-2
Directory Server Issues	E-3
Active Directory Issues.....	E-3
Active Directory Search Halts.....	E-3
ADSI Cannot Be Enabled for this DB Profile (Active Directory)	E-4
Dynamically-Linked Auxiliary Classes for Active Directory.....	E-4
ADAM Issues	E-5
ADAM: Cannot find the Config DN or Searchbase.....	E-5
ADAM Directory Server Security	E-5
ADAM Object Classes.....	E-5
ADAM Password Changes.....	E-6
ADAM Schema Updates.....	E-6
Identity System Issues	E-6
Application Has Not Been Set Up	E-6
Cannot Set Up Identity System	E-7
Checking Access Server or Identity Server Availability	E-7
Could Not Get Any DB Profile.....	E-8
Identity Server Does Not Start	E-8
Identity System Components May Fail.....	E-8
IdentityXML Calls Fail After WebGate Install	E-9
WebPass Identifier Not Available After Setup	E-9
IIS and Windows Issues	E-10
Issues with Oracle Virtual Directory Implementations	E-10
Directory Server Problems	E-10
Multi-Value Attribute Problems	E-10
Secondary Data Store Problems.....	E-11
Unexpected Group Deletion Problem	E-12
Installation Issues	E-12
Access Server Installation Halts.....	E-13
CGI Programs Do not Run After Installation	E-13

File Replace Warning When Installing on Windows.....	E-13
Installation Fails with a "bad credentials error (49)"	E-14
Installer Prompts to Replace DLL Files.....	E-14
Performing Unix Installation in GUI Mode	E-14
Quitting a Windows Installation	E-14
Running as Non-Root User When Installing on AIX.....	E-15
Specifying Installation Directories.....	E-15
Testing Your Installation.....	E-15
Unable to Leave Person Object Class Page.....	E-15
WebGate Installation with Apache Web Server on AIX	E-15
Language Issues	E-15
Garbled Password Message	E-16
Installing Additional Administrator Language Packs	E-16
Installing Language Packs for Policy Manager and WebGate in Same Directory	E-16
Removing the Default Administrator Language Pack	E-17
Login Issues	E-17
Identity Server Logged You In, Access System Logged You Out	E-17
Windows 2000 Users Cannot Log in After Installation.....	E-18
Receiving Repeated Login Prompts	E-18
Unable to log in to Oracle Access Manager on IIS	E-18
Restricting Access to Oracle Access Manager.....	E-19
Policy Manager Issues	E-19
Cannot Delete Policy Manager Policy Profile.....	E-20
Reinstalling Oracle Access Manager with Oracle Internet Directory	E-20
Removal Issues	E-21
Transport Security Mode Issues	E-21
User Directory Issues	E-22
Adding User to Replicated Directory.....	E-22
Data Corruption	E-22
Web Server Issues	E-22
Access Server Fails on an Apache Web Server	E-23
Errors, Loss of Access, and Unpredictable Behavior	E-23
Oracle HTTP Server (OHS) Fails to Start	E-23
PCLOSE Error When Starting Sun Web server	E-24
Removing and Reinstalling IIS DLLs	E-24
WebGate Issues	E-25
Access Server and WebGate Naming.....	E-25
Enabling WebGate Diagnostics.....	E-25
Error Messages After Installing WebGate.....	E-26
Installing WebGate and an Identity Server in Same Directory	E-26
Receiving Access Server Down Errors.....	E-26
WebGate Cannot Connect to Access Server.....	E-26
Miscellaneous Issues	E-26
Unable to Flush the Cache	E-27
Giving View Rights to the Master Administrator	E-27
Idle Session Time, Maximum Cookie Session Time.....	E-27
Loading the Directory in Secure Mode	E-27

Peer Does Not Use Oracle Access Protocol	E-27
Receiving Bug Report After Replication Attempt	E-28
Search and Query Error Message (Defect 4547)	E-28
Identity Server Logged You in but Access System Logged You Out	E-28

Index

Preface

This Installation Guide provides information about basic installation and setup of Oracle Access Manager components on supported platforms. Included are considerations, prerequisites, preparation worksheets that you can complete to help streamline your experience, and step-by-step instructions to help ensure your success.

Note: Oracle Access Manager was previously known as Oblix *NetPoint* and Oracle COREid.

This Preface covers the following topics:

- [Audience](#)
- [Documentation Accessibility](#)
- [Related Documents](#)
- [Conventions](#)

Audience

This guide is intended for administrators who are responsible for installing any Oracle Access Manager component.

This guide assumes that you are familiar with the following concepts:

- Operating and file systems (Windows or Unix-based)
- Sites connected to the Internet and networking protocols
- Network security: building firewalls, deploying authentication systems, and so on.
- Host security: passwords, uids, file permissions, file system integrity, and so on.
- Network security: building firewalls, deploying authentication systems, and so on.
- Web server, Web browser, and configuration details
- Database administration and your LDAP directory

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to

facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at

<http://www.oracle.com/accessibility/>

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

Related Documents

For more information, see the following documents in the Oracle Access Manager Release 10g (10.1.4.0.1) documentation set:

- *Oracle Access Manager Introduction*—Provides an introduction to Oracle Access Manager, a road map to Oracle Access Manager manuals, and a glossary of terms.
- *Oracle Application Server Release Notes*—Read these for the latest Oracle Access Manager updates. The release notes are available with the platform-specific documentation. The most current version of the release notes is available on Oracle Technology Network at:
<http://www.oracle.com/technology/documentation>.
- *Oracle Access Manager Installation Guide*—Explains how to install and configure the components.
- *Oracle Access Manager Upgrade Guide*—Explains how to upgrade earlier versions to the latest version.
- *Oracle Access Manager Identity and Common Administration Guide*—Explains how to configure Identity System applications to display information about users, groups, and organizations; how to assign permissions to users to view and modify the data that is displayed in the Identity System applications; and how to configure workflows that link together Identity application functions, for example, adding basic information about a user, providing additional information about the user, and approving the new user entry, into a chain of automatically performed steps. This book also describes administration functions that are common to the Identity and Access Systems, for example, directory profile configuration, password policy configuration, logging, and auditing.
- *Oracle Access Manager Access Administration Guide*—Describes how to protect resources by defining policy domains, authentication schemes, and authorization

schemes; how to allow users to access multiple resources with a single login by configuring single- and multi-domain single sign-on; and how to design custom login forms. This book also describes how to set up and administer the Access System.

- *Oracle Access Manager Deployment Guide*—Provides information for people who plan and manage the environment in which Oracle Access Manager runs. This guide covers capacity planning, system tuning, failover, load balancing, caching, and migration planning.
- *Oracle Access Manager Customization Guide*—Explains how to change the appearance of Oracle Access Manager applications and how to control Oracle Access Manager by making changes to operating systems, Web servers, directory servers, directory content, or by connecting CGI files or JavaScripts to Oracle Access Manager screens. This guide also describes the Access Manager API and the authorization and authentication plug-in APIs.
- *Oracle Access Manager Developer Guide*—Explains how to access Identity System functionality programmatically using IdentityXML and WSDL, how to create custom WebGates (known as AccessGates), and how to develop plug-ins. This guide also provides information to be aware of when creating CGI files or JavaScripts for Oracle Access Manager.
- *Oracle Access Manager Integration Guide*—Explains how to set up Oracle Access Manager to run with third-party products such as BEA WebLogic, the Plumtree portal, and IBM WebSphere.
- *Oracle Access Manager Schema Description*—Provides details about the Oracle Access Manager schema.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates run-in headings and information that you should pay close attention to. In some cases, boldface type indicates graphical user interface elements associated with an action.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

What's New in Oracle Access Manager

This section describes new features of the Oracle Access Manager 10g (10.1.4.0.1) and provides pointers to additional information in this book. Information from previous releases is also retained to help those users migrating to the current release.

The following sections describe the new features in Oracle Oracle Access Manager described in this book:

- [Product and Component Name Changes](#)
- [New Features in Oracle Access Manager 10g \(10.1.4.0.1\)](#)
- [Updates to Specific Chapters](#)

Note: For a comprehensive list of new features and functions in Oracle Access Manager 10g (10.1.4.0.1), and a description of where each is documented, see the chapter on What's New in Oracle Access Manager in the *Oracle Access Manager Introduction*.

Product and Component Name Changes

The original product name, Oblix NetPoint, has changed to Oracle Access Manager. Most component names remain the same. However, there are several important changes that you should know about, as shown in the following table:

Item	Was	Is
Product Name	Oblix NetPoint Oracle COREid	Oracle Access Manager
Product Name	Oblix SHAREid NetPoint SAML Services	Oracle Identity Federation
Product Name	OctetString Virtual Directory Engine (VDE)	Oracle Virtual Directory
Product Release	Oracle COREid 7.0.4	Also available as part of Oracle Application Server 10g Release 2 (10.1.2).
Directory Name	COREid Data Anywhere	Data Anywhere
Component Name	COREid Server	Identity Server
Component Name	Access Manager	Policy Manager
Console Name	COREid System Console	Identity System Console

Item	Was	Is
Identity System Transport Security Protocol	NetPoint Identity Protocol	Oracle Identity Protocol
Access System Transport Protocol	NetPoint Access Protocol	Oracle Access Protocol
Administrator	NetPoint Administrator COREid Administrator	Master Administrator
Directory Tree	Oblix tree	Configuration tree
Data	Oblix data	Configuration data
Software Developer Kit	Access Server SDK ASDK	Access Manager SDK
API	Access Server API Access API	Access Manager API
API	Access Management API Access Manager API	Policy Manager API
Default Policy Domains	NetPoint Identity Domain COREid Identity Domain	Identity Domain
Default Policy Domains	NetPoint Access Manager COREid Access Manager	Access Domain
Default Authentication Schemes	NetPoint None Authentication COREid None Authentication	Anonymous
Default Authentication Schemes	NetPoint Basic Over LDAP COREid Basic Over LDAP	Oracle Access and Identity Basic Over LDAP
Default Authentication Schemes	NetPoint Basic Over LDAP for AD Forest COREid Basic Over LDAP for AD Forest	Oracle Access and Identity for AD Forest Basic Over LDAP
Access System Service	AM Service State	Policy Manager API Support Mode

All legacy references in the product or documentation should be understood to connote the new names.

New Features in Oracle Access Manager 10g (10.1.4.0.1)

The features covered in this manual include:

- Globalization

This manual focuses on installing Oracle Access Manager 10g (10.1.4.0.1) and includes information needed to install on machines with non-English (AMERICAN) operating systems and as well as details about installing Oracle-provided Language Packs

See Also: [Chapter 3, "About Multi-Language Environments"](#)

Prerequisites in installation chapters for each component

- Oracle HTTP Server (OHS) support is provided with this release for WebPass, Access Manager, and WebGate components

See Also: Chapter 16, "Configuring the Apache v1.3 and Oracle HTTP Server Web Servers"

- Oracle Internet Directory Support is included in this release for general use.

See Also: Chapter 2, "Preparing for Installation", Chapter 4, "Installing the Identity Server", and Chapter 15, "Replicating Components"

Updates to Specific Chapters

General product and naming changes have been made throughout this book, as described in "Product and Component Name Changes" on page -xxi. Platform support details have been removed from this book and can now be located under the Certify tab at <https://metalink.oracle.com>. Other updates and changes to specific chapters include the following:

- Chapter 1, "About the Installation Task, Options, and Methods" has been streamlined.
- Chapter 2, "Preparing for Installation" includes new component installation considerations. Installation considerations that formerly resided in component installation chapters have been consolidated in this preparation chapter to eliminate redundancy and group related details together. Multi-language environment details have moved to a separate chapter.
- Chapter 3, "About Multi-Language Environments" contains new information about preparing for installation in multi-language environments as well as updated details about installing Oracle-provided Language Packs.
- Chapter 4, "Installing the Identity Server", has been updated and installation considerations moved to Chapter 2, "Preparing for Installation"
- Chapter 5, "Installing WebPass" installation considerations moved to Chapter 2, "Preparing for Installation"
- Chapter 7, "Installing the Policy Manager" has been updated and installation considerations moved to Chapter 2, "Preparing for Installation"
- Chapter 8, "Installing the Access Server" has been updated and installation considerations moved to Chapter 2, "Preparing for Installation"
- Chapter 9, "Installing the WebGate" has been updated and installation considerations moved to Chapter 2, "Preparing for Installation"
- Chapter 10, "Setting Up Oracle Access Manager with Oracle Virtual Directory" following the acquisition of OctetString by Oracle, this chapter moved from the *Oracle Access Manager Integration Guide* and includes minor changes for clarification, new information to describe graphics, and an updated table for the DN Conversion tool.
- Chapter 13, "About Installing Audit-to-Database Components" provides an introduction to this feature. Complete details are in the *Oracle Access Manager Identity and Common Administration Guide*.
- Chapter 14, "About the Software Developer Kit" has been added to provide a brief introduction to the independent installation of the Software Developer Kit (SDK). Complete details are provided in the *Oracle Access Manager Developer Guide*.

- [Chapter 15, "Replicating Components"](#) has been updated to include new syntax and commands.
- [Chapter 16, "Configuring the Apache v1.3 and Oracle HTTP Server Web Servers"](#) has been updated to include details about OHS and new information about WebGate performance.
- [Chapter 17, "Configuring Apache v2, IHS, and OHS Web Servers for Oracle Access Manager"](#) has been updated to include information about OHS and new information about Apache-based Web servers.
- [Chapter 19, "Important Notes"](#) has been added to provide details that were previously included in a file called importantnotes.txt.
- [Chapter 20, "Removing Oracle Access Manager"](#) is a new chapter that provides details about uninstalling components, including Language Packs, as well as removing schema objects and Web server configuration details
- [Chapter B, "Installing Oracle Access Manager with ADAM"](#) has been updated to reflect the requirement for a manual schema update.
- [Chapter E, "Troubleshooting Installation Issues"](#) has been updated with new information and consolidated into a single appendix.

Part I

Installation Planning and Prerequisites

This part introduces Oracle Access Manager installation concepts, requirements, and prerequisites.

Part I contains the following chapters:

- [Chapter 1, "About the Installation Task, Options, and Methods"](#)
- [Chapter 2, "Preparing for Installation"](#)
- [Chapter 3, "About Multi-Language Environments"](#)

About the Installation Task, Options, and Methods

This chapter provides an introduction to installing Oracle Access Manager 10g (10.1.4.0.1). Topics include:

- [About the Installation Task](#)
- [Installation Options](#)
- [Installation Methods](#)

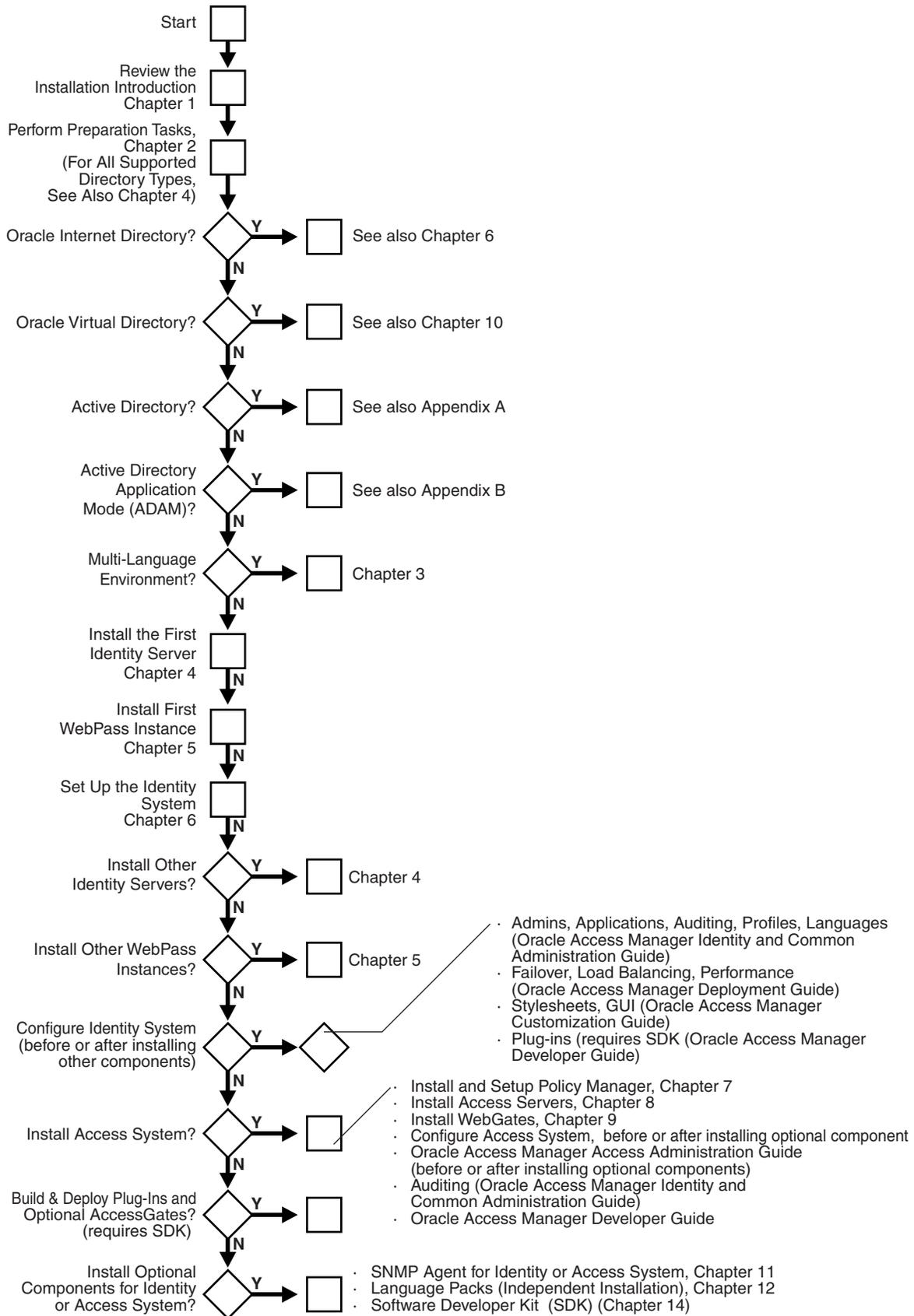
Before starting activities in this guide, be sure to read the *Oracle Access Manager Introduction*. Upgrading an older installation to 10g (10.1.4.0.1) is described in the *Oracle Access Manager Upgrade Guide*.

About the Installation Task

The Identity System is required in all installations. The Access System is optional. For an overview of both the Identity System and the Access System, including a look at a simple installation and an overview of how each system operates, see the *Oracle Access Manager Introduction*.

The sequence of tasks you must complete to install and set up Oracle Access Manager components is outlined in [Figure 1-1](#) and the expanded task overview that follows it.

Figure 1-1 *Installation Task Overview*



Task overview: Installing Oracle Access Manager

1. Review and choose your installation options, as described in ["Installation Options"](#) on page 1-5, and your methods as described in ["Installation Methods"](#) on page 1-8.
2. Complete all prerequisites in [Chapter 2, "Preparing for Installation"](#) and review the following information as needed for your environment.
 - If you are using Oracle Internet Directory in this installation, see also:
 - ["Tuning Oracle Internet Directory"](#) on page 4-13
 - ["Task overview: Ensuring full interaction between Oracle Access Manager and Oracle Internet Directory"](#) on page 6-3
 - If you are using Oracle Virtual Directory in this installation, see also [Chapter 10, "Setting Up Oracle Access Manager with Oracle Virtual Directory"](#) and complete all prerequisite tasks before you setup the Identity System.
 - If you are using Active Directory in this installation, see also [Appendix A, "Installing Oracle Access Manager with Active Directory"](#).
 - If you are including Active Directory Application Mode (ADAM) in this installation, see also [Appendix B, "Installing Oracle Access Manager with ADAM"](#).
3. If you have a multi-language environment, review information on this in [Chapter 3, "About Multi-Language Environments"](#).
4. Install the first Identity Server, as described in [Chapter 4, "Installing the Identity Server"](#).
5. Install the first WebPass, as described in [Chapter 5, "Installing WebPass"](#).
6. Set up the Identity System to ensure that object classes and attributes appear in the directory server and that the Identity Server is working correctly with the WebPass, and assign a Master Administrator who has access to the entire system, as described in [Chapter 6, "Setting Up the Identity System"](#).
7. Install other Identity Servers if needed in this environment, as described in [Chapter 4, "Installing the Identity Server"](#).
8. Install other WebPass instances if needed in this environment, as described in [Chapter 5, "Installing WebPass"](#).

Note: If you are installing multiple instances of any component, you can do this automatically after the first instance is installed and set up. See [Chapter 15, "Replicating Components"](#) for information about automated installation, cloning, and synchronizing components.

9. Start configuring and customizing your Identity System now (or after installing optional components). For example:
 - Define administrators; configure workflows, auditing, and profiles; use applications (User Manager, Group Manager, Organization Manager), configure the system to use installed languages, as described in the *Oracle Access Manager Identity and Common Administration Guide*.
 - Configure failover, load balancing, caching; performance tune the Identity System; and take a look at migration planning for a production environment as described in the *Oracle Access Manager Deployment Guide*.

- Start customizing the Identity System to change the appearance of applications and to control Oracle Access Manager by making changes to operating systems, Web servers, directory servers, directory content, or by connecting CGI files or JavaScripts to Oracle Access Manager screens, as described in the *Oracle Access Manager Customization Guide*.
- Explore how to build and deploy Identity Event Plug-ins using the software developer kit (SDK) and APIs, and how to access Identity System functionality programmatically using IdentityXML and WSDL, as described in the *Oracle Access Manager Developer Guide*.

Note: Installing the Oracle Access Manager Software Developer Kit and APIs are introduced in [Chapter 14, "About the Software Developer Kit"](#). Complete details are located in the *Oracle Access Manager Developer Guide*.

10. Install and set up the optional Access System, as follows:
 - Install and setup the Policy Manager, as described in [Chapter 7, "Installing the Policy Manager"](#).
 - Install the Access Server, which includes adding an Access Server instance in the Access System Console, as described in [Chapter 8, "Installing the Access Server"](#).
 - Install the WebGate, which includes adding a WebGate instance in the Access System Console and associating the WebGate with an Access Server before installation, as described in [Chapter 9, "Installing the WebGate"](#).
11. Start configuring the Access System now (or install other optional components first), as follows:
 - Define policy domains, authentication schemes, and authorization schemes; allow users to access multiple resources with a single login by configuring single- and multi-domain single sign-on; and design custom login forms as described in the *Oracle Access Manager Access Administration Guide*.
 - Configure the Access System for auditing, as described in the *Oracle Access Manager Identity and Common Administration Guide*.
 - Create custom WebGates (known as AccessGates), and develop custom authentication and authorization plug-ins using the software developer kit and APIs, as described in the *Oracle Access Manager Developer Guide*.

Note: Installing the Oracle Access Manager Software Developer Kit and APIs are introduced in [Chapter 14, "About the Software Developer Kit"](#). Complete details are located in the *Oracle Access Manager Developer Guide*.

12. Install any other optional Oracle Access Manager components you'd like to use, such as:
 - SNMP monitoring, as discussed in [Chapter 11, "Installing the SNMP Agent"](#)
 - Oracle-provided Language Packs, which may be installed independently, after component installation, as described in [Chapter 12, "Installing Language Packs Independently"](#)

- The Oracle Access Manager Software Developer Kit and APIs are introduced in [Chapter 14, "About the Software Developer Kit"](#). Complete details are located in the *Oracle Access Manager Developer Guide*.

Installation Options

This discussion identifies the options available to you during installation, and tells you where to find more information.

Task overview: Choosing your installation options

1. Before installation, decide whether to install components using GUI method or the command line method, as described in ["Installation Methods"](#) on page 1-8.
2. During installation you can choose to enable automatic updates of the schema using system-provided defaults, or input your own values for attributes during Identity System and Policy Manager setup, as described in ["Updating the Schema and Attributes Automatically Versus Manually"](#) on page 1-5.
3. After installation of the first instance of a component, you can choose to install multiple instances of a component manually or use an automated installation method for multiple instances, as described in ["Replicating an Installed Oracle Access Manager Component"](#) on page 1-7.
4. If you have older component files in the installation directory that you specify, you are asked if you want to upgrade to the later release. See ["Upgrading from a Earlier Release of Oracle Access Manager"](#) on page 1-8.

Updating the Schema and Attributes Automatically Versus Manually

During Identity Server and Policy Manager installation, you are asked if you want to automatically update the schema with the configuration data branch. The schema update must occur before you begin the setup process.

Note: Oracle recommends that you update the schema automatically during installation to obtain product-specific object classes and attributes. If you decline the automatic update during installation, a Schema Changes page appears at the beginning of the Identity System and Policy Manager setup process. The automatic schema update is *not* supported for the ADAM directory.

Custom schema changes must be added after the installation because the Identity Server installation changes the schema. During Identity System and Policy Manager setup, you are prompted to configure various object classes. For example, the Identity System requires attributes assigned to the Full Name, Login, and Password semantic types for Person and Group object classes. Oracle recommends that you automatically configure attributes using the Auto Configure option during setup to save time and avoid errors. You can reconfigure the attributes afterward if needed.

Automatically configuring attributes is a single step in the installation and setup processes, as shown in [Table 1-1](#). With the ADAM directory, however, you must manually update the schema and data after Oracle Access Manager component installation, as described in [Appendix B, "Installing Oracle Access Manager with ADAM"](#).

Table 1–1 Automatically Configure the Schema for All Except the ADAM Directory

Component	Automatic Schema Configuration for All Except ADAM
Identity Server installation	During the first Identity Server installation, select "Yes" to automatically update the schema. For second and subsequent Identity Servers, select No.
WebPass installation	There are no options for the schema.
Identity System set up	Select "Auto Configure" when the option is offered. After setup, you may reconfigure attributes, if needed.
Policy Manager installation and set up	Select "Auto Configure" when the option is offered. After setup, you may reconfigure attributes, if needed.
Access Server installation	There are no options for the schema update.
WebGate installation	There are no options for the schema.

If you choose to manually configure attributes, this must occur after installation during the setup process. Manually configuring attributes requires one or more ldif files located in:

IdentityServer_install_dir\identity\oblix\data.ldap\common

PolicyManager_install_dir\access\oblix\data.ldap\common

Each ldif file is prefixed with a specific directory server type, as shown in [Table 1–2](#). In most cases, you use the ldapmodify tool to perform the update. For example:

```
ldapmodify -h DS_hostname -p DS_port_number -D bind_dn -w password -a -c -f DS_type_oblix_schema_add.ldif
```

[Table 1–2](#) provides details about the schema update files needed for each directory server type. Included are any index files required for configuration data or user data.

For more information about directory requirements, see "[Meeting Directory Server Requirements](#)" on page 2-17.

Table 1–2 Manual Schema Update Files

Directory Server Type	Manual Schema Update Files
Active Directory	ADSchema.ldif (<i>Windows 2000 only</i>) ADdotNetSchema_add.ldif (<i>Windows 2003 only</i>) ADAuxSchema.ldif (<i>Windows 2003, statically-linked auxiliary classes</i>) ADUserSchema.ldif Note: The Active Directory schema is extensible using Ldifde.exe. For more information, see Appendix A, "Installing Oracle Access Manager with Active Directory" .

Table 1–2 (Cont.) Manual Schema Update Files

Directory Server Type	Manual Schema Update Files
ADAM	ADAM_oblix_schema_add.ldif ADAM_user_schema_add.ldif ADAMAuxSchema.ldif (statically-linked auxiliary classes) Note: You must manually update the ADAM schema when installing Oracle Access Manager. The ADAM schema is extensible using Ldifde.exe. For more information, see Appendix B, "Installing Oracle Access Manager with ADAM" .
Data Anywhere (Oracle Virtual Directory)	VDE_user_schema_add.ldif See Chapter 10, "Setting Up Oracle Access Manager with Oracle Virtual Directory" on page 10-1 for details about: <ul style="list-style-type: none"> ▪ Integrating Oracle Access Manager with Oracle Virtual Directory Server (VDS) ▪ Prerequisites and Oracle Access Manager installation with VDS ▪ schema.oblix.xml ▪ Adapter and mapping script templates ▪ DN conversion program and configuration file to patch user and group DNs in the configuration tree for use with VDS in existing Oracle Access Manager installations
IBM Directory Server	V3.oblix.ibm_at.ldif V3.oblix.ibm_oc.ldif V3.user.ibm_at.ldif V3.user.ibm_oc.ldif
Oracle Internet Directory	OID_oblix_schema_add.ldif OID_oblix_schema_delete.ldif OID_oblix_schema_index_add.ldif OID_user_index_add.ldif OID_user_schema_add.ldif OID_user_schema_delete.ldif
Novell Directory Server	NDS_oblix_index_add.ldif NDS_oblix_schema_add.ldif NDS_user_index_add.ldif NDS_user_schema_add.ldif
Sun Directory Servers	iPlanet_oblix_schema_add.ldif. iPlanet_user_schema_add.ldif iPlanet5_oblix_index_add.ldif iPlanet5_user_index_add.ldif

Replicating an Installed Oracle Access Manager Component

Rather than manually installing every instance of a component, you can replicate the configuration of one instance to another after installation and setup of the first instance of a particular component.

There are three methods to choose from:

- Automate the installation process using a file that contains installation parameters (known as installing in silent mode).
- Clone the configuration.
- Synchronize two components or parts of two components.

Silent Mode

Silent mode permits installation without user intervention. The Oracle Access Manager installation script takes option and configuration information from a silent mode option file.

Important: Silent mode is intended for new installations only.

For more information on silent mode, see [Chapter 15, "Replicating Components"](#).

Cloning and Synchronizing Installed Components

You can also replicate an installed component by *cloning* it, or you can *synchronize* two components or parts of two components.

For more information, see "[Cloning and Synchronizing Installed Components](#)" on page 15-29.

Upgrading from a Earlier Release of Oracle Access Manager

When you start installing a Oracle Access Manager component and specify a target installation directory that contains an earlier version, the component is detected and you are asked if you want to upgrade that component to 10g (10.1.4.0.1).

- To avoid an upgrade, you must specify a new installation directory path.
- To accept the upgrade and continue, see the *Oracle Access Manager Upgrade Guide*.

Installation Methods

You may choose to install Oracle Access Manager components using the graphical user interface (GUI method) or using the command-line console (Console method). Regardless of the method you choose, the process is similar. The sequence and prompts detailed in this manual use GUI method. Any differences will be identified as they occur. For more information, see:

- [GUI Method](#)
- [Console Method](#)

GUI Method

Different installation packages are available for Oracle Access Manager components, depending on your platform and Web server. The sequence of events and messages are the same regardless of the method you choose when launching the installation.

You obtain the Oracle Access Manager installation media from Oracle. GUI method is the default for Windows systems when you select the installation package. For example:

```
Oracle_Access_Manager10_1_4_0_1_win32_Identity_Server
```

Due to known problems with the third-party Installshield's ISMP framework, if any inputs supplied during installation contain the character \$, the installer might interpret it unpredictably. For example, if the bind password supplied during the schema update for the first Identity Server is Admin\$\$, ISMP interprets this as Admin\$ while invoking the schema update tool and the update fails citing a "bad credentials error(49)". If this problem is observed during invocation of a particular tool, you may run that tool from the command line.

Note: Every Oracle Access Manager installer that uses the same password may also fail with a credential problem of some type.

Console Method

You may use the command-line console method when installing Oracle Access Manager components on Unix platforms. Console method is the default for Unix systems. For example:

```
/ Oracle_Access_Manager10_1_4_0_1_sparc-s2_Identity_Server
```

Note: When using the console method for component installation, you are instructed to:

Press 1 for Next—1 is the default if you press the Enter key.

Press 3 to Cancel

Press 4 to Re-display the information

Occasionally, you will be asked to specify an option number then enter zero, 0, to confirm your choice.

Preparing for Installation

This chapter provides important information you need to prepare your environment before starting the installation process for Oracle Access Manager components. Topics include:

Note: Failure to complete all prerequisites may adversely affect your 10g (10.1.4.0.1) installation.

- [About Installation Prerequisites](#)
- [Synchronizing System Clocks](#)
- [Meeting Oracle Access Manager Requirements](#)
- [Meeting Web Server Requirements](#)
- [Meeting Directory Server Requirements](#)
- [Confirming Platform Requirements](#)
- [Preparing a Temporary Directory for Installers](#)
- [Uninstalling Oracle Access Manager Components](#)
- [Installation Preparation Checklists](#)

For an overview of Oracle Access Manager components, features, functions, audiences, and manuals, see the *Oracle Access Manager Introduction*. Upgrading to 10g (10.1.4.0.1) is described in the *Oracle Access Manager Upgrade Guide*.

About Installation Prerequisites

You can help ensure a successful installation by completing the following prerequisites before you install Oracle Access Manager.

Task overview: Preparing to install Oracle Access Manager

1. Review [Chapter 1, "About the Installation Task, Options, and Methods"](#) and decide which installation options are best for your environment.
2. Follow a few best practices for security, as described in "[Setting Secure Passwords and Permissions](#)" on page 2-2.
3. Synchronize the host clocks if you are installing across multiple machines, as described in "[Synchronizing System Clocks](#)" on page 2-2.
4. Review all "[Meeting Oracle Access Manager Requirements](#)" on page 2-4 and complete activities.

5. Create a Web server instance and refer to ["Meeting Web Server Requirements"](#) on page 2-14.
6. Create a supported directory server instance, define at least one administrator-level user on your directory server (see your vendor documentation), and review all topics in ["Meeting Directory Server Requirements"](#) on page 2-17.
7. Ensure that your environment meets the platform and support requirements, as described in ["Confirming Platform Requirements"](#) on page 2-26.
8. Obtain the software from the Oracle-provided installation media and prepare a temporary directory as described in ["Preparing a Temporary Directory for Installers"](#) on page 2-27.
9. Collect and document information about your environment to provide during the installation process, as described in ["Installation Preparation Checklists"](#) on page 2-28.
10. If you are installing with an Oracle-provided Language Pack or on a machine running a non-English (American) language or territory operating system, see [Chapter 3, "About Multi-Language Environments"](#) and complete any activities needed.

Setting Secure Passwords and Permissions

The following are best practices for a secure installation:

- On all platforms, create a temporary LDAP administrator password during installation and reset it after installing all components.
- On Unix, if your installation computer allows access to untrusted users or groups, perform the installation with a user and group that are accessibly only to trusted user accounts.

Create a new user and group if necessary.
- On Windows, if your installation computer allows access to untrusted users, configure all file security permissions to only allow access to trusted users.

Synchronizing System Clocks

If you plan to install Oracle Access Manager components across multiple machines, make sure all system clocks are synchronized. This is particularly important if you will be running the software in Cert or Simple mode.

WARNING: Each secure request includes a timestamp. Differences in system clocks could cause all requests to the Identity Server to be rejected.

For example, if the WebPass Web server system clock is set ahead of the Identity Server system clock, a login request sent from the WebPass plug-in on the Web server will contain a time that, to the Identity Server, has not yet occurred. The same is true for the Access System. If a Web server clock is ahead of the Access Server clock, a request sent from the Policy Manager to the Access Server will contain a time that, to the Access Server, has not yet occurred.

For successful operation:

- Ensure all machine clocks are synchronized—There is no tolerance level. If, for example, the WebGate clock is even slightly ahead of the Access Server clock, a cookie generated by the WebGate will appear to be in the future and will cause problems in the Access Server.
- Confirm that the clock on each machine running a WebGate is *not* running ahead of the Access Servers with which it is associated. The Access Server must be ahead of the WebGate clock by a maximum of 60 seconds.
- Confirm that the clock on each machine running the WebPass is not running ahead of the Identity Servers and Policy Managers with which it is associated.

About the Network Time Protocol

To synchronize 10g (10.1.4.0.1) components across geographically diverse time zones, you can use the Network Time Protocol (NTP). NTP can synchronize the time on machines to within a few milliseconds. For more information about time synchronization, go to the Web site:

<http://www.ntp.org/>

and the comp.protocols.time.ntp news group.

An ntp.conf file at minimum would contain the following:

```
server <some NTP server name>.com
driftfile /etc/ntp.drift
```

Instructions for creating the ntp.conf file can be found at the following locations:

- <http://www.sun.com/products-n-solutions/hardware/docs/html/816-3626-10/after.html>
- http://inetsd01.boulder.ibm.com/pseries/fr_FR/files/aixfiles/ntp.conf.htm
- <http://www.developer.ibm.com/tech/faq/individual/0,,2:14789,00.html>

Unix machines use UTC (also known as GMT) internally and convert to the local time that is needed on the display. Windows machines keep the clock in local time, but NTP synchronization programs compensate to ensure accurate times on Windows.

On Unix Systems

All Unix operating systems ship with a version of NTP. To configure NTP on Solaris, create an ntp.conf file. The name of the ntp.conf file to use the Solaris provided NTP daemon is /etc/inet/ntp.conf. Once this is created, xntp is started automatically when the operating system starts.

- **On HP-UX:** Use sam to start NTP.
- **On AIX:** Create an /etc/ntp.conf file and enable or create a start script.
- **For all Unix platforms:** Get the current (and more secure) version of the NTP daemon from <http://www.ntp.org/>.

On Windows Systems

Windows machines synchronize their times automatically with their domain controller using a version of NTP. The domain controller needs to be configured to synchronize with a time source.

To obtain an official time for synchronization across your network many ISPs provide a time service for their customers.

- NTP, which has a list of open stratum-1 servers available at <http://www.ntp.org>.

However, that this site may not be the most secure choice. For an example of a time-based attack, imagine unexpiring a cookie by spoofing the time to be earlier than the real time.

- GPS-based clocks, which use satellite technology to provide very accurate time, are available.

These clocks can be used to set your whole network to the same time. GPS technology requires very accurate times; each satellite contains 3 atomic clocks with continuing corrections provided from the ground that compensate for relativistic effects. This means that an accurate estimate of the current time is developed as a side effect of figuring out where the GPS receiver is.

Meeting Oracle Access Manager Requirements

The following information is provided for your convenience.

- [General Guidelines](#)
- [Preparing Linux Host Machines](#)
- [Identity System Guidelines](#)
- [Access System Guidelines](#)
- [Assessing Disk Space Requirements](#)
- [Choosing an Installation Directory](#)

General Guidelines

You need a supported host machine for each component, which you can find as described in "[Confirming Platform Requirements](#)" on page 2-26. In addition:

- **Administrative Rights:** The account that performs component installation must have administration privileges. Both the Identity Server and Access Server run as services. The user account that is used to run the Identity Server and Access Server services must have the "Log on as a service" right, which can be set through Administrative Tools, Local Security Policy, Local Policies, User Rights Assignments, Log on as a service.
 - **On Microsoft Windows**—The user account that is used to run the Identity Server service must have the right to "Log on as a service". This can be set through Administrative Tools. For example:
 - Administrative Tools, Local Security Policy, Local Policies
 - User Rights Assignments, Log on as a service
 - **On Unix Platforms**—You will be asked to specify the username and group that the Identity Server will use: typically, the defaults are "nobody"; for HP-UX, the defaults are WWW (username) and others (group). Confirm that the right commands are installed and verify the user name under which your Web server runs. For example:
 - a. Locate the following commands (usually found in /usr/bin, /usr/sbin, or /usr/sbin) and make sure their location is included in the search path:

```
sed, tar, cp, ls, mkdir, rmdir
```

- b. The user name under which your Web server runs could be nobody, root, or some other user name, such as Web. You can determine this by checking your Web server configuration files or by running your Web server's administration console and looking under View Server Settings.
- **Machines Online:** Before installation, you must be able to ping the machine on which each component will run. Also, during installation you will be asked to supply the DNS host name of the machine on which the Identity Server and Access Server are installed.
- **Linux Libraries:** Before installing components on Linux machines, you need to install additional GCC runtime libraries (libgcc_s.so.1 and libstdc++.so.5) that are compatible with GCC 3.3.2. See ["Preparing Linux Host Machines"](#) on page 2-5.
- **Component Security:** During installation, you must specify the transport security mode for communication between Oracle Access Manager components. See ["Securing Oracle Access Manager Component Communications"](#) on page 2-11.
- **Directory Security:** During installation (Identity Server, Policy Manager, and Access Server), you must specify the hostname, DN, and transport security mode for the directory server with which the components will communicate. See ["Meeting Directory Server Requirements"](#) on page 2-17 for this and other important information.
- **Existing Identity Server Name:** If you want to reuse an existing Identity Server name, see ["Recycling an Identity Server Instance Name"](#) on page 20-4.
- **Cancel Installation:** If you need to cancel an installation or remove an installed component, see ["Uninstalling Oracle Access Manager Components"](#) on page 2-28.
- **Multi-Language Environments:**
 - If you are installing on a machine with an operating system that is non-English (AMERICAN) language or locale, you may set the LANG environment variable or the *optional* NLS_LANG or COREID_NLS_LANG environment variables.
 - If you install an Identity Server with one or more Oracle-supplied Language Packs you *must* install WebPass with the same Language Packs (and install corresponding Access System Language Packs with all Access System components).
 - If you are installing components with a Language Pack on a Unix system, you must ensure that the Language Pack installer resides in the same directory as the component and that the Language Pack installer has execute permissions before launching the main installer. For example:

```
chmod +x "Oracle_Access_Manager10_1_4_0_1_FR_sparc-s2_LP_Identity_System"
chmod +x "Oracle_Access_Manager10_1_4_0_1_FR_sparc-s2_LP_Access_System"
```

For more information, see [Chapter 3, "About Multi-Language Environments"](#).

Preparing Linux Host Machines

During installation of Oracle Access Manager components on a Linux machine, you are asked to specify the location of additional GCC runtime libraries (libgcc_s.so.1 and libstdc++.so.5) that should be compatible with GCC 3.3.2. Oracle does not ship these nor make these available for download from an Oracle Web site. If you need to obtain these libraries, contact your platform vendor.

To install libgcc_s.so.1 and libstdc++.so.5 on Linux hosts

1. Obtain the libgcc_s.so.1 and libstdc++.so.5 libraries from your platform vendor.
2. Store the following files on the local Linux machine on which you will install one or more Oracle Access Manager components:

libgcc_s.so.1
libstdc++.so.5
3. During Oracle Access Manager installation, specify the location of the libraries on the local machine and continue the installation.

Identity System Guidelines

The Identity Server does not need to be on the same host system as any other Oracle Access Manager component or application. Oracle recommends you install the Identity Server and Access Server on different machines. Further, the Identity Server does not need to be on the same host system as any other Oracle Access Manager component or application. For details about installing one or more Identity Servers, see [Chapter 4, "Installing the Identity Server"](#).

Each Web server instance that communicates with the Identity Server must be configured with a WebPass. One WebPass can communicate with multiple Identity Servers. More than one WebPass can communicate with the same Identity Server, which is recommended for load balancing. See also "[Synchronizing System Clocks](#)" on page 2-2.

The WebPass instance identifier that you specify during installation must be unique. The WebPass instance identifier is not validated until after the installation, when the Web server is started.

A WebPass must also be installed with each Policy Manager on the same Web server instance, at the same directory level.

For details about installing one or more WebPass instances, see [Chapter 5, "Installing WebPass"](#).

During Identity System setup, you need to define a user who will be granted access to all Oracle Access Manager functionality. This is the Master Administrator. For more information, see [Chapter 6, "Setting Up the Identity System"](#).

Access System Guidelines

The following discussions outline Access System requirements and guidelines:

- [Policy Manager Guidelines](#)
- [Access Server Guidelines](#)
- [WebGate Guidelines](#)

Policy Manager Guidelines

The Policy Manager must be installed on the same Web server instance as a WebPass, at the same directory level as a WebPass. See also "[Synchronizing System Clocks](#)" on page 2-2.

Oracle recommends that you do not put a firewall between the Policy Manager and the directory server because no "health check" is performed. After a period of inactivity, the firewall may drop the Policy Manager connection without warning. To avoid such problems, either ensure the Policy Manager and directory server are on the

same side of the fire wall or disable the firewall connection timeout between the Policy Manager and directory server, if possible. However, not all firewalls support this.

The NETWORK account must have Modify rights at the volume root.

Depending on the directory server you use with the Policy Manager, consider the following:

- If you specify Active Directory on Windows Server 2003 as the directory server during Policy Manager installation, a new page appears asking if dynamic auxiliary classes are to be supported. If you are using ADSI, you need to set the IIS Web server Anonymous User Login Account to a Domain User after installation and before setting up the Policy Manager.
- Oracle Access Manager does not support SSL-enabled communication between the directory server and Policy Manager when the Policy Manager is installed on Solaris with a Sun (formerly Netscape) Web server.

There are several considerations depending upon the Web server you are using for your Policy Manager installation:

- **Apache:** Oracle Access Manager supports Apache with or without SSL enabled. For SSL-enabled communication, Oracle Access Manager supports Apache with mod_ssl only, not Apache-SSL. mod_ssl is a derivative of, and alternative to, Apache-SSL. configure in httpd.conf the user and group you want the Web Server to run as.
- **IIS:** The Policy Manager installer cannot update multiple Web servers instances. If you have multiple IIS Web server instances installed, be sure to install a separate Policy Manager on each Web server instance.

When installing the Policy Manager on Windows 2000 with IIS, ensure that the group named Everyone has full access to the \temp directory and the drive (for example, C or D) to which the \temp directory belongs.

The TEMP variable needs to be set to point to a valid directory, either for the entire system or for the IIS user. Oracle recommends setting the TEMP variable for the entire system.

For details about installing one or more Policy Managers, see [Chapter 7, "Installing the Policy Manager"](#).

Access Server Guidelines

Oracle recommends you install the Identity Server and Access Server on different machines. The Identity Server does not need to be on the same host system as any other Oracle Access Manager component or application.

Do not install the Access Server in the same directory as the Policy Manager. Do not install multiple Access Servers in the same directory.

Failover and Load Balancing: Oracle recommends installing multiple Access Servers for failover and load balancing.

Firewall: Oracle recommends protecting the machine on which you will install the Access Server with a firewall.

If you install a 10g (10.1.4.0.1) Access Server in an upgraded environment that includes older WebGates, you must manually change "IsBackwardCompatible" Value="true" in the Access Server's globalparams.xml file. Upgraded Access Servers are backward compatible with older WebGates automatically. See the *Oracle Access Manager Upgrade Guide* for complete details.

For details about installing one or more Access Servers, see [Chapter 8, "Installing the Access Server"](#).

WebGate Guidelines

The WebGate *must* be installed on a machine hosting a Web server. You can install the WebGate in any directory that your Web server can access.

You should install a WebGate on any Web server that you want to protect with the Access System, including the Web server on which the Policy Manager is installed. If you install the WebGate to protect an Policy Manager and WebPass, the WebGate must be installed in the same directory as the Policy Manager and WebPass. For example, if the WebPass and Policy Manager are installed in `\COREid\WebComponent`, then the WebGate must also be installed there.

The WebGate may be installed at the root level or the site level. Installing WebGate on multiple virtual sites amounts to only one instance of WebGate. The WebGate can also be installed using a non-root user if the Web server process runs as a non-root user.

The WebGate may be configured to run at either the machine level or the virtual Web server level. However, do not install at both the machine level and the virtual Web server levels. See also ["Synchronizing System Clocks"](#) on page 2-2.

Older WebGates may coexist with 10g (10.1.4.0.1) Access Servers. In this case, you need to:

- Use RC4 as the encryption scheme if you have Release 5.x and 10g (10.1.4.0.1) WebGates co-existing in the same system.
- Use RC6 as the encryption scheme if you have Release 6.x and 10g (10.1.4.0.1) WebGates co-existing in the same system.
- Use the AES encryption scheme if you have only Release 7.0 or 10g (10.1.4.0.1) WebGates co-existing in the same system.

Also, for details about Access Server backward compatibility with older WebGates, see the *Oracle Access Manager Upgrade Guide*. For details about installing one or more WebGates, see [Chapter 9, "Installing the WebGate"](#).

Web server and operating system types are not factors in WebGate-to-Access Server communication. However, there are considerations for WebGates in various environments:

- **Unix WebGates:** You may be logged in as root to install the WebGate. The WebGate can be installed using a non-root user if the Web server process runs as a non-root user.
- **Apache Web Servers:** Oracle Access Manager supports Apache with or without SSL enabled. For SSL-enabled communication, Oracle Access Manager supports Apache with `mod_ssl` only, not Apache-SSL. `mod_ssl` is a derivative of, and alternative to, Apache-SSL.
- **IHS v2 Web Servers:** Oracle Access Manager supports IHS v2 and IHS v2 Reverse Proxy servers with or without SSL enabled. For details, see ["Updating Web Server Configuration for Oracle Access Manager Web Components"](#) on page 16-8.
- **Domino Web Servers:** Before you install the WebGate with a Domino Web server, you must have properly installed and set up the Domino Enterprise Server R5. For more information, see ["Setting Up Lotus Domino Web Servers for WebGates"](#) on page 18-1
- **IIS Web Servers:** Before installing the WebGate, ensure that your IIS Web server is *not* in lockdown mode. Otherwise things will appear to be working until the

server is rebooted and the metabase re-initialized, at which time IIS will disregard activity that occurred after the lockdown.

If you are using client certificate authentication, before enabling client certificates for the WebGate you must enable SSL on the IIS Web server hosting the WebGate.

Setting various permissions for the /access directory is required for IIS WebGates only when you are installing on a filesystem that supports NTFS . For example, suppose you install the ISAPI WebGate in Simple or Cert mode on a Windows 2000 machine running the FAT32 filesystem. The last installation panel provides instructions for manually setting various permissions that cannot be set on the FAT32 filesystem. In this case, these instructions may be ignored.

Each IIS Virtual Web server can have it's own WebGate.dll file installed at the virtual level, or can have one WebGate affecting all sites installed at the site level. Either install the WebGate.dll at the site level to control all virtual hosts or install the WebGate.dll for one or all virtual hosts.

You may also need to install the postgate.dll file at the machine level. The postgate.dll is located in the `\WebGate_install_dir`, as described in "[Installing postgate.dll on IIS Web Servers](#)" on page 9-10. If you perform multiple installations, multiple versions of this file may be created which may cause unusual Oracle Access Manager behavior. In this case, you should verify that only one webgate.dll and one postgate.dll exist.

Note: The postgate.dll is always installed at the site level. If for some reason the WebGate is reinstalled, the postgate.dll is also reinstalled. In this case, ensure that only one copy of the postgate.dll exists at the site level.

To fully remove a WebGate and related filters from IIS, you must do more than simply remove the filters from the list in IIS. IIS retains all of its settings in a metabase file. On Windows 2000 and later, this is an XML file that can be modified by hand. There is also a tool available, MetaEdit, to edit the metabase. MetaEdit looks like Regedit and has a consistency checker and a browser/editor. To fully remove a WebGate from IIS, use MetaEdit to edit the metabase.

- **ISA Proxy Servers**—On the ISA proxy server, all ISAPI filters must be installed within the ISA installation directory. They can be anywhere within the ISA installation directory structure.
 1. Before installing the WebGate on the ISA proxy server:
 - Check for general ISAPI filter with ISA instructions on:
`http://msdn.microsoft.com/library/default.asp?url=/library/en-us/isa/isaisapi_5cq8.asp`
 - Ensure that the internal and external communication layers are configured and working properly.
 2. During installation you will asked if this is an ISA installation; be sure to:
 - Indicate that this is an ISA proxy server installation, when asked.
 - Specify the ISA installation directory path as the WebGate installation path.
 - Use the automatic Web server update feature to update the ISA proxy server during WebGate installation.

3. After WebGate installation, locate the file `configureISA4webgate.bat`, which calls a number of vbscripts and the process to configure the ISA server filters that must be added programmatically.

Assessing Disk Space Requirements

Table 2-1 provides estimates regarding the free disk space needed for each component are provided for your convenience.

Table 2-1 Disk Space Requirements

	Windows	Unix
Identity Server	128 MB	90 MB
WebPass	93 MB	200 MB
Policy Manager	122 MB	130 MB
Access Server	95 MB	200 MB
WebGate	76 MB	150 MB
SNMP Agent	50 MB	75 MB

Choosing an Installation Directory

You may install components in the default directory or in a directory of your choosing. When you change the path name, you may include any characters that are acceptable to your operating system. For example, you may include spaces on Windows systems but not on Unix systems.

Be sure that all file and path names include only English language characters. In file and path names, no international characters are allowed.

In any case, it is a good idea to use consistent naming regardless of platform. For example, you can use an underscore rather than a space in names on Windows platforms. Typically, the default installation directory for Oracle Access Manager is as follows:

Windows Platforms: `\Program Files\COREid\`

Unix Platforms: `/opt/coreid/` (all lowercase)

Depending on the component you are installing, the path will vary slightly as shown in Table 2-2. For example:

- `\identity` is appended automatically to all Identity component path names
- `\access` is appended automatically to all Access component path names
- `\WebComponent` is included automatically (along with either `\identity` or `\access`) in the default path name for WebPass, Policy Manager, and WebGate installations

Table 2-2 Installation Directory Path Names

Component	Installation Directory
Identity Server	Windows: <code>\Program Files\OracleAccessManager\identity</code> Unix: <code>/opt/oracleaccessmanager/identity</code> In This Guide: <code>\IdentityServer_install_dir\identity</code>

Table 2–2 (Cont.) Installation Directory Path Names

Component	Installation Directory
WebPass	Windows: \Program Files\OracleAccessManager\WebComponent\identity Unix: /opt/oracleaccessmanager/WebComponent/identity In This Guide: \WebPass_install_dir\identity
Access Server	Windows: \Program Files\OracleAccessManager\access Unix: /opt/oracleaccessmanager/access In This Guide: \AccessServer_install_dir\access
Policy Manager	Windows: \Program Files\OracleAccessManager\WebComponent\access Unix: /opt/oracleaccessmanager/WebComponent/access In This Guide: \PolicyManager_install_dir\access
WebGate	Windows: \Program Files\OracleAccessManager\WebComponent\access Unix: /opt/oracleaccessmanager/WebComponent/access In This Guide: \WebGate_install_dir\access

In this manual, the installation directory path for each Oracle Access Manager component will be expressed as `\Component_install_dir` followed by any suffix that is automatically appended to this path, as shown in Table 2–2. When the generic form is used, `Component_install_dir`, a generic suffix, `identity | access`, follows: for example, `Component_install_dir/identity | access`.

When launching a Oracle Access Manager installation on a Unix system, you can direct an installation to a directory with sufficient space using the `-is:tempdir` path parameter.

To specify a temporary directory on Unix systems

1. Use the `-is:tempdir` parameter in the following command. For example:

```
./ Oracle_Access_Manager10_1_4_0_1_sparc-s2_Identity_Server
-is:tempdir /export/home/oblix/temp
```

The path must be an absolute path, not a relative path.

2. The path `/export/home/oblix/temp` should be replaced with a file system with sufficient space.

Securing Oracle Access Manager Component Communications

Before installation, you must decide which type of transport security you will use between components. Oracle Access Manager supports three types of transport security for communication that occurs between components:

- Open: Allows unencrypted communication, see "[Open Mode](#)" on page 2-12
- Simple: Supports encryption by Oracle, see "[Simple Mode](#)" on page 2-12
- Cert: Requires a third-party certificate, see "[Cert Mode](#)" on page 2-13

Transport Security Guidelines

The following guidelines should be observed when planning and implementing transport security between Oracle Access Manager components during installation. Specifically:

- Transport security between all Identity System components (Identity Servers and WebPass instances) must match: either all open, all Simple mode, or all Cert.
- Transport security between all Access System components (Policy Managers, Access Servers, and associated WebGates) must match: either all open, all Simple mode, or all Cert.

Caveats

When access cache flushing is enabled on the Identity Server, the Identity Server communicates with the Access Server. In this case, the transport security mode between all five of the following components must be in the same mode.

- Identity Servers and WebPass instances
- Policy Managers, Access Servers, and associated WebGates

Open Mode

Use *Open* mode if transport security is not an issue in your environment. In Open mode, there is no authentication or encryption between the AccessGate and Access Server. The AccessGate does not ask for proof of the Access Server's identity and the Access Server accepts connections from all AccessGates. Similarly, Identity Server does not require proof of identity from WebPass.

Simple Mode

Use Simple mode if you have some security concerns, such as not wanting to transmit passwords as plain text, but you do not manage your own Certificate Authority (CA).

In Simple mode communications between Web clients (WebPass and Identity Server, Policy Manager and WebPass, and Access Server and WebGate) are encrypted using TLS v1. In both Simple and Cert mode, Oracle Access Manager components use X.509 digital certificates only. This includes Cert Authentication between WebGates and the Access Server where the standard cert-decode plug-in decodes the certificate and passes certificate information to the standard credential_mapping authentication plug-in.

Oracle Access Manager ships a CA with its own private key that is installed across all AccessGates and Access Server components. Oracle Access Manager does an additional password check to prevent other customers from using the same CA.

For each public key there is a corresponding private key that Oracle Access Manager stores in the `aaa_key.pem` file (or `ois_key.pem` for Oracle Access Manager). A program named `openssl` in the `\tools` subdirectory generates the private key. The `openssl` program is called automatically during installation of each AccessGate and Access Server. Unlike Cert mode, Oracle Access Manager has already generated the private key. The key is presented automatically during installation.

In Simple mode, as in Cert mode, you secure the private key with a Privacy Enhanced Mail (PEM) pass phrase that you specify during installation of each component. During installation, the PEM pass phrase may also be referred to as the Global Access Protocol pass phrase. The generic term "pass phrase" is often used in this manual.

Note: Before an AccessGate or Access Server can use a private key, it must have the correct pass phrase. The pass phrase is stored in a nominally encrypted file called `password.lst`. For Simple mode, the PEM pass phrase is the same for each WebGate and Access Server instance.

If you do not store the password in a file during Access Server installation:

- On Windows, you are prompted for the pass phrase every time you start the Access Server.
- On Unix, you must use the `-P` option to pass the password whenever you launch the `start_access_server` script.

Cert Mode

Use Cert (SSL) mode if you have an internal Certificate Authority (CA) for processing server certificates. In Cert mode, communication between WebGate and Access Server, and Identity Server and WebPass are encrypted using Transport Layer Security, RFC 2246 (TLS v1). In both Simple and Cert mode, Oracle Access Manager components use X.509 digital certificates only. This includes Cert Authentication between WebGates and the Access Server where the standard cert-decode plug-in decodes the certificate and passes certificate information to the standard credential_mapping authentication plug-in.

For each public key there exists a corresponding private key that Oracle Access Manager stores in the `aaa_key.pem` file for the Access Server (or `ois_key.pem` for Identity Server).

A program named `openssl` in the `\tools` subdirectory generates the private key. This program is called automatically during installation of each AccessGate and Access Server. During installation, you present a certificate obtained from a CA.

You secure the private key with a Privacy Enhanced Mail (PEM) pass phrase that you specify when you install each component. In this manual, the term *pass phrase* is used.

Note: Before a WebGate or Access Server can use a private key, it must have the correct PEM pass phrase. The PEM pass phrase is also referred to as WebGate Pass Phrase and Transport Password. It can be stored in a nominally encrypted file called `password.lst` (or `password.xml` for Oracle Access Manager). It can be different for each WebGate and Access Server.

During Oracle Access Manager installation, if you do not yet have a certificate you may request one. In this case, you can complete installation despite the pending certificate status. However, the component or system cannot be setup until the certificates are issued and copied into the appropriate directory.

It is important to note, that if you generate a certificate request:

- You may complete installation as usual but you cannot perform set up if a request is pending.
- You must locate the request in the component installation directory. For example:
`IdentityServer_install_dir\identity\oblix\config\ois_req.pem`

Usually, the `.pem` file contains some extra data plus the encrypted string that represents the request.

- You must copy the following information into a certificate request field from your chosen CA and send the request to your CA; Oracle does not do this:

```
*-----Begin request-----
A97C7u54Sd0000lotsofrandomstuff8640uwst
89111mmmIyoSSTKHS9670sd
*-----End request-----
```

- When the CA returns the certificate, you can copy the certificate files to the appropriate component installation directory, then restart the component server or service. For example:

```
\IdentityServer_install_dir\identity\oblix\config
```

See the *Oracle Access Manager Identity and Common Administration Guide* for details.

If you do not store the pass phrase or password in a file during Access Server installation:

- **On Windows:** You are prompted for the pass phrase every time you start the Access Server.
- **On UNIX:** You must use the `-P` option to pass the password whenever you launch the `start_access_server` script.

For more information on transport security modes, see the *Oracle Access Manager Identity and Common Administration Guide*.

Meeting Web Server Requirements

You will need one or more Web servers to host WebPass, Policy Manager, and WebGate components. The Identity Server and Access Server do *not* require a Web server instance.

If you install WebPass and the Identity Server on the same Web server, the installation destination for WebPass *cannot* be the same as for Identity Server.

If you install Policy Manager and WebPass on the same Web server, you must place them at the same directory level. For example, if you specify `C:\COREid\WebComponent` as the WebPass installation directory, you must also specify this as the Policy Manager installation directory when the two components will reside on the same machine. `\identity` is appended to the WebPass installation directory and `\access` is appended to the Policy Manager installation directory.

Be sure that your Web server meets all requirements before you begin installation. See also "[Synchronizing System Clocks](#)" on page 2-2.

Task overview: Preparing your Web server

1. Ensure your Web server version is on the list of supported platforms (Policy Manager and WebPass Web Server and WebGate Web Server), which you can find under the Certify tab on <https://metalink.oracle.com>:
 - Log in to MetaLink as directed.
 - Click the Certify tab.
 - Click View Certifications by Product.
 - Select the Application Server option and click Submit.
 - Choose Oracle Application Server and click Submit.
2. Create new instances of your Web server running with your data to make it easier to make changes without taking down the service for other applications. See your Web server documentation for details.
3. Plan the Web server installation destination, and record details on "[Installation Preparation Checklists](#)" on page 2-28.

For more information, see:

- [Web Server-Specific Installation Packages](#)
- [General Considerations for Web Servers](#)

Web Server-Specific Installation Packages

Separate Web server-specific installation packages are provided for WebPass, Policy Manager, and WebGate components. Be sure to choose the appropriate installation package for your Web server and platform:

- **ISAPI:** An Internet Web server extension that Oracle Access Manager uses to identify Web server components that communicate with the Microsoft Internet Information Server (IIS Web server for Windows environments).
- **NSAPI:** An Internet Web server extension that Oracle Access Manager uses to identify Web components that communicate with the Sun (formerly Netscape/iPlanet) Web servers running on either Windows or Solaris.
- **Apache:** An Internet Web server extension that Oracle Access Manager uses to identify Web components that communicate with the Apache Web servers running on various platforms including Windows, Solaris, and Linux. For details, see ["Confirming Platform Requirements"](#) on page 2-26

Note: Oracle Access Manager supports Apache with or without SSL enabled. For SSL-enabled communication, Apache with mod_ssl only is supported, not Apache-SSL. mod_ssl is a derivative of, and alternative to, Apache-SSL.

Oracle Access Manager 10g (10.1.4.0.1) provide a single package for components that supports Apache with or without SSL enabled. For example:

- The APACHE_WebGate supports v1.3.x with or without SSL, as described in [Chapter 16, "Configuring the Apache v1.3 and Oracle HTTP Server Web Servers"](#).
- The APACHE2_WebGate supports v2 with or without SSL (and with or without reverse proxy enabled on Solaris and Linux). See also [Chapter 17, "Configuring Apache v2, IHS, and OHS Web Servers for Oracle Access Manager"](#).
- **IHS:** An Internet Web server extension that identifies Oracle Access Manager Web components that communicate with the IBM HTTP (IHS) Web servers powered by Apache running on various platforms. For example:
 - IHS_WebGate powered by Apache v1.3.x on Solaris, Linux, and Windows
 - IHS2_WebGate powered by Apache v2 on IBM-AIX: See [Chapter 17, "Configuring Apache v2, IHS, and OHS Web Servers for Oracle Access Manager"](#).
- **OHS:** An Internet Web server extension that identifies Oracle Access Manager Web components that communicate with the Oracle HTTP Server (OHS). Oracle provides OHS Web components based on open source Apache v1.3 (named OHS_...) and v2 (named OHS2_...). 10g (10.1.4.0.1) provides WebPass, Policy Manager, and WebGate components that can be installed on a standalone Oracle HTTP Server on Linux and Windows platforms.

- An OHS or OHS2 WebGate must be installed on the Oracle Application Server to enable integration with Oracle single sign-on as described in the *Oracle Access Manager Access Administration Guide*.
- OHS or OHS2 WebPass and "Access Manager" (original name represents the Policy Manager Web component) may be used with the Oracle Application Server. However, Apache WebPass and Access Manager (Policy Manager) Web components are also supported for this application.

See [Chapter 16, "Configuring the Apache v1.3 and Oracle HTTP Server Web Servers"](#) and [Chapter 17, "Configuring Apache v2, IHS, and OHS Web Servers for Oracle Access Manager"](#) for details. For version support, see the Certify tab on <https://metalink.oracle.com>.

- **Domino:** An Internet Web server extension that Oracle Access Manager uses to identify Web components that communicate with Lotus Domino Web servers running on various platforms.

See [Chapter 18, "Setting Up Lotus Domino Web Servers for WebGates"](#) on page 18-1. For version support, see the Certify tab on <https://metalink.oracle.com>.

General Considerations for Web Servers

It's a good idea to familiarize yourself with the following general considerations for Web servers in Oracle Access Manager installations:

- Each instance of the Identity Server communicates with a Web server through a WebPass plug-in that must be installed on a Web server host.

If you install Policy Manager and WebPass on the same Web server, you *must* place them at the same directory level. For example, if you specify C:\COREid\WebComponent as the WebPass installation directory, you must also specify this as the Policy Manager installation directory when the two components will reside on the same machine. \identity is appended to the WebPass installation directory and \access is appended to the Policy Manager installation directory.

- During WebPass, Policy Manager, and WebGate installation, your Web server must be configured to work with Oracle Access Manager. You can direct this Web server configuration update to occur either automatically or manually.

Note: Oracle recommends that you use the automatic configuration option to streamline the Web server update process and avoid errors.

- When accessing the Identity System or Policy Manager, you must specify the *hostname* of the Web server for the WebPass instance that connects to the targeted Identity System or Policy Manager and the HTTP port of the WebPass Web server instance.
- Additional guidelines for WebGate Web servers are discussed in "[WebGate Guidelines](#)" on page 2-8.
- On a Unix system during WebPass, Policy Manager, and WebGate installation, you must specify the user name and group that the Web server will use. Typically, the defaults are nobody. For HP-UX, the defaults are WWW (username) and others (group).
- On Linux systems, when installing Oracle Access Manager Web components with Apache and OHS you are prompted to install as the same user under which the

Web server is running. This information is located in the httpd.conf file in the User and Group directive entries.

Meeting Directory Server Requirements

Your installation requires one or more directory servers. You need to ensure that your directory server meets requirements for Oracle Access Manager and is properly prepared before starting the installation.

Note: See also [Appendix A, "Installing Oracle Access Manager with Active Directory"](#) and [Appendix B, "Installing Oracle Access Manager with ADAM"](#), if needed.

Task overview: Preparing your directory server

1. Ensure the directory server is on the list of supported platforms, as described under the Certify tab at <https://metalink.oracle.com>.
 - Log in to MetaLink as directed.
 - Click the Certify tab.
 - Click View Certifications by Product.
 - Select the Application Server option and click Submit.
 - Choose Oracle Application Server and click Submit.

Note: The Siemens DirX directory is not supported in 10g (10.1.4.0.1). Although the installation screen may still display DirX as a possible option.

2. Identify at least one person in your directory to use as the Master Administrator to complete installation and setup, as described in "[Assigning a Bind DN](#)" on page 2-18.
3. Estimate and ensure that you have adequate directory server space, as described in "[Assessing Directory Server Space](#)" on page 2-18.
4. Determine how you will secure directory server communication with Oracle Access Manager components, as described in "[Securing Directory Server Communications](#)" on page 2-19.
5. Ensure that one or more directory server instances are available for Oracle Access Manager installation and decide if you want to store user data separately from configuration and policy data, as described in "[Data Storage Requirements](#)" on page 2-21.
6. Establish a searchbase, configuration DN, and policy base for data, as described in:
 - [User Data and the Searchbase](#)
 - [Configuration Data and the Configuration DN](#)
 - [Policy Data and the Policy base](#)
7. Record your Person and Group object classes, as described in "[About Person and Group Object Classes](#)" on page 2-26.

8. Record directory server details, as described in "[Installation Preparation Checklists](#)" on page 2-28, including:
 - a. Host name and IP address, network port, and Root DN of each directory server.
 - b. User logon id and password for the directory server.
9. Decide how you plan to update the schema (automatically or manually), as described in "[Updating the Schema and Attributes Automatically Versus Manually](#)" on page 1-5.
10. See the *Oracle Access Manager Identity and Common Administration Guide* for details about making schema data available to Oracle Access Manager.

The inheritance of all objects is based on the premise of a common super class for both the structural object class and the auxiliary class. Otherwise, object class extension is not feasible.

Assigning a Bind DN

During installation and setup of the Identity Server and Policy Manager, you are asked to provide a bind DN (also known as Root DN in Oracle Access Manager). The directory account that Oracle Access Manager binds to should have Read, Write, Add, Delete, Search, Compare, and Selfwrite permissions. The method to create a user with these privileges varies among directory vendors. See your directory documentation for details.

Be sure that the native directory access control instructions (ACIs) and access control lists (ACLs) do not restrict the Oracle Access Manager bind DN account access to the user and configuration branches. Otherwise, the Oracle Access Manager bind DN may be affected by native directory server constraints such as password policies.

In addition, the user you create as the bind DN must have access to the schema when Oracle Access Manager software upgrades are performed, because the schema may be modified during the upgrade. If the schema is not accessible to the bind DN, the upgrade will fail, then manual action will be required to complete the upgrade. This includes having the ACLs modify directory schema entries.

Take the following guidelines into account:

Oracle Internet Directory: When installing the Identity Server with Oracle Internet Directory, you must designate the Root DN as the super user `cn=orcladmin` (not the fully qualified DN `cn=orcladmin,cn=users,dc=us,dc=mycompany,dc=com`).

Sun (formerly iPlanet): Oracle recommends that the bind DN user is not Directory Manager. Instead, create another user as a bind DN. The Directory Manager account will ignore your directory server's size and timeout limits. As a result, large searches could tie up the directory server.

For more information, see also "[User Data and the Searchbase](#)".

Assessing Directory Server Space

The directory server should have at least 1 KB of RAM for each user object. Each Oracle object should have at least 16 KB of RAM.

The following information is provided to help you calculate the space that will be required for your installation:

- A directory server with 250,000 user objects requires ~250 MB of RAM.

- A directory of this size may have 5,000 Oracle objects (a high estimate for 250,000 user entries), which would require an additional 80 MB.
- The indexes for this amount of data would require about twice the space of Oracle objects, approximately 160 MB.

Securing Directory Server Communications

The Identity Server, Policy Manager, and Access Server communicate with the directory server. During Oracle Access Manager installation and setup, default directory profiles are created for the components that communicate with the directory server. Each directory profile includes a database (DB) instance profile where the directory server communication method is indicated, among other things.

Two communication methods are available between Oracle Access Manager and the directory server: unsecured or secured. Secure communication between Oracle Access Manager and the directory server is also known as *SSL-enabled*. Unsecured communication is also referred to as *Open*.

Oracle Access Manager supports CA certificates in base64 format. SSL-enabled communication requires a signer's certificate (root CA certificate) from a third-party Certificate Authority in base64 format. For example, if you want to use SSL between a Identity Server and directory server, you will be prompted to provide the path to the certificate to establish SSL-enabled communication during Identity Server installation. In this case, a certificate must be installed on your directory server according to the instructions for the directory server. The directory server should *not* require client authentication (see the directory server documentation for instructions).

When configuring SSL for the directory server, note that Oracle Access Manager supports server authentication only. Client authentication is not supported. Oracle Access Manager verifies the server certificate against the Root CA certificate that you imported during product setup.

Guidelines

When planning and configuring communication between Oracle Access Manager and the directory server, the following guidelines apply:

- Communication between Identity Servers and the directory server may differ.
- Communication between Access Servers and the directory server may differ.
- Communication between all Policy Managers and the directory server must be consistent: all SSL-enabled or all open.

Note: When storing user data on a different directory server type than configuration and policy data, multiple root CA certificates are supported. When storing user data, configuration data, and policy data on directory server types, each can use a separate root CA. For more information, see "[Data Storage Requirements](#)" on page 2-21.

Caveats

SSL-enabled communication with the directory server is *not* supported when the Policy Manager is installed on Solaris with a Sun (previously Netscape) Web server. In a heterogeneous environment that includes a Policy Manager on Solaris, be sure to specify open communication between the directory server and all Policy Managers you install.

Oracle Access Manager components can share a DB profile even when components were not installed to use the same communication mode with the directory server. For example, suppose the Identity Server and Access Server were installed in open mode and the Policy Manager was installed with SSL enabled. In this case, the cert8.db and key3.db files must exist for each Oracle Access Manager component that communicates with the directory server and must reside in the Oracle Access Manager *Component_install_dir\identity|access\oblix\config* directory. You may either copy these files from other Oracle Access Manager component directories or run genCert (Policy Manager) or other utilities to generate them.

Note: Oracle Access Manager 10g (10.1.4.0.1) works with both the cert7.db (upgraded environments) and cert8.db (new installations) certificate store. For details about upgraded environments, see the *Oracle Access Manager Upgrade Guide*.

All Directory Servers: When configuring SSL for any directory server, be aware that Oracle Access Manager supports server authentication only. Client authentication is *not* supported. Oracle Access Manager verifies the server certificate against the Root CA certificate that you imported during product setup.

Task overview: Defining directory server communication security

1. Before Oracle Access Manager installation, review all directory server requirements for 10g (10.1.4.0.1), as described here and in "[Meeting Directory Server Requirements](#)" on page 2-17.
2. Before Oracle Access Manager installation, enable SSL on your directory server, if desired, as described in the documentation for your directory server vendors and certificate. For example:
 - a. Create a directory server instance if you do not have one.
 - b. Apply to your CA for a certificate for that instance.
 - c. Install the certificate to encrypt your directory server instance and restart the directory server.

Note: When storing user data on a different directory server type than configuration and policy data, multiple root CA certificates are supported.

3. During Identity Server installation, choose the appropriate communication between the directory server and the Identity Server, as described in "[Securing Directory Server Communications](#)" on page 2-19.
4. During Identity System setup, choose the appropriate communication between the directory server and the Identity System, as described here and in [Chapter 6, "Setting Up the Identity System"](#).

Note: When using certificates generated by a subordinate CA, the root CA's certificate must be present in the xxx_chain.pem along with the subordinate CA certificate. Both certificates must be present to ensure appropriate verification and successful Identity System setup.

5. During Policy Manager installation and setup, choose the appropriate communication between the directory server and the Policy Manager, as described in "[Securing Directory Server Communications](#)" on page 2-19.
6. During Access Server installation, choose the appropriate communication between the directory server and the Access Server, as described in "[Securing Directory Server Communications](#)" on page 2-19.
7. After installation, you may change the communication mode between Oracle Access Manager and the directory server, as described in the *Oracle Access Manager Identity and Common Administration Guide*.

Data Storage Requirements

This discussion provides details about data storage options and requirements. This information affects the Identity Server, Policy Manager, and Access Server.

All Directory Server Types: Oracle Access Manager supports storing user data, Oracle Access Manager configuration data, and policy data on a single directory server.

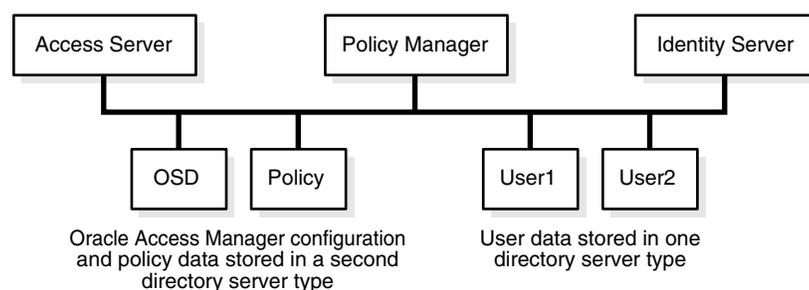
In addition, you can store user data separately on one directory server type and Oracle Access Manager configuration and policy data on a different type of directory server. For example, you may store user data in Active Directory and Oracle Access Manager configuration and policy data on ADAM (or Oracle Internet Directory).

When storing user data on a separate directory server *type* from configuration and policy data:

- All user data must be stored on the same directory server type.
- Configuration and policy data must be stored on the same type of directory server.
- With SSL, separate root CA's are supported.

[Figure 2-1](#) illustrates storing user data in a separate directory server type from configuration and policy data.

Figure 2-1 User Data in a Separate Directory Server Type



If the data is stored in different directory *types*, the user data searchbase, configuration DN, and policy base, should be unique.

During Oracle Access Manager installation and setup, you need to select proper user and configuration directory server types for your environment.

When you have configuration and policy data stored together in a different directory server type from user data, the following file comes into play:

```
IdentityServer_install_
dir\identity\oblix\data\common\ldaposdreferentialintegrityparams.xml
```

This is because the "referential_integrity_using" Value="oblix" in the `ldapreferentialintegrityparams.xml` file does *not* apply when configuration and policy data are stored on a different directory server type from user data

Also, in this case, the following files are used by the Identity System and Access System to map servers to DB profiles rather than the original `exclude_attrs` files:

```
IdentityServer_install_dir\identity\oblix\data\common\  
exclude_user_attrs.xml  
exclude_oblix_attrs.xml  
  
PolicyManager_install_dir\access\oblix\data\common\  
  
AccessServer_install_dir\access\oblix\data\common\  
exclude_oblix_attrs.lst
```

All parameters for the user data directory server are read using the DB profile. For the configuration data directory server, the DB subtype is read from:

```
Component_install_dir\identity | access\oblix\config\ldap\*DB.xml
```

Data Anywhere: This directory server option is available for only the user data directory server and implementation with Oracle Virtual Directory described in [Chapter 10, "Setting Up Oracle Access Manager with Oracle Virtual Directory"](#).

Data Anywhere is a data management layer that aggregates and consolidates *user data* from multiple sources (including RDBMS and LDAP directories) into a virtual LDAP tree that can be managed by the Identity System and used to support authentication and authorization using the Access System.

The LDAP directory branches containing Oracle Access Manager configuration and policy data must reside on one or more directory servers *other than* the one hosting VDS or user data. Oracle Access Manager applications only recognize configuration and policy information that resides outside the VDS virtual directory.

WARNING: Before you install Oracle Access Manager for use with Data Anywhere, read [Chapter 10, "Setting Up Oracle Access Manager with Oracle Virtual Directory"](#) and complete activities as specified.

IBM Directory Server (formerly SecureWay): See preceding details for all directory server types.

Oracle Internet Directory, and Sun: Oracle Access Manager supports storing user data separately from configuration and policy data with Oracle Internet Directory (multiple realms), and Sun directory servers. With these directory servers, you can store data either together on the same directory server or on different directory servers of the same type. For example:

- User data may be stored either separately or with configuration data.
- Configuration data may be stored separately or with user data.
- Policy data may either be stored separately or with user data.

If data is stored in different directories, the user data searchbase, configuration DN, and policy base should be disjoint. In other words, these DNs must be *unique* if you are storing your policy and configuration data in different Sun directories, or multiple Oracle Internet Directory realms.

Note: With Oracle Internet Directory, the configuration DN value is populated from the context of the identity management realm (cn=OracleContext). Also by default, the searchbase is the same as the configuration DN.

If you intend to have more than one user data directory and searchbase, be sure to specify the main user data directory and searchbase during installation and setup.

Active Directory, ADAM, and Novell eDirectory Caveats

Due to the strict adherence to referential integrity by Novell's eDirectory, Active Directory, and Active Directory Application Mode (ADAM), Oracle Access Manager configuration data and policy data must be stored under a common directory environment. Novell eDirectory, Active Directory, and ADAM are more rigid in the implementation of LDAP and will enforce referential integrity. These directory servers will *not* allow data in two separate trees/forests with cross references [like DN references] to each other.

With Oracle Access Manager, what is meant by having separate directories for user versus product configuration data and policy data is that you can have separate LDAP [disjoint] trees on different servers all of which happen to be in the same Novell directory server tree or Active Directory forest. Oracle Access Manager configuration data and policy data can be stored in separate parts of the overall directory environment, which does allow for a level of segregation of the Oracle Access Manager-specific information away from your user data.

On Active Directory: In an Active Directory environment you may store Oracle Access Manager configuration data on one specific domain controller and policy data on another. The policy data and configuration data domain controllers should be in the same forest. user data may reside in a different forest. It is important that replication be either avoided, or very well understood. For more information, see [Appendix A, "Installing Oracle Access Manager with Active Directory"](#).

When storing user data on a separate directory server type from configuration and policy data, auxiliary class support should match. Oracle Access Manager does not support a mixed mode for dynamic-auxiliary support. You can connect to either the user data directory server or configuration data directory server using ADSI, if they are in separate forests. ADSI does not allow a bind against both forests at the same time. With ADSI enabled for:

- **The User Data Directory Server**—During Identity Server and Policy Manager setup, if ADSI is selected for the user data directory server type then the ADSI checkbox is not available when choosing configuration directory server type details.
- **The Configuration Data Server**—When this directory type is ADSI enabled:
 - In the globalparams.xml file on the Identity Server, the parameter "IsADSIEnabled" and value "true" should appear.
 - In the globalparams.lst file on the Policy Manager, the parameter "adsiEnabled"=true appears.

The dbSubType "adsiEnabled" flags for Active Directory are read using the DB profile. Its value is ADSI when ADSI is enabled for the user data directory server. The ActiveDirectory and ADAM flags are removed from the globalparams.xml files.

See [Appendix A, "Installing Oracle Access Manager with Active Directory"](#) for more information.

If the user data directory server type is Active Directory, content of `exclude_attr-ad.xml` are copied to:

```
IdentityServer_install_dir\identity\oblix\data\common\exclude_user_attr.xml
```

If the configuration and policy data directory type is Active Directory, content of `exclude_attr-ad.xml.lst` are copied to the following locations:

```
IdentityServer_install_dir\identity\oblix\data\common\exclude_oblix_attr.xml
```

```
PolicyManager_install_dir \access\oblix\data\common\exclude_oblix_attr.lst
```

```
AccessServer_install_dir\access\oblix\data\common\exclude_oblix_attr.lst
```

Note: The `ActiveDirectory` flag longer appears in `globalparams.xml`.

With ADAM: Data can be stored as follows:

- User data may be stored on a different partition from configuration and policy data.
- User data may be stored on a separate directory server type from configuration and policy data.
- Oracle Access Manager requires a node with the object class attribute value of `organizationalUnit` (`ou`) for the configuration and policy DNSs.
- Configuration and policy data can share the same ADAM instance or be stored on different ADAM instances.

For more information, see [Appendix B, "Installing Oracle Access Manager with ADAM"](#).

Novell eDirectory: To avoid problems with `GroupOfUniqueNames`, change the class mapping for Groups in the LDAP Group object to reference `GroupOfUniqueNames` instead of `groupOfNames` (the default). Otherwise, each time you save any attribute, the schema may be violated and your groups may not work correctly. For example, for NDS the "groupOfUniqueNames" LDAP group object should be listed before the "groupOfNames" object.

To change the order using the NDS Console1

1. Expand the NDS tree.
2. For the NDS node in the left pane, right-click the "LDAP Group" object in the right pane and select Properties, Class Map tab.
3. Change the order in which the two group objects appear.

See your Novell eDirectory documentation for details about adding this mapping.

User Data and the Searchbase

User data consists of user directory entries managed by the Identity System. This data includes the information related to users, groups, locations, and other generic objects managed by the Identity System.

When installing Oracle Access Manager, you need to provide the following information to set up the main directory server profile:

- Directory server type where user data is stored
- Bind information, including the DNS host name, port, user name (bind DN), password
- Searchbase, to identify the node in the directory information tree (DIT) under which this data is stored and the highest possible base for all user data searches

Note: If you intend to have more than one user data directory and searchbase, you must specify the main user data directory and searchbase during installation and setup. After setup, you must manually add one or more database profiles to add the disjoint namespaces.

- Master Administrators (one or more)

Automatically updating the directory during setup is recommended to load Oracle Access Manager schema classes with configuration information.

Be sure to observe the following guidelines apply:

Oracle Internet Directory: When installing the Identity Server with Oracle Internet Directory, you must designate the Root DN as the super user `cn=orcladmin`, not the fully qualified DN (for example, not `cn=orcladmin,cn=users,dc=us,dc=mycompany,dc=com`).

Sun (formerly iPlanet): Oracle recommends that the bind DN user is not Directory Manager. Instead, create another user as a bind DN. The Directory Manager account will ignore your directory server's size and timeout limits. As a result, large searches could tie up the directory server.

For more information, see "[Data Storage Requirements](#)" on page 2-21.

Configuration Data and the Configuration DN

Configuration data (Oracle Access Manager configuration details), are stored in the directory. This data includes workflow and configuration information that governs the appearance and functionality of the Identity System and Access System. Configuration data is managed by the Identity System.

When installing Oracle Access Manager, you need to provide details for the directory server where you plan to store configuration data. If you store configuration data and user data together, this information will be the same. The following caveats apply:

- The bind DN for configuration data may be anywhere except at the base suffix.
- A bind DN for configuration data (known as the configuration DN) is similar to the searchbase for user data and must be specified to identify the node in the DIT under which the Oracle Access Manager schema and all configuration data is stored for the Identity and Access Systems.
- Additional caveats may apply, as described under "[Data Storage Requirements](#)" on page 2-21.

Note: With Oracle Internet Directory, the configuration DN value is populated from the context of the identity management realm (`cn=OracleContext`). Also by default, the searchbase is the same as the configuration DN.

Again, automatically updating the directory during setup is recommended to load Oracle Access Manager schema classes with configuration information.

Oracle Internet Directory: With Oracle Internet Directory, the configuration DN value is populated from the context of the identity management realm (cn=OracleContext). Also by default, the searchbase is the same as the configuration DN. For multiple realm installations, see the *Oracle Access Manager Identity and Common Administration Guide* for details about setting up a disjoint searchbase after installing and setting up the Identity System.

Policy Data and the Policy base

Policy data consists of policy definitions and rules that govern access to resources. This data is maintained in the directory server by the Policy Manager.

When installing Oracle Access Manager, you need to provide details for the directory server where you plan to install policy data. If you store policy data separately from user data, directory server details will differ from those specified for user or configuration data. For more information, see "[Data Storage Requirements](#)" on page 2-21.

During Policy Manager setup, you must also provide the policy base to identify the location in the DIT under which all Oracle Access Manager policy data is stored and the highest possible base for all policy searches. Accepting the default "/" as the policy domain when setting up the Policy Manager protects the entire Web server.

About Person and Group Object Classes

Oracle Access Manager supports User and Group as standard Person and Group object classes, respectively. In addition, Oracle Access Manager supports User and Group.

The Person object class defines the user's profile information. If you do not have a specific object class to use, Oracle Access Manager can automatically configure commonly found Person object class definitions.

Person Object Class

- **User:** Active Directory
- **InetOrgPerson:** ADAM, Data Anywhere (Oracle Virtual Directory), IBM, Oracle Internet Directory, and Sun directory servers
- **organizationalPerson:** NDS

The Group object class defines group attributes. If you do not have a specific object class to use, Oracle Access Manager can automatically configure commonly found Group object class definitions as follows:

Group Object Class

- **Group:** Active Directory
- **GroupofUniqueNames:** ADAM, Data Anywhere, IBM, NDS, Oracle Internet Directory, and Sun directory servers

Confirming Platform Requirements

Oracle Access Manager 10g (10.1.4.0.1) supports a variety of operating systems, directory servers, Web servers, compilers, and browsers, and integration with a

number of application servers, portal servers, system management products, and packaged applications.

For the latest support information, see details under the Certify tab on:

<http://metalink.oracle.com>

To use MetaLink

1. Navigate to <http://metalink.oracle.com>.
2. Log in to MetaLink as directed
3. Click the Certify tab.
4. Click View Certifications by Product.
5. Select the Application Server option and click Submit.
6. Choose Oracle Application Server and click Submit.

To download free release notes, white papers, or other collateral, please visit the Oracle Technology Network (OTN). You must register online before using OTN. Registration is free and can be accomplished at the following site:

<http://www.oracle.com/technology/membership/>

If you already have a user name and password for OTN, you can go directly to the documentation section of the OTN Web site at:

<http://www.oracle.com/technology/documentation/>

Preparing a Temporary Directory for Installers

Oracle Access Manager installation media provides all product packages that you need to install Oracle Access Manager components, including Language Packs. It is a good idea to copy these packages into a temporary installation directory that differs from the any directory where you plan to install Oracle Access Manager components. If you plan to install Oracle Access Manager components with one or more Language Packs, you must copy the needed Language packages into the same temporary directory as the component installer. Otherwise, the component installer cannot detect the Language Packs and offer these for installation. If you are not installing any Language Packs, you may install the desired Oracle Access Manager component directly from the installation media.

To store Oracle Access Manager installers for installation

1. Create a temporary directory in which to store Oracle Access Manager component installers, including all needed Language Pack installers for the Identity System and Access System.
2. Copy the Oracle Access Manager packages from the installation media into the temporary directory from which you can install the component and any Language Packs together, at the same time.
3. During component installation: Run the installer from the temporary directory as instructed in appropriate chapters of this manual.

Uninstalling Oracle Access Manager Components

During Oracle Access Manager component installation, information is saved after certain operations. Until information is saved, you may return and restate details. However, after you are informed that a component is being installed, Oracle Access Manager files are added to the file system.

Note: If you cancel the installation process after receiving the message that a component is being installed and before completing all procedures, you must remove Oracle Access Manager-related information to restore the system to its previous condition.

Some changes made for Oracle Access Manager are not handled automatically and must be manually removed when the Uninstaller program finishes. For details about removing Oracle Access Manager components, see [Chapter 20, "Removing Oracle Access Manager"](#).

Under certain circumstances, you may want to reuse an existing Identity Server name. If you do not delete the original Identity Server name from the System Console, a login following the set up of a new instance may result in the message *"Application has not been set up"*. Special steps must be taken to ensure you can set up the application and login when recycling an Identity Server name. For details, see ["Recycling an Identity Server Instance Name"](#) on page 20-4.

Installation Preparation Checklists

Installation of Oracle Access Manager requires some planning and the checklists in this discussion are provided for your convenience. For example, the checklists in [Table 2-3](#):

- Provide a space where you can map out and record your environment.
- Help you prepare to answer prompts during Oracle Access Manager installation and setup.
- Are organized according to the recommended component installation sequence described under ["About the Installation Task"](#) on page 1-1, and the installation process for each component.
- Provide a reference to a specific page number in this manual where you will find additional information.

Table 2-3 *Installation Preparation Checklists*

Task	Subtask	Checklist: Prepare for Oracle Access Manager Installation and Setup	Reference
1 to 3		Prepare for Identity System Installation and Setup	
1		Prepare for Identity Server Installation	
	1.1	Default Locale (Administrator Language)	
		Languages Language Packs	See Chapter 3, "About Multi-Language Environments"

Table 2–3 (Cont.) Installation Preparation Checklists

Task	Subtask	Checklist: Prepare for Oracle Access Manager Installation and Setup	Reference
	1.2	Transport security mode between the Identity Server and WebPass: <ul style="list-style-type: none"> ▪ Open ▪ Simple ▪ Cert 	See "Securing Oracle Access Manager Component Communications" on page 2-11
	1.3	Unique Identity Server ID to be used within Oracle Access Manager to identify this Identity Server instance:	
		Host name of the machine where the Identity Server is to be installed:	
		Port number for Identity Server/WebPass communication:	
		Is this the first Identity Server installed for this directory server? <ul style="list-style-type: none"> ▪ Yes ▪ No 	
	1.4	Security mode between directory server and Identity Server: <ul style="list-style-type: none"> ▪ SSL ▪ Open 	
		If SSL, path to the Root CA certificate:	
		Simple mode only Global Access Protocol pass phrase	
		Cert Mode Only Certificate PEM pass phrase:	
		Path of the certificate request file (Cert request only):	
		Path of the certificate file (Cert mode only):	
		Path of the key file (Cert mode only):	
		Path of the chain file (Cert mode only):	
	1.5	Prepare directory server details Location of configuration data in the directory server: <ul style="list-style-type: none"> ▪ User data directory server ▪ Separate directory server ▪ Manual install 	
		Directory server type: <ul style="list-style-type: none"> ▪ Sun Directory Server 5.x ▪ NDS ▪ Active Directory ▪ Active Directory on Windows Server 2003 ▪ Active Directory Application Mode ▪ IBM Directory Server ▪ Data Anywhere <p>Note: Data Anywhere (Oracle Virtual Directory Server) is available only for the user data directory server. Configuration and policy data must be stored in a native directory.</p>	Before you install Oracle Access Manager with Data Anywhere, see Chapter 10, "Setting Up Oracle Access Manager with Oracle Virtual Directory" .
		Directory server host machine name or IP address:	

Table 2–3 (Cont.) Installation Preparation Checklists

Task	Subtask	Checklist: Prepare for Oracle Access Manager Installation and Setup	Reference
		Directory server port #:	
		Directory server bind DN:	
		Directory server administration password:	
	1.6	(Windows only) Unique Identity Server service name that will differentiate this instance in the Services window if you install several instances of Identity Server):	
2		Prepare to install WebPass. Decide on the following:	
	2.1	Default Locale (Administrator Language)	
		Languages Language Packs Same Language Packs as the Identity Server	
		Web server user name (Unix only):	
		Web server group (Unix only):	
		WebPass installation directory. If installing on the same machine as the Identity Server, this cannot be the same as the Identity Server installation directory:	
	2.2	Transport security mode between the Identity Server and WebPass:	See task 1, this table See Securing Oracle Access Manager Component Communications on page 2-11.
		WebPass ID used by Oracle Access Manager to identify the WebPass instance:	
	2.3	WebPass host name:	
		Port # for Identity Server/WebPass communication:	See task 1, this table
		Simple mode only Global Access Protocol pass phrase	See task 1, this table
		Cert mode only Certificate PEM phrase:	
		Path of the certificate request file (Cert request only):	
		Path of the certificate file (Cert mode only):	
		Path of the key file (Cert mode only):	
		Path of the chain file (Cert mode only):	
	2.4	Automatically update your Web server with WebPass information? <ul style="list-style-type: none"> ▪ Yes ▪ No 	
		If Yes, absolute path of the Web server configuration directory containing the obj.conf file (or the httpd.conf file on Apache):	
3		Prepare for Identity System Setup Decide on the following:	

Table 2–3 (Cont.) Installation Preparation Checklists

Task	Subtask	Checklist: Prepare for Oracle Access Manager Installation and Setup	Reference
	3.1	Directory server type:	See task 1, this table
		Directory server host machine name or IP address:	See task 1, this table
		Directory server port #:	See task 1, this table
		Directory server bind DN:	See task 1, this table
		Directory server administration password:	See task 1, this table
		Security mode between directory server and Identity Server:	See task 1, this table
		Is the configuration data stored in the user data directory server?	See task 1, this table
		Configuration DN:	
		Directory searchbase where user data is stored:	
	3.2	Person object class:	
		Auto-configure the Person object class? <ul style="list-style-type: none"> ▪ Yes ▪ No 	
		Group object class:	
		Auto-configure the Group object class? <ul style="list-style-type: none"> ▪ Yes ▪ No 	
		Configure the Person object class (manual process is optional). Configure the following attributes if you chose not to auto-configure your Person object class:	
		User full name attribute:	
		User login ID attribute:	
		Password attribute:	
	3.3	Configure the Group object class (manual process is optional). Configure the following attributes if you chose not to auto-configure your Group object class:	
		Group name attribute:	
	3.4	Prepare to define Master Administrators	
		Master Administrators (The Master Administrator):	
4 to 8		Prepare for Access System Installation and Setup	
4		Prepare to install the Policy Manager. Decide on the following:	
	4.1	Install a WebPass for this Policy Manager, as described in Chapter 5, "Installing WebPass" and: <ul style="list-style-type: none"> ▪ Ensure that the WebPass is installed on the same Web server instance and at the same directory level as you will install the Policy Manager. ▪ Ensure that the Webpass has been configured to work with a particular Identity Server. 	
	4.2	Default Locale (Administrator Language)	

Table 2–3 (Cont.) Installation Preparation Checklists

Task	Subtask	Checklist: Prepare for Oracle Access Manager Installation and Setup	Reference
		Languages Language Packs Same Language Packs as Identity System	See task 1, this table
		Web server user name (Unix only):	See task 2, this table.
		Web server group (Unix only):	See task 2, this table.
		WebPass installation directory:	See task 2, this table.
	4.3	Directory server type: <ul style="list-style-type: none"> ▪ Sun Directory Server 5.x ▪ NDS ▪ Active Directory ▪ Active Directory on Windows Server 2003 ▪ Active Directory Application Mode ▪ IBM Directory Server <p>Note: Data Anywhere (Oracle Virtual Directory Server) is available only for the user data directory server and is not an option for the Policy Manager or Access Server.</p>	Configuration and policy data must be stored in a native directory, as described in Chapter 10, "Setting Up Oracle Access Manager with Oracle Virtual Directory" .
		Are you storing your policy data separate from your user data directory server? <ul style="list-style-type: none"> ▪ Yes ▪ No 	
	4.4	Transport security mode between the Policy Manager and Access Servers: <ul style="list-style-type: none"> ▪ Open ▪ Simple ▪ Cert 	See Securing Oracle Access Manager Component Communications on page 2-11.
		Simple mode only Global Access Protocol pass phrase:	
		Cert mode only Certificate PEM pass phrase:	
		Path of the certificate request file (Cert mode only):	
		Path of the certificate file (Cert mode only):	
		Path of the key file (Cert mode only):	
		Path of the chain file (Cert mode only):	
	4.5	Automatically update the Web server configuration file with Access System information? <ul style="list-style-type: none"> - Yes - No 	
		If Yes, absolute path of the Web server configuration directory containing the obj.conf file (or the httpd.conf file on Apache):	See task 2
5		Prepare to set up the Policy Manager. Fill in the following based on this Policy Manager installation:	

Table 2–3 (Cont.) Installation Preparation Checklists

Task	Subtask	Checklist: Prepare for Oracle Access Manager Installation and Setup	Reference
	5.1	Directory server type:	See task 4.
		Directory server host machine name or IP address:	See task 4.
		Directory server port #:	See task 4.
		Directory server bind DN:	See task 4.
		Directory server administration password:	See task 4.
		Security mode between the directory server and the Policy Manager: - Open - SSL	
		If SSL, path to the SSL certificate:	
		Location of configuration data in the directory server: - User data directory server - Separate directory server	
		If on separate directory server, the directory server host machine name or IP address:	
		If on separate directory server, the directory server port #:	
		If on separate directory server, the directory server bind DN:	
		If on separate directory server, the directory server administration password:	
		If on separate directory server, the security mode between the directory server and the Policy Manager: - Open - SSL	
		If SSL, path to the SSL certificate:	
		Location of policy data in the directory server: - User data directory server - Configuration data directory server - Separate directory server	
		If on separate directory server, the directory server host machine:	
		If on separate directory server, the directory server port #:	
		If on separate directory server, the directory server bind DN:	
		If on separate directory server, the directory server administration password:	
		If on separate directory server, the security mode between the directory server and the Policy Manager: - Open - SSL	
		If SSL, path to the SSL certificate:	
		Directory searchbase where user data is stored:	See task 3.
		Configuration DN:	See task 3.
		Policy base:	

Table 2–3 (Cont.) Installation Preparation Checklists

Task	Subtask	Checklist: Prepare for Oracle Access Manager Installation and Setup	Reference
		Person object class name:	See task 3.
		Policy Manager policy domain root:	
	5.2	Configure authentication schemes? - Yes - No	
		If Yes, select authentication scheme or schemes: Configure Oracle Access Manager-related authentication schemes and policy domains Authentication Schemes - Basic Over LDAP - Client Certificate - Anonymous - Oracle Access and Identity - Oracle Access and Identity for AD Forests Policy Domains - Identity Domain - Access Domain	
		Configure policies to protect Oracle Access Manager-related URLs? - Yes - No	
6		Prepare to Create an Access Server Instance in the Access System Console Before you continue, you must decide on the following:	
	6.1	Access Server name (do not include spaces):	
		Access Server host name:	
		Port # the Access Server listens to:	
		Transport security mode between the Access Server and WebGate: - Open - Simple - Cert	See Securing Oracle Access Manager Component Communications on page 2-11.
	6.2	Create an Access Server instance in the Access System Console	See " Creating an Access Server Instance in the System Console " on page 8-4
7		Prepare to Install the Access Server.	
	7.1	Default Locale (Administrator Language)	
		Languages Language Packs Same Language Packs as Policy Manager	
		Web server user name (Unix only):	

Table 2–3 (Cont.) Installation Preparation Checklists

Task	Subtask	Checklist: Prepare for Oracle Access Manager Installation and Setup	Reference
		Web server group (Unix only):	
		Access Server installation directory:	
	7.2	Transport security mode between the Access Server and the WebGate/AccessGate:	See task 6
	7.3	Security mode between the configuration data directory server and the Access Server: - Open - SSL	
		Configuration Data Directory Server host machine: Same as Policy Manager DS? --Yes --No	
		Configuration Data Directory server port #: Same as Policy Manager DS? --Yes --No	
		Configuration Data Directory server bind DN: Same as Policy Manager DS? --Yes --No	
		Configuration Data Directory server administration password: Same as Policy Manager DS? --Yes --No	
		Configuration Data Directory server type: - Sun Directory Server 5.x - NDS - Active Directory - Active Directory Application Mode - IBM Directory Server Note: You will be asked about Active Directory using ADSI whenever the Access Server installation occurs on a Windows platform.	
		Which directory server stores the configuration data?	See task 1.
		Which directory server stores the policy data?	See task 1.
		Access Server name:	See task 6.
		Configuration DN:	See task 3.
		Policy Base:	See task 4.
		Simple mode only Global Access Protocol pass phrase:	See task 4.

Table 2–3 (Cont.) Installation Preparation Checklists

Task	Subtask	Checklist: Prepare for Oracle Access Manager Installation and Setup	Reference
		Cert mode only Certificate PEM phrase:	
		Save PEM phrase in a password file? (Simple and Cert modes only): - Yes - No	
		Path of the certificate request file (Cert mode only):	
		Path of the certificate file (Cert mode only):	
		Path of the key file (Cert mode only):	
		Path of the chain file (Cert mode only):	
8		Prepare to Create a WebGate instance in the Access System Console. Before you continue, you must decide on the following:	
	8.1	WebGate name (do not include spaces):	
		WebGate host name:	
		Web server port #:	
		WebGate password/confirm password:	
		Transport security mode between the Access Server and WebGate:	See task 6.
	8.2	Define a WebGate instance in the Access System Console	See "Creating a WebGate Instance" on page 9-2
	8.3	Associate the WebGate with an Access Server	See "Associating a WebGate and Access Server" on page 9-4
9		Prepare to install the WebGate. Before you continue, you must decide on the following:	
	9.1	Default Locale (Administrator Language)	
		Languages Language Packs Same Language Packs as Policy Manager and Access Server	
		Web server user name (Unix only):	
		Web server group (Unix only):	
		WebGate installation directory (can be same as WebPass installation directory):	
	9.2	Transport security mode between the Access Server and the WebGate:	See task 6.
	9.3	WebGate ID:	See task 8.
		WebGate password:	See task 8.
		Access Server ID:	See task 6.
		Access Server host name:	See task 6.
		Access Server port #:	See task 6.

Table 2–3 (Cont.) Installation Preparation Checklists

Task	Subtask	Checklist: Prepare for Oracle Access Manager Installation and Setup	Reference
		Simple mode only Global Access Protocol pass phrase:	See task 6.
		Cert mode only Certificate PEM phrase:	
		Path of the certificate request file (Cert mode only):	
		Path of the certificate file (Cert mode only):	
		Path of the key file (Cert mode only):	
		Path of the chain file (Cert mode only):	
	9.5	Automatically update the Web server configuration file? - Yes - No	
		If Yes, absolute path of the Web server configuration directory containing the obj.conf file (or the httpd.conf file on Apache):	
10 to 14		Intended Optional Components	
	10	Oracle Virtual Directory Server (Data Anywhere components) - Yes - No	See Chapter 10, "Setting Up Oracle Access Manager with Oracle Virtual Directory"
	11	SNMP Agent (optional) - Yes - No	See Chapter 11, "Installing the SNMP Agent"
	12	Audit-to-database components - Yes - No	See Chapter 13, "About Installing Audit-to-Database Components"
	13	Language Packs, Independent Installation (optional) - Yes - No	See Chapter 12, "Installing Language Packs Independently"
	14	Software Developer Kit (SDK) is optional. - Yes - No	See the <i>Oracle Access Manager Developer Guide</i>

About Multi-Language Environments

This chapter explains how to set up Oracle Access Manager host machines when installing 10g (10.1.4.0.1) in multi-language environments. The following topics are included:

- [About Installing in Multi-Language Environments](#)
- [Setting Environment Variables for Command-Line Tools \(Optional\)](#)
- [Installing with Language Packs](#)
- [Directory Structure](#)
- [Removing Language Packs](#)

Note: The behavior of Oracle Access Manager 10g (10.1.4.0.1) is the same whether you have an English-only installation or one that includes multiple-languages. For a behavioral overview, see the *Oracle Access Manager Introduction*.

About Installing in Multi-Language Environments

When you install Oracle Access Manager without a Language Pack, English is the only language used to display product messages for Administrators and end users. When installing with Oracle-provided Language Packs, you can choose any Administrator language to be used as the default for administrative activities. English is always installed, regardless of the language (locale) that you choose as the default for Administrators and any other Language Packs you install.

Note: You can include one or more Oracle-provided Language Packs during component installation as described in "[Installing with Language Packs](#)" on page 3-5. Alternatively, you can install Language Packs after component installation as described in [Chapter 12, "Installing Language Packs Independently"](#).

Administrative information can be displayed in only installed Administrator languages. If administrative pages are requested in a language that is not supported for administrators (based upon the browser setting), the language that was selected as the default Administrator language during product installation is used to display the pages. For more information about languages, see the *Oracle Access Manager Introduction*.

Static product pages (/identity/oblix/index.html and /access/oblix/index.html) always use the default Administrator language selected during Identity Server and Access Server installation at this location.

For end-users, Oracle Access Manager 10g (10.1.4.0.1) enables the display of static application data such as error messages, and display names for tabs, panels, and attributes in installed end-user languages. After installing Language Packs, you must enable all languages that you want to use, then configure Oracle Access Manager to use the installed languages by entering display names for attributes, tabs, and panels. For details about enabling languages, see the *Oracle Access Manager Identity and Common Administration Guide*.

When installing Oracle Access Manager on a machine running an English (AMERICAN) language operating system, installation and setup messages appear in English. When installing on a machine running a supported *non*-English (AMERICAN) language operating system, installation messages appear in the locale of the operating system and setup messages appear in the language you selected as a default locale for Administrative functions during component installation (assuming that you have installed Oracle-provided Language Packs).

Note: If the machine on which you are installing is running with an operating system that specifies something other than English (AMERICAN) as the language, America as the territory, and ASCII as the character set, review "[Setting Environment Variables for Command-Line Tools \(Optional\)](#)" next.

Setting Environment Variables for Command-Line Tools (Optional)

By default, the console on each machine supports the locale and character set of the operating system. However, it may also support other character encodings such as UTF-8.

Oracle Access Manager 10g (10.1.4.0.1) console-based command-line tools automatically detect and use various environment variables to determine the language to use when processing data provided in non-English languages and non-American locales (also known as internationalized data).

Note: When *no* environment variables are set to specify the language and character set, Oracle Access Manager command-line tools use English (AMERICAN) as the language, America as the territory (AMERICA), and ASCII (US7ASCII) as the character set. In this case, Oracle Access Manager command-line tools may not properly process command-line input provided in non-English languages and non-American locales.

You may disable the auto-detect feature and specify a language to take precedence by setting the following environment variables:

- LANG—A Unix system environment variable that can be used to set the server native language, local customs, and coded character set (the language and character set used by the server also known as the *locale*). When LANG is set, Oracle Access Manager console-based command-line tools use the language and character set specified by this variable.

- **NLS_LANG**—When set, this variable disables the Oracle Access Manager command-line tools auto-detect feature and specifies a language and character set that takes precedence for data that is entered and displayed by any Oracle application (including but not limited to Oracle Access Manager).
- **COREID_NLS_LANG**—When set, this variable disables the auto-detect feature and sets the language and territory used for data that is entered and displayed by Oracle Access Manager and its command-line console only.

If the intended host machine is running a supported operating system that specifies something other than `AMERICAN_AMERICA.US7ASCII`, the auto-detection feature will determine and use the locale of the server for Oracle Access Manager command-line tools. However, when set, `NLS_LANG` takes precedence over `LANG` and `COREID_NLS_LANG` takes precedence over `NLS_LANG`.

In other words, `NLS_LANG` and `COREID_NLS_LANG` override the Oracle Access Manager automatic detection of the server's locale and convert the data passed as command-line arguments and the like from the console's character set encoding to the UTF-8 encoding used within Oracle Access Manager. This enables you to set `NLS_LANG` for the Oracle Database Server (for example) and also set `COREID_NLS_LANG` to enable proper operation of for Oracle Access Manager command-line interfaces when these products are running on the same machine.

The character set that is specified by `NLS_LANG` or `COREID_NLS_LANG` should reflect the setting for the client application. In the case of Oracle Access Manager, the client application is the console and command-line tools which are invoked during installation to perform operations such as updating the directory server schema with Oracle Access Manager configuration, policy, and user data; modifying the Web server's configuration; creating services for Identity and Access Servers on Windows platforms; registering Access Servers with the Policy Manager, and other operations; and which can be invoked by administrators at any time.

Both `NLS_LANG` and `COREID_NLS_LANG` consist of three components specified in the following format (including the punctuation):

```
NLS_LANG = language_territory.charset
COREID_NLS_LANG = language_territory.charset
```

For example:

```
NLS_LANG = FRENCH_CANADA.WE8ISO8859P1
COREID_NLS_LANG = JAPANESE_JAPAN.JA16EUC
```

Each component of `NLS_LANG` and `COREID_NLS_LANG` controls the operation of a subset of globalization and localization support features:

- **language** specifies conventions such as the language used for Oracle messages, sorting, day names, and month names. Each supported language has a unique name; for example, `AMERICAN`, `FRENCH`, or `GERMAN` to name a few. The language argument specifies default values for the territory and character set arguments. If the language is not specified, then the value defaults to `AMERICAN`.
- **_territory** specifies conventions such as the default date, monetary, and numeric formats. Each supported territory has a unique name; for example, `AMERICA`, `FRANCE`, or `CANADA`. If the territory is not specified, then the value is derived from the language value.
- **.charset** specifies the character set used by the client application (usually the Oracle character set that corresponds to the user's terminal character set or the operating system character set). Each supported character set has a unique

acronym, for example, US7ASCII or JA16EUC. Each language has a default character set associated with it.

For instance, if the database character set is AL32UTF8 and the client is running on a Windows operating system, then you should *not* set AL32UTF8 as the client character set. Instead, NLS_LANG or COREID_NLS_LANG should reflect the code page of the client. For example, on an English Windows client, the code page is 1252. An appropriate setting is AMERICAN_AMERICA.WE8MSWIN1252.

On both Unix and Windows platforms, NLS_LANG and COREID_NLS_LANG should be set as local environment variables.

For additional information about NLS_LANG, see the Oracle Database Globalization Support Guide, 10g Release 2 (10.2), Part Number B14225-02.

Setting NLS_LANG and COREID_NLS_LANG on Windows Systems

On Windows systems, the encoding scheme (character set) is specified by a code page. Code pages are defined to support specific languages or groups of languages that share common writing systems. Oracle views the terms code page and character set as the same. The Windows GUI and DOS command prompt do not use the same code page in non Chinese-Japanese-Korean environments.

Note: For more information about NLS_LANG, see the Oracle Database Globalization Support Guide, 10g Release 2 (10.2), Part Number B14225-02.

To set NLS_LANG and COREID_NLS_LANG

1. From the Start menu, select Run.
2. In the command window, type cmd, then click OK.
3. At the command prompt, type the environment variable appropriate to your system. For example:

```
C:\>set COREID_NLS_LANG = JAPANESE_JAPAN.JA16EUC
```

4. Locate and edit the entry with the name NLS_LANG, if desired.

Setting NLS_LANG and COREID_NLS_LANG on Unix Systems

As discussed earlier, Oracle Access Manager 10g (10.1.4.0.1) console-based command-line tools automatically detect and use the server locale when processing data provided in non-English languages and non-American locales (also known as internationalized data). You may disable the auto-detect feature and specify the language to take precedence by setting the Unix LANG environment variable as described in your Unix documentation, or the Oracle NLS_LANG and COREID_NLS_LANG environment variables as described here.

On a Unix system, set this as you would any other environment variable. The method will differ depending upon the shell: bash, csh, sh, and so on.

Note: For more information about NLS_LANG, see the Oracle Database Globalization Support Guide, 10g Release 2 (10.2), Part Number B14225-02.

Installing with Language Packs

Installing Oracle Access Manager without any Oracle-provided Language Packs results in English as the language for both end-user and administrative information. Installing Oracle Access Manager with one or more Oracle-provided Language Packs enables you to localize Oracle Access Manager applications to display static data such as error messages and display names for tabs, panels, and attributes to end users in their native language.

Note: Oracle Access Manager supports UTF-8 encoding for multibyte languages such as Chinese, Japanese, and provides support for bi-directional languages. See the *Oracle Access Manager Introduction* for a complete list of languages available for users and for administrative information. Contact Oracle for information about specific Language Packs.

Language Pack installers are available on the Oracle Access Manager installation media. For each language that Oracle supports (other than English, which requires no Language Pack), one Language Pack installer is provided for the Identity System and one Language Pack installer is provided for the Access System. For instance, if you install a Language Pack on the Identity Server, you must also install the Language Pack on WebPass. If you have the Access System, you must install Language Packs using the Access System installer.

Component installers call Language Pack installers silently and perform Language Pack installation at the same time. You must run appropriate Language Pack installers for each component you install. For example, use Identity System Language Pack installers when installing Identity Servers and WebPass instances and use Access System Language Pack installers for Access Server, WebGate, and Policy Manager components.

The overview that follows outlines the procedures needed to install Language Packs when you install Oracle Access Manager components. You may choose to install Language Packs independently, after Oracle Access Manager installation and setup, as described in [Chapter 12, "Installing Language Packs Independently"](#). The tasks that follow must be completed in either case.

To prepare to install Language Packs in concert with Oracle Access Manager

1. Before installation, move desired Language Pack installers into the same temporary directory as the component installer. For example:

```
Oracle_Access_Manager10_1_4_0_1_FR_Win32_LP_Identity_System.exe
Oracle_Access_Manager10_1_4_0_1_JA_Win32_LP_Identity_System.exe
Oracle_Access_Manager10_1_4_0_1_DE_Win32_LP_Identity_System.exe
```

2. **UNIX:** Ensure that each Language Pack has execute permissions before launching the main installer. For example:

```
chmod +x " Oracle_Access_Manager10_1_4_0_1_FR_sparc-s2_LP_Identity_System"
chmod +x " Oracle_Access_Manager10_1_4_0_1_JA_sparc-s2_LP_Identity_System"
chmod +x " Oracle_Access_Manager10_1_4_0_1_DE_sparc-s2_LP_Identity_System"

chmod +x " Oracle_Access_Manager10_1_4_0_1_FR_sparc-s2_LP_Access_System"
chmod +x " Oracle_Access_Manager10_1_4_0_1_JA_sparc-s2_LP_Access_System"
chmod +x " Oracle_Access_Manager10_1_4_0_1_DE_sparc-s2_LP_Access_System"
```

3. During component installation, select a Default Locale for the Administrator language and additional locales (which are listed based on the Language Pack installers detected).
4. After installation, you must enable all languages that you want to use, then configure Oracle Access Manager product applications to use the installed languages by entering display names (at the Object Class level) for attributes, tabs, and panels. For details, see the *Oracle Access Manager Identity and Common Administration Guide*.

During installation, the following processes are performed automatically:

- A `/langTag` folder is created in the `Component_install_dir/oblix/lang` directory for each installed language, as discussed in "[Language Directories](#)" on page 3-7.
- A language entry for each installed language is included under the configuration node in the LDAP directory as follows: `obid=langTag` and `configDN` (where `configDN` is the configuration DN in the directory).
- The `obnls.xml` configuration file in `\Component_install_dir\identity\access\oblix\config` is updated for each installed language (as shown) where German (de-de) and Japanese (ja-jp) Language Packs were installed.

```
<?xml version="1.0" encoding="UTF-8" ?>
- <ParamsCtlg xmlns="http://www.oblix.com" CtlgName="obnls.xml">
- <CompoundList xmlns="http://www.oblix.com" ListName="">
- <SimpleList>
  <NameValPair ParamName="default" Value="en-us" />
</SimpleList>
- <ValList xmlns="http://www.oblix.com"
ListName="languages"> <ValListMember Value="en-us" />
  <ValListMember Value="de-de" />
  <ValListMember Value="ja-jp" />
</ValList>
... </CompoundList>
</ParamsCtlg>
```

Note: For details about removing Language Packs, see "[Uninstalling Oracle Access Manager Components](#)" on page 2-28. If you need to troubleshoot language issues, see "[Language Issues](#)" on page E-15.

Directory Structure

Starting with the release of Release 6.5, a new directory structure was introduced to accommodate the addition of Language Packs that enable you to display static information to users in their native language. Oracle Access Manager provides a new directory named `\oblix\oracle\nlstrl` that is created for each component during with the automatic installation of the Oracle National Language Support Library.

OracleAccessManager\access

OracleAccessManager\identity

OracleAccessManager\webcomponent

The globalization files used internally by Oracle Access Manager are stored in the Identity Server installation directory under `\oblix\Oracle`.

Note: OracleAccessManager is the default name assigned to the top level directory. However, you can change this to be anything that you want during the installation process. In this guide, you will see path names that include `\OracleAccessManager` and references to the `Component_install_dir\`.

Language Directories

Oracle Access Manager installations include a directory named `\lang`, which includes a named subdirectory for each installed language. For example, `\lang\en-us` contains English-specific directories and files that are included with every installation.

For each installed Language Pack, a `\langtag` directory is created and named with the corresponding language tag. In the following example, the German and Japanese Language Packs were installed and appropriately named directories were created automatically:

```
Component_install_dir\identity\oblix\lang\en-us -- always present
```

```
Component_install_dir\identity\oblix\lang\de-de
```

```
Component_install_dir\identity\oblix\lang\ja-jp
```

and so on

In the preceding example, `Component_install_dir` represents the directory where the main component is installed and `identity|access` represents the appropriate suffix appended to the path during installation.

Note: Your installation will be English only unless Oracle-provided Language Packs were installed.

Each `\langTag` directory contains .xml message catalog files for various applications, which you may customize, as well as other .html files. For more information, see the *Oracle Access Manager Customization Guide*.

Removing Language Packs

You must remove (uninstall) each installed Language Pack individually using the appropriate file in the component's uninstall directory. Do not remove (uninstall) the Language Pack associated with the default Administrator language selected during installation.

For more information, see [Chapter 20, "Removing Oracle Access Manager"](#).

Part II

Identity System Installation and Setup

This part provides all the information you need to successfully install and setup the Identity System.

Part II contains the following chapters:

- [Chapter 4, "Installing the Identity Server"](#)
- [Chapter 5, "Installing WebPass"](#)
- [Chapter 6, "Setting Up the Identity System"](#)

Installing the Identity Server

The Identity System must be installed first, before installing the Access System. The Identity Server must be the first Oracle Access Manager component you install. This chapter covers the following topics:

- [About the Identity Server and Installation](#)
- [Identity Server Prerequisites Checklist](#)
- [Installing the Identity Server](#)
- [Tuning Oracle Internet Directory](#)

Upgrading an older release to 10g (10.1.4.0.1) is described in the *Oracle Access Manager Upgrade Guide*.

About the Identity Server and Installation

The Identity Server must be the first Oracle Access Manager component you install. The Identity Server provides applications through a Web-based interface and processes all requests related to user, group, and organization identification.

Each instance of the Identity Server receives requests through a WebPass plug-in installed on a Web server host. Each instance of the Identity Server reads and writes to your LDAP directory server across a network connection. For more information, see the *Oracle Access Manager Introduction*

Separate platform-specific installation packages are provided for the Identity Server in \win32 and \solaris subdirectories. Platform differences are noted in steps as needed. For example:

Windows: \Software\Win32\OracleAccessManager\...

Solaris: /Software/Solaris/OracleAccessManager/...

Note: If you intend to reuse a Identity Server instance name, see ["Recycling an Identity Server Instance Name"](#) on page 20-4.

The installation process follows the same sequence regardless of the operating system and whether you choose GUI mode or Console mode.

During installation, the transport security mode you choose will impact the scope of communication details you will be asked for in a later procedure. Also, you will be asked if this is the first Identity Server being installed for the directory server. Your response will determine the scope of activities in later procedures. Any caveats are

identified and may be skipped when they do not apply to your environment. For example:

Information is saved at various points during the installation. Should an error be detected in the information you supply, you will be offered the opportunity to restate information or complete a sequence again. After information is saved, you cannot return and restate information.

- **Simple:** Complete step 4.
- **Certificate:** Continue with step 5.

Two procedures are provided to guide you as you specify directory server details:

- One procedure walks you through installing the first Identity Server for the directory server.
- A second procedure walks you through specifying details for additional Identity Servers installed on a Windows system. When you install multiple Identity Servers on a Unix system, no additional directory server details are needed.

A default directory profile is created for this Identity Server based on the information you supply. This profile will be available after you setup the Identity System, as described in [Chapter 6, "Setting Up the Identity System"](#).

If you cancel the installation before completing all procedures and after being informed that the Identity Server is being installed, you must uninstall the Identity Server as described in ["Uninstalling Oracle Access Manager Components"](#) on page 2-28.

For more information, see:

- [The Identity Server and the Software Developer Kit](#)
- [About Installing Multiple Identity Servers](#)
- [Adding a New Identity Server to an Upgraded Environment](#)

The Identity Server and the Software Developer Kit

Certain functions in the Identity System require the Oracle Access Manager Software Developer Kit (SDK). By default, the SDK is installed in a subdirectory under `\IdentityServer_install_dir\identity`. Following Identity System set up, you must manually configure the SDK for the Identity System to enable required functions, as described in the *Oracle Access Manager Identity and Common Administration Guide*.

See the *Oracle Access Manager Developer Guide* for details about installing the SDK to construct simple AccessGate servlets or applications for each of the supported development platforms.

About Installing Multiple Identity Servers

You may want to install multiple Identity Servers, all associated with the same directory server.

Task overview: Installing additional Identity Servers

1. Install your first Identity Server, as explained in this chapter.
2. Install a WebPass, as explained in [Chapter 5, "Installing WebPass"](#) on page 5-1
3. Set up the first Identity Server in the Identity System, as explained in [Chapter 6, "Setting Up the Identity System"](#) on page 6-1.

4. Add a new Identity Server instance in the Identity System Console, as described in the *Oracle Access Manager Identity and Common Administration Guide*.
5. Associate the new Identity Server instance with a WebPass and specify the priority as Primary, as described in the *Oracle Access Manager Identity and Common Administration Guide*.
6. Modify the WebPass instance to set the maximum connections to the appropriate number to communicate with all primary Identity Servers, as described in the *Oracle Access Manager Identity and Common Administration Guide*.

You must wait at least one minute before proceeding to Step 7 to ensure that the WebPass configuration file, `webpass.xml`, is updated with the new instance information. Otherwise, the WebPass instance may not receive the new information and cannot connect to the new Identity Server instance.

7. Wait at least one minute before stopping all installed Identity Servers.
8. Install the new Identity Server and indicate that this is not the first Identity Server for this directory server.

You do not need to update the schema again.

9. Set up the new Identity Server, as explained in "[Setting Up Other Identity Server Instances](#)" on page 6-12.
10. Configure this Identity Server as a failover server, if desired, as explained in the *Oracle Access Manager Deployment Guide*.

Adding a New Identity Server to an Upgraded Environment

Starting with 10g (10.1.4.0.1), the Identity Server uses UTF-8 encoding and plug-in data will contain UTF-8 data. Earlier plug-ins send and receive data in Latin-1 encoding.

Backward compatibility between an upgraded Identity Server and earlier Identity Event plug-ins is automatic when you upgrade an earlier Identity Server to 10g (10.1.4.0.1). In this case, a new flag (`encoding`) is added to the `oblixpppcatalog.lst` file automatically to ensure backward compatibility with earlier plug-ins. A backward-compatible Identity Server continues to send data to earlier plug-ins in Latin-1 encoding. The format of this is as follows:

```
actionName;exectype;netpointparam1,...;path;execparam,...;apiVersion;encoding;
```

When you add a new 10g (10.1.4.0.1) Identity Server to an upgraded environment, you must manually set the `encoding` flag in the Identity Server `oblixpppcatalog.lst` to enable communication with earlier plug-ins and interfaces that need backward compatibility for Latin-1 data. For backward compatibility with Latin-1 data you must set the `encoding` flag to `Latin-1`. As shown above, this must follow the `ApiVersion` flag, which specifies the version of the Event API used by the event handler. If the `ApiVersion` parameter is set to `preNP60`, then Latin-1 encoding is assumed by default. If no `ApiVersion` flag is set, you must include an additional semi-colon before the `Latin-1` flag to indicate that there is no value for `ApiVersion`. See the example in the following procedure to see how this is done.

To add a new Identity Server to an upgraded environment

1. Upgrade the environment as described in the *Oracle Access Manager Upgrade Guide*.
2. Perform activities in "[About Installing Multiple Identity Servers](#)" on page 4-2.

3. Locate and open the new Identity Server oblixpppcatalog.lst file in *IdentityServer_install_dir\identity\oblix\apps\common\bin\oblixpppcatalog.lst*.
4. Set encoding to Latin-1 after the `ApiVersion` flag (if there is one) to provide backward compatibility for Latin-1 data. For example:

From:

```
userservcenter_view_pre;lib;;;..\..\..\unsupported\ppp\ppp_dll\
ppp_dll.dll;Publisher_USC_PreProcessingTest_PPP_Automation;
```

To:

```
userservcenter_view_pre;lib;;;..\..\..\unsupported\ppp\ppp_dll\
ppp_dll.dll;Publisher_USC_PreProcessingTest_PPP_Automation;;Latin-1
```

5. Repeat as needed for entries in this file.
6. Save the file.
7. Restart the Identity Server service.
8. Repeat for each new Identity Server in an upgraded environment as long as backward compatibility is needed.

Note: When all plug-ins and customizations have been successfully upgraded and backward compatibility is no longer needed, Oracle recommends that you manually reset the `encoding` flag in all Identity Server `oblixpppcatalog.lst` files.

Identity Server Prerequisites Checklist

Before you begin installing the Identity Server, check the tasks in [Table 4–1](#) to ensure they have been completed. Failure to complete prerequisites may adversely affect your Oracle Access Manager installation.

Table 4–1 Identity Server Installation Prerequisites Checklist

Checklist	Identity Server Installation Prerequisites
	Review and complete all prerequisites and requirements that apply to your environment, as described in Part I, "Installation Planning and Prerequisites"

Installing the Identity Server

Refer to your completed installation preparation worksheets as you install the Identity Server. The installation task is divided into the following procedures:

Task overview: Installing an Identity Server

1. Start the installation as described in ["Starting the Installation"](#) on page 4-5.
2. Continue by ["Installing the Identity Server"](#) on page 4-6.
3. Continue with ["Specifying a Transport Security Mode"](#) on page 4-7.
4. Identify the Identity Server, as described in ["Specifying Identity Server Configuration Details"](#) on page 4-7.
5. Define communication details, as described in ["Defining Communication Details"](#) on page 4-8.

6. Define directory server details, as described in ["Defining Directory Server Details"](#) on page 4-9.
7. Conclude with ["Finishing the Identity Server Installation"](#) on page 4-13.

Starting the Installation

You can start the installer in either GUI or console mode, as described in:

- [To start the installation in GUI mode](#)
- [To start the installation in Console mode](#)

Following the program launch, one set of procedures will be provided because the sequence is similar regardless of your platform.

Note: Skip any details that do not apply to your installation. If you are installing with Microsoft Active Directory, see [Appendix A, "Installing Oracle Access Manager with Active Directory"](#) on page A-1 before proceeding.

To start the installation in GUI mode

1. Log in as a user with administrator privileges.
2. Copy the Oracle Access Manager packages from the installation media into a temporary directory from which you can install the component and any Language Packs together, at the same time.
3. Locate and launch the Identity Server installer (including any Identity System Language Packs you want to install).

For example:

GUI Method, Windows: Oracle_Access_Manager10_1_4_0_1_Win32_Identity_Server.exe

The Welcome screen appears.

4. Dismiss the Welcome screen by clicking Next, then continue as described in ["Installing the Identity Server"](#) on page 4-6.

WARNING: Due to a problem with Installshield, passwords containing \$ or other special character sequences may not be interpreted properly. See ["GUI Method"](#) on page 1-8.

To start the installation in Console mode

1. Log in as a user with administrator privileges.
2. Copy the Oracle Access Manager packages from the installation media into a temporary directory from which you can install the component and any Language Packs.
3. Locate and launch the Identity Server installer (including any Identity System Language Packs you want to install).

For example:

Console Method, Solaris: ./ Oracle_Access_Manager10_1_4_0_1_sparc-s2_Identity_Server

The Welcome screen appears.

4. Dismiss the Welcome screen by clicking Next, then continue as described in ["Installing the Identity Server"](#) next.

Installing the Identity Server

During this sequence, you must specify the installation directory for your Identity Server. If you have a Language Pack in the same directory as the Identity Server installation package, you will be asked to choose a language.

To install the Identity Server

1. Respond to the question about administrator rights based upon your platform. For example.
 - **Windows:** If you are logged in with administrator rights, click Next (otherwise click Cancel, log in as a user with administrator privileges, then restart the installation).
 - **Unix:** Specify the username and group that the Identity Server will use, then click Next. Typically, the defaults are "nobody".

For HP-UX, the defaults are WWW (username) and others (group).

You are asked to specify the installation directory for the Identity Server. When you do this and click Next, the installation will begin and you will not be able to return to restate the name.

2. Accept the default directory by clicking Next (or change the destination, then click Next). For example:

```
\OracleAccessManager
```

You complete step 3 to choose a locale (base language) and other locales (languages) to install. Otherwise, skip to step 4.

3. **Language Pack:** Choose a Default Locale to use for the Administrator language and any other Locales to install, then click Next. For example:

```
English
French
Arabic
```

A summary identifies the installation directory and required disk space and asks you to make a note of this information for future reference.

4. Write the installation directory name in the preparation worksheet if you haven't already, then click Next to continue.

You are notified that the Identity Server is being installed, which may take several seconds. On Windows systems, the Microsoft Managed Interfaces are being configured.

Note: If a previous version of a Oracle Access Manager component or file is detected, you must specify a new installation directory path or uninstall the existing version.

You are now asked to specify the transport security mode. At this point you cannot return to restate previous details.

Specifying a Transport Security Mode

Transport security between all Identity System components (Identity Servers and WebPass instances) must match: either all open, all Simple mode, or all Cert. For more information, see "[Securing Oracle Access Manager Component Communications](#)" on page 2-11.

To specify a transport security mode

1. Choose the desired mode to use between the Identity Server and its clients: Open, Simple, or Cert.

If you chose either Simple or Cert, you will be asked for more information later.

2. Click Next.

You are now asked for Identity Server configuration details.

Specifying Identity Server Configuration Details

You are asked to identify this Identity Server by entering a unique name that will appear in the Identity System Console. The name you specify must differ from the name of any other Identity Server that accesses the same instance of your LDAP directory server, and cannot contain any blank spaces. You may use this name as a Windows Service name for the Identity Server.

In addition, you are asked to identify the DNS hostname where this Identity Server will be installed and the port number on which this Identity Server communicates with the WebPass (and by extension, with your Web server).

After you describe the Identity Server, you will be asked if this is the first Identity Server to be installed for the directory server. Your answer will determine the scope of activities now and during the setup process after WebPass installation. Selecting Yes indicates that this is the first Identity Server and you will be asked about directory server communication, schema updates, and directory server configuration details.

- Selecting Yes indicates that this is the first Identity Server. You will be asked about directory server communication, schema updates, and directory server configuration details.
- Selecting No indicates that an Identity Server has already been set up with this directory server. You will be asked only about directory server communication.
- On a Windows system, you will also be asked for Active Directory details.

To identify this Identity Server

1. Enter a unique name for this Identity Server that adheres to the preceding guidelines. For example:

IdentityServer_1014_6025

2. Enter the DNS hostname where this Identity Server will be installed. For example:

DNS_hostname.domain.com

3. Enter the port number on which this Identity Server communicates with its clients, then click Next. For example:

6025

4. Respond when asked if this is the first Identity Server to be installed for the directory server, then click Next.

For example, when you are installing the first Identity Server only, choose:

Yes

Regardless of your response to the question about this being the first Identity Server, you are now asked to specify communication details for the directory server and for the transport security mode you chose earlier.

Defining Communication Details

During this sequence, you are asked about securing communication between the Identity Server and your directory server. You may answer No during this installation and set up an SSL connection to the directory later as described in the *Oracle Access Manager Identity and Common Administration Guide*. In addition, you will be asked to specify Oracle Access Manager transport security details based on the information you supplied earlier.

Unix Systems: If you are installing on a Unix system using either Open or Simple transport security for the Identity Server, and this is not the first Identity Server, there are few security options and no directory server details required. In this case, complete the following steps, as needed, then skip to "[Finishing the Identity Server Installation](#)" on page 4-13.

To define communication details

1. Check the box beside the appropriate option if you have a certificate and want to enable SSL between the Identity Server and the directory server, then click Next. For example:

Directory Server ... user data is in SSL

Directory Server ... configuration data is in SSL

Note: Ensure you have a check mark beside each option if you have a certificate and want to enable SSL for each.

2. **SSL:** Specify the path to the root CA certificate, and click Next.

If you are installing on an Active Directory forest, enter the directory and file name of the retrieved CA certificate. See [Appendix A, "Installing Oracle Access Manager with Active Directory"](#) on page A-1.

3. Complete the transport security dialog according to the mode you chose earlier. For example:
 - **Open:** Skip to "[Defining Directory Server Details](#)" on page 4-9 unless you are installing on a Unix system and this is not the first Identity Server. In the later case, skip to "[Finishing the Identity Server Installation](#)" on page 4-13
 - **Simple:** Complete step 4.
 - **Certificate:** Continue with step 5
4. **Simple:** Enter and confirm the Pass Phrase to authenticate between the Identity Server and WebPass, then click Next and continue as follows:
 - If this is the first Identity Server or if you are installing an additional Identity Server on a Windows system, skip to "[Defining Directory Server Details](#)" on page 4-9

- If you are installing on a Unix system and this is not the first Identity Server, skip to "[Finishing the Identity Server Installation](#)" on page 4-13
- 5. **Certificate:** Indicate if you are requesting or installing a certificate, then click Next and continue.
- 6. **Certificate:** Enter and confirm the Pass Phrase to authenticate between the Identity Server and WebPass, then click Next.
 - If you are installing a certificate, skip to step 10
 - If you are requesting a certificate, continue with step 7
- 7. **Request Certificate:** Enter the requested information, then click Next and issue your request for a certificate to your CA. For example:
- 8. **Request Certificate:** Record certificate file locations, if they are displayed.
- 9. **Request Certificate:** Click Yes if your certificates are available and continue with step 10 (otherwise click No and skip to "[Defining Directory Server Details](#)" on page 4-9).

If you selected No, instructions are provided.

Note: You do not need a certificate in hand to finish the installation. However, the Identity System cannot be setup until the certificates are copied to `\IdentityServer_install_dir\identity\oblix\config` and the Identity Server is restarted. See the *Oracle Access Manager Identity and Common Administration Guide* for details.

10. **Install Certificate:** Specify the full paths to the following three files, then click Next:

`IdentityServer_install_dir\identity\oblix\config`

- Certificate file (ois_cert.pem)
- Key file (ois_key.pem) the installer may know where this is.
- Chain file (ois_chain.pem)

Note: When using certificates generated by a subordinate CA, the root CA's certificate must be present in the xxx_chain.pem along with the subordinate CA certificate. Both certificates must be present to ensure appropriate verification and successful Identity System setup.

The information you provided has been saved and you are asked if you want to update the schema. You cannot return to restate details.

11. Continue with "[Defining Directory Server Details](#)", next.

Defining Directory Server Details

What you see and do during this sequence depends in part upon how you responded when asked if this was the first Identity Server to be installed for this directory server. Refer to the following topics and choose the one for this installation:

- [Installing the First Identity Server](#)
- [Installing Additional Identity Servers on Windows](#)

Note: If you are installing on a Unix system and this is not the first Identity Server, skip to "[Finishing the Identity Server Installation](#)" on page 4-13

Installing the First Identity Server

If you indicated that this is the first Identity Server being installed for the directory server, you will be asked if you want to update your directory server with the Oracle Access Manager schema. This will include Oracle Access Manager-specific workflow definitions, attribute policies, tab and panel configurations, configuration attributes, and the like.

Schema Extension: Oracle recommends that you automatically extend the schema during installation of the first Identity Server. You update the schema only once. Either Yes response will result in questions about directory server type and specifications.

A No response on a Windows system will lead to questions for Active Directory. A No response on a Unix system will conclude the installation.

Separate Data Storage: If you plan to store user data separately from configuration data, see "[Data Storage Requirements](#)" on page 2-21 for more information.

By default, configuration and user data are presumed to be on the same directory server. With certain directory servers, such as Sun directory servers, data may be stored either together on the same directory server or on different directory servers of the same type.

Note: The Siemens DirX directory is not supported in 10g (10.1.4.0.1). Although the installation screen may still display DirX as a possible option.

To specify directory server details for the first Identity Server

1. Select the option that describes your environment. For example:
Configuration data will be in the user data directory
2. Select the appropriate schema update option for your environment, then click Next. For example:
Yes
 - If Yes, continue with step 3.
 - If No and you are installing on a Windows system, skip to "[Installing Additional Identity Servers on Windows](#)" on page 4-12
 - If No and you are installing on a Unix system, skip to "[Finishing the Identity Server Installation](#)" on page 4-13
3. Select your directory server type for automatic configuration, and click Next. For example:
Sun
You are asked for directory server configuration details. If you chose Active Directory for Windows 2003, you will be asked about dynamic auxiliary class support.
4. Specify your directory server configuration details, then click next. For example:

- **Host name:** The DNS hostname of the directory server machine
- **Port number:** On which the directory server listens (for SSL connections, provide the encrypted port)
- **Bind DN:** For the user data directory server

Note: The distinguished name you enter as the bind DN must have full permissions for the user and configuration branches of the directory information tree (DIT). Oracle Access Manager will access the directory server as this account. Examples are provided in [Table 4-2](#). Your directory server configuration may differ.

Table 4-2 Bind DN for Various Directory Servers

Directory Server	Bind DN
Active Directory or Active Directory on Windows Server 2003	cn=administrator,cn=users,<domain DN> Note: This information is required even if you are using ADSI with implicit bind. See Appendix A, "Installing Oracle Access Manager with Active Directory" on page A-1 and the <i>Oracle Access Manager Identity and Common Administration Guide</i> for more information.
ADAM	cn=administrator,o=domain.com The values represent: A Windows security principal user name. Domain name of the machine where ADAM is installed. Notes: The Master Administrator must be an ADAM user with administrative privileges, not a Windows Security Principal. See Appendix B, "Installing Oracle Access Manager with ADAM" on page B-1 for more information.
Data Anywhere (Oracle Virtual Directory)	cn=admin
IBM Directory Server	cn=root
NDS	cn=admin,o=nds
Oracle Internet Directory	cn=orcladmin Note: this is the default, unless you change the person object class during Identity System set up.
Sun Directory Server	cn=administrator Note: Oracle recommends that you do not use cn=Directory Manager. For details, see "Meeting Directory Server Requirements" on page 2-17.

- Password: The password for the user data directory server bind DN
5. Click Next and continue as indicated:
- If Active Directory 2003: You are asked about ADSI (for user data).
 - If configuration data is Separate: Repeat step 4 to specify details for the configuration data directory. The SSL sequence will repeat for this directory, if needed.

If the schema cannot be updated, you are offered the opportunity to run the sequence again and restate information. If you decline, you must manually update

the schema using the ldapmodify utility that ships with LDAP SDK or the following file:

```
\IdentityServer_install_dir\identity\oblix\tools\ldap_tools\ds_conf_update.exe
```

Note: All ldapmodify options can be viewed by using -h option. All ds_conf_update options can be viewed by using the --help option. Both utilities may be used with the Identity Server and Policy Manager installations.

For an example of the ldapmodify command, see "[Updating the Schema and Attributes Automatically Versus Manually](#)" on page 1-5. If you choose to update the schema with Oracle Access Manager configuration data using ds_conf_update, the command is:

```
ds_conf_update -h DS_hostname -p 389 -D cn=admin, o=my-company -w passwd  
-i C:\np\ois\identity -d 8 -e C:\errFile.txt -n 3
```

For more information on the -d option and directory server type input, see "[Silent Mode Parameters](#)" on page 15-6.

6. Continue with "[Finishing the Identity Server Installation](#)" on page 4-13

Installing Additional Identity Servers on Windows

In this sequence you are asked to supply information related to Active Directory. This sequence occurs only when:

- You indicated that this is not the first Identity Server in the installation
- You declined the automatic schema update on a Windows system

Note: Your responses determine the scope of this sequence. Whenever your sequence ends, skip to "[Finishing the Identity Server Installation](#)" on page 4-13

To specify Active Directory details on a Windows system

1. Select No when asked if you want to update the schema, then click Next.
2. Click Yes if you are using Active Directory with ADSI (or No if you are not), then click Next. For example:

Yes

If Yes, continue with step 3. If No, skip to "[Finishing the Identity Server Installation](#)" on page 4-13

3. Click Yes if the machine on which you are installing this Identity Server is in a separate Active Directory domain from the Oracle Access Manager data (otherwise, click No), then click Next. For example:

No

If No, continue with step 4. If Yes, skip to "[Finishing the Identity Server Installation](#)" on page 4-13.

4. Click Yes if you want to use implicit bind with the directory server (or No if you don't), then click Next. For example:

Yes

Finishing the Identity Server Installation

You complete the first step only if you are installing on Microsoft Windows. Otherwise, skip to step 2

To finish the installation

1. **Windows:** Specify a unique service name to identify your Identity Server in the Windows Services window, then click Next.

If the name is already registered as a Windows Service name on this host, you will be asked if you want to try again. In this case, you can either choose Yes to provide a unique name now or No to set this up manually using `\IdentityServer_install_dir\identity\oblix\apps\common\bin\config_ois.exe`.

ReadMe information appears.

2. Scroll through the ReadMe information.
3. Click Next to display an installation summary.

The installation summary provides the details that you specified during this installation and instructs you to start the Identity Server at the conclusion of this installation.

4. Write the details about this installation, if needed, then click Next.
5. Click Finish to complete the sequence.
6. Start the Identity Server service, as follows, to confirm that the Identity Server is installed and operating properly:

- **Windows:** Open the Services Window then locate and start the Identity Server service.

By default, the Identity Server (also known as the Oracle Identity Server (OIS)) starts manually, but you can set its startup type to Automatic. See the Microsoft Windows Help for details.

- **Unix:** Execute the following command:

```
/IdentityServer_install_dir/identity/oblix/apps/common/bin/start_ois_server
```

7. Proceed as appropriate for your environment: in
 - If you have installed the Identity Server with Oracle Internet Directory, complete activities in "[Tuning Oracle Internet Directory](#)", next.
 - Otherwise, install the first WebPass as described in [Chapter 5, "Installing WebPass"](#).

Tuning Oracle Internet Directory

When you have installed Oracle Access Manager with Oracle Internet Directory, you need to execute the `ldapmodify` command below to ensure that Oracle Internet Directory is properly tuned.

To tune Oracle Internet Directory for Oracle Access Manager

1. Run the following ldap command:

```
ldapmodify -D cn=orcladmin -w <adminPsswd> -h <OID_host> -p <OID_port> << eof
dn: cn=dsconfig, cn=configsets, cn=oracle internet directory
changetype: modify
add: orclinmemfiltprocess
```

```
orclinmemfiltprocess:  
(|(obuseraccountcontrol=activated)(!(obuseraccountcontrol=*))  
(|(!(obuseraccountcontrol=*)) (obuseraccountcontrol=activated))  
eof
```

2. When you install Oracle Access Manager with Oracle Internet Directory, additional steps are needed during Identity System setup, as described in [Chapter 6, "Setting Up the Identity System"](#).

Installing WebPass

The WebPass is second in the sequence of Oracle Access Manager components to install. This chapter explains how to install the WebPass and configure your Web server to work with it. For details, see:

- [About WebPass and Installation](#)
- [WebPass Prerequisites Checklist](#)
- [Installing the WebPass](#)
- [Manually Configuring Your Web Server](#)
- [Verifying WebPass Permissions on IIS](#)
- [Establishing Communication with the Identity Server](#)
- [Confirming WebPass Installation](#)

Upgrading an older version to 10g (10.1.4.0.1) is described in the *Oracle Access Manager Upgrade Guide*.

About WebPass and Installation

The WebPass is a Web server plug-in that shuttles information back and forth between the Web server and the Identity Server as described in the *Oracle Access Manager Introduction*. (A WebPass must also be installed with each Policy Manager as discussed in "[Identity System Guidelines](#)" on page 2-6.)

Installing a WebPass follows a similar sequence and includes a number of the same procedures as the Identity Server installation. However, the WebPass does not communicate with the directory server. Therefore, no directory server details are needed.

The WebPass does communicate with a Web server. Be sure to choose the proper package for your Web server and platform, as described in "[Confirming Platform Requirements](#)" on page 2-26. The Web server configuration needs to be updated either automatically during WebPass installation or manually after installation.

Important: The WebPass cannot reside in the same directory as the Identity Server (or Policy Manager). For example, if the Identity Server is installed in C:\OracleAccessManager\, consider installing the WebPass in C:\OracleAccessManager\WebComponent.

Task overview: Installing a WebPass

1. Install the WebPass and specify a unique identifier for WebPass (different than Identity Server identifier), as described in ["Installing the WebPass"](#) on page 5-2.
2. Conclude with the appropriate procedures for your installation. For example:
 - [Manually Configuring Your Web Server](#) (if you don't do this automatically during installation)
 - [Verifying WebPass Permissions on IIS](#)
 - [Confirming WebPass Installation](#)

The installation process is similar regardless of the installation method you choose and your operating system. Differences for specific operating systems and Web servers are noted within the installation procedures when appropriate. Again, any caveats are identified and may be skipped when they do not apply to your environment.

During WebPass installation on a Windows system, you will not be asked to specify a Windows Service name. Rather than starting and stopping a WebPass service, you will start and stop the WebPass Web server.

About Installing Multiple WebPass Instances

If you plan to install multiple WebPass instances, pay close attention to the following items:

- Oracle Access Manager supports one WebPass for each Web server instance. This means that each WebPass instance must have its own Web server instance.
- All WebPass instances must be installed with the same transport security mode as the Identity Server to which they are connecting.
- You must have at least one WebPass instance installed before you can perform the Identity Server setup described in [Chapter 6, "Setting Up the Identity System"](#) on page 6-1.
- After the first Identity Server is set up, you can install any number of WebPass instances. For each additional WebPass, you must add information about the new instance in the Identity System Console. For details and instructions, see the *Oracle Access Manager Identity and Common Administration Guide*.

WebPass Prerequisites Checklist

Before you begin installing the WebPass, check the tasks in Table to ensure they have been completed. Failure to complete prerequisites may adversely affect your Oracle Access Manager installation

Table 5–1 WebPass Installation Prerequisites Checklist

Checklist	WebPass Installation Prerequisites
	Review and complete all prerequisites and requirements that apply to your environment, as described in Part I, "Installation Planning and Prerequisites"
	Complete all activities in Chapter 4, "Installing the Identity Server" .

Installing the WebPass

Refer to your completed installation preparation worksheets as you install the WebPass. The procedures in this sequence cover both GUI and console method.

Following the program launch, one set of procedures will be provided because the sequence is similar.

The following procedures must be completed to install the WebPass:

Task overview: Installing a WebPass

1. Choosing the installation method and initiating the process as described in ["Starting the Installation"](#) on page 5-3
2. Choosing a transport security option for WebPass as discussed in ["Specifying a Transport Security Mode"](#) on page 5-4
3. Identifying WebPass configuration details as described in ["Specifying WebPass Configuration Details"](#) on page 5-4
4. Performing automatic Web server configuration updates as explained in ["Updating the WebPass Web Server Configuration"](#) on page 5-5
5. Completing the process as discussed in ["Finishing the WebPass Installation"](#) on page 5-6

Starting the Installation

Be sure to choose the appropriate package for your Web server and platform.

To start the WebPass installation

1. Log in as a user with administrator privileges.
2. Locate the WebPass installer (including any Identity System Language Packs you want to install) in the temporary directory you created.
3. Launch the WebPass installer for your preferred platform, installation method, and Web server. For example:
 - **GUI Method**
Windows: Oracle_Access_Manager10_1_4_0_1_Win32_API_WebPass.exe
 - **Console Method**
Solaris: ./ Oracle_Access_Manager10_1_4_0_1_sparc-s2_API_WebPass

The Welcome screen appears.
4. Dismiss the Welcome screen by clicking Next.
5. Respond to the question about administrator rights based upon your platform. For example:
6. Choose the installation destination, then click Next. For example:
 - \OracleAccessManager\Webcomponent
7. **Language Pack:** Choose a Default Locale to use for the Administrator language and any other Locales to install, then click Next.

A summary identifies the installation directory and required disk space and asks you to make a note of this information for future reference.
8. Write the installation directory name, if needed, then click Next to continue.

You are notified that the WebPass is being installed and kept informed about the status of the process, which may take several seconds. On Windows systems, the Microsoft Managed Interfaces are also being configured.

You are asked to specify a transport security mode to use between the WebPass and Identity Server. At this point, you cannot return to restate the installation directory.

Specifying a Transport Security Mode

Transport security between all Identity System components (Identity Servers and WebPass instances) must match: either all open, all Simple mode, or all Cert. For more information, see ["Securing Oracle Access Manager Component Communications"](#) on page 2-11.

To specify a transport security mode

1. Choose the same transport security mode for the WebPass as you did for the Identity Server.
2. Click Next.

When you specify Simple or Cert, you will be asked for additional information later. You are asked now for WebPass configuration details.

Specifying WebPass Configuration Details

Now, you are asked to enter a unique name to use for this WebPass, which will appear in the Identity System Console after setup.

Each WebPass must have a unique name that identifies it. The WebPass name you specify cannot contain any blank spaces and must uniquely identify this WebPass in the Identity System Console and LDAP directory.

You are also asked to identify the DNS hostname and port number of a Identity Server with which this WebPass should communicate. In addition, you may be asked to specify additional information about the transport security mode you selected when you selected either Simple or Certificate mode only.

To specify WebPass configuration details

1. Enter a unique name for this WebPass that adheres to the preceding guidelines. For example:

WebPass_1014_1_72

2. Enter the DNS hostname of the Identity Server with which this WebPass should communicate. For example:

Identity_DNS_hostname

3. Enter the port number of the Identity Server with which this WebPass should communicate, then click Next. For example:

Identity_port

4. Perform the following operations according to the transport security mode you chose earlier.

- **Open:** Skip to ["Updating the WebPass Web Server Configuration"](#) on page 5-5.
- **Simple:** Specify and confirm the Pass Phrase to authenticate between the Identity Server and WebPass, click Next, then continue with ["Updating the WebPass Web Server Configuration"](#) on page 5-5.
- **Certificate:** Continue with step 5.

5. **Certificate:** Indicate if you are requesting or installing a certificate, then click Next and continue as follows:
 - If you are requesting a certificate, enter information about your organization, click Next, issue the request to your CA, and continue with step 6.
 - If you are installing a certificate, skip to step 8.
6. **Request Certificate:** Record the location of the private key and certificate request files, if displayed, then click Next.
7. **Request Certificate:** Click Yes if your certificates are available (otherwise click No), then click Next and continue with step 8.

If certificates are not ready, complete the installation. When you receive the certificates, copy these to the `\WebPass_install_dir\identity\oblix\config` directory and restart the WebPass Web server.

Note: With an IIS Web server, consider using `net stop iisadmin` and `net start w3svc` to stop and start IIS after installing WebPass. This is a good way to ensure that the Metabase does not become corrupted.

8. **Install Certificate:** Specify the full paths to the requested files, then click Next and continue with "[Updating the WebPass Web Server Configuration](#)" on page 5-5.

You are notified that the WebPass is being configured, which may take a few seconds. The information has been saved and you may not return to previous screens to restate details.

You are now asked to update the WebPass Web server configuration.

Updating the WebPass Web Server Configuration

Your WebPass Web server must be configured with product-related configuration information to use the WebPass component. You can direct this update to occur either automatically or manually. Updating the Web server configuration:

- On Sun Web servers a configuration update involves updating the `obj.conf` and `magnus.conf` files.
- On IIS Web servers a configuration update involves updating the Web server directly by adding the ISAPI filter and creating extensions required by Oracle Access Manager.
- On Apache Web servers a configuration update involves updating the `httpd.conf` file.

Oracle recommends automatically updating your Web server configuration. However, instructions for manual configuration are included.

To automatically update your Web server configuration

1. Click Yes to automatically update your Web server, then click Next. For example:
 - **Most Web Servers:** Specify the absolute path of the directory containing the Web server configuration files.
 - **IIS Web Servers:** The process begins immediately and may take more than a minute.

A screen appears when the Web server configuration has been updated.

2. **Sun Web Servers:** Apply the changes in the Web server Administration console *before* you continue.
3. Stop the WebPass Web server instance, then stop the Identity Server service.
4. Start the Identity Server service, then start the WebPass Web server instance.

Note: With IIS, using `net stop iisadmin` and `net start w3svc` are good ways to stop and start the Web server after installing WebPass, to ensure that the Metabase does not become corrupted.

5. Click Next to dismiss the announcement, then continue with ["Finishing the WebPass Installation"](#) on page 5-6.

ReadMe information appears.

To manually update your Web server configuration

1. Click No when asked if you want to proceed with the automatic update, then click Next.

ReadMe information appears along with a new screen to assist you in manually setting up your Web server for Oracle Access Manager.

2. Return to the WebPass installation screen and click Next to finish the installation.
3. Complete ["Manually Configuring Your Web Server"](#) on page 5-6.

Finishing the WebPass Installation

The ReadMe information provides details about documentation and Oracle.

To finish the WebPass installation

1. Review the ReadMe information.
2. Click Next to complete the installation.
3. Continue with the following procedures, as needed:
 - [Manually Configuring Your Web Server](#) if you did not do this automatically during WebPass installation.
 - [Verifying WebPass Permissions on IIS](#)
 - [Establishing Communication with the Identity Server](#)
 - [Confirming WebPass Installation](#)
 - [Chapter 6, "Setting Up the Identity System"](#)

Manually Configuring Your Web Server

If you do not want the installation wizard to update your Web server configuration during WebPass installation, you must do it manually before you set up the Identity Server.

Note: You complete step 1 only if needed to display online instructions.

To configure your Web server for the WebPass

1. Launch your Web browser, and open the following file, if needed. For example:
`\WebPass_install_dir\identity\oblix\lang\langTag\docs\config.htm`
 where `\WebPass_install_dir` is the directory where you installed the WebPass and `langTag` is a language, `en-us`, for example.
2. Select the appropriate Web server interface configuration protocol from the table on the screen
3. Follow all instructions specific to your Web server type and:
 - Make a back up copy of any file that you are required to modify during Web server set up, so it is available if you need to start over.
 - Some setups launch a new browser window or require you to launch a Command window to input information, so ensure that you return to and complete all original setup instructions to enable your Web server to recognize the appropriate Oracle Access Manager files.

Note: If you accidentally close the window, you can open the `\WebPass_install_dir\identity\oblix\apps\common\docs\config.htm` file in a browser window and click the appropriate link again.

4. Continue with the appropriate task for your environment when you finish your Web server update. For example:
 - [Verifying WebPass Permissions on IIS](#)
 - [Confirming WebPass Installation](#)

Verifying WebPass Permissions on IIS

Once you have installed WebPass and updated the Web server configuration, you should ensure that the WebPass installation directory has the proper permissions to run correctly.

To verify the WebPass IIS Web server configuration

1. Locate the following directory:
`\WebPass_install_dir\identity\oblix\apps\webpass\bin`
2. Right click the `\bin` directory, then select Properties.
3. Select the Security tab and ensure that "Allow" for "Read" and "Write" rights are granted to user "SERVICE".

To verify when WebPass was set up in Simple or Cert mode

1. Locate `\WebPass_install_dir\identity\oblix\config\password.xml`.
2. Right click `password.xml`, then select Properties.
3. Select the Security tab and ensure that "Allow" for "Read" rights are granted to users:

"IUSR_<machine_name>"

"IWAM_<machine_name>"

"NETWORK SERVICE"

"IIS_WPG" (only for IIS 6.0)

Establishing Communication with the Identity Server

After installation, you must establish communications between WebPass and its Identity System when the Web server restarts using the following procedure.

To establish communications between WebPass and its Identity Server

1. Stop the WebPass Web server instance.
2. Stop then restart Identity Server service.
3. Start the WebPass Web server instance.

Confirming WebPass Installation

A good way to ensure that the WebPass is installed correctly is to complete the following procedure.

To confirm your WebPass installation

1. Make sure your Identity Server and WebPass Web server are running.
2. Navigate to the Identity System Console from your browser by specifying the following URL. For example:

```
http://hostname:port/identity/oblix
```

where *hostname* refers to machine that hosts the Web server; *port* refers to the HTTP port number of the WebPass Web server instance; */identity/oblix* connects to the Identity System Console.

The Identity System landing page should appear.

Note: Do not select any link on the Identity System landing page, because the system has not yet been set up. See [Chapter 6, "Setting Up the Identity System"](#).

Setting Up the Identity System

After you install the Identity Server and the WebPass, you must set up and configure the Identity System to work within your environment.

This chapter explains how to set up the Identity System and configure the required attributes. See the following topics:

- [About Setting Up the Identity System](#)
- [Identity System Setup Considerations](#)
- [Identity System Setup Prerequisites Checklist](#)
- [Setting up the Identity System](#)
- [Configuring Attributes Manually](#)
- [Setting Up Other Identity Server Instances](#)

Caution: During setup, specifications are saved whenever you click the Next button. If you leave setup and restart it later, you are returned to the same place.

About Setting Up the Identity System

After the first Identity Server and WebPass are installed, you need to setup the Identity System to complete associations and make the system functional. This process is completed using a Web browser.

During the setup process, you enter information about your directory server and configure required LDAP person and group object classes with Oracle Access Manager-specific information. This associates the Identity Server with the WebPass and extends the directory server schema to include the product branch and attributes. For example, the Identity System requires attributes assigned to the Full Name, Login, and Password semantic types for Person and Group object classes. For details, see "[About Oracle Access Manager Object Classes](#)" on page 6-7.

You must complete the entire setup process before you can use the Identity System applications. During setup, the information that you supply is saved as you progress from one page to the next. You may leave the setup process and restart it at any time. In this case, you will continue with the question that follows your last entry.

Some information may appear in the setup pages automatically based on the updated schema. If you did not automatically update your schema during Identity Server installation, a sequence of Schema Changes pages appear when you begin the setup. The pages are self explanatory and are not covered here.

Identity System Setup Considerations

The setup process described in this chapter applies only to the first Identity Server instance that connects to a given directory server. You may install multiple Identity Servers, all associated with the same directory server. The setup process for the second or successive Identity Server instances is described in ["About Installing Multiple Identity Servers"](#) on page 4-2.

Note: Be sure to review the following important considerations before starting to set up the Identity System

Following are important considerations that you need to be aware of before setting up the Identity System:

Certificates Generated by a Subordinate CA—The root CA's certificate must be present in the `ois_chain.pem` along with the subordinate CA certificate. Both certificates must be present to ensure appropriate verification for successful Identity System setup.

Multiple User Data Directories—If you intend to have more than one user data directory and searchbase, specify the main user data directory and searchbase during Identity System setup. Add one or more database profiles for the disjoint name spaces after setup is complete, as described in the *Oracle Access Manager Identity and Common Administration Guide*.

Active Directory—Read ["Installation and Setup Considerations for Active Directory"](#) on page A-7 before proceeding. When you are installing Oracle Access Manager within a Microsoft Active Directory forest, additional steps are needed during setup:

- Check the box beside Dynamic Auxiliary Object Classes, to enable this feature when asked.
- Ensure the semantic-type "Login" has been assigned to one attribute and that the people you select as a Master Administrator all have a value for the login attribute. For more information, see ["Configuring Master Administrators"](#) on page 6-8 and the *Oracle Access Manager Identity and Common Administration Guide*.

If you are using Active Directory with ADSI, you must:

- Complete the ADSI setup procedure before Identity System setup, as described in ["Setting Up ADSI \(Optional\)"](#) on page A-17.
- Check the Enable ADSI option when you specify the directory server type during setup to enable native integration with Active Directory and allow implicit failover and native password changes.

This creates a default directory profile and an associated database agent. With this configuration, the directory profile (db agent) is automatically assigned a name using a default Identity-machinename convention. You should modify this name to reflect your respective domain name to facilitate user authentication. The resulting directory profile enables the associated Identity Server to perform all operations with a primary domain controller in your Active Directory tree using an Implicit Bind.

Active Directory Application Mode—Read [Appendix B, "Installing Oracle Access Manager with ADAM"](#) before proceeding.

Data Anywhere (Oracle Virtual Directory Server)—Before you setup the Identity System for use with Data Anywhere, read [Chapter 10, "Setting Up Oracle Access Manager with Oracle Virtual Directory"](#) and complete activities as specified.

Oracle Internet Directory: For multiple realm installations and working with multiple directory searches, see the *Oracle Access Manager Identity and Common Administration Guide* for post-installation details. See the task overview that follows for additional information about ensuring full interaction between Oracle Access Manager and Oracle Internet Directory.

Note: Be sure that you have completed the Oracle Internet Directory tuning procedure in "[Tuning Oracle Internet Directory](#)" on page 4-13.

Task overview: Ensuring full interaction between Oracle Access Manager and Oracle Internet Directory

1. During Identity System setup, specify the user objectclass and group objectclass used by Oracle Internet Directory when prompted to specify these object classes.
2. During Identity System setup, configure the orclUserV2 objectclass and associate this auxiliary object class with the Employees tab in the User Manager. This enables you to manage orclUserV2 attributes in Oracle Internet Directory user entries through the Identity System Console.

You can complete the next step either during Identity System setup or later by later adding the Oracle Internet Directory searchbases using the Identity System Console.

3. Configure the Identity System to use both the user and group searchbases that Oracle Internet Directory uses to ensure that both Oracle Access Manager and Oracle Internet Directory can "see" every new user or group within a given Oracle Internet Directory instance, regardless of the application (Oracle Access Manager or Oracle Internet Directory) that created the user or object.
4. Group Objects to be Managed through Oracle Internet Directory: Attach the orclGroup auxiliary class to the group objects created through Oracle Access Manager as follows:
 - After Identity System setup, manually configure the auxiliary class (orclGroup) through the Identity System Console.
 - Configure a new group type for this auxiliary objectclass using the Oracle Access Manager Group Manager interface. See the *Oracle Access Manager Identity and Common Administration Guide* for details about configuring a group type.
 - Include at least one attribute from orclGroup in the workflow defined for creating group objects through the Identity System. This ensures that groups created through the Group Manager belong to the orclGroup objectclass and can be managed through Oracle Internet Directory Oracle Delegated Administration Services.
5. In Oracle Internet Directory, index all the attributes previously marked as "searchable" through Oracle Access Manager. This ensures that all attributes used in an LDAP filter can be searched by Oracle Internet Directory.

Complete the following activities to determine which attributes have been marked as searchable for the User Manager, Group Manager, and Org. Manager:

- From the Identity System Console, click the appropriate application's Configuration tab (User Manager Configuration, for example).
- On the application's configuration page, click Tabs then click a name in the Existing Tabs list (Employees, for example).

- On the View Tab page, click the View Search Attributes button.
 - Repeat the preceding steps for all applications (User Manager, Group Manager, and Org. Manager) and for each existing tab within the application.
6. Use the Oracle Directory Manager or the Oracle Internet Directory Self-Service Console to ensure that EMailAdminsGroup is not a member of UAdminsGroup. This allows Nested Group searches, while also preventing endlessly recursive searches that may cause the Access Server to fail.

Note: LDAP referrals and continuation references are not supported when Oracle Access Manager is used in conjunction with Oracle Internet Directory.

Identity System Setup Prerequisites Checklist

Before you begin installing the WebGate, confirm that you have completed the tasks in [Table 6-1](#). Failure to complete all prerequisites may adversely affect your installation.

Table 6-1 Identity System Setup Prerequisites Checklist

Checklist	Identity System Setup Prerequisites
	Review and complete all prerequisites and requirements that apply to your environment, as described in Part I, "Installation Planning and Prerequisites" .
	Complete all activities in Chapter 4, "Installing the Identity Server" .
	Complete all activities in Chapter 5, "Installing WebPass" .

Setting up the Identity System

Refer to your completed installation preparation worksheets as you complete Identity Server setup. The setup process has been divided into the following procedures to help guide you

Task overview: Setting up the Identity System

1. Initiate the process as described in ["Starting the Setup Process"](#) on page 6-4
2. Identify the directory server and data locations as discussed in ["Specifying Directory Server and Data Location Details"](#) on page 6-5
3. Define Person and Group object class details as explained in ["Specifying Object Class Details"](#) on page 6-6
4. Verify the changes to object classes as discussed in ["Confirming Object Class Changes"](#) on page 6-8
5. Identify the person to manage the entire system as described in ["Configuring Master Administrators"](#) on page 6-8
6. Finish setup as described in ["Completing Identity System Setup"](#) on page 6-9

Starting the Setup Process

You complete this procedure to start the Identity System setup.

Caution: If you just confirmed your WebPass installation and the Identity System Console setup page is currently available, skip to step 2.

To start setup

1. Navigate to the Identity System Console from your browser by specifying the following URL for your environment. For example:

```
http://hostname:port/identity/oblix
```

where *hostname* refers to machine that hosts the WebPass Web server; *port* refers to the HTTP port number of the WebPass Web server instance; */identity/oblix* connects to the Identity System Console.

2. Click the Identity System Console link.
The System Console setup page appears.
3. Click the Setup button.
 - If You Updated the Schema During Identity Server Installation—Skip to ["Specifying Directory Server and Data Location Details"](#) on page 6-5.
 - If You Did Not Update the Schema During Identity Server Installation—A Schema Changes page appears and you complete step 5. For additional information, see ["Updating the Schema and Attributes Automatically Versus Manually"](#) on page 1-5.
4. **Schema Changes**—Complete activities described on the Schema Changes page, if this appears, then continue.
5. Complete the procedures in following discussions and see [Chapter 19, "Important Notes"](#) for more information.

Specifying Directory Server and Data Location Details

You need to specify details about the directory server where user data and configuration data are stored.

Note: The Data Anywhere directory server option is available for only the user data directory server and integration with Oracle Virtual Directory Server (VDS). Before you setup the first Identity Server for use with Data Anywhere, read [Chapter 10, "Setting Up Oracle Access Manager with Oracle Virtual Directory"](#) on page 10-1 and complete activities as specified.

Typically, details about the user data are requested first, then details about configuration data. Information you supplied during the schema update usually appears on setup pages.

When user data and configuration data are stored separately, you repeat the sequence to specify directory server details.

To specify directory server details

1. Specify your user data directory server type. For example:

```
Sun
```

Next, you are asked for the location of the user data directory server. If you updated the schema during installation, most details will be filled in already.

2. Specify the user data directory server details based on your installation, then click Next. For example:
 - **Host**—The user data directory server DNS hostname
 - **Port Number**—The user data directory server port number
 - **Root DN**—The user data directory server bind DN
 - **Root Password**—Password for the bind DN
 - **Directory Server Security Mode**—Unsecured or SSL-enabled between the user data directory server and Identity Server
 - **Is Oracle data stored in this directory also?**—Yes (default) or No

Note: If user data is stored separately from configuration data, a similar page appears where you can enter information for the configuration data directory. However, that sequence is not repeated here.

A new page asks you to specify the location of user and configuration data.

3. Enter the configuration bind DN and user data searchbase to be used.

For example, when the data is stored in the same directory:

- **Configuration DN**—o=my-company,c=us
- **Searchbase**—o=my-company,c=us

Note: When user data and configuration data are stored separately, the configuration DN and searchbase must be unique. Also, you will see details about each directory to the right of each field.

4. Click Next and continue with "[Specifying Object Class Details](#)".

Specifying Object Class Details

The next sequence in the Identity System setup process asks for details about your Person and Group object classes. This discussion is divided into the topics:

- [About Oracle Access Manager Object Classes](#) provides an overview, which you may skip if you are already familiar with these concepts.
- [Specifying Person and Group Object Classes](#) provides the procedure to accomplish this task.

Note: For details about setting up the Identity System to operate with Oracle Internet Directory, see "[Task overview: Ensuring full interaction between Oracle Access Manager and Oracle Internet Directory](#)" on page 6-3.

About Oracle Access Manager Object Classes

In the directory server, Oracle Access Manager stores data as objects. Each object is composed of attributes and their values, which are displayed on Profile pages for each application in the Identity System. All objects are associated with an object class.

The Identity System includes its own object classes, which must be added to your directory server schema. These object classes begin with the prefix "ob" and contain functional information for the Identity System. You may configure additional object classes after you setup the Identity System.

Oracle Access Manager requires at least one Person object class and one Group object class, which must be setup before you can log in to Identity Systems applications. For more information, see "[About Person and Group Object Classes](#)" on page 2-26.

Note: To save time and avoid errors, Oracle recommends that you automatically configure both the person and Group object classes during Identity System setup.

Automatic configuration adds attributes to the Person and Group object classes. Specifically, the attributes for default display name, semantic type, and display type are added. Before you can log in to Identity System applications, attributes must be assigned to the following semantic types: Full Name, Login, and Password.

You may reconfigure attributes after setup, if needed, to define your own object classes and attributes and to incorporate unique requirements for your enterprise.

Specifying Person and Group Object Classes

You complete the following procedure to specify Person and Group object class details. If you do not use the recommended Auto configure option, you must do this manually, as described in "[Configuring Attributes Manually](#)" on page 6-10. Only partial pages are shown here to illustrate a completed setup page.

To specify Person and Group object class details

1. Enter your Person object class for the User Manager. For example:

Person Object Class—InetOrgPerson

As shown, the Auto configure objectclass feature is enabled by default to help streamline the configuration process. Later during this setup process, you can verify and accept, or change, the automatic configuration. You may disable this feature to manually configure the object class.

These instructions are based on automatic configuration of both Person and Group object classes.

2. Click Next to complete the Person object class configuration (or disable Auto configure object class, then click Next).

The Group object class page appears.

3. Enter your Group object class for the Group Manager, then click Next to complete the Group object class configuration. For example:

Group Object Class—GroupofUniqueNames

The next page that appears asks you to restart your Identity System. The time it takes for the Identity Server to automatically configure object classes may exceed

your Web browser's timeout. If your browser times out waiting for the Identity Server, wait a minute or two and click your browser's Refresh button to continue.

4. Stop the WebPass Web server instance.
5. Stop, then restart the Identity Server service.
6. Start the WebPass Web server instance.
7. Return to the Identity System setup window and click Next.

What you do after restarting the Identity System depends upon the update method you chose earlier in the setup. For example:

- If you chose to automatically configure the Person or Group object class, continue ["Confirming Object Class Changes"](#) on page 6-8.
- If you disabled automatic configuration of the Person or Group object class, continue with ["Configuring Attributes Manually"](#) on page 6-10.

Confirming Object Class Changes

You are presented with object class changes made automatically during this setup. Just review the changes for the specified object class, then click Yes to accept them. You may click No to launch the Configure Attributes function where you can make any corrections.

The following procedure presumes that you enabled automatic configuration for both the Person and Group object classes:

To confirm object class changes

1. Review the Person object class attribute list.
2. Click Yes to accept the changes (or No to launch the Configure Attributes function).
 - If Yes, continue with step 3.
 - If No, continue with ["Configuring Attributes Manually"](#) on page 6-10.
3. Review the Group object class attribute list, then click Yes to accept the changes (or click No to decline the changes) and continue as follows:
 - If Yes, continue with ["Configuring Master Administrators"](#) on page 6-8.
 - If No, continue with ["Configuring Attributes Manually"](#) on page 6-10.

Configuring Master Administrators

After you configure object classes and attributes, you are asked to identify one or more people as a Master Administrator for the entire installation and system.

Note: Be sure to select a person with the appropriate Person object class as the Master Administrator.

The Master Administrator has access to all configuration and management functions. This includes the rights to assign other administrators and perform all tasks other administrators can perform. For example, after the set up process a Master Administrator can assign one or more:

- Master Identity Administrators who have rights to configure the Identity System and assign individuals to be Delegated Identity Administrators.
- Master Access Administrators who have rights to configure the Access System, including WebGates, Access Servers, authentication parameters, and the initial set of policy domains. This includes the rights to assign individuals to the role of Delegated Access Administrators.

For more information, see the *Oracle Access Manager Identity and Common Administration Guide* and *Oracle Access Manager Access Administration Guide*.

To assign Master Administrators

1. On the Configure Administrators setup page, click the Select User button beside Administrators.

The Selector page appears providing two search criteria lists, an empty field where you enter at least three characters on which to search, and buttons to display results.

2. Locate the person or persons you want by choosing search criteria from the two drop-down lists on the top left (Full Name and That Contains, for example), then entering at least three characters in the empty field, and click the Go button.

The results of your search appear beneath the criteria. By default, 8 results are listed (as indicated in the field beside the Go button). You can use the Previous and Next buttons to navigate through the results, if needed.

Included on the left are control buttons that you can use to add everyone in the list (Add all), or add individuals (by choosing the Add button beside the desired name). When you choose one of these buttons, the name or names you add will appear on the right side of the window under "Selected".

3. Click the Add button beside the name of the person you want to assign as a Master Administrator.
4. Confirm that the name you added now appears on both the right side of the window under "Selected", and on the left side.

You may continue to add names as you did in step 3. Also, you can remove names from the "Selected" list using the DEL button beside the name or using the Delete All button.

5. Click Done to return to the original Configure Administrators page and confirm that the person or persons you wanted to add appear beside Administrators.
6. Click Next.

The Securing Data Directories page appears explaining things to do following Identity System setup.

Completing Identity System Setup

The Securing Data Directories page lists the Oracle Access Manager directories that you should protect to maintain the security of the Identity System and to both:

- Restrict access both from browsers and network users who access the directory through the file system. See the documentation for your Web server and operating system if you need instructions on how to protect directories.
- Protect the Identity System within a Oracle Access Manager policy domain. See the *Oracle Access Manager Access Administration Guide* for more information.

To complete the Identity System setup

1. Click Done to complete Identity System setup.

The login page for the Identity System appears. Your Identity System setup and minimum configuration are complete.

A default directory profile for this Identity Server is available in the Identity System Console.

2. Perform any of the following tasks.
 - a. Set up more than one Identity Server instance, as described under "[Setting Up Other Identity Server Instances](#)" on page 6-12.
 - b. Start install the first Access System component as described in "[Installing the Policy Manager](#)" on page 7-2.
 - c. Log in to the Identity System as a Master Administrator and complete any of the following tasks, as described in the *Oracle Access Manager Identity and Common Administration Guide*.

For example:

- View the directory server profile for this Identity Server by selecting Identity System Console, System Configuration, Directory Profiles, *link_to_this_profile*.
- Set up panels in the User Manager, Group Manager, Organization Manager.
- Set up object-based searchbases in the User Manager.
- Set up access controls in the User Manager, Group Manager, or Organization Manager.
- Create workflow definitions.
- Configure options such as the mail server and session settings.

Configuring Attributes Manually

The attribute configuration function helps you either manually complete the minimum configuration necessary to make the Identity System functional or fine-tune attributes that were configured automatically during setup. You can use the procedures here to modify attributes at any time after setup.

The Configure Attributes page appears in the following situations:

- You disabled Auto configure object class during Identity System setup, and restart your Identity Server and Web server.
- You enabled Auto configure object classes during Identity System setup, restart your Identity Server and Web server, then click No when asked if the configuration is correct.
- You navigated to the Modify Attributes page after setup by selecting Identity System Console, select Common Configuration, select Object Classes, then select *object_class_link*, and select Modify Attributes.

Novell Directory Server Considerations

Novell Directory Server (NDS) maps attribute and object class names from the native directory server to the LDAP layer of NDS. Some attributes or object classes will have

multiple mappings (aliases) in the LDAP layer. For example, the native NDS object class is Group, while the LDAP layer of NDS maps two aliases called GroupofNames and GroupofUniqueNames.

To ensure that Oracle Access Manager and the NDS work correctly

1. Confirm that the object class or attribute name you provide during configuration is the one that occurs ahead of the other mappings for the same object class or attribute.
2. Check the mapping order through consoleOne.

Configuring or Refining Attributes

Use these instructions to manually setup Person and Group object classes.

To define the minimum Person object class attribute set

1. On the Configure Attributes page, Attribute list, select or enter the following Person object class attribute details:
 - a. **Attribute**—The class attribute for your Person object class; often cn.
 - b. **Display Name**—Name or Full Name
 - c. **Semantic Type**—DN Prefix and Full Name
 - d. **Display Type**—Single Line Text
 - e. **Attribute Value(s)**—Single
2. Click Save, then click OK to close the confirmation message.
3. In the Attribute List, select or enter the following details to define the login ID attribute:
 - **Attribute**—The attribute that defines the login ID of your users; often the uid attribute.
 - **Display Name**—Such as Login ID
 - **Semantic Type**—Login
 - **Display Type**—Single Line Text
 - **Attribute Value(s)**—Single
4. Click Save, then click OK to close the confirmation message.
5. In the Attribute List, select or enter the following details to define the surname attribute:
 - **Attribute**—The attribute that defines the surname of your users; often sn.
 - **Display Name**—(such as Last Name)
 - **Display Type**—Single Line Text
 - **Attribute Value**—Single
 - Do not specify a Semantic Type
6. Click Save, the click OK to close the confirmation message.
7. In the Attribute List, select or enter the following details to define the user password attribute:

- **Attribute**—The attribute that defines the user password; often the password or userPassword attribute.
 - **Display Name**—Such as Password
 - **Display Type**—Password
 - **Attribute Value**—Password
 - **Attribute Value(s)**—Single
8. Click Save, then click OK to close the confirmation message.
 9. Click Next to proceed to the page where you configure the Group object class.

To specify the minimum set of Group object class attributes

1. In the Attribute List, select or enter the following details:
 - **Attribute**—The attribute that defines the Group name; often the cn attribute.
 - **Display Name**—Such as Group Name
 - **Semantic Type**—DN Prefix and Full Name
 - **Display Type**—Single Line Text
 - **Attribute Value(s)**—Single
2. Click Save, then click OK.
3. Continue with the following, as needed:
 - [Setting Up Other Identity Server Instances](#), next
 - [Installing the Policy Manager](#) as described in [Chapter 7, "Installing the Policy Manager"](#)
 - Configuring the Access Server SDK for the Identity System, as described in the *Oracle Access Manager Identity and Common Administration Guide*.

Certain functions in the Identity System require the Access Server SDK. By default, the Access Server SDK is installed in a subdirectory under `\IdentityServer_install_dir\identity`. After Identity System set up, you must manually configure the Access Server SDK for the Identity System to enable these functions.

Setting Up Other Identity Server Instances

[Table 6-2](#) lists the tasks that should be completed before you set up additional Identity Server instances.

Table 6-2 Preparing to Set Up Additional Identity Servers

Checklist	Setting Up Additional Identity Server Prerequisites
	Review and complete all prerequisites and requirements that apply to your environment, as described in Part I, "Installation Planning and Prerequisites"
	Install Identity Servers, as described in Part II, "Identity System Installation and Setup" on page 4-4
	Complete all activities in "Setting up the Identity System" on page 6-4
	Install additional Identity Servers, as described in "Installing the Identity Server" on page 4-4
	Stop all Identity Server services, if you haven't already.

Table 6–2 (Cont.) Preparing to Set Up Additional Identity Servers

Checklist	Setting Up Additional Identity Server Prerequisites
	Start the new Identity Server service only and with the procedure " To setup the Identity Server and associate it with a WebPass " on page 6-13.

Setting up additional Identity Servers involves only a subset of the original setup process.

To setup the Identity Server and associate it with a WebPass

1. Navigate to the Identity System Console, as usual.

`http://hostname:port/identity/oblix/`

The WebPass will attempt to connect to the original Identity Server. When it is unavailable, the WebPass will connect to the new Identity Server and launch the setup page.

2. Click Setup and follow the instructions to set up the Identity Server, and see "[Setting up the Identity System](#)" on page 6-4 for more information.
3. Restart the new Identity Server service when instructed to do so during setup.
4. Restart other Identity Server services.
5. Repeat as needed for each additional Identity Server.

Part III

Access System Installation and Setup

This part provides all the information you need to successfully install and setup the Access System.

Part III contains the following chapters:

- [Chapter 7, "Installing the Policy Manager"](#)
- [Chapter 8, "Installing the Access Server"](#)
- [Chapter 9, "Installing the WebGate"](#)

Installing the Policy Manager

After you install the Identity System, you can begin to install the Access System, which includes three components: the Policy Manager, the Access Server, and the WebGate. The Policy Manager is the first component that must be installed, as topics in this chapter describe:

- [About Policy Manager Installation and Setup](#)
- [Policy Manager Prerequisites Checklist](#)
- [Installing the Policy Manager](#)
- [Manually Configuring Your Web Server](#)
- [Verifying Policy Manager Permissions on IIS](#)
- [Setting Up the Policy Manager](#)
- [Confirming Policy Manager Setup](#)

Upgrading to 10g (10.1.4.0.1) is described in the *Oracle Access Manager Upgrade Guide*.

About Policy Manager Installation and Setup

The Policy Manager provides the login interface for the Access System, communicates with the directory server to write policy data, and communicates with the Access Server over the Oracle Access Protocol to update the Access Server when you make certain policy modifications. Master Access Administrators and Delegated Access Administrators use the Policy Manager to define resources to be protected and to group resources into policy domains. An overview is provided in the *Oracle Access Manager Introduction*.

The Policy Manager installation includes the Access System Console. Installing the Policy Manager combines elements of both Identity Server and WebPass installation. For instance, when installing the Policy manager you must identify where to store Oracle Access Manager policy data. A default Policy Manager directory profile is created and becomes available after setup. You also need to update your Web server configuration for the Policy Manager, as you did for the WebPass. Rather than starting and stopping an Policy Manager service, you will start and stop the Policy Manager Web server.

Again, separate Web server-specific installation packages are provided for the Policy Manager in platform-specific directories. The installation process is similar regardless of the installation method you choose and your operating system. Information is saved at certain points during installation. If you cancel the installation after being informed that the Policy Manager is being installed, you must uninstall the component, as

described in ["Upgrading from a Earlier Release of Oracle Access Manager"](#) on page 1-8.

After installation, you must complete the Policy Manager setup process before installing other Access System components. As with Identity System set up, your information is saved as you progress from one page to the next during setup. You may return to previous pages at any time and you may leave the setup process and restart it at any time. If you restart the setup process, you will continue with the question that follows your last saved entry.

For installation considerations, see ["Policy Manager Guidelines"](#) on page 2-6.

About Installing Multiple Policy Managers

Oracle recommends you install multiple Policy Managers for fault tolerance. To install multiple Policy Managers, you simply perform the installation and setup described in this chapter for each new Policy Manager instance.

A Policy Manager installed with an IIS Web server depends on the Registry to obtain the `\PolicyManager_install_dir`. To avoid a conflict in the Registry when you install two Policy Managers on a single machine, one with an IIS Web server and the other with a Sun Web server, you must install the Policy Managers as outlined in the following procedure.

To avoid a conflict with IIS and Sun Web server instances

1. Install the Policy Manager with the Sun Web server first.
2. Install the Policy Manager with the IIS Web server second.

Policy Manager Prerequisites Checklist

Before you begin installing the WebGate, confirm that you have completed the tasks in [Table 7-1](#) Failure to complete all prerequisites may adversely affect your Oracle Access Manager installation.

Table 7-1 Policy Manager Prerequisites Checklist

Checklist	Policy Manager Prerequisites
	Review and complete all prerequisites and requirements that apply to your environment, as described in Part I, "Installation Planning and Prerequisites"
	Complete all activities in Part II, "Identity System Installation and Setup"
	Install a WebPass for this Policy Manager, as described in Chapter 5, "Installing WebPass" and: <ul style="list-style-type: none"> ▪ Ensure that the WebPass is installed on the same Web server instance and at the same directory level as you will install the Policy Manager. ▪ Ensure that the Webpass has been configured to work with a particular Identity Server.

Installing the Policy Manager

You must install your Policy Manager in the same directory as your WebPass. If you specify a directory that does not include a WebPass, you will be asked if you want to install a WebPass or specify a different directory. If you choose to install a WebPass, this may launch automatically.

Refer to your completed installation preparation worksheets as you install the Policy Manager. The installation task has been divided into the following procedures:

Task overview: Installing the Policy Manager

1. Choose your installation method and start the process as described in "[Starting the Installation](#)" on page 7-3.
2. Identify the directory server and data location as explained in "[Defining a Directory Server Type and Policy Data Location](#)" on page 7-4.
3. Identify the transport security mode as described in "[Specifying a Transport Security Mode](#)" on page 7-6.
4. Update the Web server configuration as explained in "[Updating Your Policy Manager Web Server Configuration](#)" on page 7-7.
5. Complete installation as discussed in "[Finishing the Policy Manager Installation](#)" on page 7-8.

Starting the Installation

Be sure to choose the appropriate installation package for your Web server.

To start the Policy Manager installation

1. Log in as a user with administrator privileges.
2. Locate the Policy Manager installer (including any Access System Language Packs you want to install) in the temporary directory you created.
3. Launch the Policy Manager installer for your preferred platform, installation method, and Web server.

For example:

- **GUI Method**

`Oracle_Access_Manager10_1_4_0_1_Win32_API_Policy_Manager.exe`

- **Console Method**

`./ Oracle_Access_Manager10_1_4_0_1_sparc-s2_API_Policy_Manager`

The Welcome screen appears.

4. Dismiss the Welcome screen by clicking Next.
5. Respond to the question about administrator rights based upon your platform. For example:

You are asked to specify the installation directory for the Policy Manager.

6. Choose the installation destination, then click Next.

For example:

`\OracleAccessManager\WebComponent`

7. **Language Pack:** Choose a Default Locale and any other Locales to install, if this screen appears, then click Next.

A summary identifies the installation directory and required disk space and asks you to make a note of this information for future reference.

8. Write the installation directory name, if needed, then click Next.

You are notified that the Policy Manager is being installed, which may take several seconds. On Windows systems, you are informed that the Microsoft Managed Interfaces are being configured. Information is saved and you cannot return to previous screens to restate information.

The installation process is not complete. You are asked about the location for Oracle Access Manager policy data.

Defining a Directory Server Type and Policy Data Location

Oracle Access Manager policy data includes the rules that govern access to resources. You are asked to specify where policy data will be stored and if you want to add the Oracle Access Manager schema now or later. If your policy data is stored on the:

- **Same Directory Server:** Respond with No. When Oracle Access Manager policy data will be stored in the same directory server as configuration data or user data, an update is not needed because the schema was added during the Identity Server installation.
- **Separate Directory Server:** When Oracle Access Manager policy data will be stored in a separate directory server than either the configuration data or user data, the Oracle Access Manager schema must be added. You can direct this addition to occur either:
 - **Automatically:** Respond with Yes to automatically update the schema now.
 - **Manually:** Respond with No to update the schema manually later. For additional information, see "[Updating the Schema and Attributes Automatically Versus Manually](#)" on page 1-5.

To identify the location of policy data

1. Select your directory server type, then click Next

For example:

Sun

2. Respond to the question about where policy data will be stored:
 - **No:** Answer No if policy data will be stored with user and configuration data or if you want to manually update the schema later.
 - **Yes:** Answer Yes when policy data will be stored separately and you want to automatically update the schema now.

This information will be saved and you will not be allowed to return restate it.

3. Click Next and skip to the appropriate procedure for your environment:
 - [Continuing on Solaris Without Updating the Schema](#)
 - [Continuing on Windows Without Updating the Schema](#)
 - [Storing Policy Data Separately and Updating the Schema](#)

Continuing on Solaris Without Updating the Schema

During installation on a Solaris system, when policy data is stored with other Oracle Access Manager data you will be asked about the communication method for the existing directory server.

To specify directory server communication details

1. Respond to the question about securing directory server communication with SSL, then click Next.

Note: SSL is not supported on Policy Managers installed on Solaris with Sun Web servers.

2. **SSL:** Specify the path to the certificate, then click Next.
3. Continue with "[Specifying a Transport Security Mode](#)" on page 7-6.

Continuing on Windows Without Updating the Schema

During installation on a Windows system, when policy data is stored with other Oracle Access Manager data you will be asked about communication with the directory server.

Note: When this sequence concludes, you will be asked for transport security details. When this occurs, skip to "[Specifying a Transport Security Mode](#)" on page 7-6.

To specify details about the existing directory server

1. Click Yes if you are using Active Directory with ADSI (or No if you are not), then click Next. For example:

No

Next you are asked about the communication between the directory server and the Policy Manager for each of the three types of data: user, configuration, and policy data.

2. Check the box beside each type of data for which SSL communication with the directory server is needed, then click Next. For example:

Directory Server ... user data is in SSL

Directory Server ... configuration data is in SSL

Directory Server ... Policy data is in SSL

3. **SSL:** Specify the path to each certificate, then click Next.
4. Continue with "[Specifying a Transport Security Mode](#)" on page 7-6

Storing Policy Data Separately and Updating the Schema

When your policy data is stored separately you need to identify the type of directory server and other relevant details. For additional information, see "[Data Storage Requirements](#)" on page 2-21.

To specify directory server type and configuration details

1. Specify your directory server type for policy data stored separately, then click Next. For example:

Sun

2. Specify the following directory server configuration information, then click Next. For example:

- **Host name:** The DNS hostname of the policy data directory server machine
- **Port number:** The port on which the policy data directory server listens (for SSL connections, provide the encrypted port)
- **Bind DN:** The DN for the policy data directory server

Note: The distinguished name you enter as the bind DN must have full permissions for the policy data branch of the directory information tree (DIT). Oracle Access Manager will access the directory server as this account. Examples are provided in [Table 7-2](#). Your configuration may be different.

Table 7-2 Sample Bind DNs for Supported Directory Servers

Directory Server	Bind DN
Sun Directory Server 5.x	<i>cn=administrator</i> Note: Oracle recommends that you do not use <code>cn=Directory Manager</code> . For details, see " Meeting Directory Server Requirements " on page 2-17.

- **Password:** The password for the user data directory server bind DN
- **Update through SSL connection?** (Yes or No): If you are installing on Solaris with a Sun Web server, SSL is not supported and communication must be Open.

You complete step 3 when you indicated SSL.

3. **SSL only:** Enter the certificate path, then click Next.

If there is an error in the information you provide, the schema cannot be updated. You can either restate the configuration information during installation or manually update the schema later using the file: `\PolicyManager_install_dir\access\oblix\tools\ldap_tools\ds_conf_update`. See also, "[Updating the Schema and Attributes Automatically Versus Manually](#)" on page 1-5.

Next, you are asked about transport security.

Specifying a Transport Security Mode

You must specify a transport security mode for the Policy Manager and its WebPass. Transport security between all Access System components (Policy Managers, Access Servers, and associated WebGates) must match: either all open, all Simple mode, or all Cert.

To specify a transport security mode

1. Specify the transport security mode this Policy Manager will use to communicate with the rest of the Access System.
2. Click Next and perform the following operations according to the transport security mode you chose. For example:
 - **Open:** Skip to "[Updating Your Policy Manager Web Server Configuration](#)" on page 7-7.

- **Simple:** Specify and confirm the Access System Pass Phrase, click Next, then continue with "[Updating Your Policy Manager Web Server Configuration](#)" on page 7-7.
 - **Certificate:** Specify and confirm the certificate password (PEM phrase), click Next, and continue with step 3.
3. **Certificate:** Indicate if you are requesting or installing a certificate, complete the sequence, then continue with "[Updating Your Policy Manager Web Server Configuration](#)" on page 7-7.

Note: You cannot setup the Policy Manager until the certificates are copied to the `\PolicyManager_install_dir\access\oblix\config` directory, and the Policy Manager Web server is restarted. See the *Oracle Access Manager Access Administration Guide* for more information.

You are ready to update the Policy Manager Web server configuration.

Updating Your Policy Manager Web Server Configuration

Your Web server must be configured to work with the Policy Manager. You can direct this Web server configuration update to occur either automatically or manually.

Note: Oracle recommends automatically updating your Web server configuration. However, instructions for manual configuration are also provided.

To automatically update your Web server configuration

1. Click Yes to automatically update your Web server, then click Next.
 - **Most Web Servers:** Specify the absolute path of the directory containing the Web server configuration file, then click Next.
 - **IIS Web Servers:** The process begins immediately and may take more than a minute.

A screen announces that the Web server configuration has been updated.

2. **Sun Web Servers:** Apply the changes in the Web server Administration console before you continue.
3. Stop the Policy Manager Web server instance, stop and restart the Identity Server service, then start the Policy Manager Web server instance.

Note: With an IIS Web server, using `net stop iisadmin` and `net start w3svc` are good ways to stop and start the Web server, especially after installing the Policy Manager. The net commands help to ensure that the Metabase does not become corrupted following an installation.

4. Click Next to dismiss the announcement and continue with "[Finishing the Policy Manager Installation](#)" on page 7-8
ReadMe information appears.

To manually update your Web server configuration

1. Click No when asked if you want to proceed with the automatic update, then click Next.

A new window opens to assist you in manually setting up your Web server for Oracle Access Manager.

2. Return to the Policy Manager installation and click Next.
3. Refer to "[Manually Configuring Your Web Server](#)" on page 7-8 after you finish the installation and before you setup the Policy Manager.

Finishing the Policy Manager Installation

The ReadMe information provides details about documentation and contacting Oracle.

To finish the Policy Manager installation

1. Review the ReadMe information, then click Next.
You are informed that the Policy Manager has been successfully installed.
2. Click Finish to close the wizard.
3. Continue with the following procedures, as needed:
 - [Manually Configuring Your Web Server](#) if you did not do this automatically during installation
 - [Verifying Policy Manager Permissions on IIS](#)
 - [Setting Up the Policy Manager](#)

Manually Configuring Your Web Server

During Policy Manager installation you are asked if you want to automatically update your Web server installation. If you selected No, you must do this manually before you set up the Policy Manager.

Note: If the manual configuration process was launched during Policy Manager installation, you can skip step 1.

To manually configure your Web server for the Policy Manager

1. Launch your Web browser, and open the following file, if needed:
`\\PolicyManager_install_dir\access\oblix\lang\langTag\docs\config.htm`
where `\\PolicyManager_install_dir` is the directory where you installed the Policy Manager.
2. Select the appropriate supported Web server interface configuration protocol from the table on the screen.
3. Follow all instructions that appear, which are specific to each type of Web server, and note the following:
 - Make a back up copy of any file that you are required to modify during Web server set up, so it is available if you need to start over.
 - Some setups launch a new browser window or require you to launch a Command window to input information, so ensure that you return to and

complete all original setup instructions to enable your Web server to recognize the appropriate Oracle Access Manager files.

Note: If you accidentally closed the window, return to step1 and click the appropriate link again.

4. Continue with the following procedures:
 - [Verifying Policy Manager Permissions on IIS](#) if needed.
 - [Setting Up the Policy Manager](#)

Verifying Policy Manager Permissions on IIS

Whether you updated your configuration automatically during Policy Manager installation or manually, you can easily verify that the directory permissions are properly set for Oracle Access Manager.

To verify the Policy Manager IIS Web server configuration

1. Launch your Web browser, and open the following file, if needed. For example:
`\PolicyManager_install_dir\access\oblix\lang\langTag\docs\config.htm`
2. Select the appropriate Web server interface configuration protocol from the table on the screen, also shown under "[Manually Configuring Your Web Server](#)" on page 7-8.
3. Review the directory permissions and compare them to those set on the Policy Manager Web server.

Setting Up the Policy Manager

The Policy Manager must communicate with your directory server to write the new policies you create. The following procedures guide you as you make the connections that are necessary for this communication.

During setup, specifications are saved whenever you click the Next button. If you leave setup and restart it later, you are returned to the same place.

Task overview: Setting up the Policy Manager

1. Start the process, as described in "[Starting the Setup Process](#)" on page 7-9.
2. Define directory details, as described in "[Specifying Directory Server Details and Data Locations](#)" on page 7-10.
3. Set up authentication schemes, as described in "[Configuring Authentication Schemes](#)" on page 7-13.
4. Finish the setup process, as described in "[Completing Policy Manager Setup](#)" on page 7-14.

Starting the Setup Process

Policy Manager setup cannot be completed if the directory server used to store policy information is not loaded with the Oracle Access Manager schema.

You must manually update the policy data directory server schema before you begin the setup process, when the following conditions are both true:

- You plan to store policy data in a separate directory server
- You did not update this directory server schema during Identity System setup

If you need to do this, use the instructions in the following file:

```
\PolicyManager_install_dir\access\oblix\lang\langTag  
\ldap_schema_changes_directory_server.html
```

where *directory_server* in the path name refers to your specific directory server type and *langTag* refers to the language you are using, for example `\en-us`.

To start setting up the Policy Manager

1. Make sure your Web server is running.
2. Navigate to the Access System Console from your browser by specifying the URL of the WebPass instance that connects to the Policy Manager. For example:

```
http://hostname:port/access/oblix
```

where *hostname* refers to machine that hosts the Web server; *port* refers to the HTTP port number of the WebPass Web server instance; `/access/oblix` connects to the Access System Console.

You will see the main Access System page.

3. Click the Access System Console link.
You are informed that the application is not yet set up.
4. Click the Setup button.
The next page asks about the directory server type.
5. Continue with "[Specifying Directory Server Details and Data Locations](#)" on page 7-10 and see [Chapter 19, "Important Notes"](#) for additional details:

Specifying Directory Server Details and Data Locations

You need to specify details about the directory servers where user data, configuration data, and policy data are stored. You will be asked to provide information about the directory server for each type of data:

- User data
- Configuration data
- Policy data

Your directory server type affects the scope of activities. With Sun directory servers, you may store policy data on a different directory server than configuration or user data. All policy data must be stored together on the same directory server.

With Active Directory, a pure ADSI configuration is created and communication to the directory servers will be configured over ADSI when you select the ADSI option. If you want to enable Dynamic Auxiliary Object Classes (Windows 2003 only), see "[About Dynamically-Linked Auxiliary Classes](#)" on page A-3.

The information you see during setup will depend on your environment. In this example, user data, configuration data, and policy data are stored together on the same directory server. Your environment may be different.

To specify directory server details during Policy Manager setup

1. Select your user data directory server type, then click Next. For example:

Sun

Now you specify details for the user data directory server to help the Policy Manager locate your directory server and copy information into it.

2. Specify the user data directory server details based on your installation, then click Next. For example:
 - **Machine:** The user data directory server DNS hostname
 - **Port Number:** The user data directory server port number
 - **Root DN:** The user data directory server bind DN
 - **Root Password:** The password for the bind DN

Note: For Active Directory, a Domain Name field is included to fill in. With ADSI, a User-Principle-Name field is included where you enter the UserPrincipleName of the Root DN, such as :admin@mycompany.com.

You are asked about where the user data and configuration data are stored.

3. Select your configuration data directory server type, then click Next. For example:

Sun

Next you are informed that you can store your user data and configuration data either in the same directory or in separate directories and asked to choose a configuration for your deployment.

4. Choose the item that describes where your user data and configuration data are stored (together or separately), then click Next.
 - If the data is stored together, you are asked where policy data should be stored. In this case, continue with step 5.
 - If the data is stored separately, you are asked to specify details for the configuration data directory server before you continue.
5. Choose the item that describes where your policy data and configuration data are stored (together or separately), then click Next.
 - If the data is stored together, continue with step 6.
 - If the data is stored separately, you are asked to specify details for the policy data directory server before you continue.

The Setup Help button appears on the next page, which you can select to obtain additional information during the setup process. You are now asked to specify the location of the configuration DN, searchbase, and policy base.

Note: The configuration DN, searchbase, and policy base may be at the same level or at different levels of the directory tree. However, when the searchbase and the policy base are in separate directories, they must have unique DNs. That is, the searchbase cannot be `o=oblix,<Policy Base>` or `ou=oblix,<Policy Base>` if they are in separate directories. Similarly, the policy base and the configuration DN cannot be same if they are in separate directories.

6. Specify the appropriate information for your installation, then click Next. For example:
 - **Searchbase:** `o=my-company,c=us`
This *must* be the same searchbase you specified during Identity System configuration.
 - **Configuration DN:** `o=my-company,c=us`
This *must* be the same configuration DN you specified during Identity System configuration.
 - **Policy Base:** `o=my-company,c=us`
This node resides within the policy directory server. If this node does not already exist, create it manually.

You are now asked to specify the Person object class, which must match the one you specified during Identity System setup. For more information, see your preparation worksheets and ["To specify Person and Group object class details"](#) on page 6-7.

7. Enter the Person object class name, then click Next.

For example:

Person Object Class: `gensiteOrgPerson`

At this point, you are prompted to restart your Web server.

Note: If you are using IIS, be sure to follow additional on-screen instructions. Consider using `net stop iisadmin` and `net start w3svc` to stop and start IIS. The net commands help to ensure that the Metabase does not become corrupted.

8. Stop and restart your WebPass/Policy Manager Web server instance and the related Identity Server instance, as usual, then click Next to continue.

Now you are asked to specify the root directory for Oracle Access Manager policy domains.

Oracle recommends that you accept the default value `/"` unless you want to restrict the Master Administrator's ability to define and protect policy domains. For more information, see the *Oracle Access Manager Access Administration Guide*

9. Accept the default root directory for policy domains (or specify a new root directory), then click Next. For example:

Policy Domain Root /

The next page asks about configuring authentication schemes.

Configuring Authentication Schemes

During Policy Manager setup, the following two authentication schemes are configured automatically:

- **Oracle Access and Identity:** Used to protect Oracle Access Manager-related resources (URLs) and Oracle Access Manager-related resources (URLs) for Active Directory.
- **Anonymous:** Used to unprotect specific Oracle Access Manager URLs.

The Anonymous authentication method is especially useful because it provides for anonymous users. Users are allowed access to Oracle Access Manager-specific URLs you do not want protected with the Access System, such as Self Registration and Lost Password Management.

In addition, you can automatically configure a Basic and a Client Certificate authentication scheme based on the configuration information from your user directory:

- **Basic Over LDAP:** This built-in Web server challenge mechanism requires the user to enter their login ID and password. The credentials supplied are then compared to the users profile in the LDAP directory server.
- **Client Certificate:** This is a certificate-based user identification method. To use this method, a certificate must be installed on your browser and the Web server must be SSL-enabled.

The fields on the setup page for each scheme must be completed with information that is consistent with the Oracle Access Manager environment you are setting up. In most cases, appropriate defaults will appear on the setup page. You can modify these parameters later using the Access System Console.

You are also asked if you want to setup policies. If you accept this option, the following two policy domains are created automatically:

- Access Domain
- Identity Domain

Of course, you can decline automatic configuration and set up Basic over LDAP and Client Certificate authentication schemes in the Access System Console later. For more information about authentication schemes and policy domains, see the *Oracle Access Manager Access Administration Guide*.

To complete the authentication scheme sequence

1. Select Yes to initiate the automatic configuration sequence, or No to set up all authentication schemes yourself, then click Next.
 - If you selected Yes, continue with step 2.
 - Otherwise, skip to step 5.
2. Choose the authentication scheme or schemes you want to configure automatically, then click Next.
 - If you chose Basic Over LDAP, a page appears with its definition, which you can change now or later. In this case, continue with step 3.
 - If you chose only Client Certificate, skip to step 4.
3. Review and change Basic Over LDAP parameters, as needed, then click Next.
4. Review and change Client Certificate parameters, as needed, then click Next.

Next you are asked if you want to configure policies to protect Oracle Access Manager-related (URLs). The default is No.

5. Select Yes to configure the policies or No, then click Next. For example:

You must associate and install WebGates and Access Servers before you can use the policy domains. For more information about policy domains, see the *Oracle Access Manager Access Administration Guide*.

The next page provides instructions to complete the Policy Manager setup.

Completing Policy Manager Setup

The Securing Data Directories page lists the Oracle Access Manager directories that you must protect to maintain the security of the Identity System.

- You must restrict access both from browsers and from network users who access the directory through the file system. See the documentation for your Web server and operating system if you need instructions on how to protect directories.
- You can also protect the Access System within a policy domain. See the *Oracle Access Manager Access Administration Guide* for more information.

The second half of the page on-screen provides additional information about configuring Oracle Access Manager policy domains.

To complete the Policy Manager setup

1. Read all information on the page before you continue.

Note: If you are using Active Directory, see "[Installing and Setting Up the Access System](#)" on page A-19 for additional information before you continue.

2. Restart the Web server and Identity Server service in the following order:
 - a. Stop the WebPass Web server instance, which is the same as the Policy Manager.
 - b. Stop, then restart the Identity Server service for the WebPass.
 - c. Restart the WebPass/Policy Manager Web server instance.
3. After the Web server restarts, click Done.

The Policy Manager home page appears.
4. Review the following information; you may perform any of the following procedures:
 - a. [Confirming Policy Manager Setup](#) on page 7-14
 - b. [Chapter 8, "Installing the Access Server"](#)
 - c. Protect the directories as indicated on the Securing Directories page during setup, as described in the *Oracle Access Manager Access Administration Guide*.

Confirming Policy Manager Setup

An easy way to confirm your Policy Manager setup is to log in and review the authentication schemes automatically configured during the setup process. You may also begin to use the Access System Console to setup the Access Server instance and

define other administrators, as described in the *Oracle Access Manager Access Administration Guide*.

Note: If the Policy Manager home page is on your screen, you may skip step 2

To confirm Policy Manager setup

1. Navigate to the Access System Console from your browser. For example:

`http://hostname:port/access/oblix`

where *hostname* refers to machine that hosts the Web server; *port* refers to the HTTP port number of the WebPass Web server instance; `/access/oblix` connects to the Access System Console.

2. Select the Access System Console link.
3. Log in as a user with Master Administrator privileges.

The Access System Console appears.

You can click a tab in the top navigation bar to display a list of options, which will appear along the left side of the on-screen page. For example, complete step 4 to display a list of currently configured authentication schemes.

4. Select the Access System Configuration tab, then click Authentication Management when it appears in the left column.

A list of currently configured authentication schemes appears in the main body of the new page. If you did not choose to automatically configure schemes, none will be listed.

At this point, you can:

- Display configuration details for an authentication scheme by clicking the link that corresponds to the scheme.
- Add an Access Server instance by selecting Access Server Configuration in the side navigation bar (this is a prerequisite to installing an Access Server). For more information, see ["Installing the Access Server"](#) on page 8-5.
- Continue to explore the Access System Console and Policy Manager.

For example, you can define or modify policy domains as described in the *Oracle Access Manager Access Administration Guide*. The fact that Access Server or WebGate has not yet been installed has no impact on your ability to define them. Once these components are installed, the policy domains will be in affect.

- Log out by selecting Logout in the side navigation bar.

For more information, see the *Oracle Access Manager Access Administration Guide*

- Install the Access Server. For details, see [Chapter 8, "Installing the Access Server"](#).

Installing the Access Server

This chapter explains how to install the Access Server, which is the second Access System component you must install. See the following topics:

- [About the Access Server and Installation](#)
- [Access Server Prerequisites Checklist](#)
- [Creating an Access Server Instance in the System Console](#)
- [Installing the Access Server](#)

Upgrading an older version to 10g (10.1.4.0.1) is described in the *Oracle Access Manager Upgrade Guide*. For an overview of Oracle Access Manager components, see the Introduction to *Oracle Access Manager Introduction*.

About the Access Server and Installation

The Access Server is a stand-alone component that provides dynamic policy evaluation services for both Web-based and non-Web resources and applications. The Access Server receives requests from an access client, either a WebGate or a custom AccessGate; queries your LDAP directory for authentication, authorization, and auditing rules; and validates credentials, authorizes users, and manages user sessions for Oracle Access Manager. For more information, see *Oracle Access Manager Introduction*.

Before you install the Access Server you need to create an instance for it within the Access System Console.

Task overview: Adding an instance and installing the Access Server

1. Create an Access Server instance in the Access System Console, as described in "[Creating an Access Server Instance in the System Console](#)" on page 8-4.
2. Install the Access Server, as described in "[Installing the Access Server](#)" on page 8-5.
3. Install additional Access Servers, if needed, as described in "[About Installing Multiple Access Servers](#)" on page 8-2.
4. Add new Access Servers to an upgraded environment and be sure to manually set the appropriate parameter in the `globalparams.xml` file to ensure backward compatibility with older plugs-ins, as described in "[Adding New Access Servers to an Upgraded Environment](#)" on page 8-2.

Installing the Access Server is similar to installing the Identity Server. You will specify directory server details during this installation and a default directory profile is created for this Access Server. The default profile is available after you create an

Access Server instance; the completed profile is available after installation. There is no Web server involved in Access Server installation.

The following two installation packages are provided for the Access Server:

Windows: Oracle_Access_Manager10_1_4_0_1_win32_Access_Server

Unix: Oracle_Access_Manager10_1_4_0_1_sparc-s2_Access_Server

Again, platform-specific packages are available and installation is similar regardless of the platform or installation mode you choose. Information is saved at certain points. If you cancel the installation after being informed that the Access Server is being installed, you must uninstall the component as described in "[Upgrading from a Earlier Release of Oracle Access Manager](#)" on page 1-8. Any caveats are identified and may be skipped when they do not apply to your environment.

For more information, see "[Access Server Guidelines](#)" on page 2-7.

About Installing Multiple Access Servers

Oracle recommends you install multiple Access Servers for failover and load balancing. The procedures to do this are similar to those for installing a single Access Server.

Task overview: Installing multiple Access Servers

1. Create instances for each Access Server in the Access System Console, as described in "[Creating an Access Server Instance in the System Console](#)" on page 8-4

Note: Do not install multiple Access Servers in the same directory.

2. Install the Access Server, as described in "[Installing the Access Server](#)" on page 8-5, and specify a different installation directory for each Access Server.

You can replicate an existing installation using an options file, as described in "[Replicating Components](#)" on page 15-1.

3. Install one or more AccessGates/WebGates and assign the Access Servers to them as either primary or secondary Access Servers, as described in "[Installing the WebGate](#)" on page 9-1.

Refer to the *Oracle Access Manager Access Administration Guide* for complete instructions on how to enable these features.

Adding New Access Servers to an Upgraded Environment

Starting with 10g (10.1.4.0.1), the Access Server uses UTF-8 encoding and plug-in data will contain UTF-8 data. Earlier plug-ins send and receive data in Latin-1 encoding.

In releases before 10g (10.1.4.0.1), cookie encryption and decryption was handled by WebGate/AccessGate. However, cookie encryption and decryption is now handled by the Access Server.

When you upgrade an earlier Access Server to 10g (10.1.4.0.1), a new parameter "IsBackwardCompatible" value="true" is set in the Access Server globalparams.xml file automatically. This provides backward compatibility that enables the Access Server to continue to send (and receive) data to earlier custom authentication and authorization plug-ins in Latin-1 encoding (and earlier custom plug-ins will set data in Latin-1 encoding). In addition, the Access Server maintains

backward compatibility with earlier WebGates and custom AccessGates that continue to encrypt/decrypt cookies.

However, when you add a new 10g (10.1.4.0.1) Access Server to an upgraded environment, you must manually set "IsBackwardCompatible" Value="true" in the new Access Server globalparams.xml file to enable communication with earlier plug-ins and interfaces, as well as earlier WebGates and custom AccessGates.

To add a new Access Server to an upgraded environment

1. Upgrade the environment as described in the *Oracle Access Manager Upgrade Guide*.
2. Review details in "[About Installing Multiple Access Servers](#)" on page 8-2.
3. Perform activities in "[Creating an Access Server Instance in the System Console](#)" on page 8-4.
4. Add the new Access Server, as described in "[Installing the Access Server](#)" on page 8-5.
5. Locate and open the new Access Server globalparams.xml file in *AccessServer_install_dir\access\oblix\apps\common\bin\globalparams.xml*.
6. Set "IsBackwardCompatible" Value="true". For example:

```
<SimpleList
  <NameValPair
    ParamName="IsBackwardCompatible"
    Value="true">
  </NameValPair>
</SimpleList>
```

7. Save the file.
8. Restart the Access Server service.
9. Repeat for each new Access Server you add to an upgraded environment as long as your earlier plug-ins and WebGates require backward compatibility.

Note: When all plug-ins and WebGates have been successfully upgraded and backward compatibility is no longer needed, Oracle recommends that you manually set "IsBackwardCompatible" Value="false" in all Access Server globalparams.xml files.

For more information, see the discussion on the Access Server and backward compatibility in the *Oracle Access Manager Upgrade Guide*.

Access Server Prerequisites Checklist

Before you begin installing the Access Server, confirm that you have completed the tasks in [Table 8-1](#). Failure to complete all prerequisites may adversely affect your Oracle Access Manager installation.

Table 8-1 Access Server Prerequisites Checklist

Checklist	Access Server Prerequisites
	Review and complete all prerequisites and requirements that apply to your environment, as described in Part I, "Installation Planning and Prerequisites" .
	Complete all activities in Part II, "Identity System Installation and Setup" .

Table 8–1 (Cont.) Access Server Prerequisites Checklist

Checklist	Access Server Prerequisites
	Install, set up, and confirm that you have a working Policy Manager, as described in Chapter 7, "Installing the Policy Manager" .

Creating an Access Server Instance in the System Console

Before you can install the Access Server you must create an instance for it within the Policy Manager, Access System Console. This can be accomplished by either the Master Administrator or the Master Access Administrator if one has been defined.

The Access Server ID you specify when you create the instance must be unique and cannot contain spaces, a colon ":", the pound sign "#", or non-English keyboard characters. On Windows systems, this Access Server ID will be used as the Windows Service name, with "NetPoint AAA Server" as a prefix.

To create an Access Server instance

1. Log in to the Access System Console. For example:

```
http://hostname:port/access/oblix
```

where *hostname* refers to machine that hosts the Web server; *port* refers to the HTTP port number of the WebPass Web server instance; `/access/oblix` connects to the Access System Console.

The Access System main page appears.

2. Click the Access System Console link, then log in as a Master Administrator.

The Access System Console main page provides three tabs across the top and information about the functions in the center.

3. Click the Access System Configuration tab, then click Access Server Configuration when the side navigation bar appears.

If this is the first Access Server, the main page will inform you that no Access Servers were found in the directory server. Otherwise, Access Servers that have been added will be listed.

4. Click the Add button to display the Add a new Access Server page with some defaults.

You need only supply basic information to create the instance. After installation, you can complete additional configuration as discussed in the *Oracle Access Manager Access Administration Guide*. Online help is also provided.

5. Specify the following parameters for the Access Server you plan to install. For example:
 - **Name:** Descriptive name for the Access Server that is different from any others already in use on this directory server. Do not include spaces, a colon (":"), or the pound sign ("#") in the name.
 - **Hostname:** Name of the machine where the Access Server will be installed. The Access Server does not require a Web server instance.
 - **Port:** Port on which the Access Server will listen.
 - **Transport Security:** Transport security between all Access Servers and associated WebGates must match: either all open, all Simple mode, or all Cert.

6. Click Save.
The List All Access Servers page appears with a link to this instance.
7. Click the link to the Access Server instance, print the Details page for later reference, then click the Back button at the bottom of the page.
8. Repeat step 3 through step 7 for each additional Access Server instance you want to install.
9. Click Logout, close the browser window, and continue with "[Installing the Access Server](#)"

Installing the Access Server

Refer to your completed installation preparation worksheets as you install the Access Server. The following procedures must be completed for each Access Server:

Task overview: Installing the Access Server includes

1. Choosing the installation method and starting the process as described in "[Starting the Installation](#)" on page 8-5
2. Defining the transport security mode as discussed in "[Specifying a Transport Security Mode](#)" on page 8-6
3. Identifying directory server details as explained in "[Specifying Directory Server and Communication Details](#)" on page 8-6
4. Completing the process as discussed in "[Finishing the Access Server Installation](#)" on page 8-7

Starting the Installation

The Access Server installation sequence is similar to those you have performed for other Oracle Access Manager components.

Note: Do not install the Access Server in the same directory as the Policy Manager. Do not install multiple Access Servers in the same directory.

To start the Access Server installation

1. Log in as a user with Administrator privileges.
2. Locate the Access Server installer (including any Access System Language Packs you want to install).
3. Launch the Access Server installer for your preferred platform and installation method.

For example:

- **GUI Method**

Windows: Oracle_Access_Manager10_1_4_0_1_Win32_Access_Server.exe

- **Console Method**

Solaris: ./ Oracle_Access_Manager10_1_4_0_1_sparc-s2_Access_Server

4. Dismiss the Welcome screen by clicking Next

5. Respond to the question about administrator privileges based upon your platform. For example:
6. Identify the installation directory, then click Next.

For example:

`\OracleAccessManager`

7. **Language Pack:** Choose a Default Locale and any other Locales to install, then click Next.
8. Record the installation directory name, then click Next.

The Access Server is installed, which may take a few seconds. On Windows systems, a screen appears informing you that the Microsoft Managed Interfaces are being configured.

You are asked to specify the transport security mode.

Specifying a Transport Security Mode

Transport security between all Access System components (Policy Managers, Access Servers, and associated WebGates) must match: either all open, all Simple mode, or all Cert.

To specify a transport security mode

1. Choose a transport security mode: Open, Simple, or Cert.
2. Click Next.

Regardless of your transport security choice, you are asked to specify directory server details next.

Specifying Directory Server and Communication Details

During this sequence, you are asked to provide details about your environment and the Oracle Access Manager configuration and policy data directory servers. Oracle Access Manager adds additional configuration entries to the directory server.

To specify directory server details

1. Provide the information requested for the configuration data directory server, then click Next.
 - Open or SSL
 - **Host machine:** The DNS hostname of the directory server with configuration data
 - **Port number:** Port on which the directory server with configuration data listens (for SSL connections, provide the encrypted port)
 - **Root DN:** Bind DN for the directory server with configuration data
 - **Root Password:** Bind DN password for the directory server with configuration data
 - **Configuration Directory:** *Type of directory server with Oracle Access Manager configuration data.* For example:
 - Sun
2. **SSL Only:** Enter the path to the SSL certificate.

You need to identify where Oracle Access Manager policy data is stored: either with Oracle Access Manager configuration data or in a separate directory server. For more information, see "[Data Storage Requirements](#)" on page 2-21.

3. Identify where the Oracle Access Manager policy data is stored. For example:
Configuration Directory

Note: If your policy data is stored separately, you need to provide information for the policy data directory server. The configuration DN and policy base must be unique. See "[Data Storage Requirements](#)" on page 2-21.

You are now asked for the Access Server instance ID that you specified in the Access System Console and the configuration DN and policy base.

4. Enter the requested details, then click Next. For example:

Access Server ID: *Access_Server_1014_A*

Configuration DN: *o=my-company,c=us*

Policy Base: *o=my-company,c=us*

5. Perform the following operations according to the transport security mode you chose earlier:
 - **Open:** Skip to "[Finishing the Access Server Installation](#)" on page 8-7
 - **Simple:** Continue with step 6.
 - **Certificate:** Indicate whether you are requesting or installing a certificate, click Next, then continue with step 6.
6. Specify and confirm the Pass Phrase, click Yes (or No) when asked to store the password in a file, click Next.

Note: When you select No on Windows, you are prompted for the PEM phrase every time you start the Access Server. When you select No on Unix, you must use the -P option to pass the password whenever you launch the start_access_server script.

7. **Simple:** Skip to "[Finishing the Access Server Installation](#)" on page 8-7.
8. **Certificate:** Complete your certificate request and installation sequence, then continue with "[Finishing the Access Server Installation](#)" on page 8-7.

Note: If you requested certificates and they are not ready during this installation, the Access Server cannot be used until the you copy certificates to the `\AccessServer_install_dir\access\oblix\config` directory, and restart the Access Server.

You are informed that the Access Server is being configured, then ReadMe information appears.

Finishing the Access Server Installation

The ReadMe information provides details about documentation and contacting Oracle.

To finish the Access Server installation

1. Review the ReadMe information, then click Next to dismiss it.

You are informed that the installation is complete and that you need to start your Access Server.

2. Click Finish to close the wizard.

You need to start your Access Server, which confirms that the Access Server installation was successful and prepares for WebGate installation.

3. Start your Access Server to confirm that it is installed and operating properly:
 - **Windows:** The Access Server ID you specified in the Access System Console is used as the Service name (including an Oracle Access Manager prefix).
 - **Unix:** Go to your `/AccessServer_install_dir/access/oblix/apps/common/bin` directory and execute `./start_access_server`.

Note: For installations that do not use a password file, you must start the Access Server locally. Attempting this remotely (through a terminal emulator such as NetMeeting or Windows 2000 remote service restart) will fail.

What you do next depends on your environment:

- Add additional Access Servers as described in this chapter.
- Set up ADSI, as described in ["Setting Up ADSI \(Optional\)"](#) on page A-17.
- Install a WebGate, as described in [Chapter 9, "Installing the WebGate"](#).

Installing the WebGate

This chapter explains how to install WebGate and how to configure the WebGate to work with the Web server. This chapter covers the following topics:

- [About WebGate Installation](#)
- [WebGate Prerequisites Checklist](#)
- [Creating a WebGate Instance](#)
- [Associating a WebGate and Access Server](#)
- [Installing the WebGate](#)
- [Manually Configuring Your Web Server](#)
- [Completing IIS WebGate Installations](#)
- [Completing httpd.conf Updates](#)
- [Confirming WebGate Installation](#)

Upgrading to 10g (10.1.4.0.1) is described in the *Oracle Access Manager Upgrade Guide*. For an overview of Oracle Access Manager components, see the Introduction to *Oracle Access Manager Introduction*.

About WebGate Installation

A WebGate is a Web server plug-in that is shipped out-of-the-box with Oracle Access Manager. The WebGate intercepts HTTP requests from users for Web resources and forwards them to the Access Server for authentication and authorization. An AccessGate is an Oracle Access Manager access client that processes requests for Web and non-Web resources and is developed using the Software Developer Kit.

The terms AccessGate and WebGate may be used interchangeably. Before you can install a WebGate, you must associate it with an Access Server.

Task overview: Adding an instance and installing a WebGate

1. Create an instance, as described in "[Creating a WebGate Instance](#)" on page 9-2.
2. Associate the instance, as described in "[Associating a WebGate and Access Server](#)" on page 9-4.
3. Install the WebGate, as described in "[Installing the WebGate](#)" on page 9-5.
4. Complete the following procedures as needed:
 - [Manually Configuring Your Web Server](#)(if you did not do this automatically during installation)

- [Completing IIS WebGate Installations](#), if needed
 - [Completing httpd.conf Updates](#), if needed
5. Finish by "[Confirming WebGate Installation](#)" on page 9-13, which is a good practice.

Installing the WebGate is similar to installing the WebPass. There are no directory server details to specify and the WebGate Web server configuration must be updated.

Separate Web server-specific installation packages are provided for the WebGate on various platforms. Be sure you choose the one for your environment.

You must complete all procedures for a successful installation. Information is saved at certain points during the installation process. If you cancel the installation after being informed that the WebGate is being installed, you must uninstall the component, as described in "[Upgrading from a Earlier Release of Oracle Access Manager](#)" on page 1-8. Any caveats are identified and may be skipped when they do not apply to your environment.

For more information, see "[WebGate Guidelines](#)" on page 2-8.

About Installing Multiple WebGates

Oracle recommends you install multiple WebGates for failover and load balancing. Oracle recommends you use the cloning feature to facilitate installation on multiple systems, as described in [Chapter 15, "Replicating Components"](#) on page 15-1.

Installing multiple WebGates follows the same process as described in this chapter.

WebGate Prerequisites Checklist

Before you begin installing the WebGate, confirm that you have completed the tasks in [Table 9–1](#). Failure to complete all prerequisites may adversely affect your Oracle Access Manager installation.

Table 9–1 WebGate Prerequisites Checklist

Checklist	WebGate Prerequisites
	Review and complete all prerequisites and requirements that apply to your environment, as described in Part I, "Installation Planning and Prerequisites"
	Complete all activities in Part II, "Identity System Installation and Setup" .
	Install, set up, and confirm that you have a working Policy Manager, as described in Chapter 7, "Installing the Policy Manager" .
	Install and confirm that you have a working Access Server as described in Chapter 8, "Installing the Access Server"

Creating a WebGate Instance

Before you install an AccessGate or WebGate, you must define an instance of the new WebGate using the Access System Console. The WebGate ID you specify in the Access System Console must be unique and cannot contain spaces, a colon ":", the pound sign "#", or non-English keyboard characters.

To define a WebGate instance in the Access System Console

1. Navigate to the Access System Console. For example:

```
http://hostname:port/access/oblix
```

where *hostname* refers to machine that hosts the Web server; *port* refers to the HTTP port number of the WebPass Web server instance; `/access/oblix` connects to the Access System Console.

The Access System main page appears.

2. Click the Access System Console link, then log in as a Master Administrator.

The Access System Console main page appears.

3. Click Access System Configuration, then select Add New Access Gate.
4. Specify the following parameters for your WebGate (also known as an AccessGate) and click Save:
 - **AccessGate Name**—A unique, descriptive name for this WebGate/AccessGate. Do not include spaces in the name.
 - **Description**—This is optional; you can add it later. This is case insensitive; if you change capitalization of information in this field it will not be accepted unless you include new information.
 - **Hostname**—The name of the machine where the WebGate/AccessGate will be installed.
 - **Port**—The port the WebGate Web server is listening to. For more information, see "[WebGate Prerequisites Checklist](#)" on page 9-2.
 - **AccessGate Password and Re-type AccessGate Password**—This is an optional, unique password to verify and identify the component regardless of the transport security mode. This should differ for each WebGate instance.
 - **Transport Security**—The level of transport security between the Access Server and associated WebGates. The default value is Open. For details see, "[Securing Oracle Access Manager Component Communications](#)" on page 2-11. You can change the mode later, as described in the *Oracle Access Manager Identity and Common Administration Guide*.
 - **Preferred HTTP Host**—This parameter is now required before WebGate installation. It defines how the host name appears in all HTTP requests as users attempt to access the protected Web server. The host name in the HTTP request is translated into the value entered into this field, regardless of the way it was defined in a user's HTTP request.

The Preferred Host function prevents security holes that can be inadvertently created if a host's identifier is not included in the Host Identifiers list.

However, it cannot be used with virtual Web hosting. For virtual hosting, you must use the Host Identifiers feature. For more information, see the *Oracle Access Manager Access Administration Guide*.

Details for your WebGate appear and you are asked to associate an Access Server or Access Server cluster with this AccessGate (also known as a WebGate). Buttons at the bottom of this page help you modify the specifications, List Access Servers, or go back to the previous page.

5. Print this page, then click the Back button.
6. Continue with "[Associating a WebGate and Access Server](#)" on page 9-4.

Associating a WebGate and Access Server

Each Access Server functions as either a primary server or secondary server in association with a WebGate/AccessGate. If this is the only Access Server you are associating with this WebGate it should be a primary server. Multiple primary servers share incoming requests as they arrive. Secondary servers become active only if the primary servers go down. When you have multiple Access Servers, define at least one primary Access Server for this WebGate and define other Access Servers as either primary or secondary servers.

The number of connections identifies the number of Access Servers this WebGate can connect to, and the relative priority of the Access Servers for requests that come through the WebGate. For example, if you have two primary Access Servers and specify 2 connections for the first and 1 connection for the second, the first would receive two requests for every one the second receives. The default is 1. The number of requests the WebGate receives at one time is controlled by the Maximum Connections parameter in the AccessGate Configuration page.

Note: If you are continuing from step 5 above, you may skip step 1.

To assign an Access Server to the WebGate

1. Navigate to the Details for AccessGate page, if needed: Access System Console, Access System Configuration, AccessGate Configuration, *WebGate_Link*.

You may associate this WebGate with an individual Access Server or with a cluster of Access Servers. For information about clusters, see the *Oracle Access Manager Access Administration Guide*.

2. On the Details for AccessGate page, click the List Access Servers (or List Clusters) button at the bottom of the page.

A page appears saying that there are no primary or secondary Access Servers currently configured for this WebGate.

3. Click the Add button to advance to the Add a new Access Server page.
4. Select an Access Server from the Select Server list, specify a priority, and define the number of Access Servers (connections) to which this WebGate can connect.

For example:

Select server—*Your_Choice*

Select priority—Primary Server

Number of connections—1

If the Access Server you want is not listed, you may need to configure it. For details, see "[Creating an Access Server Instance in the System Console](#)" on page 8-4.

5. Click the Add button to complete the association.
A page appears listing the Access Server associated with this WebGate.
6. Click the link to display a summary and print this page for use later.
7. Repeat step 3 through step 6 to associate another WebGate and Access Server, if needed.
8. Logout and continue with "[Installing the WebGate](#)" on page 9-5.

Installing the WebGate

Once you have created a WebGate instance and associated it with an Access Server, you are ready to install the WebGate. Refer to your completed installation preparation worksheets as you complete the following procedures:

Task overview: Installing the WebGate includes

1. "Starting the Installation" on page 9-5
2. "Specifying a Transport Security Mode" on page 9-6
3. "Specifying WebGate Configuration Details" on page 9-6
4. "Updating the WebGate Web Server Configuration" on page 9-7
5. "Finishing the WebGate Installation" on page 9-8

Starting the Installation

The WebGate installation sequence is similar to those you have performed for other Oracle Access Manager components.

To start the installation

1. Log in as a user with Administrator privileges.
2. Locate the WebGate installer (including any Access System Language Packs you want to install) in the temporary directory you created.
3. Launch the WebGate installer for your preferred platform, installation mode, and Web server. For example:

GUI Method

Windows— Oracle_Access_Manager10_1_4_0_1_Win32_API_WebGate.exe

Console Method

Solaris—./ Oracle_Access_Manager10_1_4_0_1_sparc-s2_API_WebGate

Linux—./ Oracle_Access_Manager10_1_4_0_1_linux_API_WebGate

where *API* refers to the API used by your Web server. For example ISAPI for IIS Web servers.

On HP-UX and AIX systems, you can direct an installation to a directory with sufficient space using the `-is:tempdir` path parameter. The path must be an absolute path to a file system with sufficient space.

4. Dismiss the Welcome screen by clicking Next.
5. Respond to the question about administrator privileges based upon your platform. For example:
6. Specify the installation directory for the WebGate. For example:
`\OracleAccessManager\WebComponent\`
7. **Language Pack**—Choose a Default Locale and any other Locales to install, then click Next. For example:
8. Record the installation directory name in the preparation worksheet if you haven't already, then click Next to continue.

The WebGate is installed, which may take a few seconds. On Windows systems, a screen appears informing you that the Microsoft Managed Interfaces are being configured.

The installation process is not yet complete. You are asked to specify a transport security mode. At this point, you cannot go back to restate information.

Specifying a Transport Security Mode

Transport security between all Access System components (Policy Managers, Access Servers, and associated WebGates) must match: either all open, all Simple mode, or all Cert.

To specify a transport security mode

1. Choose Open, Simple, or Cert for the WebGate.
2. Click Next.

You are now asked to specify WebGate configuration details.

Specifying WebGate Configuration Details

It's a good idea to refer to the printed pages from your Access System Console as you complete the following procedure. During this sequence, you are asked to provide details about your WebGate and its associated Access Server.

To provide WebGate configuration details

1. Provide the information requested for the WebGate as specified in the Access System Console.
 - **WebGate ID**—The unique ID specified in the Access System Console
 - **WebGate password**—The password you defined in the Access System Console (if no password was entered, leave the field blank)
 - **Access Server ID**—The Access Server ID associated with this WebGate
 - **DNS hostname**—For the Access Server associated with this WebGate
 - **Port number**—On which the Access Server that listens for this WebGate

Note: If you specified the Simple transport security mode, you are also asked for the Global Network Protocol pass phrase. If you specified Cert mode, you are asked for the password phrase.

2. Click Next to continue.
3. Perform the following operations according to the transport security mode you chose earlier:
 - **Open or Simple**—Skip to ["Updating the WebGate Web Server Configuration"](#) on page 9-7.
 - **Certificate**—Complete your certificate sequence, then continue with ["Updating the WebGate Web Server Configuration"](#) on page 9-7.

If you requested certificates and they are not ready during this installation, be sure to copy them to the `\WebGate_install_dir\access\oblix\config` directory and restart the WebGate when they arrive.

WARNING: The certificate request for WebGate generates the certificate-request file `aaa_req.pem`. You need to send this WebGate certificate request to a root CA that is trusted by the AAA server. The root CA returns the WebGate certificates, which can then be installed either during or after WebGate installation.

Updating the WebGate Web Server Configuration

Your Web server must be configured to operate with the WebGate. Oracle recommends automatically updating your Web server configuration during installation. However, procedures for both automatic and manual updates are included.

To automatically update your Web server configuration

1. Click Yes to automatically update your Web server, then click Next.
 - **Most Web servers**—Specify the absolute path of the directory containing the Web server configuration file.
 - **IIS Web Servers**—The process begins immediately and may take more than a minute.

A screen announces that the Web server configuration has been updated.

2. **Sun Web Servers**—Be sure to apply the changes in the Web server Administration console before you continue.
3. **IIS Web Servers**—You may receive special instructions to perform before you continue.

Note: Setting various permissions for the `/access` directory is required for IIS WebGates only when you are installing on a filesystem that supports NTFS . The last installation panel provides instructions for manually setting various permissions that cannot be set on the FAT32 filesystem. In this case, these instructions may be ignored.

4. Stop and restart your Web server to enable configuration updates to take affect.

Note: With an IIS Web server, consider using `net stop iisadmin` and `net start w3svc` after installing the WebGate to help ensure that the Metabase does not become corrupted.

5. Click Next and continue with "[Finishing the WebGate Installation](#)" on page 9-8.

To manually update your Web server configuration

1. Click No when asked if you want to proceed with the automatic update, then click Next.

ReadMe information appears and a new screen also appears to assist you in manually setting up your Web server for the WebGate.

2. Return to the WebGate installation screen and click Next.
3. Continue with "[Manually Configuring Your Web Server](#)" on page 9-8.

Finishing the WebGate Installation

The ReadMe information provides details about documentation and Oracle.

To finish the WebGate installation

1. Review the ReadMe information, then click Next to dismiss it.
2. Click Finish to conclude the installation.
3. Restart your Web server now or at a later time.

With an IIS Web server, consider using `net stop iisadmin` and `net start w3svc` after installing the WebGate to help ensure that the Metabase does not become corrupted.

4. Continue with the appropriate procedures, as needed. For example:
 - [Manually Configuring Your Web Server](#) (if you did not do this automatically during installation)
 - [Installing postgate.dll on IIS Web Servers](#)
 - [Completing httpd.conf Updates](#)
5. Finish by "[Confirming WebGate Installation](#)" on page 9-13.

Manually Configuring Your Web Server

During WebGate installation you are asked if you want to automatically update your Web server installation. If you selected No, you must do this manually.

Note: If the manual configuration process was launched during WebGate installation, you can skip the step 1 in the following procedure.

To manually configure your Web server for the WebGate

1. Launch your Web browser, and open the following file, if needed. For example:
`\WebGate_install_dir\access\oblix\lang\langTag\docs\config.htm`
where `\WebGate_install_dir` is the directory where you installed the WebGate.
2. Select from the following supported Web servers.
3. Follow all instructions that appear, which are specific to each Web server type, and:
 - Make a back up copy of any file that you are required to modify during WebGate set up, so it is available if you need to start over.
 - Some setups launch a new browser window or require you to launch a Command window to input information, so ensure that you return to and complete all original setup instructions to enable your Web server to recognize the appropriate Oracle Access Manager files.

Note: If you accidentally closed the window, return to step 1 and click the appropriate link again.

4. Continue with one of the following, if needed:

- [Completing IIS WebGate Installations](#)
- [Completing httpd.conf Updates](#)

Completing IIS WebGate Installations

On IIS, if you are using client certificate authentication you must enable SSL on the IIS Web server hosting the WebGate before enabling client certificates for WebGate. You must also ensure that various filters are installed in a particular order. In addition, you may need to install the postgate.dll as an ISAPI filter.

Task overview: Completing IIS WebGate installations includes

1. ["Enabling SSL on the IIS Web Server"](#) on page 9-9
2. ["Ordering the ISAPI Filters"](#) on page 9-10
3. ["Installing postgate.dll on IIS Web Servers"](#) on page 9-10
4. ["Protecting a Web Site When the Default Site is Not Setup"](#) on page 9-12

Enabling SSL on the IIS Web Server

Use the following procedures as a guide, which reflects the sequence for IIS v5.

To enable SSL on the IIS Web server

1. Start the Internet Information Services console, if needed: Click Start, Programs, Administration Tools, Internet Information Services.
2. Expand the local computer to display your Web Sites.
3. Expand the Default Web Site or the appropriate Web site, then expand \access\oblix\apps\webgate\bin.
4. Right click cert_authn.dll and select Properties.
5. Select the File Security tab in the Properties panel.
6. In the Secure Communications sub-panel, click Edit.
7. In the Client Certificate Authentication sub-panel, click Accept Certificates and click OK.
8. Click OK in the cert_authn.dll Properties panel.

If you select client certificate authentication during setup, you must also add the cert_authn.dll as one of the ISAPI filters.

To add cert_authn.dll as an ISAPI filter

1. Start the Internet Information Services console, if needed: Click Start, Programs, Administration Tools, Internet Information Services.
2. Expand the local computer to display your Web Sites.
3. Right click the appropriate Web Site to display the Properties panel.
4. Click the ISAPI Filters tab, then click the Add button to display the Filter Properties panel.
5. Enter filter name "cert_authn".
6. Click the Browse button and navigate to the following directory:
`\WebGate_install_dir\access\oblix\apps\webgate\bin`

7. Select cert_authn.dll as the executable.
8. Click OK on the Filter Properties panel.
9. Click Apply on the ISAPI Filters panel.
10. Click OK.
11. Ensure the filters are listed in the correct order.

Ordering the ISAPI Filters

It is important to ensure that the WebGate ISAPI filters are included in the right order.

To order the WebGate ISAPI filters

1. Start the Internet Information Services console, if needed: Click Start, Programs, Administration Tools, Internet Information Services.
2. Expand the local computer to display your Web Sites.
3. Right-click the Web Site and select Properties.
4. Click Properties, select ISAPI filters.
5. Confirm the following .dll files appear.

For example:

```
cert_authn.dll
webgate.dll
oblixlock.dll
transfilter.dll
```

6. Add any missing filters, if needed, then select a filter name and use the up and down arrows to arrange the filter order as shown in step 5.

WARNING: Confirm that there is only one webgate.dll and one postgate.dll filter.

Installing postgate.dll on IIS Web Servers

Following WebGate installation, you may need to install the postgate.dll manually. POST data is required for pass through during a form login on the IIS Web server when using the WebGate extension method (where the WebGate is the action of the form). In other words, if a form authentication scheme on the IIS Web server is configured with the passthrough option, and the target of the login form requires the data posted by the form, the WebGate extension method (where the WebGate DLL is the action of the form) cannot be used. The WebGate filter method (where the action of the form is a protected URL that is not the WebGate DLL) must be used instead, and the postdate DLL must be installed and enabled.

POST data is used in an authorization decision that include rule parameters for the AzMan authorization plug-in. In this case, postgate.dll must be installed.

The following procedures presume that you are familiar with the IIS Web server commands. Two procedures are provided:

- [Setting Up IIS Web Server Isolation Mode](#)
- [Installing the Postgate ISAPI Filter](#)

Setting Up IIS Web Server Isolation Mode

On IIS 6 Web servers only, you must run the WWW service in IIS 5.0 isolation mode. This is required by the ISAPI postgate filter.

To set IIS 5.0 isolation on IIS 6 Web servers

1. Start the Internet Information Services console, if needed: Click Start, Programs, Administration Tools, Internet Information Services.
2. Expand the local computer to display your Web Sites.
3. Right-click the Web Site and select Properties.
4. Select the Service tab in the Web Site Properties window.
5. Check the box beside Run WWW service in IIS 5.0 Isolation Mode.
6. Click OK.

Installing the Postgate ISAPI Filter

If you perform multiple WebGate installations on one machine, multiple versions of the postgate.dll file may be created which may cause unusual Oracle Access Manager behavior.

There can only be one postgate.dll configured at the (top) Web Sites level of a machine. You may have multiple webgate.dlls configured at different levels from the top level Web Sites. However, they share the same postgate.dll. Install the filters in the following order:

- The ISAPI Webgate filter should be installed after the sspifitt filter and before any others.
- The postgate filter should be installed before the WebGate filter, only if needed.
- All other Oracle Access Manager filters can be installed at the end.

Note: Before installation (or after uninstallation) the filters must be removed manually. If multiple copies of a filter are installed, this means that they were not manually removed before installing the new filters.

The following procedures guide as you install and position the postgate ISAPI filter.

To install the postgate ISAPI filter

1. Start the Internet Information Services console, if needed: Click Start, Programs, Administration Tools, Internet Information Services.
2. Expand the local computer to display your Web Sites.
3. Right-click the Web Site and select Properties.
4. Select the ISAPI Filters tab in the Web Site Properties window.
5. Click the Add button to display the Filter Properties panel.
6. Enter the filter name "postgate".
7. Click the Browse button and navigate to the following directory:
`\WebGate_install_dir\access\oblix\apps\webgate\bin`
8. Select postgate.dll as the executable.

9. Click OK on the Filter Properties panel.
10. Click Apply on the ISAPI Filters panel.

To restart IIS and reposition the postgate ISAPI filter

1. Start the Internet Information Services console, if needed.
2. Right-click your local computer, then select All Tasks, select Restart IIS.
3. Select the ISAPI Filters tab on the Properties panel.
4. Select the postgate filter and move it above WebGate, using the up arrow.

For example:

```
postgate.dll  
webgate.dll  
oblixlock.dll
```

5. Restart IIS or proceed with ["Protecting a Web Site When the Default Site is Not Setup"](#) next.

Note: Consider using `net stop iisadmin` and `net start w3svc` to help ensure that the Metabase does not become corrupted.

Protecting a Web Site When the Default Site is Not Setup

When you install a WebGate on an IIS Web server that does not have the "Default Web Site" configured, the installer does not create "Virtual Directory -> access", which must be done manually using the following procedure:

To protect a Web site (not the default site)

1. Start the Internet Information Services console, if needed
2. Select the name of the Web site to protect.
3. Right-click the name of the Web site to protect and select New -> Virtual Directory in the menu.
4. Click Next.
5. Select Alias: access, then click Next.
6. Directory: Enter the full path to the /access directory, then click Next.
7. Select Read, Run Scripts, and Execute, then click Next.
8. Click Finish.
9. Restart IIS. For example:

```
Select Start, then Run  
Type net start w3svc  
Click OK
```

Completing httpd.conf Updates

You must complete the following procedure to update the Apache httpd.conf file after you finish the WebGate installation and automatic Web server updates conclude.

To update the WebGate section in httpd.conf

1. Locate the updated httpd.conf file on the machine hosting the WebGate.
2. Ensure the section that loads WebGate in the httpd.conf file appears as shown next (tailored for your environment, which will differ from the example). For example:

```

**** BEGIN WEBGATE SPECIFIC ****
# The path to this library may need to be changed to suit your installation
LoadFile
"/home/usr/sparc-s2/obdevsun1_wp_apache/identity/oblix/lib/libgcc_s.so.1"
LoadFile
"/home/usr/sparc-s2/obdevsun1_wp_apache/identity/oblix/lib/libstdc++.so.5"
<IfModule mod_ssl.c>
ObWebGateInstalldir
"/home/usr/sparc-s2/obdevsun1_wp_apache/identity"
ObWebGateMode PEER
Obwebgateload obWebgateModule
"/home/usr/sparc-s2/obdevsun1_wp_
apache/identity/oblix/apps/webgate/bin/webgatessl.so"
</IfModule>
<IfModule !mod_ssl.c>
ObWebGateInstalldir
"/home/usr/sparc-s2/obdevsun1_wp_apache/identity"
ObWebGateMode PEER
Obwebgateload obWebgateModule
"/home/usr/sparc-s2/obdevsun1_wp_
apache/identity/oblix/apps/webgate/bin/webgate.so"
</IfModule>
<Location /access/oblix/apps/webgate/bin/webgate.cgi>
SetHandler obwebgateerr
</Location>
<Location "/oberr.cgi">
SetHandler obwebgateerr
</Location>
<LocationMatch "/*">
AuthType Oblix
require valid-user
</LocationMatch>
**** END Oblix NetPoint Specific ****

```

3. Use the `chmod -r username:groupname directory/file` to change the User Name and Group Name of a directory or a file.

When you do this, you need to change the User and Group parameters in the httpd.conf file accordingly.

For more information, see:

- [Chapter 16, "Configuring the Apache v1.3 and Oracle HTTP Server Web Servers"](#)
- [Chapter 17, "Configuring Apache v2, IHS, and OHS Web Servers for Oracle Access Manager"](#)

Confirming WebGate Installation

After WebGate installation and Web server updates, you can enable WebGate diagnostics to confirm that your WebGate is running properly.

To enable WebGate diagnostics

1. Make sure your Identity Server, WebPass Web server, and Access Server are running.
2. Specify the following URL for WebGate diagnostics. For example:

Most Web Servers—`http(s)://hostname:port/access/oblix/apps/webgate/bin/webgate.cgi?progid=1`

IIS Web Servers—`http(s)://hostname:port/access/oblix/apps/webgate/bin/webgate.dll?progid=1`

where *hostname* refers to the name of the machine hosting the WebGate; *port* refers to the Web server instance port number.

The WebGate diagnostic page should appear. If it does not open, the WebGate is not functioning properly and should be uninstalled and reinstalled.

At this point, you are ready to:

- Configure Oracle Access Manager, as described in the *Oracle Access Manager Identity and Common Administration Guide* and *Oracle Access Manager Access Administration Guide*.
- Customize Oracle Access Manager, as described in the *Oracle Access Manager Customization Guide*.
- Integrate third-party products, as described in the *Oracle Access Manager Integration Guide*.

Part IV

Installing Optional Components

This part provides all the information you need to successfully install other optional components.

Part V contains the following chapters:

- [Chapter 10, "Setting Up Oracle Access Manager with Oracle Virtual Directory"](#)
- [Chapter 11, "Installing the SNMP Agent"](#)
- [Chapter 12, "Installing Language Packs Independently"](#)
- [Chapter 13, "About Installing Audit-to-Database Components"](#)
- [Chapter 14, "About the Software Developer Kit"](#)

Setting Up Oracle Access Manager with Oracle Virtual Directory

This chapter focuses on implementing Oracle Access Manager with the Oracle Virtual Directory to enable Data Anywhere. It includes the following topics:

- [About Oracle Access Manager Implementations with Oracle Virtual Directory](#)
- [Implementation Limitations](#)
- [Implementation Architecture](#)
- [About Schema Extension](#)
- [Implementation Scenarios and Limitations](#)
- [Implementation Requirements](#)
- [About the Implementation Process](#)
- [Preparing Your Environment](#)
- [Installing and Configuring Oracle Virtual Directory and Virtual Directory Manager](#)
- [Installing the First Identity Server](#)
- [Extending Directory Schemas](#)
- [Creating Mapping Files for Adapters](#)
- [Creating Data Store Adapters](#)
- [Customizing Adapters and Mapping Files](#)
- [Completing Identity System Installation and Setup](#)
- [Testing Your Implementation](#)
- [Reference Information](#)
- [Oracle Access Manager-Oracle Virtual Directory Implementation Templates](#)
- [Tips](#)
- [Troubleshooting Implementations with Oracle Virtual Directory](#)

The Oracle Access Manager-Oracle Virtual Directory implementation uses only a subset of Oracle Virtual Directory functions; therefore not all of the information provided in the Oracle Virtual Directory documentation applies to the Oracle Access Manager-specific configurations described in this chapter.

This chapter should be used in conjunction with the following manuals:

- *Oracle Virtual Directory and Virtual Directory Manager Installation Guide*
- *Oracle Virtual Directory Product Manual*

About Oracle Access Manager Implementations with Oracle Virtual Directory

The Oracle Virtual Directory combines the user data from multiple data sources to create an aggregated, virtual directory.

From the point of view of Oracle Access Manager applications, the virtual directory looks and behaves just like any other LDAP directory, and the Oracle Access Manager user usually does not receive any obvious indications that the data retrieved by Oracle Access Manager has come from heterogeneous sources.

From the perspective of the target data store owners, the impact of Oracle Virtual Directory is minimal; the data store owners do not relinquish ownership of their data, Oracle Virtual Directory does not reformat the native data structures, and no permanent copies of the original data are maintained by Oracle Virtual Directory.

To enable certain Oracle Access Manager features, you must extend the schema of the target LDAP directories or add columns simulating Oracle Access Manager auxiliary user attributes to the primary database tables included in your virtual directory. For more information, see "[About Schema Extension](#)" on page 10-15.

Note: Oracle Access Manager implementations with Oracle Virtual Directory are intended for use with the user profile and group repository. However, the implementation does not support Oracle Access Manager metadata such as policy rules, workflows, and the like. This metadata must be stored in a certified LDAPv3 directory (Oracle Internet Directory, Sun, or Microsoft Active Directory, for example).

Key Terms and Features

To explain precisely the issues surrounding Oracle Access Manager-Oracle Virtual Directory implementations, this document uses the following terms in very specific fashion.

Terms

Virtual Directory: A logical, aggregated directory that presents user data drawn from multiple sources, just as if all that data came from a standard LDAP directory to which a customer-defined schema has been uniformly applied. For the purposes of the Oracle Access Manager implementation, Oracle Virtual Directory does not create permanent copies of user profiles outside the native data sources. Rather, Oracle Virtual Directory retrieves and transforms each user profile as it is requested by a Oracle Access Manager application.

You can configure your virtual directory as a single, contiguous searchbase or as multiple disjoint searchbases. For details, see "[About Searchbase Options](#)" on page 10-5.

Super Directory: A special type of virtual directory that facilitates namespace mapping. It can contain any combination of federated LDAP directories, RDBMS databases, and embedded virtual data sources. The embedded virtual data sources can be split profiles, native RDBMS Joins, and native RDBMS Views. The super directory,

which is the only supported method for producing a single, contiguous searchbase aggregated from multiple data stores, connects to Oracle Access Manager by means of a Oracle Virtual Directory local store adapter.

Federation: A method by which Oracle Virtual Directory makes a data source visible in the virtual directory it presents to Oracle Access Manager. All the data for a given user profile comes from a single data store such as an LDAP directory, a single-table database, or an embedded virtual data source.

Different user profiles can come from different federated data stores, which incorporate any combination of the following types of data sources:

- Multiple, heterogeneous LDAP directories
- Multiple relational databases that store all user data in a single table
- Embedded virtual data sources, which fall into the following three categories:
- Split profiles involving any combination of directories and databases
- Native RDBMS Views involving multiple database tables
- Native RDBMS Joins involving multiple database tables

For more information, see "[Federated Data Stores](#)" on page 10-4.

Embedded Virtual Data Source: A virtual object that Oracle Virtual Directory "sees" as a target data store it can present to Oracle Access Manager or federate in a virtual directory, then present to Oracle Access Manager. Each embedded virtual data store aggregates two or more target data stores. The three types of embedded virtual data stores are:

- Split profile
- Native RDBMS Join
- Native RDBMS View

In general, embedded virtual data stores are suitable for authentication and authorization activities only, because they necessarily involve secondary data sources, which are sometimes not available for the full range of identity management activities.

Split Profile: A special type of embedded virtual data source created from more than one data source. Split Profiles draw the user profile attributes for each user from multiple sources, including LDAP directories and multiple database tables.

Each data store contributes some of the attributes necessary to complete the full set of user profile attributes that gets mapped into the Oracle Virtual Directory virtual directory. These attributes can come from LDAP directories or database tables. All Oracle Access Manager user attributes must reside in the primary data store, because not all Oracle Access Manager operations can be performed on the attributes in the secondary stores. Oracle Virtual Directory can make a split profile visible to Oracle Access Manager as a standard LDAP directory. Alternatively, a split profile can be federated as part of a virtual directory.

For more information, see "[Split Profiles](#)" on page 10-7 and [Figure 10-8, "Oracle Virtual Directory Implementation Layers"](#) on page 10-12.

Single-Table Database: A single-table database does not necessarily refer to a database that contains just one table, but rather, a database that stores in just one table all the user profile attributes that get mapped into the top level virtual directory.

Multi-Table Database: A database that stores in more than one table the user profile attributes that get mapped into the virtual directory.

Virtual Directory Schema: This is the schema developed by the customer for use by the top-level directory that Oracle Virtual Directory makes visible to Oracle Access Manager. It must be extended with the Oracle Access Manager attributes. See ["Virtual Directory Schema"](#) on page 10-17.

Optionally, you can further extend the virtual directory schema with customer attributes drawn from the target data sources. For details, see ["Customer Schemas"](#) on page 10-18.

Features

Virtual directory technology enhances Oracle Access Manager capabilities with four major features:

- Federated Data stores, mentioned earlier and discussed in more detail in ["Federated Data Stores"](#) on page 10-4.
- Split Profiles, mentioned earlier and discussed in more detail in ["Split Profiles"](#) on page 10-7.
- Aggregated Namespaces: Map target data stores and embedded virtual data sources to nodes in the super directory. You must install a local store adapter to create the super directory. For details, see ["Aggregated Namespaces"](#) on page 10-8.
- Schema Mapping: Transforms the data from all the target data stores according to the customer-defined schema in the top-level virtual directory. For details, see ["Aggregated Schema Mapping"](#) on page 10-8.

For background discussion of the general advantages offered by virtual directories, see *the Oracle Virtual Directory Product Manual*

Federated Data Stores

Oracle Virtual Directory allows Oracle Access Manager users to access and manipulate user data from disparate, multiple sources, just as if all user accounts came from a single, uniformly "schematized" data store. It can incorporate user data from LDAP directories even if the host directory servers come from different vendors and use different schema [Figure 10-1](#) illustrates how Oracle Access Manager connects to multiple LDAP directories.

Figure 10-1 Oracle Virtual Directory Implementation with Federated LDAP Directories

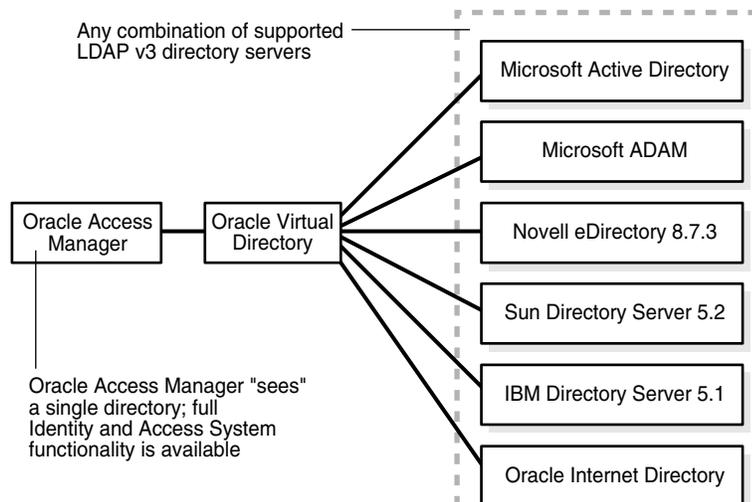


Figure 10–1 shows Oracle Access Manager "seeing" a single directory (Oracle Virtual Directory), which accesses any combination of supported LDAP v3 directory servers from Microsoft, Novell, Sun, Oracle Internet Directory, and IBM.

Your Oracle Virtual Directory virtual directory can also incorporate RDBMS databases that store all user data in a single table. For details on integrating databases that spread user data across multiple tables, see "Split-Profiles" on page 10-22.

Figure 10–2 illustrates how Oracle Access Manager connects to multiple relational databases. See also "Database Connectivity Tips" on page 10-82.

Figure 10–2 Oracle Virtual Directory Implementation Involving Federated RDBMS Applications

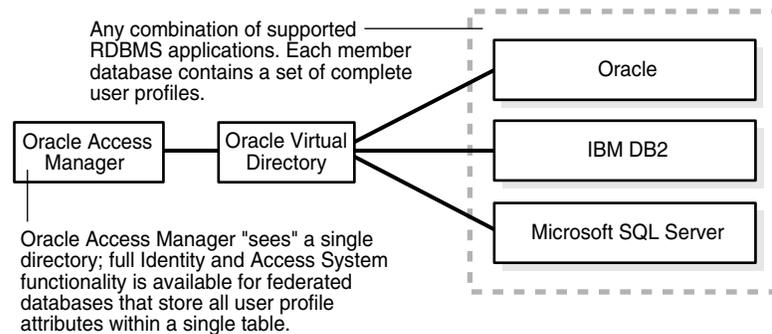


Figure 10–2 shows Oracle Access Manager "seeing" the Oracle Virtual Directory directory, which is accessing Oracle, IBM DB2, and Microsoft SQL Server RDBMS applications. Full Identity and Access System functionality is available for federated databases that store all user profile attributes within a single table. Each member database contains a set of complete user profiles.

About Searchbase Options

Oracle Access Manager supports two options for federating target data stores through Oracle Virtual Directory:

- Oracle Virtual Directory Disjoint Searchbases
- Unified Searchbase

Disjoint Searchbases: You can configure your Oracle Virtual Directory implementation so that Oracle Access Manager "sees" each target data store as a distinct, disjoint searchbase within the virtual directory. Namespace aggregation is not possible for such a configuration. Also, each target data store resides behind a different top-level mapping adapter, so global directory searches across all the data sources are not possible.

Figure 10–3 illustrates a virtual directory configured for disjoint searchbases.

Figure 10-3 A Virtual Directory with Disjoint Searchbases

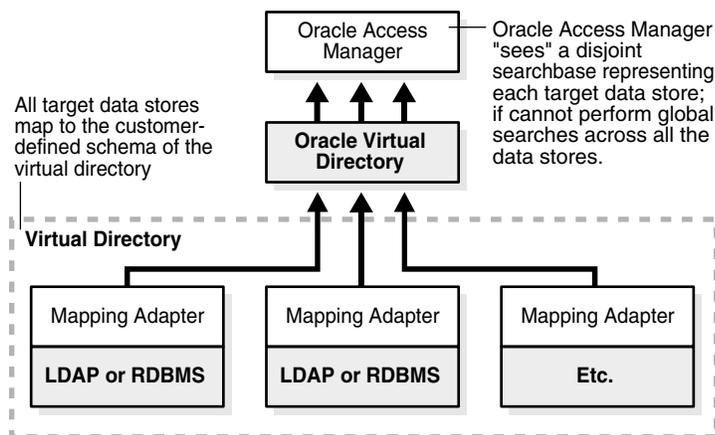


Figure 10-3 shows Oracle Access Manager accessing the Oracle Virtual Directory virtual directory. The virtual directory is comprised of a set of target data stores, each being a mapping adapter and its LDAP or RDBMS. All target data stores map to the customer-defined schema of the virtual directory. Oracle Access Manager "sees" a disjoint searchbase representing each target data store. It cannot perform global searches across all the data stores.

Unified Searchbase: You can create a super directory by installing a Local Store Adapter at the top level, then creating nodes to which you map your target data stores. This option allows for both global directory searches and powerful namespace aggregation.

Figure 10-4 provides an illustration of super directory with a unified, contiguous searchbase.

Figure 10-4 A Super Directory with a Unified, Contiguous Searchbase

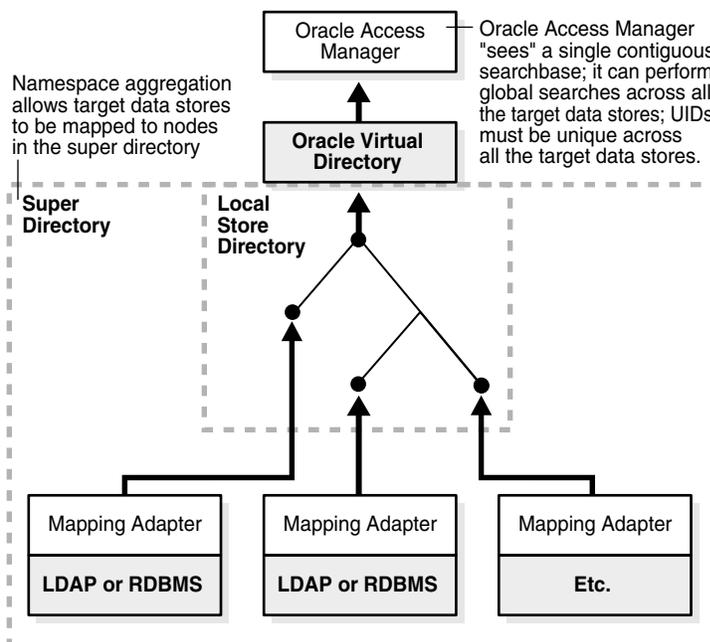


Figure 10-4 shows Oracle Access Manager accessing a super directory that has a Local Store Adapter at the top. Each node in the super directory contains a mapping adapter and LDAP or RDBMS. They access Oracle Virtual Directory through the Local Store Adapter: Oracle Access Manager "sees" a single contiguous searchbase that performs global searches across all the target data stores. Note that UIDs must be unique across the target data stores.

Split Profiles

In addition to providing Oracle Access Manager users with access to federated data stores, Oracle Virtual Directory can provide access to virtualized split profile data sources which draw user profile attributes from multiple data sources, such as LDAP directories and relational database tables.

For example, you can store attributes such as user login password and office phone number in an Active Directory account maintained by Information Technology, while storing other attributes such as home phone number and health plan affiliation in a relational database account maintained by Human Resources.

Because this distribution of attributes across multiple data sources precludes the execution of certain identity management functions on secondary data stores, split profile configurations are suitable primarily for authentication and authorization (Access System) operations.

Figure 10-5 illustrates a simple implementation involving a split profile.

Figure 10-5 Oracle Access Manager-Oracle Virtual Directory Implementation for a Simple Split Profile

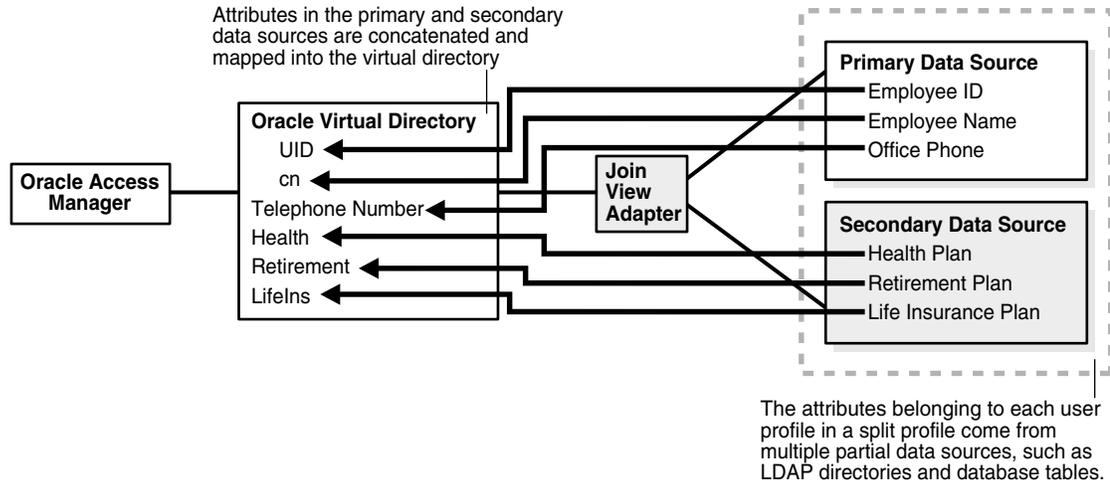


Figure 10-5 shows a simple split profile. Oracle Access Manager accesses Oracle Virtual Directory, which in turn accesses a primary data source and a secondary data source by using a join view adapter. In this example, the primary data source provides an employee ID, employee name, and office phone; and the secondary data source provides data for a health place, retirement plan, and life insurance plan. Hence, these six attributes are concatenated and mapped into the virtual directory

The primary data source contains the Oracle Access Manager user branch schema attributes, while the secondary data sources usually contain customer attributes.

All Access System and Identity System operations can be performed on the attributes in the primary data source. All Access System operations can also be performed on the

data in the secondary sources, but certain Identity System operations cannot be performed on attributes residing in the secondary data stores.

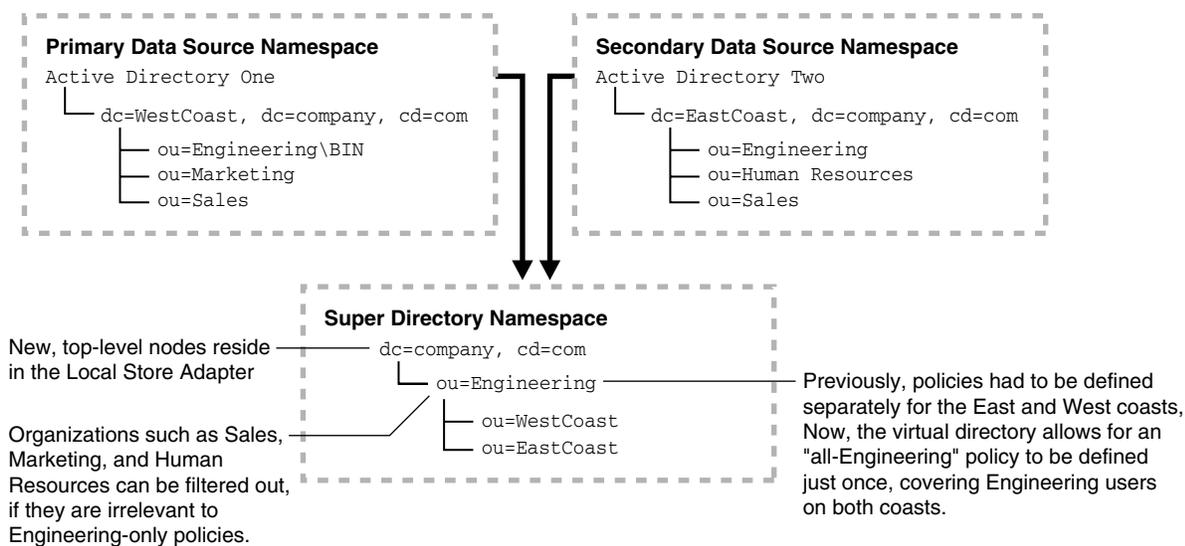
For more information, see ["Implementation Limitations"](#) on page 10-9.

Aggregated Namespaces

When you create a super directory, you can specify a namespace hierarchy ideally suited to your identity management and policy management needs. This new hierarchy can differ from the native namespace hierarchies used by the constituent data stores in the virtual directory.

As [Figure 10-6](#) illustrates, attributes can be reorganized and assigned to new levels.

Figure 10-6 Namespace Aggregation for a Simple Super Directory



[Figure 10-6](#) shows a super directory namespace accessing data in from a primary data source namespace and a secondary data source namespace. In this example, the primary data source namespace contains Active Directory One, which has Engineering, Marketing, and Sales attributes under a West Coast node. Similarly, the secondary data source namespace has Active Directory Two, with its Engineering, Marketing, and Sales attributes under an East Coast node. The super directory namespace reorganizes and assigns the attributes as follows: company, then Engineering, then attributes for either West Coast or East Coast.

In this example, organizations such as Sales, Marketing, and Human Resources can be filtered out if they are irrelevant to the Engineering-only policies. Previously, policies had to be defined separately for the East and West coasts. Now, the virtual directory allows for an "all-Engineering" policy to be defined just once, covering Engineering users on both the East Coast and West Coast.

Aggregated Schema Mapping

When you create your Oracle Virtual Directory virtual directory, you map the native schema used by the constituent data stores to the schema used by your virtual directory [Figure 10-7](#) illustrates this mapping on a simple Oracle Virtual Directory system.

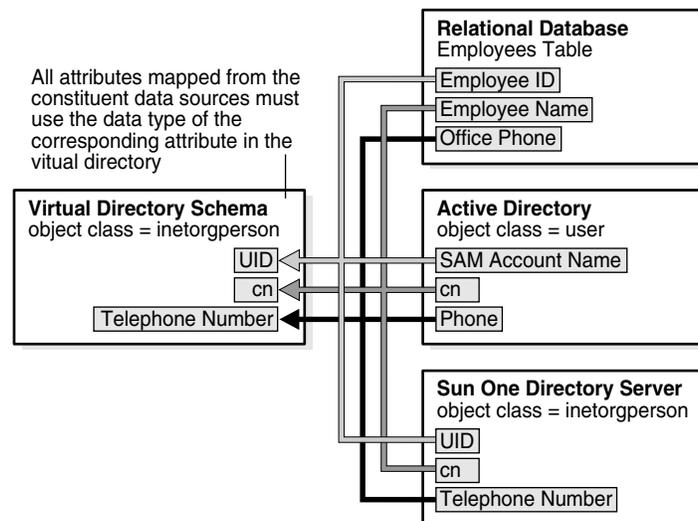
Figure 10–7 Aggregated Schema Mapping for a Simple Virtual Directory

Figure 10–7 shows a virtual directory schema mapping to the following data sources:

- **Relational Database:** An employee table with rows for the Employee ID, Employee Name, and employee’s Office Phone number.
- **Active Directory:** An object class called user that has attributes for the SAM Account Name, cn, and Phone number.
- **SunOne Directory Server:** An object class called inetorgperson, with attributes for the UID (user ID), cn, and Telephone Number.

All attributes from the constituent data sources must use the data type of the corresponding attribute in the virtual directory. For example, the Telephone Number attribute in the virtual directory schema maps to Office Phone in the Employees table of the relational database, Phone in the user object class of Active Directory, and Telephone Number in the inetorgperson object class of Sun One Directory Server.

Implementation Limitations

Oracle Virtual Directory extends Oracle Access Manager functionality to multiple, heterogeneous directories and databases, but with certain limitations. When you deploy Oracle Access Manager with Oracle Virtual Directory, you must carefully observe the limitations stated in this document. Table 10–1 lists the virtual directory configurations subject to limitation and provides references to detailed discussions of these issues.

Table 10–1 Oracle Access Manager Feature Availability for Virtual Directory Configurations

Data Source	Oracle Access Manager Feature Availability	
	Access System	Identity System
Federated LDAP directory	Full	Full
Federated "single table" database	Full	Full

Table 10–1 (Cont.) Oracle Access Manager Feature Availability for Virtual Directory Configurations

Data Source	Oracle Access Manager Feature Availability	
Federated "multi-table" database (using the native RDBMS Join feature)	Full	Full functionality for primary data stores. Add, Modify, and Delete are also available for secondary data stores if the native RDBMS Join feature supports these functions
Federated "multi-table" database (using the native RDBMS View feature)	Full	Full functionality for primary data stores, but Add and Delete are not available for secondary data stores. (Modify is available for secondary data stores).
Split-Profile directory (using the Oracle Virtual Directory Join View adapter)	Full	Full functionality for primary data stores, but Add, Modify and Delete are not available for secondary data stores.

For more information, see:

- [About Limitations on Multi-Value Attributes](#)
- [About Limitations on Embedded Virtual Data Sources](#)
- [Database Connectivity Tips](#)

About Limitations on Multi-Value Attributes

Individual attributes stored in standard LDAP directories can take multiple values. For instance, you can record each user's password history or assign multiple subscriptions to a user account stored in an LDAP directory.

By contrast, properly normalized data tables in SQL-compliant RDBMS applications cannot store multiple values for the same user attribute within a single table. Therefore, Oracle Access Manager-Oracle Virtual Directory implementations involving database tables support only limited functionality for multi-valued attributes. For details, consult Oracle customer care.

Note: If your virtual directory incorporates LDAP directories exclusively, no restrictions apply to multi-valued attributes.

User profiles already stored in existing RDBMS databases are most likely to have been implemented entirely with single-value attributes, so no restrictions apply, as long as all the database tables you incorporate into your virtual directory contain single-value attributes only.

In rare situations where multi-valued attributes were used to implement the user accounts in a non-normalized RDBMS data store, you can incorporate the unsupported tables containing multi-valued attributes into your virtual directory as long as you carefully observe the following limitations on User and Group Manager operations:

- The password history function is not supported
- No more than one administrator can be configured for each group
- No more than one subscription can be configured for each group

- Either group subscription or unsubscription notification can be activated, but you cannot activate both simultaneously
- No more than one dynamic filter can be configured for each group
- No more than one group type can be configured for each group
- No more than one subscription type can be configured for each group (The possible types are: Open, Close, Open with filter, and Controlled through workflow).
- A database table in the virtual directory can contain no more than one multi-valued attribute.
- If you are creating new data stores for your virtual directory, Oracle strongly recommends that you use LDAP directories whenever possible. This is because the limited ability of relational databases to handle multi-valued attributes restricts the functionality available in your identity management application. If you are working with existing user data stored in a relational database, please familiarize yourself thoroughly with the restrictions on multi-valued attribute handling within the virtual directory.

About Limitations on Embedded Virtual Data Sources

Table 10–2 lists the limitations on embedded virtual data sources involving multiple database tables.

Table 10–2 *identity Management Function Availability for Multi-table Configurations*

Identity Management Function	Table Aggregation Method		
		Join View Adapter (Split Profile)	Native RDBMS Join Feature
Modify	No	Yes, if supported by the native RDBMS Join feature	Yes
Add	No	Yes, if supported by the native RDBMS Join feature	No
Delete	No	Yes, if supported by the native RDBMS Join feature	No

Implementation Architecture

The Oracle Access Manager-Oracle Virtual Directory implementation consists of three layers, shown in Figure 10–8.

Figure 10–8 Oracle Virtual Directory Implementation Layers

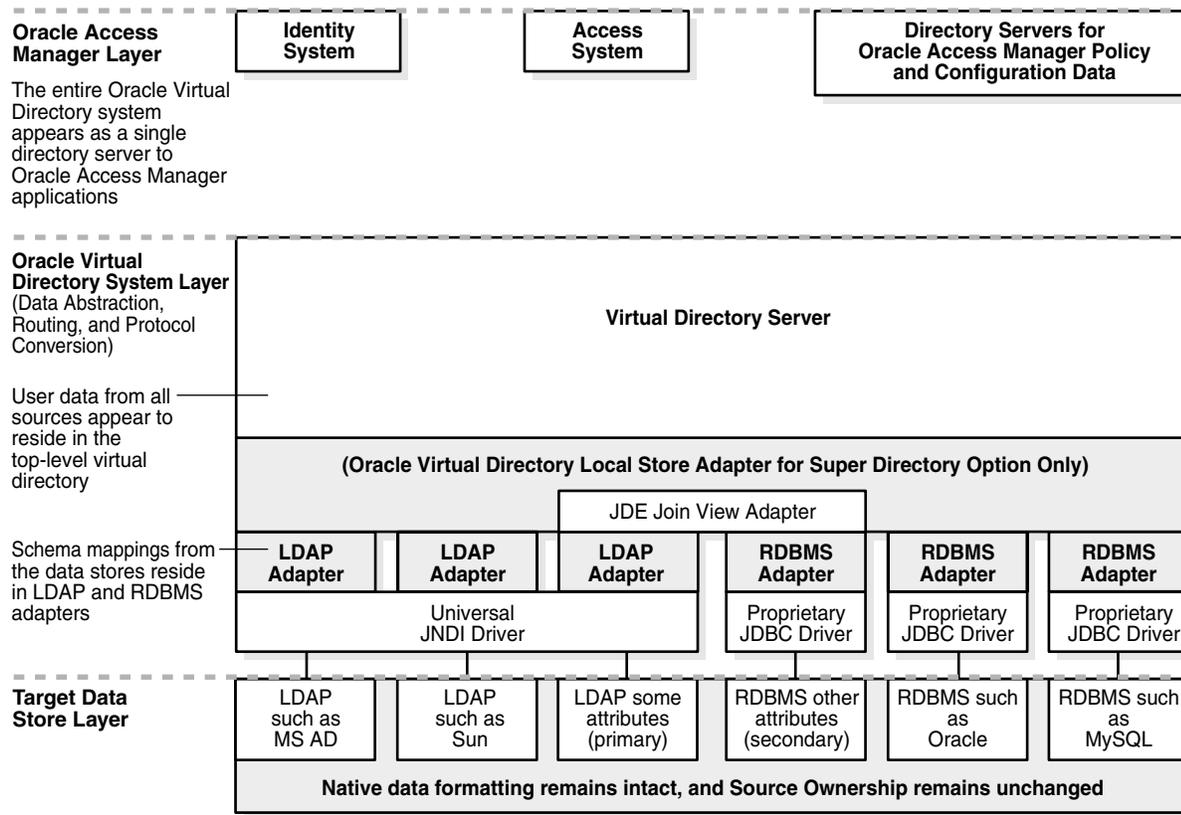


Figure 10–8 shows the three Oracle Virtual Directory integration layers:

- Oracle Access Manager Layer:** This layer contains the identity system, access system, and directory servers for Oracle Access Manager policy and configuration data. In this layer, the entire Oracle Virtual Directory system appears as a single directory server to Oracle Access Manager applications.
- Oracle Virtual Directory System Layer:** This layer contains the virtual directory server for a Oracle Virtual Directory Local Store Adapter for the Super Directory option. It uses a Oracle Virtual Directory join view adapter to access LDAP adapters and RDBMS adapters. To connect to their corresponding target data stores in the third layer (the target data store layer), the LDAP adapters use a JNDI driver and each RDBMS adapter uses its own proprietary JDBC driver. Schema mappings from the data stores reside in the LDAP and RDBMS adapters.
- Target Data Store Layer:** This layer contains LDAP and RDBMS data sources that map to their corresponding LDAP and RDBMS adapters in the Oracle Virtual Directory system layer. In this layer, native data formatting remains intact, and that source ownership remains unchanged.

To users and applications in the Oracle Access Manager layer, the Oracle Virtual Directory system appears to be a single LDAP directory that includes the standard schema, plus the extended Oracle Access Manager attributes.

Within the virtual directory layer, Oracle Virtual Directory accepts requests for user data from the Oracle Access Manager applications, retrieves the requested data from the constituent data stores, transforms that data so that it conforms to the Oracle Access Manager schema, then passes the processed data back to the requesting Oracle Access Manager application. Figure 10–9 illustrates the steps in this process.

Figure 10–9 Data Request Handling in a Simple Oracle Access Manager-Oracle Virtual Directory Implementation

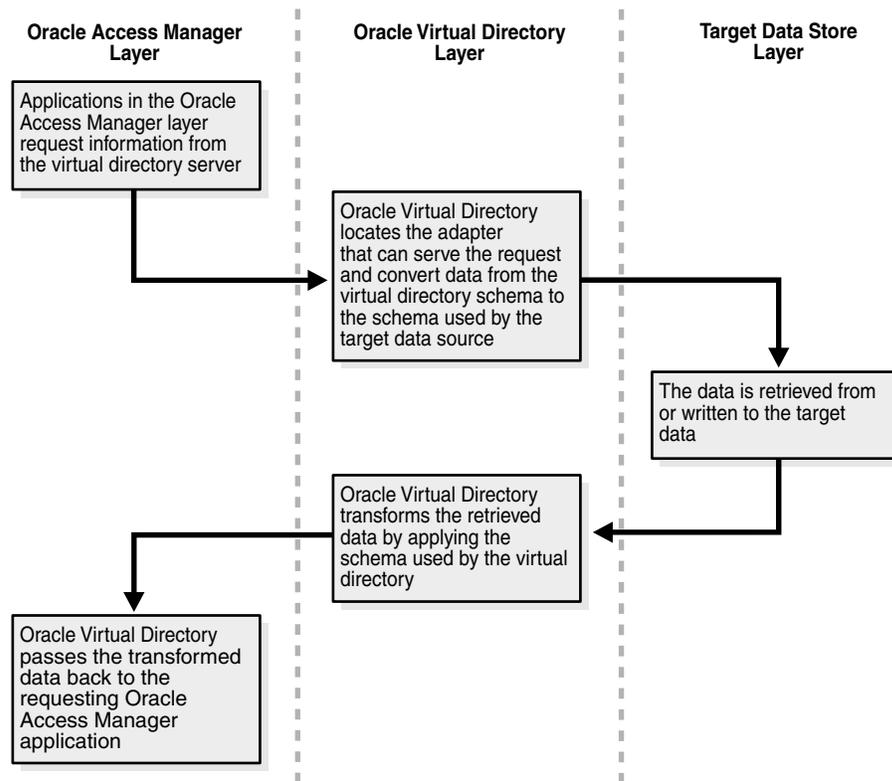


Figure 10–9 shows the data request handling steps as follows:

Process overview: Request Handling

1. In the Oracle Access Manager layer, applications request information from the virtual directory server.
2. In the virtual directory layer, Oracle Virtual Directory locates the adapter that can serve the request and convert data from the virtual directory schema to the schema used by the target data source.
3. In the target data store layer, the data is retrieved from or written to the target data stores.
4. In the virtual directory layer, Oracle Virtual Directory transforms the retrieved data by applying the schema used by the virtual directory.
5. In the Oracle Access Manager layer, Oracle Virtual Directory passes the transformed data back to the requesting Oracle Access Manager application.

To the administrators of the directories and databases in the target data store layer, the Oracle Access Manager-Oracle Virtual Directory implementation appears to have minimal impact, because implementation does not require permanent alteration of the native namespaces or data structures for either directories or databases.

Note: Depending on the features you wish to use, you may need to add certain Oracle Access Manager auxiliary attributes as columns in target database tables. You also must extend the target LDAP directory schema with the user branch of the Oracle Access Manager schema. For details, see "[About Schema Extension](#)".

Furthermore, the virtual directory does not require that the data be copied permanently to a location beyond the control of the original data owner. Finally, data security is maintained or even enhanced, because access to individual data stores and even individual user profiles can now be controlled through Identity System attribute access control.

About Oracle Virtual Directory Drivers and Adapters

Oracle Virtual Directory uses special drivers and adapters to connect to the data sources it incorporates in its virtual directory.

JNDI Driver: The JNDI driver is shipped as part of the Oracle Virtual Directory installation package. A JNDI driver connects Oracle Virtual Directory to the LDAP directories, and a JDBC driver connects Oracle Virtual Directory to the RDBMS sources. You install these drivers on the machine that hosts Oracle Virtual Directory.

JDBC Driver: You must install the appropriate version of the JDBC driver for each RDBMS application you use. See the *Oracle Virtual Directory Product Manual* and the *Oracle Virtual Directory and Virtual Directory Manager Installation Guide* for details. Oracle Virtual Directory Database Adapters support any database that provides a JDBC driver.

Adapters: In addition to installing the appropriate driver for each data source in your virtual directory, you must configure an LDAP or RDBMS adapter for each directory or relational database that connects to Oracle Virtual Directory. These adapters contain the mapping information Oracle Virtual Directory uses to transform user profile information from the native data stores with the virtual directory schema. For details, see "[Creating Data Store Adapters](#)" on page 10-44.

About Oracle Access Manager-Specific Data

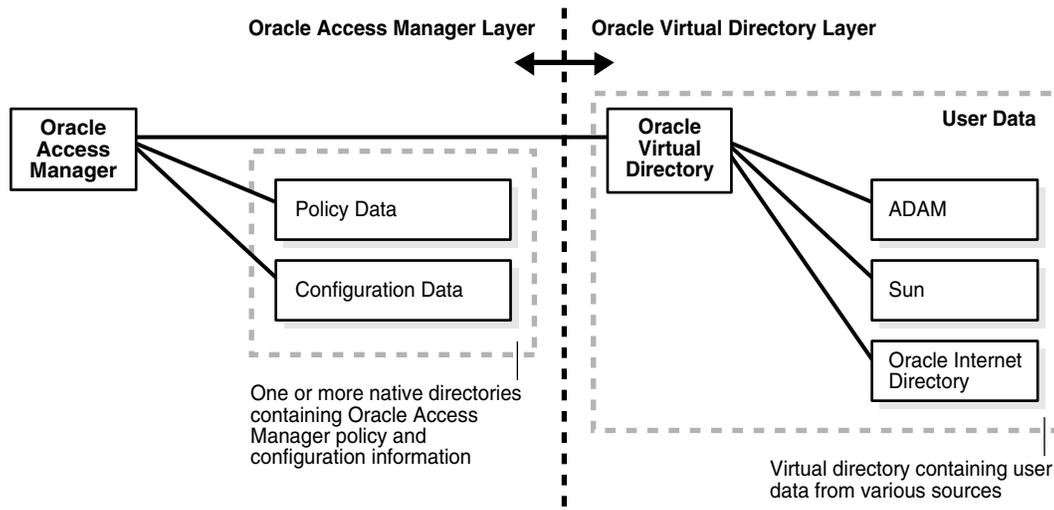
Oracle Access Manager user, policy, and configuration data each occupy an LDAP directory information tree (DIT) branch. The Oracle Virtual Directory implementation requires that each branch exist in a specific location. [Table 10-3](#) lists these requirements.

Table 10-3 Required locations for the branches of the Oracle Access Manager directory

Branch	Location
Policy Configuration	These branches must reside on one or more directory servers within the Oracle Access Manager Layer. The host directories are native to Oracle Access Manager.
User Data	All user data stores appear to be part of the top-level directory, which resides on the machine hosting Oracle Virtual Directory within the Oracle Virtual Directory layer.

[Figure 10-10](#) illustrates the policy and configuration branch location for a Oracle Virtual Directory implementation.

Figure 10–10 Policy and Configuration Branch Location for a Oracle Virtual Directory Implementation



describes the contents of this diagram.

About Schema Extension

For proper functioning, you need certain Oracle Access Manager attributes like `userid`, `userpassword`, and others to be extended to your schema.

Regardless of the native schemas or table structures used by the data stores in the virtual directory, Oracle Access Manager "sees" only the schema used by the Oracle Virtual Directory virtual directory. This is because Oracle Virtual Directory automatically maps the native object classes and attributes used by the various data stores to the corresponding logical object classes and attributes used by the virtual directory.

Oracle Virtual Directory supplies a default virtual directory schema, which is quite similar to an industry-standard schema for LDAP directories. You can use this as a starting point when developing a virtual directory schema optimized for the needs of your enterprise.

The files required for schema extension are located in:

```
IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit\oblix\tools
\DataAnyWhere\OblixUserSchema\*.ldif
```

Figure 10–11 illustrates both the required and optional schema extension tasks involved with Oracle Access Manager-Oracle Virtual Directory implementation.

Figure 10–11 Schema Extension Tasks for Implementation

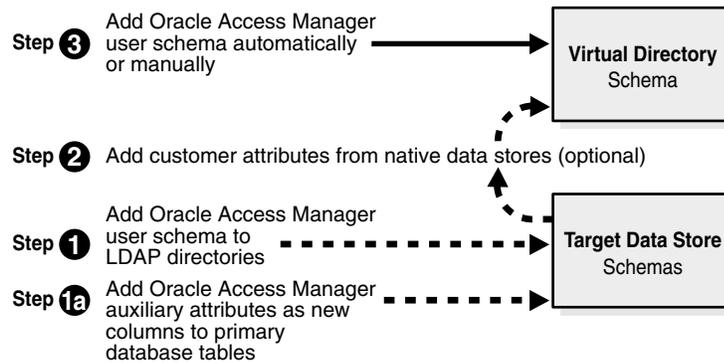


Figure 10–11 shows the following steps:

- Step 1a: Add Oracle Access Manager auxiliary attributes to the target data store schemas as new columns to primary database tables.
- Step 1: Add Oracle Access Manager user schema to LDAP directories in the target data store schemas.
- Step 2: Add customer attributes from native data stores (optional).
- Step 3: Add Oracle Access Manager user schema automatically or manually to the virtual directory schema.

Table 10–4 lists the schema requirements for various components of the virtual directory.

Table 10–4 Schema Requirements for Components of this Implementation

Component	Schema Requirements
The virtual directory	Must be extended with the Oracle Access Manager user schema. Should also be extended with customer attributes.
LDAP directories connected to Oracle Virtual Directory as federated data sources	Must be extended with the Oracle Access Manager user schema.
Databases connected to Oracle Virtual Directory as federated data sources	Table columns must be added to the database tables serving as primary data stores. Each column represents a Oracle Access Manager auxiliary attribute that enables a feature you plan to use. For details on the specific Oracle Access Manager features enabled by each attribute in the user branch schema, see " Oracle Access Manager Auxiliary Attributes " on page 10-67.
Split profile	The schema of the primary data store must be extended with the Oracle Access Manager user schema. If the primary data store is a database table, a column must be added for each auxiliary attribute that enables a Oracle Access Manager feature you plan to use. For details on the specific features enabled by each attribute in the Oracle Access Manager user branch schema, see " Oracle Access Manager Auxiliary Attributes " on page 10-67.

For more information, see:

- [Virtual Directory Schema](#)
- [Target Directory Schemas](#)
- [About Adding Attributes to Target Database Tables](#)
- [Customer Schemas](#)

Virtual Directory Schema

To enable your virtual directory to connect to and make use of all Oracle Access Manager features, you must extend it with appropriate attributes. For example, Oracle Access Manager requires attributes assigned to the Full Name, Login, and Password semantic types for Person and Group object classes. This and other essential user data occupies a branch in each Oracle Access Manager-enabled user directory. For further discussion, see the section about object classes in the chapter on setting up the Identity System.

For a detailed listing of the Oracle Access Manager schema, see the *Oracle Access Manager Schema Description*. You can update the Oracle Virtual Directory schema with Oracle Access Manager user attributes in the following two ways:

- **Automatically:** This occurs if you select the "Auto configure objectclass" checkbox during Identity System setup, as explained earlier.
- **Manually:** For details, see the section on configuring attributes manually in "[Configuring Attributes Manually](#)" on page 6-10.

Target Directory Schemas

Just as you extend the schema of your parent virtual directory, you must also extend the native schema used by the LDAP directories included in your virtual directory. You achieve this with the ldapmodify.exe utility. For details, see "[Extending Directory Schemas](#)" on page 10-40.

Note: The Oracle Access Manager attributes must also be added to the primary data stores in any split profile included in your virtual directory. For details, see [Table 10-4, "Schema Requirements for Components of this Implementation"](#) on page 10-16.

About Adding Attributes to Target Database Tables

For any databases included in your virtual directory, you must add table columns to simulate the auxiliary attributes that enable Oracle Access Manager features you plan to use. This applies only to database tables used as primary data sources. For example, you add an Out of Office Indicator column to enable the Surrogate feature in Oracle Access Manager workflows.

Note: SQL-compliant databases do not possess any means to implement LDAP object classes directly. However, you can simulate an object class by mapping, for example, all the rows (user accounts) in a primary database table to the person object class used by the virtual directory.

For a listing of the Oracle Access Manager auxiliary attributes that enable specific functions, see the following:

- [Table 10-7, "Extended Attributes Required by User Manager Functions"](#)
- [Table 10-8, "Extended Attributes Required by Group Manager Function"](#)

Customer Schemas

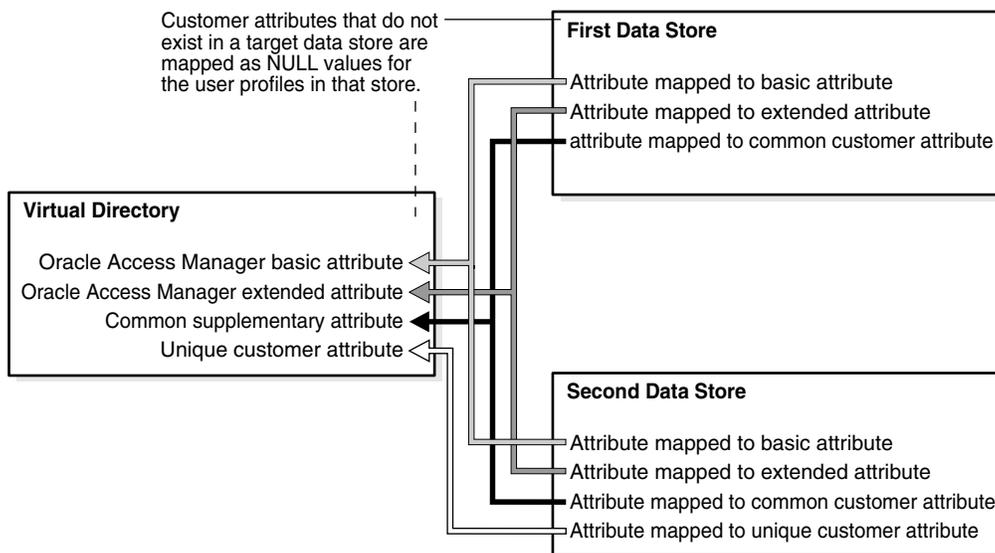
Optionally, you can further extend the default Oracle Virtual Directory schema by adding customer attributes from your native data stores. For example, the default Oracle Virtual Directory person object class is UID, but you can add InetOrgPerson, then specify InetOrgPerson as the Person object class when you run Oracle Access Manager setup.

For details on modifying the default Oracle Virtual Directory schema using the Oracle Virtual Directory Manager (VDM, formerly known as the DME) interface, see the section on schema configuration in the chapter on configurations and settings in the Oracle Virtual Directory Product Manual.

For details on specifying the person object class, see the section about object classes in ["Specifying Person and Group Object Classes"](#) on page 6-7.

Note: When a customer attribute exists on one target data store, but not in the others, Oracle Virtual Directory returns the user profiles from those other data stores with that supplementary attribute set to NULL. [Figure 10-12](#) illustrates this situation.

Figure 10-12 Mapping Supplementary Attributes to a Simple Virtual Directory



[Figure 10-12](#) shows the following supplementary attributes mapped to a simple virtual directory:

- First Data Store: Contains attributes mapped to a basic attribute, an extended attribute, and a common customer attribute.
- Second Data Store: Contains the same attributes as the first data store, but with an attribute mapped to a unique customer attribute.

Hence, the virtual directory contains the following attributes:

- Oracle Access Manager basic attribute, mapped to both first and second data stores
- Oracle Access Manager extended attribute, mapped to both first and second data stores
- Common supplementary attribute (mapped to both first and second data stores)
- Unique customer attribute, mapped to second data store only, since the first data store does not have this type of attribute

Implementation Scenarios and Limitations

This discussion introduces the three scenarios that are supported when implementing Oracle Access Manager with Oracle Virtual Directory. The following sections also explain the limitations you encounter with each scenario.

- [Heterogeneous LDAP Directories](#)
- [Multiple RDBMS Databases](#)
- [Split-Profiles](#)

Heterogeneous LDAP Directories

The Oracle Access Manager-Oracle Virtual Directory implementation can connect to multiple LDAP v3 directories from one or more vendors. Each directory can use a different schema (attributes and object classes). Because Oracle Virtual Directory transforms data at run-time, Oracle Access Manager sees the aggregated directories as a single directory to which the Oracle Access Manager schema has been uniformly applied.

When Oracle Access Manager is connected to a Oracle Virtual Directory system that includes just LDAP directories (and no RDBMS data stores), all Access System and Identity System functionality is available. However, you should observe several restrictions when combining directories.

Restrictions

- Oracle Access Manager supports only a single Person object class and a single Group object class associated with each user profile. Therefore, the various (and possibly multiple) Person and Group object classes in the native directories must be mapped to just one Person object class and one Group object class in the virtual directory.
- The native namespaces of the constituent directories can be identical, but those namespaces must map to different namespaces within the virtual directory.
- The login IDs for all users supported by the virtual directory must be unique across all the included directories. [Figure 10-13](#) illustrates this situation:

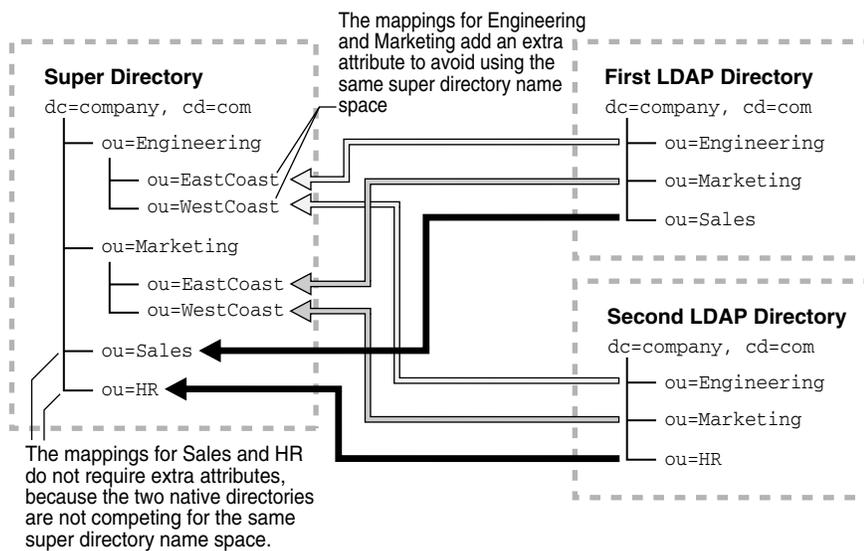
Figure 10–13 Mapping Identical Namespaces into a Simple Super Directory

Figure 10–13 shows the mapping of two identical LDAP directory name spaces to a super directory. The first LDAP directory contains attributes for Engineering, Marketing, and Sales. The second LDAP directory contains mappings for Engineering, Marketing, and HR.

The super directory maps to the Engineering, Marketing, Sales, and HR attributes. Its Engineering and Marketing mappings add an extra attribute (for EastCoast and WestCoast) to avoid using the same super directory namespace. This super directory also has mappings for Sales and HR, but these do not require extra attributes because the two native directories are not competing for the same super directory namespace.

- All the attributes mapped from the constituent directories to a given attribute in the virtual directory must use a common data type. For example, the `ObOutOfOfficeIndicator` attribute cannot be a binary value in one data source, but a date (indicating when the user will return) in another.
- If a native directory enforces referential integrity, references such as `Manager` or `Group Member` can only come from the same native directory. If the native directory doesn't enforce referential integrity, and that native directory also supports external references, references can reside in other directories.
- RDN (relative distinguished name) is not supported.

Multiple RDBMS Databases

Databases that include a single table that contributes all the user profile attributes mapped by Oracle Virtual Directory can be federated into the virtual directory and made visible to Oracle Access Manager. In situations where more than one data table contributes attributes to a given user profile, four options exist for joining the tables so as to create a virtual data source containing the complete set of user attributes.

For more details about these options and the limitations each entails, see ["About Joining Database Tables in an Embedded Virtual Data Source"](#) on page 10-21. See also ["Database Connectivity Tips"](#) on page 10-82.

About Joining Database Tables in an Embedded Virtual Data Source

When more than one data table contributes attributes to a given user profile, four options exist for joining the tables to create a virtual data source containing the complete set of user attributes:

- The native RDBMS Join feature
- The native RDBMS View feature
- The Oracle Virtual Directory Join View adapter (split profile)
- A custom joiner based on your preferences

Figure 10–14 illustrates the four methods for joining multiple database tables within one of the three supported types of embedded virtual data sources.

Figure 10–14 Methods for Joining Tables within a Virtual Directory

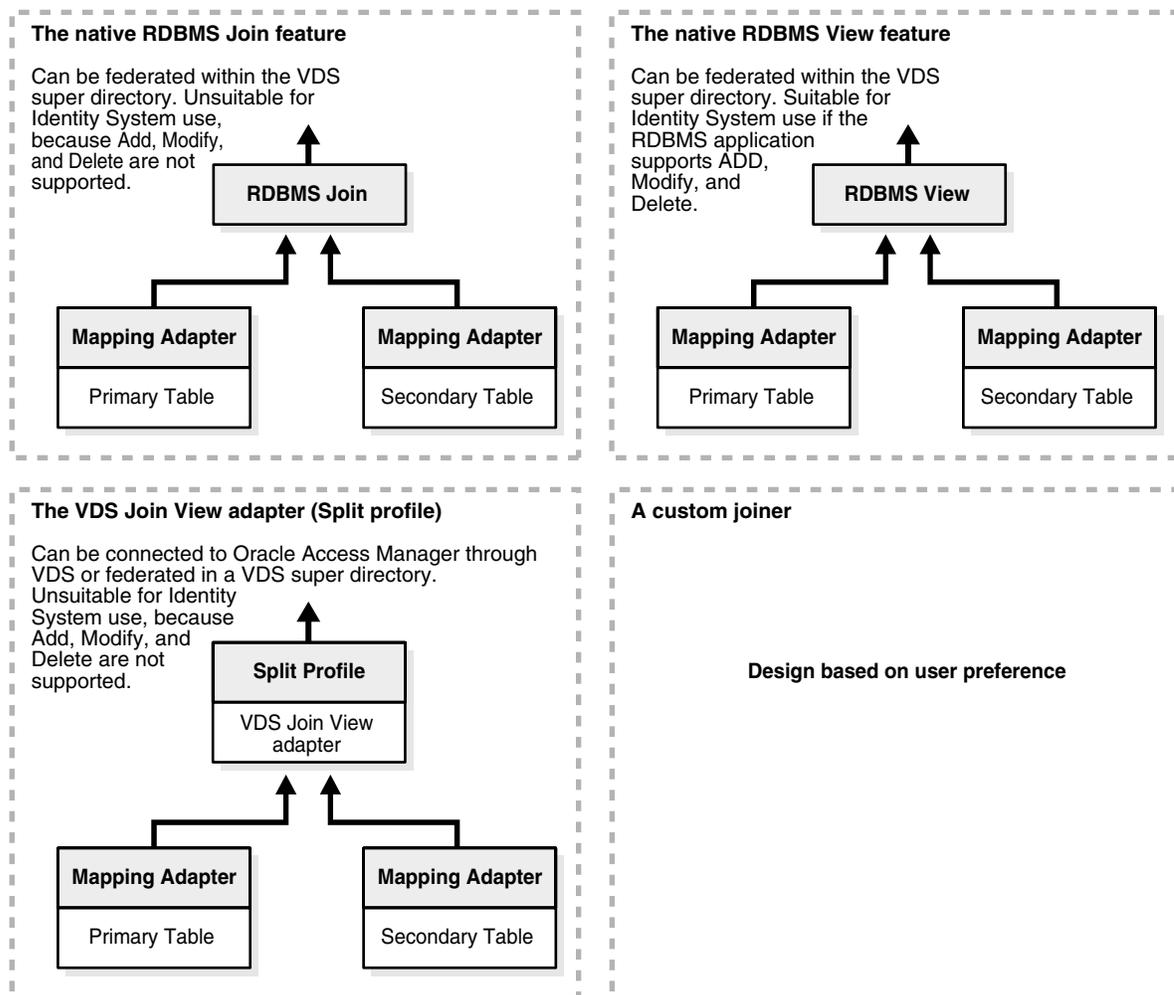


Figure 10–14 provides the following detailed information about the four methods:

- The native RDBMS join feature: This method shows an RDBMS join created by a primary table and a secondary table, each with its own mapping adapter. It cannot be federated within the Oracle Virtual Directory super directory. It is unsuitable

for Identity System use because it does not support Add, Modify, and Delete operations.

- The native RDBMS view feature: This method shows an RDBMS view created by a primary table and a secondary table, each with its own mapping adapter. It can be federated within the Oracle Virtual Directory super directory, and it is suitable for Identity System use if the RDBMS application supports Add, Modify, and Delete operations.
- The Oracle Virtual Directory Join Viewer adapter (split profile): This method shows a split profile with its Oracle Virtual Directory join view adapter created from a primary table and a secondary table, each with its own mapping adapter. It cannot be connected to Oracle Access Manager through Oracle Virtual Directory or federated in a Oracle Virtual Directory super directory. It is unsuitable for Identity System use because it does not support Add, Modify, or Delete operations.
- A custom joiner: Its design is based on user preference.

Table 10-5 lists the specific limitations associated with each method:

Table 10-5 Methods for Joining Database Tables within a Virtual Directory

Method	Suitability and Limitations
<p>The native Join feature of the host RDBMS application</p> <p>The resulting Join is then federated as part of the virtual directory</p>	<ul style="list-style-type: none"> ■ Suitable for Access System use, because Oracle Access Manager Read and Search operations are both supported. ■ Not suitable for Identity System use, because Oracle Access Manager Add, Modify and Delete operations are not supported.
<p>The native View feature of the host RDBMS application</p> <p>The resulting View is then federated as part of the virtual directory.</p>	<ul style="list-style-type: none"> ■ Suitable for Access System use, because Oracle Access Manager Add and Search operations are supported. ■ Suitable for Identity System use only if the native View feature of the RDBMS application supports Add, Modify, and Delete operations.
<p>The Oracle Virtual Directory Join View adapter method</p> <ul style="list-style-type: none"> ■ Each data source connects to the Join View adapter through an RDBMS adapter. ■ The result is a split profile, which can be connected to Oracle Access Manager through Oracle Virtual Directory or be federated as part of the virtual directory, which is then connected to Oracle Access Manager. 	<ul style="list-style-type: none"> ■ Suitable for Access System use, because Oracle Access Manager Add and Search operations are supported. ■ Suitable for Identity System use, but Subtype Search, Add, and Delete operations can be performed only on the primary table. ■ All limitations to LDAP directories joined with the Join View adapter also apply to databases joined with the Join View adapter. For details, see "Join View Adapter Requirements and Limitations" on page 10-23.
<p>A custom joiner method</p>	<p>You can write a custom joiner to overcome the limitations imposed by the standard Join View adapter or the native Join and View features of your RDBMS application.</p> <p>This involves custom programming. For details, consult the section on joiners in Oracle Virtual Directory Product Manual.</p>

Split-Profiles

Virtual directories which draw the attributes for each user profile from two or more data sources are known as split profiles. These data sources can include any combination of LDAP directories and relational databases.

One data store serves as the primary data source. The schema of this data store must be extended with the Oracle Access Manager-specific user data. For details, see "[About Schema Extension](#)" on page 10-15

All additional data stores are secondary data sources. Not all Identity System functionality is supported for these secondary data stores. See "[Join View Adapter Requirements and Limitations](#)" on page 10-23. It is not necessary to extend the Oracle Access Manager user schema to secondary data stores.

You can join the data sources in a split profile either through the standard Oracle Virtual Directory Join View tool (recommended method), or through a custom joiner. For details on creating a custom joiner, consult the Oracle Virtual Directory Product Manual.

Join View Adapter Requirements and Limitations

The Join View adapter supports all Access System operations on attributes that reside in either the primary or secondary data stores. This includes authentication, authorization, auditing, and single sign-on.

Note: Identity System operations are supported, with the following restrictions.

Restrictions

- The user login ID attribute for the split directory must reside in the primary data store. The user login password and the user full name attribute must also reside in the primary data store
- The Oracle Access Manager user schema must reside in the primary data store.
- Base-level searches are supported for both the primary and secondary data stores.
- Sub-tree searches are supported only for the primary data store. By implication, the following restrictions apply:
 - Attributes residing in a secondary data store must not be configured as searchable.
 - Attributes residing in a secondary data store must not be configured for filter operations involving sub-tree searches. This includes:
 - domain filters
 - dynamic group filters
 - group subscription filters
 - Query Builder filters
- Modify is supported for all attributes, without regard to the specific data store in which those attributes reside.
- Users, groups, and other objects can be created only in the primary data store.
- Only users, groups, and other objects in the primary data store can be deleted. (Oracle Access Manager applications cannot delete objects in the secondary data store. This can be accomplished only through the target RDBMS application or LDAP directory, which may lead to synchronization problems in real-time environments.)
- A given attribute can be configured from only one data store. Join values are not supported.

Implementation Requirements

Oracle Access Manager connects to the virtual directory just as it connects to any other LDAP directory; therefore, most supported Oracle Access Manager configurations should integrate smoothly with Oracle Virtual Directory. The following sections list support and requirement details for various aspects of the Oracle Access Manager-Oracle Virtual Directory implementation.

Oracle Access Manager: The LDAP directory branches containing Oracle Access Manager configuration and policy data must reside on one or more directory servers native to Oracle Access Manager. In other words, the configuration and policy branches cannot reside anywhere in the virtual directory, which contains any and all user data visible to Oracle Access Manager.

You specify the locations of your configuration, policy, and user data during installation and setup of the Identity Server, Policy Manager, and Access Server, as explained earlier.

Oracle Virtual Directory: For the latest operating support details for the host machine on which you are installing Oracle Virtual Directory, see Oracle *MetaLink* or Oracle Technology Network (OTN) at the following sites:

<https://www.metalink.oracle.com/>

<http://www.oracle.com/technology/deploy/security/index.html>

Operating System: You can implement Oracle Virtual Directory with Oracle Access Manager components installed on host machines running any of the 10g (10.1.4.0.1) supported operating systems

The LDAP directories and RDBMS databases supported by your virtual directory can be installed on any of the host platforms supported by Oracle Virtual Directory.

Java Runtime Environment: The host machine on which you install Oracle Virtual Directory must have the Java Runtime Environment installed.

The Oracle Access Manager-Oracle Virtual Directory implementation has been tested with JRE v1.4.

JNDI Driver: Use the JNDI driver that comes with your supported Oracle Virtual Directory installation package.

JDBC Driver: On the machine that hosts Oracle Virtual Directory, you must install a version of the JDBC driver appropriate for the RDBMS application you connect to the virtual directory. You can obtain the proper driver from the vendor of your RDBMS application. Oracle Virtual Directory Database Adapters support any database that provides a JDBC driver

If your virtual directory includes databases from multiple vendors, you must install a JDBC driver for each vendor represented. See the *Oracle Virtual Directory Product Manual* for details.

Data Set: The Oracle Access Manager-Oracle Virtual Directory implementation uses the UTF-8 character set. Oracle Virtual Directory is localization ready; however, only English is supported explicitly.

Relational Database: In general, Oracle Access Manager can connect to any virtual directory that includes RDBMS databases supported by Oracle Virtual Directory. For more information, see [Figure 10–2, "Oracle Virtual Directory Implementation Involving Federated RDBMS Applications"](#) on page 10-5 for a listing of RDBMS applications that are supported for the Oracle Access Manager-Oracle Virtual Directory implementation. See also ["Database Connectivity Tips"](#) on page 10-82.

Directory Server: In general, Oracle Access Manager can connect to any LDAP directory server supported by Oracle Virtual Directory. [Figure 10–1, "Oracle Virtual Directory Implementation with Federated LDAP Directories"](#) on page 10-4 lists the LDAP directory servers specifically supported for the Oracle Access Manager-Oracle Virtual Directory implementation.

Caching: Oracle Virtual Directory does not provide explicit caching support.

For additional details on Oracle Access Manager-Oracle Virtual Directory implementation requirements and support, see:

- [Security Connection Support](#)
- [Authentication Support](#)
- [Access Control Support](#)
- [Failover Support](#)

Security Connection Support

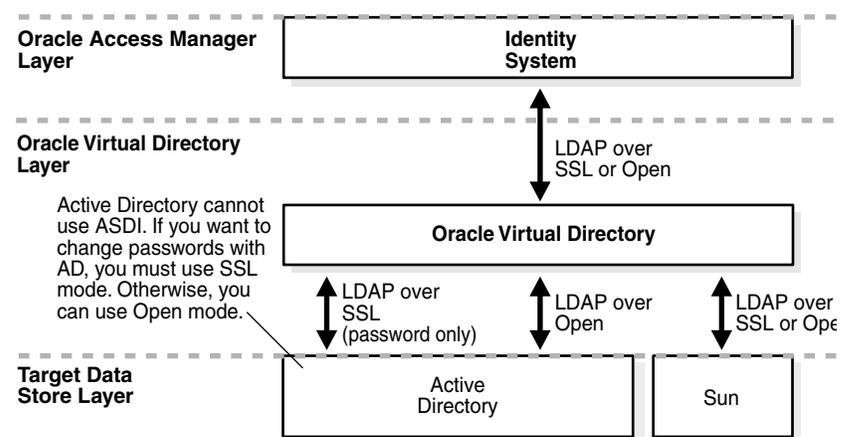
Open or SSL connections are supported between Oracle Access Manager and Oracle Virtual Directory and between Oracle Virtual Directory and the native data stores.

For Active Directory, ADSI is not supported. You must use SSL, if you want to change passwords within your Active Directory. Otherwise, you can use Open mode.

Note: For best performance when connecting an Active Directory to Oracle Virtual Directory, specify Password Only SSL as the security connection mode. For this scenario, you will also need to create an Open connection between Oracle Virtual Directory and the Active Directory.

[Figure 10–15](#) illustrates the protocols used by the connections within the Oracle Access Manager-Oracle Virtual Directory implementation.

Figure 10–15 Protocol support for a simple Oracle Access Manager-Oracle Virtual Directory Implementation



This diagram shows protocol support for the three layers discussed in detail under [Figure 10–8](#). Under each is the following:

- **Oracle Access Manager Layer:** Shows the identity system.
- **VDS Layer:** Shows the Oracle Virtual Directory layer accessing the identity system in the Oracle Access Manager layer. In turn, this layer accesses the Active Directory and Sun in the Target Data Store layer.
- **Target Data Store Layer:** Shows the Active Directory and Sun. The active directory accesses the Oracle Virtual Directory layer using LDAP SSL (password only) and LDAP over Open. Sun accesses Oracle Virtual Directory using LDAP over SSL or Open. The Active Directory cannot use ASDL. If you want to change passwords with the Active Directory, you must use SSL mode. Otherwise, you can use Open mode.

Authentication Support

Oracle Virtual Directory supports the following authentication methods:

- Pass credential authentication
- Pure proxy

About Pass Credential Authentication

If you use Pass Credential authentication for your Oracle Access Manager-Oracle Virtual Directory implementation, you must set Pass Credentials to "Always" (or Bind Only) to ensure that Oracle Virtual Directory passes the user distinguished name and password supplied by Oracle Access Manager to the proxied LDAP directory.

For background details, consult the section on directory namespace and attribute mapping in the chapter covering configurations and settings in the Oracle Virtual Directory Product Manual.

Access Control Support

Make sure that both Oracle Access Manager and Oracle Virtual Directory access control are turned on and the default settings are in effect for the connection between Oracle Access Manager and Oracle Virtual Directory. For background details, consult the chapter on security and access control in the Oracle Virtual Directory Product Manual.

For the connections between Oracle Virtual Directory and the target data stores, turn on the access control supported each target data store. (Because Oracle Virtual Directory is an LDAP client, it must use the access control implementation native to each target directory server.) For details, consult the section on access control and the LDAP adapter in the configuration and settings chapter of the Oracle Virtual Directory Product Manual.

Failover Support

The Oracle Access Manager-Oracle Virtual Directory implementation implements failover support using the existing failover capabilities in the Oracle Access Manager, Oracle Virtual Directory, directory server, and RDBMS applications. You can implement failover on the following three levels:

- Oracle Access Manager failover
- Oracle Virtual Directory target source failover
- Target data store failover

Oracle Access Manager Failover: An Identity or Access Server can connect to one or more primary virtual directory instances and one or more secondary Oracle Virtual Directory instances.

- See the section on adding database instances to an LDAP server profile in the chapter on Managing and Configuring the Identity System in the *Oracle Access Manager Identity and Common Administration Guide*.
- See the *Oracle Access Manager Deployment Guide* for details about configuring failover in Oracle Access Manager.
- See the section on fault-tolerant deployments in the chapter on virtual directory planning in the Oracle Virtual Directory Product Manual.

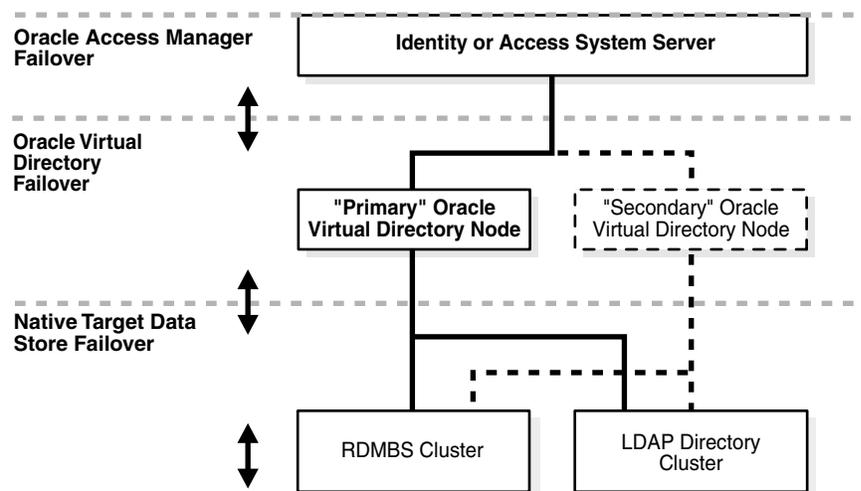
Oracle Virtual Directory Target Source Failover: Oracle Virtual Directory can implement failover protection between your virtual directory and your target data stores. For details, see the section on fault-tolerant deployments in the chapter on virtual directory planning in the *Oracle Virtual Directory Product Manual*.

Target Data Store Failover: Often, RDBMS applications and LDAP directory servers support failover in the form of clustering at the target data store level. In general, the mechanisms that implement this capability operate automatically and are not visible to Oracle Virtual Directory or Oracle Access Manager. For details, consult the documentation for your RDBMS application or LDAP directory server.

Note: This chapter does not provide any specific procedures for configuring failover for your environment. You can set up failover as you usually do, according to your product documentation.

Figure 10–16 illustrates the types of failover potentially available within a Oracle Access Manager-Oracle Virtual Directory implementation.

Figure 10–16 Failover Options for Oracle Access Manager-Oracle Virtual Directory Implementations



About the Implementation Process

The following topics introduce two different situations. Be sure to choose the topic that corresponds to your environment:

- Oracle Access Manager Not Installed: See "[Implementing Oracle Virtual Directory when Installing Oracle Access Manager](#)"
- Oracle Access Manager Installed: See "[Implementing Oracle Virtual Directory with Existing Oracle Access Manager Installations](#)"

Implementing Oracle Virtual Directory when Installing Oracle Access Manager

When you have not yet installed Oracle Access Manager, you need to complete the activities outlined to complete the implementation with Oracle Virtual Directory.

Task overview: Implementing Oracle Virtual Directory when Installing Oracle Access Manager includes

1. "[Preparing Your Environment](#)" on page 10-30
2. "[Installing and Configuring Oracle Virtual Directory and Virtual Directory Manager](#)" on page 10-34
3. "[Installing the First Identity Server](#)" on page 10-39
4. "[Extending Directory Schemas](#)" on page 10-40
5. "[Creating Mapping Files for Adapters](#)" on page 10-43
6. "[Creating Data Store Adapters](#)" on page 10-44
7. "[Customizing Adapters and Mapping Files](#)" on page 10-52
8. "[Completing Identity System Installation and Setup](#)" on page 10-66
9. "[Testing Your Implementation](#)" on page 10-67

Implementing Oracle Virtual Directory with Existing Oracle Access Manager Installations

When you have a working Oracle Access Manager installation (version 5.2.x or later) with a native directory or RDBMS, you can integrate Oracle Virtual Directory to provide access to the originally installed native directory server and additional user directories and databases.

In this case, you complete activities in the following task to upgrade your older Oracle Access Manager installation to 10g (10.1.4.0.1), prepare Oracle Virtual Directory, extend directory schemas, then reconfigure Oracle Access Manager to use the Data Anywhere option as a user data directory server.

Note: The DN Conversion Toolkit is installed automatically as part of the Identity Server installation. See *IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit\oblix*.

Task overview: Implementing Oracle Virtual Directory with existing Oracle Access Manager installations

1. In your existing Oracle Access Manager installation, confirm that user data and configuration data are stored on separate directory servers.

2. Locate the DN Conversion Toolkit in the following directory:

```
IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit\oblix\
```

3. Review the README file in the following directory for Build and Runtime requirements as well as a listing of the files and directories provided in the kit.

```
IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit
\oblix\tools\DataAnyWhere\README
```

4. Upgrade your existing Oracle Access Manager installation with the native directory to 10g (10.1.4.0.1) using instructions in the *Oracle Access Manager Upgrade Guide*.

5. Prepare any additional directories and databases to be integrated with Oracle Virtual Directory, as described in:
 - [Preparing Your Environment](#)
 - [Extending Directory Schemas](#)

6. Install and configure Oracle Virtual Directory/Virtual Directory Manager, as described in the *Oracle Virtual Directory and Virtual Directory Manager Installation Guide* and in "[Installing and Configuring Oracle Virtual Directory and Virtual Directory Manager](#)" on page 10-34.

7. Use the Oracle-provided DN Conversion Tool to clean up existing Oracle Access Manager configuration and policy data in the native directory, as follows:

- a. Locate the conversion tool in the following directory and review the README file in the Data AnyWhere directory:

```
README: IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit
\tools\DataAnyWhere\README
```

```
Conversion tool: IdentityServer_install_dir\identity\oblix\tools
\DNConversionToolkit\tools\DataAnyWhere\conversion_tools\obmigratedn.exe
```

- b. **Same Object Class:** If the object class for the new Oracle Virtual Directory virtual directory remains the same as the object class used by the original Oracle Access Manager installation, and the only change involves the namespace hierarchy, the configuration branch (Oracle Access Manager configuration data) can be patched using the obmigratedn.exe tool. This tool saves the original configurations, and then converts the native DN used by the existing Oracle Access Manager installation to the new virtual (logical) DN used by the Oracle Virtual Directory virtual directory. DN references such as the UIDs in the policy and the Oracle Access Manager Master Administrators, Master Identity Administrators and Master Access Administrators, and directory server administrators need to be converted from the old native DN to the new virtual DN.

Note: The config.lst file is generated at runtime depending on the arguments given while executing the DN Conversion Tool (obmigratedn). This same file is generated while executing the data migration tool (obmigratedata).

- c. **Different Object Class:** If the virtual object class that Oracle Virtual Directory will make visible to Oracle Access Manager differs from the virtual object class used by your original (pre-Oracle Virtual Directory) installation, you might

need to reconfigure your entire Oracle Access Manager installation, during which you typically remove the entire configuration and policy branches of the LDAP DIT.

When you complete step 7, next, to manually complete a new Identity System setup procedure be aware that you will lose any configured policies and the like.

8. **All:** Re-run setup manually for Oracle Access Manager using the following specifications, as needed:
 - a. **User Data Directory Server:** Select Data Anywhere as the directory type.
 - b. **User Data -Host:** Specify the machine hosting Oracle Virtual Directory.
 - c. **User Data -Port:** Specify the Oracle Virtual Directory LDAP licensing port.
 - d. **Searchbase:** Specify the Oracle Virtual Directory virtual DN, which may be any part of the virtual tree.
 - e. **Oracle Data Directory Server:** Select the native directory, as you normally do, because Oracle Access Manager configuration and policy data must be stored outside the Oracle Virtual Directory virtual directory, which contains user data only, as far as Oracle Access Manager is concerned.
 - f. **Automatically Update Schema:** Select Yes to automatically update the Oracle Virtual Directory schema with Oracle Access Manager auxiliary attributes.
 - g. **Automatically Configure Person and Group Object Classes:** Choose Yes or No, as you normally do, to configure the Oracle Virtual Directory schema.

Note: For details about manually configuring Person and Group object classes, see "[Specifying Person and Group Object Classes](#)" on page 6-7

Preparing Your Environment

Preparing your environment so that Oracle Access Manager can be implemented with Oracle Virtual Directory (also known as Data Anywhere), includes the following activities.

Task overview: Preparing your environment includes

1. Obtaining the DN Conversion Toolkit when you get the 10g (10.1.4.0.1) installation packages from the installation media.
2. "[Identifying Factors for Designing Your Implementation](#)" on page 10-30.
3. "[Preparing Directory Servers for Implementation](#)" on page 10-33.
4. "[Preparing Relational Databases for Implementation](#)" on page 10-33.

Identifying Factors for Designing Your Implementation

Before you start the implementation, you need to collect information and make decisions to guide the design of your Oracle Access Manager-Oracle Virtual Directory implementation.

Consider and answer the following questions, performing background investigation, as necessary.

To identify factors for this implementation

1. Determine the data stores do you want to access through Oracle Virtual Directory.

You can federate LDAP directories and RDBMS databases within your virtual directory. You can also create and federate embedded virtual data sources such as split profiles, native RDBMS Joins, and native RDBMS Views.

Qualifying LDAP Directories as Target Data Sources: The incorporation of LDAP directories is relatively straightforward, because Oracle Access Manager supplies both an adapter template and a schema mapping template for each of the LDAP directory servers Oracle Access Manager supports for Oracle Virtual Directory implementation. See "[About DN Conversion Toolkit](#)" on page 10-70

The only major restriction is that two directories cannot occupy the same namespace in a super directory, but you can prevent this sort of collision by mapping the name spaces used by the native directories to unique name spaces within the super directory. For details, see "[Aggregated Namespaces](#)" on page 10-8.

Qualifying RDBMS Databases as Target Data Sources: For RDBMS databases, you must first determine whether all the essential information Oracle Access Manager needs to operate exists within a single table. This includes the database columns that are mapped to the following attributes:

- UID (user login id)
- User password
- Full Name
- Person object class, which is generally implicit through the name of the table in which the essential fields reside. For example, all the user accounts in the Employee table of a database can be mapped to the inetorgperson person object class in the virtual directory. If you have another table such as Consultants, you can also map all of its user accounts to inetorgperson, then use the native RDBMS Join or View features to concatenate the user accounts. The important principle to observe is that all user accounts must be associated with the single person object class specified by the virtual directory.
- Whatever Oracle Access Manager user branch attributes are necessary to enable the specific features you plan to use. For instance, you can add a column labelled OOO (Out of Office) to the Employee table so that you can run workflows against the virtual directory.
- If all essential information exists within a single table, you can federate the database as part of the virtual directory. If not all the essential information exists in the database, or that information is spread across more than one table, the database might not be suitable for inclusion in the virtual directory, or you may have to use one of three available methods to transform the database into an embedded virtual data source, which you then federate within the virtual directory.

Oracle Virtual Directory RDBMS applications Native LDAP directory servers

2. Determine the items of virtual directory information you want to make visible to Oracle Access Manager.

To ensure that Oracle Access Manager can interact with the virtual directory, you must make the essential items listed in step 1 visible to Oracle Access Manager. You should also determine what customer attributes you want to make visible to Oracle Access Manager. For instance, you can make employee cell phone numbers or birthdays visible by adding those attributes to the virtual directory.

3. Determine the best approach to mapping the object class and attributes.

This depends on the native schema used by your target data stores. You are free to create virtually any schema you wish Oracle Virtual Directory to make visible to Oracle Access Manager, but you should keep in mind the following points:

- Information from two different target data stores can never occupy the same namespace in a super directory
- If your virtual directory includes any embedded virtual data stores, you should avoid workflows that create, delete, or otherwise modify user accounts, because you generally cannot change information in the secondary data stores
- The secondary data stores in embedded virtual directories (split profiles, RDBMS Joins, and RDBMS Views) cannot be included in filters or sub-searches. In other words, embedded virtual data stores such as split profiles, RDBMS Joins, and RDBMS Views are suitable for Access system operations (authentication and authorization), but they are generally inappropriate for identity management operations, because key functionality such as Add, Delete, and Modify are often not available for data in the secondary data stores.
- The Oracle Access Manager-Oracle Virtual Directory implementation does not simultaneously support attributes with multiple values and databases used as target data stores. You can use multi-valued attributes if your virtual directory contains LDAP directories exclusively.

Note: If you must use multi-valued attributes in conjunction with a database, see "[Oracle Access Manager-Oracle Virtual Directory Implementation Templates](#)" on page 10-73 and "[Multi-Value Attribute Problems](#)" on page E-10.

4. Decide what operations you want Oracle Access Manager to perform on each piece of information in the virtual directory.

If you want to use certain Oracle Access Manager features such as surrogates in workflows, then you have to add columns to the primary database tables in your embedded virtual data stores. For lists that correlate specific User Manager and Group Manager functions to auxiliary Oracle Access Manager user attributes which that must be added to the primary tables in databases serving as target data stores, see "[About Adding Attributes to Target Database Tables](#)" on page 10-17 and "[Oracle Access Manager Auxiliary Attributes](#)". on page 10-67.

5. From the standpoint of policy management and identity management, what is the optimal DIT hierarchy for your virtual directory?

You can choose between disjoint searchbases or a unified searchbase. For details, see "[About Searchbase Options](#)" on page 10-5.

If you chose the super directory option, you can optimize namespace aggregation and schema mapping to fit the needs of your organization. For instance, the engineering directories of two companies can be aggregated following a corporate merger so that only one set of access policies has to be configured for all the engineers in the new corporation. See "[Aggregated Namespaces](#)" on page 10-8 for an simple example of namespace mapping that handles such a scenario.

6. Decide what machines will host components and where to install:

- Oracle Access Manager
 - Oracle Virtual Directory
 - RDBMS applications
 - Native LDAP directory servers
7. Continue with:
- [Preparing Directory Servers for Implementation](#)
 - [Preparing Relational Databases for Implementation](#)

Preparing Directory Servers for Implementation

You need to install and configure any native directory servers that you plan to integrate into Oracle Virtual Directory. This you can accomplish now.

Note: A second requirement, which you will perform later, is to extend the native schema of each back-end directory server with Oracle Access Manager-related user and group information so your Oracle Virtual Directory and native schemas include the same Oracle Access Manager attributes.

To prepare each directory server

1. Review "[Implementation Requirements](#)" on page 10-24.
2. Install back-end directory servers according to vendor instructions.
Later you will extend the native schema with Oracle Access Manager-related attributes.
3. Proceed as follows, depending on your environment:
 - [Preparing Relational Databases for Implementation](#)
 - [Installing and Configuring Oracle Virtual Directory and Virtual Directory Manager](#)

Preparing Relational Databases for Implementation

Before you continue, you must begin preparing each of your relational databases for inclusion in the directory. This procedure is a prerequisite for creating an RDBMS-specific adapter.

To prepare each relational database for implementation

1. Install and configure your RDBMS according to vendor instructions.
2. Verify that the database contains, in a single table, all the fields that must be mapped to the essential attributes in the Oracle Access Manager schema used by the virtual directory, which are the following:
 - UID
 - User Password
 - Full Name
3. Consider the following:

- If the database does not contain all essential fields, it may not be suitable for inclusion in the virtual directory.
- If all the essential fields are not in the same table, you cannot include that table in the virtual directory (because the essential fields residing in secondary tables are not searchable). Optional, customer fields can reside the secondary tables.
- If all the essential fields are in the same table, you might have to create an embedded virtual data store using one of the following methods:
 - The Join View adapter
 - The View feature native to your RDBMS application
 - The Join feature of your RDBMS application

Note: The three methods for incorporating multi-table databases mentioned earlier, necessarily limit certain Identity System functionality.

4. From the Web site of your RDBMS application vendor, download the necessary driver libraries.

For example, for Oracle you need to download the Oracle JDBC thin driver: ojdbc14.jar (or newer).

5. Install the driver according to your vendor instructions.
6. Skip to "[Installing and Configuring Oracle Virtual Directory and Virtual Directory Manager](#)" on page 10-34.

Later you will be instructed to deploy the JDBC driver for your target database on the machine hosting Oracle Virtual Directory.

See also [Database Connectivity Tips](#) on page 10-82.

Installing and Configuring Oracle Virtual Directory and Virtual Directory Manager

After you have selected an implementation configuration and prepared your data sources, you must install both the Oracle Access Manager and Oracle Virtual Directory software necessary for the implementation. The following task overview summarizes the procedures you need to complete.

Task overview: Installing and configuring Oracle Virtual Directory and Virtual Directory Manager includes

1. ["Installing Oracle Virtual Directory"](#) on page 10-35
2. ["Installing Virtual Directory Manager"](#). on page 10-35
3. ["Creating a Project Space and Server"](#) on page 10-35
4. ["Obtaining/Updating Sample Adapter and Mapping Templates"](#) on page 10-36
5. RDBMS: ["Deploying JDBC Driver Libraries for Your RDBMS"](#) on page 37
6. ["Configuring the Oracle Virtual Directory SSL Listener \(Optional\)"](#) on page 37

Installing Oracle Virtual Directory

You install the Oracle Virtual Directory as usual. There are no specific measures you need to take to facilitate implementation with Oracle Access Manager.

To install Oracle Virtual Directory

1. Install and setup Oracle Virtual Directory following the instructions in the *Oracle Virtual Directory and Virtual Directory Manager Installation Guide*.
2. Use the default settings provided in the Oracle Virtual Directory documentation.
3. Record information about your Oracle Virtual Directory installation for use when you install the first Identity Server, including:
 - **Host Name:** The DNS hostname of the machine hosting Oracle Virtual Directory.
 - **Port Number--**The Oracle Virtual Directory LDAP licensing port.
 - **Bind DN for the user data directory server:** The Oracle Virtual Directory virtual DN, which may be any part of the virtual tree.
 - **Password:** The password for the user data bind DN.
4. Configure Oracle Virtual Directory as described in your Oracle Virtual Directory documentation.

Note: If you want to configure an SSL connection between the Oracle Virtual Directory and Oracle Access Manager, see "[Configuring the Oracle Virtual Directory SSL Listener \(Optional\)](#)" on page 10-37.

5. Proceed with "[Installing Virtual Directory Manager](#)" on page 10-35

Installing Virtual Directory Manager

You install the Virtual Directory Manager as usual. There are no specific measures you need to take to facilitate implementation with Oracle Access Manager.

To install the Virtual Directory Manager

1. Follow the instructions in the *Oracle Virtual Directory and Virtual Directory Manager Installation Guide*.
2. Use the default settings provided in your Oracle Virtual Directory documentation.
3. Configure the Virtual Directory Manager as described in your Oracle Virtual Directory documentation.
4. Proceed with "[Creating a Project Space and Server](#)" on page 10-35.

Creating a Project Space and Server

You create a project space and server using the Virtual Directory Manager, as you usually do. Some sample steps are provided here; however, these are not intended as a complete tutorial.

For additional information, see your Oracle Virtual Directory documentation.

To create a project space and server

1. From Start click VDM.

2. From the menu under the Server Navigator window, select Directory Management Project.
3. Specify a unique project name.
4. Right-click the project name, then select New > Server.
5. Enter a unique server name.
6. Click Finish.
7. Proceed to ["Obtaining/Updating Sample Adapter and Mapping Templates"](#) on page 10-36.

Obtaining/Updating Sample Adapter and Mapping Templates

You need to complete the procedure that follows to ensure that the appropriate Oracle-provided sample templates are available in the Adapter Template list within Oracle Virtual Directory. You can create your own templates from scratch if you like. Oracle provides two types of sample templates specific to each data store and also to a specific user-defined schema:

Sample Adapter Templates: Sample templates for vendor-specific adapter files that you can use as the basis for individual adapters that connect native data stores to Oracle Virtual Directory. The Oracle-provided sample adapter template files are stored in:

```
IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit  
\tools\DataAnyWhere\plugins\OracleAccessManagerOVIDTEmples  
\adapter_templates
```

Sample Mapping Templates: Sample Mapping files that you can use so that Oracle Virtual Directory can transform the schema (or database fields) used by native data stores to the logical schema used by the aggregated virtual directory made visible to Oracle Access Manager. Oracle-provided sample mapping templates are provided in:

```
IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit  
\tools\DataAnyWhere\plugins\OracleAccessManagerOVIDTEmples  
\mapping_templates
```

Note: At this point you just obtain and update the sample templates. Later, you use these to configure each connector and perform the schema and the namespace mapping. For more information, see ["Creating Mapping Files for Adapters"](#) on page 10-43 and ["Creating Data Store Adapters"](#) on page 10-44

To copy the sample templates to the Virtual Directory Manager

1. Obtain and update Oracle-provided templates using the appropriate step for your environment.
 - a. **No Samples Yet:** In the Virtual Directory Manager, Click Help, Software Updates, Find and Install.
 - b. **Automatic Update, Oracle Samples Installed:** Complete the appropriate step based on your installation:
Choose "Search for updates of currently installed features".
Choose "Search for new features to install".

- c. **Manual Update of sample templates:** Unzip the distributed zip file (for example, COREidFeatures_10.1.4.bin.dist.zip) to the Virtual Directory Manager directory, then restart the Virtual Directory Manager:

```
VDM_install_dir\  
for example  
C:\Oracle\OVid Manager
```

2. Proceed to one of the discussions that follow, depending on your environment:
 - [Deploying JDBC Driver Libraries for Your RDBMS](#)
 - [Configuring the Oracle Virtual Directory SSL Listener \(Optional\)](#)
 - [Installing the First Identity Server](#)

Deploying JDBC Driver Libraries for Your RDBMS

You complete this procedure only if your implementation will include an RDBMS.

After downloading and installing the JDBC driver libraries and completing the preceding steps, you need to complete the procedure that follows to deploy each library. For example, for Oracle you need to deploy the Oracle JDBC thin driver: ojdbc14.jar (or newer).

Again, the instructions here give you an idea of how to proceed. For more information, see your Oracle Virtual Directory documentation.

To deploy a JDBC driver library

1. Complete activities in "[Preparing Relational Databases for Implementation](#)" on page 10-33.
2. Launch the Virtual Directory Manager, as usual, and navigate to the Server Navigator window.
3. Right click Oracle Virtual Directory, then select Manage Server Libraries from the menu.
4. Select the file menu.
5. From the file menu select New, then select Deploy.

The JDBC Driver files are stored in *JDBC_Driver_install_dir/lib* (for example, C:\Program Files\JDBC).

6. Select the libraries for your environment.
 - msbase.jar
 - mssqlserver.jar
 - msutil.jar
7. Deploy as usual.

Configuring the Oracle Virtual Directory SSL Listener (Optional)

The procedure that follows is required only if you wish to set up an SSL connection between Oracle Access Manager and the Oracle Virtual Directory. Skip this section if you plan to use an Open connection.

To configure the Oracle Virtual Directory SSL listener

1. Generate a private key, as follows:

- a. Right-click server and select Server-Manager Server keys.
 - b. Click Generate Key.
 - c. Fill in the key information.

The Common Name you use must be exactly the host name you use in Oracle Access Manager later on.
2. Generate a certificate request, as follows:
 - a. Select the key just generated from Key/Certificate window.
 - b. Click Request Certificate.
3. Sign the certificate request, as follows:
 - a. Start MicroSoft certificate service using `http://machine/certsrv`
 - b. Click the link Request a Certificate.
 - c. Click the link Advanced Certificate Request.
 - d. Click the link Submit a Certificate Request by Using Base64....
 - e. In an editor, open the certificate request file generated in step 2.
 - f. Copy the text and past it to the Base64-encoded window in the certificate service.
 - g. In Certificate Template, select Web Server and then Submit.
 - h. Download the CA certificate in Base64 encoded format.
4. Import the signed certificate to Oracle Virtual Directory, as follows:
 - a. On Virtual Directory Manager Key/Certificate window, click Import.
 - b. Select the certificate file obtained in step 3.
 - c. Provide the alias exactly the same as the alias given for the key in step 1.
 - d. Once you Finish, you should see the Issuer of the key entry is updated to the CA.
5. Configure LDAP listener with SSL, as described next:
 - a. In Virtual Directory Manager Server Navigation Pane, right click Listeners and select New - Ldap Listener.
 - b. Provide a port.
 - c. In Server Key Alias, select the key entry created in 4.
 - d. Save to Server.
6. Install the certificate in Oracle Access Manager, according to the following conditions:
 - **Identity Server Not Installed:** In this case, you can install the certificate automatically during Identity Server installation. In this case, skip to "[Installing the First Identity Server](#)" on page 10-39.
 - **Identity Server Installed:** In this case, complete the following steps to create and import the certificate.
7. Create the cert8.db, if needed:
 - a. Navigate to `IdentityServer_install_dir\identity\oblix\tools\certutil`.
 - b. Issue the following command:

```
certutil -d IdentityServer_install_dir\identity\oblix\config -N -f
```

8. Import the root CA to the Identity Server using the following command:

```
certutil -d IdentityServer_install_dir\identity\oblix\config -A -n ldap -a -t  
-C -i root_ca_file
```

9. Reconfigure the Identity Server as follows:
 - a. From the Identity System Console, select System Configuration, Directory Options.
 - b. Locate the user profile and DB instance, as described in the *Oracle Access Manager Identity and Common Administration Guide*.
 - c. Mark SSL, then enter the secure port of Oracle Virtual Directory.
 - d. Restart the Identity Server.
 - e. Repeat this for all the instances for which you want to use SSL.
 - f. Rerun Identity System setup manually, as described in the *Oracle Access Manager Identity and Common Administration Guide*.

Installing the First Identity Server

For successful implementation with Oracle Virtual Directory, you must complete Oracle Access Manager installation in stages. During this first phase you install only the first Identity Server. This installation provides the ldif files you need to extend the native schema of directory servers you plan to integrate with Oracle Virtual Directory.

During Identity Server installation for Oracle Virtual Directory, you must specify the following:

- **User Data Directory Server:** Select Data Anywhere when prompted for the user data directory server.
- **Configuration and Policy Data Directory Server:** Specify a native directory server when prompted for the location of configuration and policy data. The configuration and policy branches cannot reside on the same host machine as your Oracle Virtual Directory installation.

When you finish the procedure that follows, you can extend the schema of native directory servers you plan to integrate with Oracle Virtual Directory.

Important: If you have the Identity Server installed with a native directory server, see ["Implementing Oracle Virtual Directory with Existing Oracle Access Manager Installations"](#) on page 10-28, then skip to ["Extending Directory Schemas"](#) on page 10-40

To install the first Identity Server

1. Review installation prerequisites, requirements, options, and Identity Server installation considerations in [Part I, "Installation Planning and Prerequisites"](#).
2. Start installing the first Identity Server, and proceed through defining directory server details, as described in [Chapter 4, "Installing the Identity Server"](#).

Storing Oracle Access Manager configuration data separately is required. Later you are asked to provide details about both the user data directory server and configuration data directory server.

3. When asked where data is to be stored, indicate that configuration data is to be stored separately.

Configuration data stored separately

Oracle recommends that you automatically extend the schema during installation of the first Identity Server. You update the schema only once. Either Yes response will result in questions about directory server type and specifications.

4. When asked about updating the schema, select the Automatic schema update option for separate storage of user and configuration data.
5. When asked about user data directory server details, specify the following for this implementation:

- a. **User Data Directory Server:** Data Anywhere.

- b. User Data Directory Server Details:

Host name: The DNS hostname of the machine hosting Oracle Virtual Directory.

Port number: on which the directory server listens: Specify the Oracle Virtual Directory LDAP licensing port.

Bind DN: For the user data directory server, specify the Oracle Virtual Directory virtual DN, which may be any part of the virtual tree.

Password: Specify the password for the user data bind DN.

6. When asked about configuration data directory server details, specify the following for this implementation:

- a. **Configuration Data Directory Server:** A native directory server type.

- b. **Configuration Data Directory Server Details:**

Host name: The DNS hostname of the machine hosting a native directory where you will store Oracle Access Manager configuration data.

Port number: Specify the port on which the configuration data directory server listens.

Bind DN: For the configuration data directory server.

Password: The password of the configuration data bind DN.

7. Finish installing the first Identity Server as usual.
8. Proceed to the next topic, "[Extending Directory Schemas](#)" on page 10-40 and continue through all topics.

Important: Finishing the Identity Server installation and setup is the last thing you do to complete this implementation. If you do not complete all other activities first, your implementation with Oracle Virtual Directory may not be successful.

Extending Directory Schemas

The Identity System requires attributes assigned to the Full Name, Login, and Password semantic types for Person and Group object classes.

Before you continue, you need to complete the procedure that follows to ensure that you:

- Extend back-end native directory schemas with Oracle Access Manager attributes using the appropriate ldif file in:

```
IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit
\tools\DataAnyWhere\OblixUserSchema\directory_user_schema_add.ldif
```

- Extend your Oracle Virtual Directory schema with Oracle Access Manager attributes using the VDE_user_schema_add.ldif file in:

```
IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit
\tools\DataAnyWhere\OblixUserSchema\VDE_user_schema_add.ldif
```

To extend directory schemas

1. Locate the ldif files to use when you extend the schema of a back-end directory server you are preparing for inclusion in the virtual directory, as follows:

```
IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit
\tools\DataAnyWhere\OblixUserSchema/
```

2. Use [Table 10–6](#) as a guide to manually configure attributes for each specific back-end directory server you will include in the virtual directory.

Note: If you do not find the appropriate *.ldif file in *IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit*, you may use the corresponding *.ldif located in *IdentityServer_install_dir\identity\oblix\data\common*.

Table 10–6 Files and Commands to Extend Native Schemas with Oracle Access Manager Attributes

Directory Server and ldif File	Manual Schema Update Commands
Active Directory ADUserSchema.ldif or ADAuxSchema.ldif depending on your environment	ldifde -s host -t port -a bind-dn -w password -c fromDN toDN -i -f ADUserSchema.ldif
ADAM ADAM_user_schema_add.ldif or ADAMAuxSchema.ldif depending on your environment	ldifde -s host -t port -a bind-dn -w password -c fromDN toDN -i -f ADAM_user_schema_add.ldif
SunONE ■ iplanet_user_schema_add.ldif ■ iplanet5_user_index_add.ldif	ldapmodify -h host -p port -D bind-dn -w password -a -f iplanet_user_schema_add.ldif ldapmodify -h host -p port -D bind-dn -w password -a -f iplanet5_user_index_add.ldif

Table 10–6 (Cont.) Files and Commands to Extend Native Schemas with Oracle Access Manager Attributes

Directory Server and Idif File	Manual Schema Update Commands
eDirectory <ul style="list-style-type: none"> ▪ NDS_user_schema_add.ldif ▪ NDS_user_index_add.ldif 	<pre>ldapmodify -h host -p port -D bind-dn -w password -a -f NDS_user_schema_add.ldif</pre> <pre>ldapmodify -h host -p port -D bind-dn -w password -a -f NDS_user_index_add.ldif</pre>
IBM <ul style="list-style-type: none"> ▪ v3.user.ibm_at.ldif ▪ v3.user.ibm_oc.ldif 	<pre>ldapmodify -h host -p port -D bind-dn -w password -a -f v3.user.ibm_at.ldif</pre> <pre>ldapmodify -h host -p port -D bind-dn -w password -a -f v3.user.ibm_oc.ldif</pre>

3. Repeat this procedure for each directory server in your installation.

Important: If you are working with an existing Oracle Access Manager installation, you need to manually extend the Oracle Virtual Directory schema using the following step.

4. Manually extend the Oracle Virtual Directory schema with Oracle Access Manager attributes using the VDE_user_schema_add.ldif file as follows:

```
IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit\oblix/tools
/DataAnyWhere/OblidUserSchema/VDE_user_schema_add.ldif
ldapmodify -h host -p port -D bind-dn -w password -a -f
```

5. Extend your Oracle Virtual Directory schema to represent all your back-end data sources as follows:

- Either add attributes from your back-end directory to the Oracle Virtual Directory schema:

Active Directory Example: Update/add attributes from "user" or "inetOrgPerson" object class to Oracle Virtual Directory inetOrgPerson object class.

Database Example: Add attributes from your Oracle Employees table to the inetOrgPerson object class.

- Or create a new object class having all visible attributes from the native data store.

Active Directory Example: Create a new object class (MyCompanyPerson) having all needed attributes from "user" or "inetOrgPerson" object class.

Database Example: Create a new object class (MyCompanyPerson) having all visible attributes from your Oracle Employees table to the inetOrgPerson object class.

Note: The schema extension can be done using the VDM user interface (VDM > *Your_Project*, *Your_Server*, Engine, Schema. When your extended schema is in an ldif file, use ldapmodify to load it into your Oracle Virtual Directory instance.

Creating Mapping Files for Adapters

You are ready to create the mapping files needed for the data store adapters you will develop later:

- Each mapping file results in a filter that converts a back-end schema to the front-end (Oracle Virtual Directory) schema.
- Each mapping file enables you to map inbound and outbound data from the data store to remove anything inappropriate.

You can create your own mapping files from scratch or you can use the Oracle-provided sample files, which include several plug-ins. See "[Oracle Access Manager-Oracle Virtual Directory Implementation Templates](#)" on page 10-73.

The steps that follow use the Oracle-provided samples, and are included here as a guide. The procedure is similar for both LDAP and RDBMS mapping files.

For details about using the VDM, see your Oracle Virtual Directory documentation.

To create a mapping file for a data store adapter

1. Complete activities in "[Obtaining/Updating Sample Adapter and Mapping Templates](#)" on page 10-36 so you have the sample mapping files provided by Oracle.
2. In the VDM Server Navigator window, select your Oracle Virtual Directory server.
3. From the Oracle Virtual Directory select Engine, from engine menu select Mapping.
4. Right-click Mapping, then select New Mapping.

In the New Mapping window that opens, a File list contains the names of the sample mapping templates you copied to the VDM earlier.

5. Specify the information requested:
 - File Name: Enter a unique name for the version you will modify
 - Server: Specify the server containing your project.
 - File template: Choose the sample mapping template you want.

You need to assign a new name so that your changes do not over write the sample template, which you should preserve for future use.

6. In the Name field, enter a unique name for the version you will modify.
7. Click Finish.

The name appears in the window on the left.

8. In the Virtual Directory Manager Server Navigator window, select the name of the file you just created to display it in the window on the right.

Before you finish Identity System installation and setup you must customize your mapping file and add it to the data store adapter. You can customize the mapping file now, or later.

9. Continue as follows:
 - Deploy your mapping file to the server, as usual.
 - Proceed to "[Creating Data Store Adapters](#)" on page 10-44.

You can modify the mapping script for your needs now or after you create the data store adapter. See ["Customizing Adapters and Mapping Files"](#) on page 10-52. When you customize the file and include it in an adapter:

- If you are not using the Oracle-provided sample file, you may need to create a dummy user (see ["Unexpected Group Deletion Problem"](#) on page E-12.)
- If you are using the Oracle-provided sample, this occurs automatically.

Creating Data Store Adapters

You now need to create an adapter for each data store you want to connect:

- **For Directories:** Create an LDAP adapter as discussed in ["Creating Adapters for LDAP Directories"](#) on page 10-44.
- **For Databases:** Create a database adapter as described in ["Configuring a Database Adapter"](#) on page 10-47.
- **For Split Profiles:** Create an individual adapter for each data store, then create an adapter to join the two data sources into a single view. See ["Creating a Split-Profile Adapter"](#) on page 10-49.
- **For Multiple Directories:** Create a separate adapter to connect each data source to the Oracle Virtual Directory adapter, as described in ["Creating a Multiple-Directories Adapter"](#) on page 10-51.

You can create an adapter from scratch or use the sample templates provided by Oracle, which provide you with a quick start. When you use a sample adapter template provided by Oracle, you need to fill in connection and credential information, logical root, remote root, and so on, to create the adapter. You also need to modify and tailor the settings for each data store. Once the adapter is created, the information defined in the template will be set for this adapter.

For details about Oracle templates, see ["Oracle Access Manager-Oracle Virtual Directory Implementation Templates"](#) on page 10-73. For details about modifying templates, see ["Customizing Adapters and Mapping Files"](#) on page 10-52.

Creating Adapters for LDAP Directories

The procedure for creating an LDAP adapter is similar regardless of the host directory server. You first create the LDAP adapter, then add plug-ins such as your mapping file.

As described in the following discussion, ADAM and Active Directory adapters do include some requirements that other adapters do not:

About Active Directory and ADAM Adapters: Active Directory and ADAM require two adapters each: one for SSL that must be created first and a second for an open connection that must be created second. Oracle provides individual sample templates for each of these. Setting up this environment involves:

- Creating an Active Directory or ADAM adapter for an SSL connection
- Creating an Active Directory or ADAM adapter for an open connection

There are two plug-ins that Active Directory and ADAM adapters require. If you use the Oracle-provided sample templates, the following two plug-ins are already included. However, if you create your own templates, you must add the following two plug-ins manually.

- **Active Directory Password Plug-In:** Active Directory and ADAM require the use of the secure mode to set or change a password.

To address performance concerns, a Password Only SSL mode is supported so that while normal operations are going through the adapter with the Open connection, operations related to password change/set functions are redirected to the adapter with the SSL connection.

Note: If you do not use the Oracle-provided sample templates, you need to add the SSL adapter you create as the Active Directory Password plug-in to the open-connection adapter you create, as described in "[To create an adapter for LDAP](#)" on page 10-45

- **Active Directory Ranged Attributes Plug-In:** Active Directory and ADAM require the use of the Active Directory Ranged Attributes plug-in to handle the group page issue.

This plug-in concatenates all the group pages returned by Active Directory/ADAM and returns the information to the Oracle Virtual Directory client as one result.

Note: If you use the Oracle-provided sample templates, you simply edit the value. If you do not use the Oracle-provided sample templates, you must create and add the Active Directory Ranged Attributes plug-in to the open-connection adapter you create.

One generic procedure is given for all LDAP directories. This example includes steps to create two adapters for ADAM (assuming that you are not using an Oracle-provided example).

Note: When you use Oracle-provided templates for Active Directory or ADAM, see step 17 for details about the open connector you need to create. The SSL connector is included in the Oracle template.

To create an adapter for LDAP

1. Complete activities in "[Customizing Adapters and Mapping Files](#)" on page 10-52 so you have the sample adapter templates provided by Oracle.
2. In Virtual Directory Manager, navigate to Adapters and click New, LDAP Adapter to display the Adapter configuration screen.
3. Select the appropriate adapter type from the Adapter Template list.

For example:

Adapter Template: OblixADAMSSLAdapterUsingMapper

4. Enter a unique adapter name.

For example:

Adapter Name: CustomAdamSSLAdapter

5. Fill in the server address, server proxy port, and server proxy bind DN for the LDAP Server to which you wish to connect.
6. Supply a Proxy Password and Passthrough credentials.

For example:

Proxy Password: xxxxxxxx

Passthrough credentials: Always

Specifying "Always" may impact performance; however, using "Bind Only" or "Never" is less secure.

Next you specify the connections options.

Note: When you do not use Oracle-provided templates or ADAM or Active Directory, you create an SSL adapter to include as a plug-in to an open connection adapter.

7. For Connection Options in an SSL version (not needed when using Oracle-provided templates), select:

Connection Options: Secure SSL/TLS

This step connects to the data store and downloads the certificate automatically.

8. For Remote Base, click the button labeled with an ellipsis (...).

A screen appears showing the searchbase (root DN) of the LDAP directory server you connected to.

At this point, you need to map the physical namespace to the logical namespace.

9. Select the remote physical namespace (searchbase) from the back-end data store. For example:

`ou=company,c=us,dc=intranet,dc=pspl,dc=co,dc=in`

10. In the Mapped Namespace field, enter the logical Oracle Virtual Directory namespace.

For example, if your Oracle Virtual Directory root suffix is `o=MyCompany,c=us`, you can have a mapped namespace of:

`ou=ADAM,o=MyCompany,c=us`

11. Click Finish.

You should see the newly created LDAP adapter in the Server Navigator window, under the Adapter list.

12. Click the new Adapter name in the Server Navigator window:

- a. Click the Routing tab in the right pane.
- b. In General Settings, ensure that visibility is set to internal if this is an SSL adapter for Active Directory or ADAM.

For example:

General Settings

Visibility: Internal

- c. Click Finish.

For Oracle Access Manager to function properly, DN attributes used in the product (for example, manger, secretary, uniqueMembers, and so on) need to be converted to the logical view format when viewed then back to the physical format when stored.

13. Optional: DN Attributes:
 - a. Double click the adapter you created.
 - b. In the right window, click the "Config" tab.
 - c. Under Settings, specify DN Attributes in a comma separated list of Oracle Virtual Directory attributes for all object classes/tables.
14. Right-click the adapter name in the Server Navigator window, then select Save to Server.

You have completed your first adapter and need to repeat this procedure for each data store in your implementation.

When, you need to repeat this procedure to create a second adapter for ADAM, this time with an open connection.

Note: In the following step, only the differences between the SSL adapter you created earlier and the open connection adapter you need to create for ADAM and Active Directory are identified. All other specifications remain the same.

15. ADAM/Active Directory Open Connection Adapters: Create the required open connection adapter by repeating the earlier procedure with the following differences. For example:
 - Adapter Template:** OblixADAMAdapterUsingMapper
 - Adapter Name:** CustomAdamOpenAdapter
 - Port:** open_port
 - Connection Options:** (Neither box should be checked)
 - Searchbase:** Same as the SSL adapter
 - Visibility:** Yes
16. **Optional:** DN Attributes: For Active Directory or ADAM adapters created without the Oracle-supplied sample template, this should be the same list as used in the SSL adapter created earlier.
 - a. Double click the Active Directory adapter you created.
 - b. In the right window, click the "Config" tab.
 - c. Under Settings, specify DN Attributes in a comma separated list of Oracle Virtual Directory attributes for all object classes/tables.
17. Save and Deploy, as usual.
18. Continue as follows:
 - See ["Editing an Adapter Plug-in to Refer to Your Mapping File"](#) on page 10-64.
 - See also ["Customizing Adapters and Mapping Files"](#) on page 10-52.
 - Create other LDAP adapters as needed, or create any database adapters as described next.

Configuring a Database Adapter

You can skip this procedure if it is not relevant to your environment.

The following procedure is a generic example. Your environment will vary.

To configure a database adapter

1. Complete activities in "[Deploying JDBC Driver Libraries for Your RDBMS](#)" on page 10-37
2. In Virtual Directory Manager, navigate to Adapters > New > Database Adapter to display the Adapter configuration screen.
3. Select the `OblxDBAdapterUsingScript` identified in "[Database Template: OblxDBAdapterUsingScript](#)" on page 10-80.
4. Enter a unique Adapter Name.
5. Enter logical namespace DN for the mapping.
6. Select "use predefined database".
7. Select the Type of the database you plan to connect, such as MS SQL Server.
8. Fill in the host, port, database name, username and password for the database server.
9. Click Validate Connection to see if the connection information is correct, then click Next.
10. Proceed as follows for your environment:
 - **Other Templates:** When you are not using Oracle-provided templates, complete steps 11, 12, 13, and 14 as indicated.
 - **Oracle-Provided Templates:** When using the Oracle-provided "[Database Template: OblxDBAdapterUsingScript](#)" on page 10-80 during steps 11, 12, 13, and 14 you need only click Next.
11. On the database adapter mapping Choose table screen, complete the following steps:
 - a. Select the table you want to use from the left pane
 - b. Click ">" to move it to right pane.
 - c. Click Next.
12. On database adapter mapping: Build Joins screen, click Next to skip it.
13. On the database adapter mapping: map attributes screen, complete the following steps:
 - a. Click the logical DN you specified before.
 - b. Click Add to add the hierarchy.
 - c. In the pop up window, complete the following activities:
 - In object class, fill in the LDAP object class (such as `inetorgperson`) to which you want to map.
 - In the RDN field, fill in the RDN attribute name (such as `cn`).
 - Click OK.
14. On database adapter mapping: map attributes screen, complete the following steps:
 - a. Click the node you just created (for example, "`cn= inetorgperson`").
 - b. Click Add.

- c. In the pop up window, select ldap attribute, table name, and table column.
You can type in the LDAP attribute name (such as obuseraccountcontrol) if it is not on the list.
- d. Continue this until all the attributes that you want to map are mapped.

Note: You need to map at least the cn, uid, password, and obuseraccountcontrol fields for Oracle Access Manager to work properly. Be sure that obuseraccountcontrol is Activated.

- e. Add password and obuseraccountcontrol columns to an existing table in the database if table does not contain those columns.

15. Click Finish.

You should now see the newly created DB adapter in the Server Navigator window, under the Adapter list.

16. Right-click the Adapter name in the Server Navigator window, select Save to Server.

17. Check the Client view in the Browser pane to verify the configuration.

18. All: Continue as follows:

- See also "[Customizing Adapters and Mapping Files](#)" on page 10-52.
- Add a mapping file to this adapter, as described in "[Editing an Adapter Plug-in to Refer to Your Mapping File](#)" on page 10-64.
- Create other adapters as described next.

Creating a Split-Profile Adapter

A split profile is one where you have the same users with different attributes stored on different directory servers.

The primary data store contains essential attributes while secondary data stores provide optional attributes. For example, suppose you have two different directory server types and an RDBMS. In this case, you need to create a split-profile adapter to join the views together.

Before you can create a split-profile adapter, you need to have the individual data store adapters created. Each primary and secondary adapter must have "Visibility" set to "Internal" so that only Oracle Virtual Directory will see them.

While you create the split-profile adapter, you identify the primary adapter and bind to that primary adapter. After creating the split-profile adapter, you specify join rules to identify the primary adapter and the first secondary adapter to be joined. While you specify join rules, you can indicate that you want to join the primary adapters to many secondary adapters (one to many).

Note: Oracle provides a Join View adapter template.

The searchbase (Oracle Virtual Directory refers to this as the root base) for a split-profile should be the same as that of the primary directory.

You need to ensure that the logical view of the split-profile adapter is the same as the primary data store. The split profile adapter does not map the values of DN attributes from Primary logical view to the split-profile logical view and vice versa.

The procedure that follows provides a general guide using the Join View method. For more information, see your Oracle Virtual Directory documentation. The adapters in this example include ADAM as the primary, and Sun as the secondary. In this case, you use the open-connection adapter you created for ADAM because it includes the appropriate plug-ins, including the SSL adapter. Your environment will vary.

To create a split-profile adapter

1. Create an adapter for each data store you plan to join, as described in "[Creating Adapters for LDAP Directories](#)" on page 10-44.
2. In the Virtual Directory Manager Server Navigator window, select an Oracle Virtual Directory server, then select Engine.
3. Right-click Adapters, then select New and then select the Join View Adapter.
4. In the dialog box, Adapter Template, select a default Join View template.
5. In the Adapter Name field, enter a unique name for your customized template.

For example:

Adapter Name: CustomJoinADAMSun

6. In the Adapter Suffix/Namespace list, enter the same namespace (base DN) with the DN of the primary adapter.

In this example, ADAM is the primary adapter. You must specify the name of the open-connection adapter, which includes the SSL adapter as a plug-in.

7. In Primary Adapter field, select your primary adapter.

For example:

Primary Adapter: CustomAdamAdapter

8. In the Binding Adapter list, select the same adapter.

For example:

Binding Adapter: CustomAdamAdapter

9. Click Finish.

The adapter name appears in the left pane and the Join View Primary Adapter Configuration window appears on the right.

10. In the Join View Primary Adapter Configuration window, Settings area, enter settings for the primary and binding adapters.

For example:

Settings

Primary Adapter: CustomAdamAdapter

Binding Adapter: CustomAdamAdapter

11. Beside Join Rules, click the New button to display the Enter Join Rules dialog.
12. In the Enter Join Rules dialog, select the secondary adapter to join, then select a type class, and conditions for your environment.

For example:

Joined Adapter: CustomSunAdapter

Type Class: One to Many Joiner

Conditions: cn=cn

13. Repeat step 12 to join another adapter; otherwise, skip this step.
14. Right-click the adapter name in the Server Navigator window, select Save to Server.
15. Confirm the new configuration in the Browser window, Client View.

Creating a Multiple-Directories Adapter

When you have multiple directory servers behind Oracle Virtual Directory, you need to create a local data store adapter entry within Oracle Virtual Directory, then add an entry for the Oracle Virtual Directory virtual root that is used as the searchbase for the Identity System, as outlined in the following.

Task overview: Creating a multiple-directories adapter

1. Create an adapter for each data store you plan to include, as described in:
 - [Creating Adapters for LDAP Directories](#)
 - [Configuring a Database Adapter](#)
2. Ensure that each directory server uses the same searchbase so the multiple-directories adapter will be the root.
3. Complete activities in "[Creating an Local Data Store Adapter](#)" on page 10-51.
4. Complete activities in "[Creating a Physical Node for the Virtual Root](#)" on page 10-52.

Creating an Local Data Store Adapter

The only time a local store adapter is needed with Oracle Access Manager is to create a virtual entry that is the parent of entries in multiple adapters so that it can appear as if a single contiguous tree exists.

For example, suppose you have two directories and want to create a directory tree with them:

Directory 1: ou=Marketing,o=Company

Directory 2: ou=Product,o=Company

In this case, to search from the o=Company level and have a search that covers both Directory 1 and Directory 2 you can use the local store adapter and create a single entry o=Company as its only entry. You would then have a full tree that looks like the following one:

o=Company: Oracle Access Manager can now search from here

/ \

ou=Marketing ou=Product

The local data store adapter is needed only when:

- You want a unified searchbase for all individual data store adapters
- All individual data store adapters have the same root searchbase

- No duplicate entries exist in any data store (either remove the duplicate entries or filter them out)

The steps that follow are general. For more information, see your Oracle Virtual Directory documentation.

To create an adapter entry in Oracle Virtual Directory

1. In Virtual Directory Manager, navigate to Adapters.
2. Right-click Adapters.
3. Select New, Local Store Adapter to display the Adapter configuration screen.
4. Provide the adapter suffix (the common virtual root base for all the adapters).
5. Save to the server, as usual.
6. Proceed to ["Creating a Physical Node for the Virtual Root"](#) on page 10-52.

Creating a Physical Node for the Virtual Root

After creating a local data store adapter entry for multiple directories, you need to create a physical node in the Oracle Virtual Directory directory because Identity System setup reads the configured node as the global searchbase. You create a physical node for the virtual root using the ldp utility, as usual.

For details about using the Virtual Directory Manager, see your Oracle Virtual Directory documentation.

To create a physical node for the virtual root

1. Locate the ldp or ldapmodify utility.
2. Add an entry for the Oracle Virtual Directory virtual root.

For example, if your virtual root is o=Company, c=us, you add the entry:

```
dn:o=Company,c=us
Objectclass: organization
o: Company
```

3. Ensure that each directory server uses the same searchbase so the multiple-directories adapter will be the root.
4. Proceed to on page 10-52.

Customizing Adapters and Mapping Files

The following discussions provide specifics related to use with Oracle Access Manager:

- [Customization Examples](#)
- [Customizing General Settings for Oracle Access Manager](#)
- [Customizing Routing Settings](#)
- [Editing an Adapter Plug-in to Refer to Your Mapping File](#)

Customization Examples

As mentioned earlier, you can create your own templates from scratch or customize the samples provided by Oracle. The two types of samples that Oracle provides are:

Sample Adapter Templates: Sample templates for vendor-specific adapter files that you can use as the basis for individual adapters that connect native data stores to Oracle Virtual Directory.

Note: Oracle-provided sample templates are specific to both a single data store and also to a specific user-defined schema. Your environment will vary.

Sample Mapping Files: Sample mapping files that you can use so that Oracle Virtual Directory can transform the schema (or database fields) used by native data stores to the logical schema used by the aggregated virtual directory made visible to Oracle Access Manager.

The following examples illustrate the type of modifications to Oracle-provided samples that you may want to make for your environment. The information contained in the examples, and the specific modifications made, are for illustration only. Your environment will vary.

- [Customized Mapping Script for Active Directory](#)
- [Customized Mapping Script for Oracle Database](#)
- [Customized Adapter for Oracle Database](#)

Customized Mapping Script for Active Directory

The example in this discussion shows a customized version of the Active Directory directory server mapping file. The starting point for this example is the Oracle-provided sample template for the Active Directory directory server and a specific user-defined schema that is not included here.

Note: The DN Conversion Toolkit is installed automatically as part of the Identity Server installation. See *IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit\oblix*.

To see the types of mapping script changes made for Active Directory

1. In your Virtual Directory Manager console, create a mapping file using the sample `OblixADMMapping` file as a base (see "[Creating Mapping Files for Adapters](#)" on page 10-43).

```
IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit
\oblix\tools\DataAnyWhere\plugins\OracleAccessmanagerOVIDTemplates
\mapping_templates\OblixADMMapping_mpy.xml
```

2. Modify your mapping file for your environment and compare your customized version to the one for Active Directory shown under step 4.
3. Save and deploy your mapping script, as usual.
4. Save your mapping script as a template for future use:
 - Right click the new mapping template name, for example `MyADMMapping`.
 - Select `Save as template`.

```
<?xml version="1.0" encoding="UTF-8" ?> ...
# Mapping template for: Custom Data sets
#
```

```

# Target DS: AD : - using static Auxiliary objectclass
# Target user objectclasses: User and group
# Target custom schema:
#   AD_custom_schema_add.ldif
#   AD.NET_custom_schema_add.ldif
#
# Functions:
# a. maps AD user to inetOrgPerson
# b. maps AD group to groupofuniqueNames
# c. filters out auxiliary class from objectclass in add/modify
# d. filter out AD system attributes
# e. set native flag useraccountcontrol when user is activated/deactivated
# f. set grouptype to 8
#
def inbound():
    #first rename the attributes

renameAttribute({'uniqueMember':'member','owner':'managedby','uid':'samaccountname'})

renameAttribute({'carlicense':'gencarlicense','departmentnumber':'gendepartmentnumber'})

    #temporary.
    removeAttribute('nsaccountlock')

    #map object class names
    revalueAttribute('objectclass','groupofUniqueNames','group')
    revalueAttribute('objectClass','inetOrgPerson','user')

    #If static auxiliary class is used on AD, AD does not like to mention
    #the auxiliary classes in the objectclass attribute. If dynamic auxiliary
    #class is used on AD, comment these out.
    removeAttributeValue('objectclass','person')
    removeAttributeValue('objectclass','organizationalPerson')
    #removeAttributeValue('objectclass','inetOrgPerson')
    removeAttributeValue('objectclass','oblixOrgPerson')
    removeAttributeValue('objectclass','oblixpersonpwdpolicy')
    removeAttributeValue('objectclass','oblixadvancedgroup')
    removeAttributeValue('objectclass','oblixgroup')
    removeAttributeValue('objectclass','oblixAuxLocation')
    #--- Remove custom data auxiliary object classes
    removeAttributeValue('objectclass','genAuxLocation')
    removeAttributeValue('objectclass','genAuxUserEquipment')
    removeAttributeValue('objectclass','genAuxUserNetwork')
    removeAttributeValue('objectclass','genAuxUserPersonal')
    removeAttributeValue('objectclass','genAuxUserSecurity')

    #If static auxiliary class is used in AD, remove the objectclass attribute
    #during modify. AD does not like the mentioning of the auxiliary class.
    if operation == 'modify':
        removeAttribute('objectClass')

    #set the native flag useraccountcontrol based on the value of
    obuseraccountcontrol.
    if haveAttribute('obuseraccountcontrol'):
        copyAttribute('obuseraccountcontrol','userAccountControl')
        #during modify, read the user entry first.
        if operation == 'modify':
            currentUser = getByName(name)

```

```

        val = int(`getAttributeValues(currentUser, 'userAccountControl')[0]`)
    else:
        val = 546
    #Deactivate - set the 2nd bit
    revalueAttribute('userAccountControl', 'ObWfPendingActivate', `val |
0x0002`)
    revalueAttribute('userAccountControl', 'DEACTIVATED', `val | 0x0002`)
    revalueAttribute('userAccountControl', 'ObWfPendingDeactivate', `val |
0x0002`)
    #Activate - set the 2nd bit
    revalueAttribute('userAccountControl', 'ACTIVATED', `val & ~0x0002`)

#when adding a group entry, add the grouptype and samaccountname.
#groupType is hard coded here. If multiple group types are to be supported,
#configured grouptype in VDE for user to enter.
if operation == 'add':
    if haveAttributeValue('objectClass', 'group'):
        addAttributeValue('groupType', '8')
        if not haveAttribute('samaccountname'):
            copyAttribute('cn', 'samaccountname')
        #remove these attributes as they are not in AD group. It is better not
        #configure them in COREid if not used.
        #removeAttribute ('businessCategory')
        removeAttribute ('seeAlso')
        removeAttribute ('o')

    #if haveAttributeValue('objectClass', 'user'):
        #removeAttributeValue('objectclass', 'person')
        #removeAttributeValue('objectclass', 'organizationalPerson')
        #removeAttributeValue('objectclass', 'inetOrgPerson')

if operation == 'modify':
    currentEntry = getByName(name)
    val = getAttributeValues(currentEntry, 'objectclass')
    if DirectoryString('group') in val:
        #removeAttribute ('businessCategory')
        removeAttribute ('seeAlso')
        removeAttribute ('o')

#filter out obgroupcreator otherwise iplanet user cannot create ad group.
if haveAttribute ('obgroupcreator'):
    removeAttribute ('obgroupcreator')

return

def outbound():
    #first rename the attributes

    renameAttribute({'member': 'uniqueMember', 'managedby': 'owner', 'samaccountname': '
uid'})

    renameAttribute({'gencarlicense': 'carlicense', 'gendepartmentnumber': 'department
number'})

    #map object class names
    revalueAttribute('objectClass', 'group', 'groupofUniqueNames')
    revalueAttribute('objectClass', 'user', 'inetOrgPerson')

    #filter out AD system attributes
    removeAttribute ('allowedAttributes')

```

```
removeAttribute ('allowedAttributesEffective')
removeAttribute ('allowedChildClasses')
removeAttribute ('allowedChildClassesEffective')
removeAttribute ('assistant')
removeAttribute ('bridgeheadServerListBL')
removeAttribute ('canonicalName')
removeAttribute ('createTimeStamp')
removeAttribute ('department')
removeAttribute ('distinguishedName')
removeAttribute ('dSASignature')
removeAttribute ('dSCorePropagationData')
removeAttribute ('extensionName')
removeAttribute ('flags')
removeAttribute ('fromEntry')
removeAttribute ('frsComputerReferenceBL')
removeAttribute ('frsMemberReferenceBL')
removeAttribute ('fsmORoleOwner')
removeAttribute ('generationQualifier')
removeAttribute ('instanceTyp')
removeAttribute ('isCriticalSystemObject')
removeAttribute ('isDeleted')
removeAttribute ('isPrivilegeHolder')
removeAttribute ('lastKnownParent')
removeAttribute ('managedObjects')
removeAttribute ('modifyTimeStamp')
removeAttribute ('mS-DS-ConsistencyChildCount')
removeAttribute ('mS-DS-ConsistencyGuid')
removeAttribute ('name')
removeAttribute ('netbootSCPBL')
removeAttribute ('nonSecurityMemberBL')
removeAttribute ('nTSecurityDescriptor')
removeAttribute ('objectCategory')
removeAttribute ('objectGUID')
removeAttribute ('objectVersion')
removeAttribute ('partialAttributeDeletionList')
removeAttribute ('partialAttributeSet')
removeAttribute ('possibleInferiors')
removeAttribute ('queryPolicyBL')
removeAttribute ('replPropertyMetaData')
removeAttribute ('replUpToDateVector')
removeAttribute ('revision')
removeAttribute ('sDRightsEffective')
removeAttribute ('serverReferenceBL')
removeAttribute ('showInAdvancedViewOnly')
removeAttribute ('siteObjectBL')
removeAttribute ('subRefs')
removeAttribute ('subSchemaSubEntry')
removeAttribute ('systemFlags')
removeAttribute ('uSNChanged')
removeAttribute ('uSNCreated')
removeAttribute ('uSNDSALastObjRemoved')
removeAttribute ('USNIntersite')
removeAttribute ('uSNLastObjRem')
removeAttribute ('uSNSource')
removeAttribute ('wbemPath')
removeAttribute ('wellKnownObjects')
removeAttribute ('whenChanged')
removeAttribute ('whenCreated')
removeAttribute ('instanceType')
removeAttribute ('ms-sql-olapcube')
```

```
removeAttribute ('ms-sql-database')
removeAttribute ('ms-sql-server')
removeAttribute ('ms-sql-sqlpublication')
removeAttribute ('ms-sql-sqldatabase')
removeAttribute ('ms-sql-sqlrepository')
removeAttribute ('ms-sql-sqlserver')
removeAttribute ('acpolarity')
removeAttribute ('acsubnet')
removeAttribute ('msexchconfigurationcontainer')
removeAttribute ('msmqconfiguration')
removeAttribute ('msmqenterpriseettings')
removeAttribute ('msmqmigrateduser')
removeAttribute ('msmqqueue')
removeAttribute ('msmqsettings')
removeAttribute ('msmqsitelink')
removeAttribute ('ntdsconnection')
removeAttribute ('ntdsdsa')
removeAttribute ('ntdsservice')
removeAttribute ('ntdssitesettings')
removeAttribute ('ntfrsmember')
removeAttribute ('ntfrsreplicaset')
removeAttribute ('ntfrssettings')
removeAttribute ('ntfrssubscriber')
removeAttribute ('ntfrssubscriptions')
removeAttribute ('accountExpires')
removeAttribute ('aCSPolicyName')
removeAttribute ('adminCount')
removeAttribute ('badPasswordTime')
removeAttribute ('badPwdCount')
removeAttribute ('codePage')
removeAttribute ('controlAccessRights')
removeAttribute ('dBCSPwd')
removeAttribute ('defaultClassStore')
removeAttribute ('desktopProfile')
removeAttribute ('dynamicLDAPServer')
removeAttribute ('groupMembershipSAM')
removeAttribute ('groupPriority')
removeAttribute ('groupsToIgnore')
removeAttribute ('homeDirectory')
removeAttribute ('homeDrive')
removeAttribute ('lastLogoff')
removeAttribute ('lastLogon')
removeAttribute ('lmPwdHistory')
removeAttribute ('localeID')
removeAttribute ('lockoutTime')
removeAttribute ('logonCount')
removeAttribute ('logonHours')
removeAttribute ('logonWorkstation')
removeAttribute ('mSMQDigests')
removeAttribute ('mSMQDigestsMig')
removeAttribute ('mSMQSignCertificates')
removeAttribute ('mSMQSignCertificatesMig')
removeAttribute ('msNPAllowDialin')
removeAttribute ('msNPCallingStationID')
removeAttribute ('msNPSavedCallingStationID')
removeAttribute ('msRADIUSCallbackNumber')
removeAttribute ('msRADIUSFramedIPAddress')
removeAttribute ('msRADIUSFramedRoute')
removeAttribute ('msRADIUSServiceType')
removeAttribute ('msRASSavedCallbackNumber')
```

```

removeAttribute ('msRASSavedFramedIPAddress')
removeAttribute ('msRASSavedFramedRoute')
removeAttribute ('networkAddress')
removeAttribute ('ntPwdHistory')
removeAttribute ('operatorCount')
removeAttribute ('otherLoginWorkstations')
removeAttribute ('preferredOU')
removeAttribute ('primaryGroupID')
removeAttribute ('profilePath')
removeAttribute ('pwdLastSet')
removeAttribute ('scriptPath')
removeAttribute ('servicePrincipalName')
removeAttribute ('userAccountControl')
removeAttribute ('userParameters')
removeAttribute ('userSharedFolder')
removeAttribute ('userSharedFolderOther')
removeAttribute ('userSMIMECertificate')
removeAttribute ('userWorkstations')
removeAttribute ('masteredBy')
removeAttribute ('maxStorage')
removeAttribute ('userPrincipalName')
removeAttribute ('objectSid')
removeAttribute ('samaccounttype')
removeAttribute ('badPasswordCount')
removeAttribute ('sAMAccountControl')
removeAttribute ('ADsPath')
removeAttribute ('directReport')

return
</ldap>
</adapters>

```

Customized Mapping Script for Oracle Database

The example in this discussion shows the sample `OblxDBMapping` file as it looks after being customized for an Oracle database that uses the SQL server as a back end with a user-defined schema. The original sample is located in:

```

IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit
\oblix\tools\DataAnyWhere\plugins\OracleAccessmanagerOVidTemplates
\mapping_templates\OblxDBMapping_mpy.xml

```

Note: The DN Conversion Toolkit is installed automatically as part of the Identity Server installation. See `IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit\oblix\`. Be sure to see the README file that comes with the toolkit.

You can compare the original Oracle-provided sample with the following one, to see the types of changes that are needed.

To see mapping script changes for the Oracle Database

1. In your Virtual Directory Manager console, create a mapping file using the sample `OblxDBMapping` file as a base (see "[Creating Mapping Files for Adapters](#)" on page 10-43).

2. Modify the mapping file for your environment and include the work around shown in the example under "[Unexpected Group Deletion Problem](#)" on page E-12.
3. Save and deploy your mapping script, as usual.
4. Save your mapping script as a template for future use:
 - Right click the new mapping template name, for example MyOracleDBMapping.
 - Select Save as template.
5. See also the customized adapter for the Oracle database next.

```
<?xml version="1.0" encoding="UTF-8"?>
<variables>
</variables>
<content>
def inbound():
    #These Oblix attributes are not being used. Remove them.
    removeAttribute('obver')
    removeAttribute('nsaccountlock')

    # More custom mapping
    # ....

    # If your user password is stored as character type, for example
    # NVARCHAR, CHAR, VARCHAR, etc, you need to map userPassword attribute
    # from binary syntax.
    mapSyntax('userPassword', 'IA5String')

    # This is a workaround ... for more information, see " Unexpected Group
    Deletion Problem" on page E-12.
    # Need to prevent COREid from writing dummy user to backend database
    if haveAttributeValue('uniqueMember', 'cn=Dummy User'):
        #removeAttributeValue('uniqueMember', 'cn=Dummy User')
        if operation != 'modify':
            removeAttributeValue('uniqueMember', 'cn=Dummy User')
        else:
            change = removeAttribute('uniqueMember')[0]
            change.values.remove(DistinguishedName('cn=Dummy User'))
            addEntryChange(change)

    #Filter out objectclass. Only mention the structure class during add.
    if operation == 'modify':
        removeAttribute('objectClass')
    if operation == 'add':
        newobj = ''
        if haveAttributeValue('objectClass', 'inetOrgPerson'):
            newobj = 'inetOrgPerson'
        if haveAttributeValue('objectClass', 'groupOfUniqueNames'):
            newobj = 'groupOfUniqueNames'
            removeAttribute ('businessCategory')
            removeAttribute ('seeAlso')
            removeAttribute ('o')
        if haveAttributeValue('objectClass', 'oblixlocation'):
            newobj = 'oblixlocation'
        if not newobj == '':
            removeAttribute('objectClass')
            addAttributeValue('objectClass', newobj)

return
```

```

def outbound():
    #code here for handling outbound mapping
    # ....
    # This is a workaround to bug #18865
    if operation=='entry':
        # Add the following workaround for each multiple value DN attribute
        if haveAttribute('uniqueMember') and len(findFilters('uniqueMember')) > 0:
            addAttributeValue('uniqueMember', 'cn=Dummy User')
    return
</content>

```

Customized Adapter for Oracle Database

The following example shows a sample adapter after being customized for the Oracle Database. The Oracle-supplied sample template was used as a starting point.

```

<?xml version="1.0" encoding="UTF-8"?>
<adapters dirty="" version="0"
  xmlns="http://www.octetstring.com/schemas/Adapters" xmlns:adapters="http://
www.w3.org/2001/XMLSchema-instance">
  <dataBase dirty="" id="DB Adapter Company Employees" version="0">
    <root>ou=Employees,o=MyCompanyDB,c=us</root>
    <active>true</active>
    <routing>
      <critical>true</critical>
      <priority>50</priority>
      <inclusionFilter/>
      <exclusionFilter/>
      <plugin/>
      <retrieve/>
      <store>
        <exclude>carlicense</exclude>
        <exclude>street</exclude>
        <exclude>employeeType</exclude>
      </store>
      <visible>Internal</visible>
      <levels>-1</levels>
      <bind>true</bind>
      <bind-adapters/>
      <views/>
      <dnpattern/>
    </routing>
    <pluginChains xmlns="http://www.octetstring.com/schemas/Plugins">
      <plugins>
        <plugin>
          <name>MyOracleDBMapping</name>
          <class>com.octetstring.VDE.chain.plugins.mapper.Mapper</class>
          <initParams>
            <param name="mapfile" value="MyOracleDBMapping.mpy" />
          </initParams>
        </plugin>
        <plugin>
          <name>Dump after</name>
          <class>com.octetstring.VDE.chain.plugins.DumpTransactions.DumpTransactions</
class>
          <initParams>
            <param name="loglevel" value="info" />
          </initParams>
        </plugin>

```

```

    <plugin>
      <name>Dump before</name>

<class>com.octetstring.VDE.chain.plugins.DumpTransactions.DumpTransactions</
class>
    <initParams>
      <param name="loglevel" value="info"/>
    </initParams>
  </plugin>
</plugins>
<default>
  <plugin name="Dump before"/>
  <plugin name="MyOracleDBMapping"/>
  <plugin name="Dump after"/>
</default>
</pluginChains>
<driver>oracle.jdbc.driver.OracleDriver</driver>
<url>jdbc:oracle:thin:@127.0.0.1:1521:QA2</url>
<user>CUSTDATA</user>
<password>oblix</password>
<ignoreObjectClassOnModify>>false</ignoreObjectClassOnModify>
<includeInheritedObjectClasses>>true</includeInheritedObjectClasses>
<maxConnections>10</maxConnections>
<mapping>
  <joins/>
  <objectClass name="inetOrgPerson" rdn="cn">
    <attribute field="EMPLOYEE_ID" ldap="uid"
      table="CUSTDATA.EMPLOYEES" type="VARCHAR"/>
    <attribute field="NAME" ldap="cn" table="CUSTDATA.EMPLOYEES"
type="VARCHAR"/>
    <attribute field="FIRST_NAME" ldap="givenName"
      table="CUSTDATA.EMPLOYEES" type="VARCHAR"/>
    <attribute field="LAST_NAME" ldap="sn"
      table="CUSTDATA.EMPLOYEES" type="VARCHAR"/>
    <attribute field="TITLE" ldap="title" table="CUSTDATA.EMPLOYEES"
type="VARCHAR"/>
    <attribute field="USERPASSWORD" ldap="userPassword"
      table="CUSTDATA.EMPLOYEES" type="VARCHAR"/>
    <attribute field="PREFERREDLANGUAGE"
      ldap="PreferredLanguage" table="CUSTDATA.EMPLOYEES" type="CHAR"/>
    <attribute field="MAIL" ldap="mail" table="CUSTDATA.EMPLOYEES"
type="VARCHAR"/>
    <attribute field="CHALLENGEPHRASE" ldap="ChallengePhrase"
      table="CUSTDATA.EMPLOYEES" type="CHAR"/>
    <attribute field="PHOTO" ldap="Photo"
      table="CUSTDATA.EMPLOYEES" type="BLOB"/>
    <attribute field="DESCRIPTION" ldap="Description"
      table="CUSTDATA.EMPLOYEES" type="VARCHAR"/>
    <attribute field="OBUSERACCOUNTCONTROL"
      ldap="OBUSERACCOUNTCONTROL" table="CUSTDATA.EMPLOYEES" type="VARCHAR"/>
    <attribute field="OBLOGINTRYCOUNT" ldap="oblogintrycount"
      table="CUSTDATA.EMPLOYEES" type="NUMERIC"/>
    <attribute field="OBPASSWORDCREATIONDATE"
      ldap="obpasswordcreationdate" table="CUSTDATA.EMPLOYEES" type="VARCHAR"/
>
    <attribute field="OBPASSWORDHISTORY" ldap="obpasswordhistory"
      table="CUSTDATA.EMPLOYEES" type="VARCHAR"/>
    <attribute field="OBPASSWORDCHANGEFLAG"
      ldap="obpasswordchangeflag" table="CUSTDATA.EMPLOYEES" type="VARCHAR"/>
    <attribute field="OBPASSWORDEXPIRATIONDATE" ldap="obpasswordexpirationdate"
      table="CUSTDATA.EMPLOYEES" type="VARCHAR"/>
  </objectClass>
</mapping>

```

```

        table="CUSTDATA.EMPLOYEES" type="VARCHAR" />
<attribute field="OBLOCKOUTTIME" ldap="oblockouttime"
        table="CUSTDATA.EMPLOYEES" type="VARCHAR" />
<attribute field="OBFIRSTLOGIN" ldap="obfirstlogin"
        table="CUSTDATA.EMPLOYEES" type="VARCHAR" />
<attribute field="OBRESPONSETRIES" ldap="obresponsetries"
        table="CUSTDATA.EMPLOYEES" type="VARCHAR" />
<attribute field="OBLASTLOGINATTEMPTDATE"
        ldap="oblastloginattemptdate" table="CUSTDATA.EMPLOYEES" type="VARCHAR" /
>
<attribute field="OBLASTRESPONSEATTEMPTDATE"
        ldap="oblastresponseattemptdate" table="CUSTDATA.EMPLOYEES"
type="VARCHAR" />
<attribute field="OBRESPONSETIMEOUT" ldap="obresponsetimeout"
        table="CUSTDATA.EMPLOYEES" type="VARCHAR" />
<attribute field="MANAGER_DN" ldap="Manager"
        table="CUSTDATA.EMPLOYEES" type="VARCHAR" />
</objectClass>
<objectClass name="groupOfUniqueNames" rdn="cn">
<attribute field="GROUP_NAME" ldap="cn"
        table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="OWNER_DN" ldap="owner"
        table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="MEMBER_DN" ldap="uniqueMember"
        table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="MAIL" ldap="mail" table="CUSTDATA.GROUPS"
type="VARCHAR" />
<attribute field="OBGROUPCREATOR" ldap="obgroupcreator"
        table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="OBGROUPCREATIONDATE"
        ldap="obgroupcreationdate" table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="OBGROUPTYPE" ldap="obgroupstype"
        table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="OBSUBSCRIPTIONTYPES"
        ldap="obsubscriptiontypes" table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="OBVER" ldap="obver"
        table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="OBGROUPSUBSCRIPTIONTYPE"
        ldap="obgroupsubscriptiontype" table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="OBGROUPEXPANDEDYNAMIC"
        ldap="obgroupexpandeddynamic" table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="OBGROUPEXPANDEDYNAMIC" ldap="obgroupexpandeddynamic"
        table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="OBGROUPEXPANDEDYNAMIC" ldap="obgroupexpandeddynamic"
        table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="OBGROUPADMINISTRATOR"
        ldap="obgroupadministrator" table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="OBGROUPSUBSCRIBEMESSAGE"
        ldap="obgroupsubscribemessage" table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="OBGROUPUNSUBSCRIBEMESSAGE"
        ldap="obgroupunsubscribemessage" table="CUSTDATA.GROUPS" type="VARCHAR" /
>
<attribute field="OBGROUPSUBSCRIPTIONFILTER"
        ldap="obgroupsubscriptionfilter" table="CUSTDATA.GROUPS" type="VARCHAR" /
>
<attribute field="OBGROUPSUBSCRIBENOTIFICATION"
        ldap="obgroupsubscribenotification"
        table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="OBGROUPDYNAMICFILTER"
        ldap="obgroupdynamicfilter" table="CUSTDATA.GROUPS" type="VARCHAR" />
<attribute field="OBGROUPSIMPLIFIEDACCESSCONTROL"
        ldap="obgroupsimplifiedaccesscontrol "

```

```

        table="CUSTDATA.GROUPS" type="VARCHAR"/>
    <attribute field="GROUP_DESC" ldap="description"
        table="CUSTDATA.GROUPS" type="VARCHAR"/>
</objectClass>
<objectClass name="oblixlocation" rdn="obid">
    <attribute field="OBID" ldap="obid"
        table="CUSTDATA.OBLIXLOCATION" type="VARCHAR"/>
    <attribute field="OBLOCATIONNAME" ldap="oblocationname"
        table="CUSTDATA.OBLIXLOCATION" type="VARCHAR"/>
    <attribute field="OBLOCATIONTITLE" ldap="oblocationtitle"
        table="CUSTDATA.OBLIXLOCATION" type="VARCHAR"/>
    <attribute field="OBPARENTLOCATIONDN" ldap="obparentlocationdn"
        table="CUSTDATA.OBLIXLOCATION" type="VARCHAR"/>
    <attribute field="OBRECTANGLE" ldap="obrectangle"
        table="CUSTDATA.OBLIXLOCATION" type="VARCHAR"/>
    <attribute field="OBPHOTO" ldap="obphoto"
        table="CUSTDATA.OBLIXLOCATION" type="BLOB"/>
</objectClass>
</mapping>
</dataBase>
</adapters>

```

Customizing General Settings for Oracle Access Manager

General adapter configuration and setup information is provided in the Oracle Virtual Directory/Virtual Directory Manager product manual. Once an adapter is created, default values are valid for most places. The following highlights are concerns with Oracle Access Manager:

- DN attributes
- Connection Information

DN Attributes:

- DN attributes should be set with the attribute names that are in DN syntax.

These DN attributes could exist in the customer schema, or could be introduced by Oracle Access Manager auxiliary classes. This tells Oracle Virtual Directory to store the values of these DN's in their native form instead of logical form.

- The DN attributes are related to the object classes you are using:
 - **For inetorgperson and groupofuniquenames**, use
uniquemember, manager, secretary, owner
 - **For user and group**, use
member, memberOf, managedObjects, distinguishedname, objectcategory, manager, secretary, managedby
 - **For Oracle Access Manager introduced auxiliary classes**, use
obgroupadministrator, obgroupcreator
 - In Pass-Through Mode, select
Always

Connection Information: Set the following connection information in Oracle Virtual Directory based on the estimated workload:

Operation timeout
Max Pool Connection

Max Pool Wait
Max Pool Tries

To customize general settings for Oracle Access Manager

1. Review the preceding information.
2. In the Virtual Directory Manager Server Navigator window, select a server, then locate and select the name of the adapter.
3. In the right pane, click the Routing tab.
4. In General Settings, ensure that visibility is set to internal.

For example:

General Settings

Visibility: Internal (for split profile adapters)

5. Ensure that your adapter uses the settings discussed in this section.
6. Click Finish.
7. Right-click the adapter name in the Server Navigator window, then select Save to Server.

Customizing Routing Settings

If you are setting up an adapter that will be used by the Join View adapter, the visibility of the primary and secondary adapters should be set to "internal" so they are only invoked by the Join View adapter.

Once your adapters are working, you should observe the performance and evaluate the log to see if there is a pattern that an adapter is unnecessarily invoked by certain operation. If yes, you should try to use the following to filter block the unnecessary operation for that particular adapter. This step is very important to improve the overall performance:

- Filter to include
- Filter to exclude
- DN matching

To customize routing settings

1. Select the Adapter name in the Server Navigator window.
2. Click the Routing tab in the right pane.
3. Select the options for your environment:
 - Filter to include
 - Filter to exclude
 - DN matching
4. Save to the server, as usual.

Editing an Adapter Plug-in to Refer to Your Mapping File

The Oracle-provided sample adapter templates include several plug-ins hooked in already, as discussed in "[Oracle Access Manager-Oracle Virtual Directory Implementation Templates](#)" on page 10-73. There are two types of plug-ins:

- **Plug-in:** A predefined plug-in that provides a parameter-based user interface for configuration.
- **Mapping Plug-in:** A plug-in for a mapping script.

Keep the following in mind as you work:

- If you use the Oracle-provided sample templates, you need only modify plug-ins.
- If you do not use the Oracle-provided sample templates you need to add plug-ins to your adapter. For example:

As discussed earlier, Active Directory and ADAM adapters require two specific plug-ins, which are included in the Oracle-provided sample templates.

If you do not use the Oracle-provided sample templates, you need to add the SSL adapter you create as the Active Directory Password plug-in to the open-connection adapter you create and also add the Active Directory Ranged Attributes plug-in. Be sure to specify the same mapped namespace as the SSL adapter. See "[Creating Adapters for LDAP Directories](#)" on page 10-44.

The following is an example only. For details about using Virtual Directory Manager, see your Oracle Virtual Directory documentation.

To edit an adapter plug-in to refer to your mapping file

1. Complete activities in:
 - [Creating Mapping Files for Adapters](#)
 - [Creating Data Store Adapters](#)
2. In the Virtual Directory Manager Server Navigator window, select a project and server, then locate and select the name of the adapter to which you want to add or verify a plug-in.
3. In the right pane, click the Plug-ins tab.
4. On the Adapter Plug-ins screen:
 - a. For ADAM or Active directory, you need the following plug-ins in the following order:
 - Active Directory Ranged Attributes OblixADMMapping (should be the mapping file you created earlier)
 - Active Directory Password
 - b. Arrange plug-ins for your environment using the up and down arrows.
 - c. When the mapping file you created earlier is not listed:

Select the current mapping, for example, OblixADMMapping, then click the Edit button.

In the Name field, change the name to your mapping file name (for example MyADMMapping).

In the Name field, change the name to your mapping file name (for example MyADMMapping).

- d. When using non-Oracle templates for Active Directory or ADAM adapters, add the SSL adapter you created earlier to handle the password.

Click the New Plug-in button.

Click **Select from Server**, then select the **Active Directory Password** plug-in.

Specify your SSL adapter name as the value of the parameter, for example: `CustomAdamSSLAdapter`.

Select a parameter line, then click **Edit**.

Specify your ADAM or Active Directory SSL adapter name as the value, for example: `CustomAdamSSLAdapter`

5. Finish, save, and deploy as usual.

Completing Identity System Installation and Setup

Now that you have completed all other essential activities, described earlier, you are ready to complete Identity System installation and setup.

To complete Identity System installation and setup

1. Complete all preceding tasks.
2. **WebPass:** Install WebPass as described in [Chapter 5, "Installing WebPass"](#).
3. **Identity System Set Up:** Set up the Identity System using the specifications that follow, then finish setup as you normally do:
 - a. **User Data Directory Server:** Select **Data Anywhere** as the directory type.
 - b. **User Data ... Host:** Specify the machine hosting Oracle Virtual Directory.
 - c. **User Data ... Port:** Specify the Oracle Virtual Directory LDAP licensing port.
 - d. **Searchbase:** Specify the Oracle Virtual Directory virtual DN, which may be any part of the virtual tree.
 - e. **Configuration Data Directory Server:** Select the native directory you specified during Identity Server installation. Configuration (and policy data) must be stored outside the Oracle Virtual Directory virtual directory.
 - f. **Automatically Update Schema:** Select **Yes** to automatically update the Oracle Virtual Directory schema with Oracle Access Manager auxiliary attributes.
 - g. **Automatically Configure Person and Group Object Classes:** Choose **Yes** or **No**, as you normally do, to configure the Oracle Virtual Directory schema.

Note: For details about manually configuring Person and Group object classes, see ["Specifying Person and Group Object Classes"](#) on page 6-7.

4. **Policy Manager:** Install and set up the Policy Manager as described in [on page 7-1](#) using specific details that follow, then complete setup as usual:
 - a. Specify user data directory server details during setup, as described earlier.
 - b. Specify the following during Policy Manager setup:

Searchbase: Must be the same searchbase you specified during Identity System setup.

Configuration DN: Must be the same configuration data DN you specified during Identity System setup.

Policy Base: Must be the same policy data DN you specified during Access Manager installation.

5. **Access Server:** Install the Access Server as described in [Chapter 8, "Installing the Access Server"](#) and:
 - a. Provide information for the configuration data directory server.
 - b. Identify where the Oracle Access Manager policy data is stored.
 - c. Provide the following information when asked:

Access Server ID

Configuration DN Must be the same as specified earlier.

Policy Base: Must be the same as specified earlier.

6. **WebGate:** Install the WebGate as described in [Chapter 9, "Installing the WebGate"](#).
7. **Failover:** Configure failover as described in:
 - [Failover Support](#)
 - *Oracle Access Manager Deployment Guide*
 - Your product documentation.

Testing Your Implementation

To test your implementation, simply perform a Oracle Access Manager function that requires obtaining user data from a native directory.

If the operation works, the implementation is a success.

Reference Information

The following sections provide technical details related to several aspects of the Oracle Virtual Directory implementation for Oracle Access Manager.

- [Oracle Access Manager Auxiliary Attributes](#)
- [About DN Conversion Toolkit](#)
- [Oracle Access Manager-Oracle Virtual Directory Implementation Templates](#)

Oracle Access Manager Auxiliary Attributes

Certain Oracle Access Manager functions require that specific attributes exist in the schema of both your top-level virtual directory and in the schema (or database equivalent) of each target data store.

- You can extend your virtual directory schema automatically as follows:
 - When you install and set up the Identity System you have the opportunity to automatically (or manually) extend the Oracle Virtual Directory schema when you choose Data Anywhere as the user data directory server. For Identity System installation and setup, see [Part II, "Identity System Installation and Setup"](#).

- After you upgrade an older installation to Oracle Access Manager 10g (10.1.4.0.1) you must manually extend the Oracle Virtual Directory schema using the ldapmodify utility as discussed in ["Extending Directory Schemas"](#) on page 10-40.
- You extend the target LDAP directory schemas by running the ldapmodify.exe utility with the appropriate ldif file, as described in ["Extending Directory Schemas"](#) on page 10-40.
- You simulate the object classes of your virtual directory by mapping all the user accounts in your primary database tables to the appropriate classes. See ["About Adding Attributes to Target Database Tables"](#) on page 10-17.
- You simulate the auxiliary user attributes that enable special Oracle Access Manager features by creating extra columns in your primary database tables. See ["About Adding Attributes to Target Database Tables"](#) on page 10-17.

You use NVARCHAR with length of 1000 for all unbounded or user data, and VARCHAR with a length of 240 for all Oracle Access Manager-specific valued attributes.

[Table 10-7](#) correlates the Oracle Access Manager auxiliary attributes required for specific User Manager functions.

Note: The use of Oracle Virtual Directory does not support the Location object; therefore the User Location function is not supported for the User Manager application.

Table 10-7 Extended Attributes Required by User Manager Functions

User Manager Function	Required Attributes	Suggestions for Attribute Type and Length
User Add/Activate/Deactivate	Obuseraccountcontrol	VARCHAR (240)
Workflow Surrogate	oboutofofficeindicator	VARCHAR (240)
Password Change on Reset	obpasswordchangeflag	VARCHAR (240)
Password Number of login tries allowed	oblogintrycount	VARCHAR (240)
Password Validity period	obpasswordcreationdate	VARCHAR (240)
Password Expiry Notice Period	obpasswordcreationdate	VARCHAR (240)
Password Lockout Duration	oblockouttime	VARCHAR (240)
Password Login tries reset	oblastloginattemptdate	VARCHAR (240)
Password minimum age	obpasswordcreationdate	VARCHAR (240)
Password history	obpasswordhistory password history support optional	NVARCHAR (1000)
Challenge Response	Customer attributes for Challenge phrase and response	NVARCHAR (1000)
Challenge Response Login tries reset	oblastresponseattemptdate	VARCHAR (240)

Table 10–7 (Cont.) Extended Attributes Required by User Manager Functions

User Manager Function	Required Attributes	Suggestions for Attribute Type and Length
Challenge Response Lockout Duration	Obresponsetimeout oblockouttime	VARCHAR (240) VARCHAR (240)
Challenge Response Number of login tries allowed	obresponsetries	VARCHAR (240)

Table 10–8 correlates the Oracle Access Manager auxiliary attributes required for specific Group Manager functions.

Table 10–8 Extended Attributes Required by Group Manager Function

Group Manager Function	Required Attributes	Suggestions for Attribute Type and Length
Subscription type	obgroupsubscriptiontype	VARCHAR (240)
Group expansion	obgroupexpandeddynamic	VARCHAR (240)
Pure dynamic group	obgrouppuredynamic	VARCHAR (240)
Group administrators	obgroupadministrator The virtual directory must support exactly one administrator for each group.	NPVARCHAR (1000)
Subscription message	obgroupsubscribemessage	NPVARCHAR (1000)
Unsubscription message	obgroupunsubscribemessage	NPVARCHAR (1000)
Subscription filters	obgroupsubscriptionfilter	NPVARCHAR (1000)
Subscription notification types	The virtual directory must support exactly one subscription for each group.	NPVARCHAR (1000)
Dynamic filters	obgroupsubscribenotification Either subscription or unsubscription notification can be implemented, but both functions cannot be implemented simultaneously.	NPVARCHAR (1000)
Simplified access control	obgroupdynamicfilter The virtual directory must support exactly one dynamic filter for each group	
Group types	obgroupstype The virtual directory must support exactly one group type for each group.	
Selectable subscription types	obsubscriptiontypes The virtual directory must support exactly one subscription type for each group. The possible subscription types are: Open, Close, Open with filter, and Controlled through workflow.	NPVARCHAR (1000)

About DN Conversion Toolkit

DN Conversion Toolkit is for use when you have an existing Oracle Access Manager installation and you want to integrate Oracle Virtual Directory. It will convert all the user data-related native DN suffixes in the configuration/policy tree to logical DN suffixes.

The DN Conversion Toolkit is installed automatically as part of the Identity Server installation. [Table 10-9](#) identifies the contents of the toolkit.

Note: Be sure to see the README file that comes with the toolkit. Any *.ldif files that are not in the DN Conversion Toolkit may be found in *IdentityServer_install_dir\identity\oblix\data\common*.

Table 10-9 Contents of the DN Conversion Toolkit for Oracle Access Manager

DNConversionToolkit oblix	File Components	Description
\apps\common\bin\	globalparams.xml	A file that controls the scope of searches in the searchbase, among other things.
\lib	obxmlengine.dll (windows) obxerces-c21.dll (windows) msvci70.dll (windows) msvci70d.dll (windows) msvcr70.dll (windows) msvcr70d.dll (windows) libxmlengine.so (solaris) libstdc++.so.5 (solaris) libgcc_s.so.1 (solaris) all required ldap sdk libraries for solaris	Libraries for Windows and Solaris.
\tools\DataAnyWhere	README	An overview of the content, runtime requirements, and simple usage examples for the components of the DN Conversion Toolkit.
\tools\DataAnyWhere \conversion_tools	obmigrateDN.exe obmigrateDNmsg.lst	The DNConversion binary and configuration file. When you integrate Oracle Virtual Directory with existing Identity Server installations, obmigrateDN converts the user and group DN in the Oracle Access Manager configuration tree by internally calling obmigratedata for handling the Oracle Virtual Directory DN specific operations, which then refers to the ldapmodify executable.
\tools\DataAnywhere \features\ \OracleAccessManage rOVIDFeatures	feature.xml	

Table 10-9 (Cont.) Contents of the DN Conversion Toolkit for Oracle Access Manager

DNConversionToolkit oblix	File Components	Description
\tools\DataAnyWhere \OblidUserSchema	ADUserSchema.ldif ADAuxSchema.ldif ADAM_user_schema_add.ldif ADAMAuxSchema.ldif f iPlanet_user_schema_add.ldif iPlanet_user_schema_delete.ldif NDS_user_schema_add.ldif NDS_user_schema_delete.ldif v3.user.ibm_at.ldif v3.user.ibm_oc.ldif schema.oblix.xml VDE_user_schema_add.ldif VDE_user_schema_delete.ldif Note: When possible, use the ldif files provided with your Identity Server installation.	schema.oblix.xml extends the Oracle Access Manager user schema into your virtual directory VDE_user_schema_add.ldif extends the Oracle Virtual Directory schema with Oracle Access Manager attributes. The other files extend the Oracle Access Manager user schema to the directory servers supported by the Oracle Access Manager-Oracle Virtual Directory implementation.
\tools\DataAnywhere \plugins \OracleAccessmanage rOVidTemplates	plugin.xml	
\tools\DataAnywhere \plugins \OracleAccessmanage rOVidTemplates \adapter_templates	OblixADAdapterUsingMapper_adapter_ template.xml OblixADAdapterUsingScript_adapter_ template.xml OblixADSSLAdapterUsingMapper_adapter_ template.xml OblixADAMAdapterUsingMapper_adapter_ template.xml OblixADAMAdapterUsingScript_adapter_ template.xml OblixADAMSSLAdapterUsingMapper_a dapter_template.xml OblixSunOneAdapterUsingMapper_adapter_ template.xml OblixSunOneAdapterUsingScript_adapter_ template.xml For additional information, see ""Oracle Access Manager-Oracle Virtual Directory Implementation Templates" on page 10-73.	Oracle Virtual Directory adapter templates for directory servers that require templates. These can serve as a starting point for adapter creation. Each includes basic settings, pre-configured data, plug-ins, and plug-in parameters. See "Creating Data Store Adapters" on page 10-44.

Table 10–9 (Cont.) Contents of the DN Conversion Toolkit for Oracle Access Manager

DNConversionToolkit oblix	File Components	Description
\tools\DataAnywhere \plugins \OracleAccessmanage rOVidTemplates \mapping_templates	OblixADAMMapping_mpy.xml OblixADMMapping_mpy.xml OblixDBMapping_mpy.xml OblixeDirectoryMapping_mpy.xml OblixSunOneMapping_mpy.xml	Mapping script templates. These sample mappings achieve the same configuration as those produced by the Object Class Mapper plug-in in the adapter template. Mapping scripts are more flexible and can produce a fine level of adjustment not available through the plug-in. See Creating Mapping Files for Adapters on page 10-43.
\tools \ldap_tools	ldapmodify.exe libobnspr4.dll libobplc4.dll libobplds4.dll obnsldap32v50.dll obnsldappr32v50.dll obnsldapssl32v50.dll obnss3.dll obsoftokn3.dll obsoftokn3.dll	The ldapmodify tool for use with the DN Conversion Toolkit. Library files for Windows.
\tools \migration_tools \obmigratedata	at_DN_Conversion_map_osd_offline.lst oc_DN_Conversion_map_osd_offline.lst at_DN_Conversion_map_osd_online.lst oc_DN_Conversion_map_osd_online.lst obmigratedata.exe obmigratedatamsg.lst	Files required by obmigratedata during runtime. They have the objectclass and attribute details that need special handling. This is similar to typical upgrades, except the DNConversion tool requires special handling in offline and online mode (as mentioned in the README file).

The conversion tool changes the native DNs in the configuration and policy branches of the Oracle Access Manager configuration and policy tree into logical (virtual) DNs that can be used by the virtual directory. The following tool is used:

```
IdentityServer_install_dir\identity\oblix\tools\DNConversionToolkit\oblix\tools
\DataAnywhere\conversion_tools \obmigrateDN.exe
```

Conditions

For the conversion to occur successfully, your Oracle Access Manager directory must meet the following two conditions:

- Oracle Access Manager configuration and policy data resides in a native directory outside Oracle Virtual Directory.
- Oracle Access Manager must see this configuration and policy data as belonging to a directory distinct from the Oracle Virtual Directory virtual directory, which contains all the user data seen by the product.

Requirements

- A file containing the list of DN attributes to be converted
- A mapping list you create, which correlates native DNs to logical DNs
- Host name, Port number, Bind DN, Password, Directory type, Config DN, Oblix node, Install Dir, Native DN, Logical DN, Mode (online, offline, test)

Details

- The tool only performs conversion when the domains differ in Oracle Virtual Directory. For example:

If the DN on Oracle Access Manager is:

```
o=company, c=us
```

And the DN for iPlanet on Oracle Virtual Directory is:

```
o=iPlanet, o=company, c=us
```

Then the conversion takes place by referring to the mapping details given as input.

- If only the attributes themselves differ, no mapping occurs. For example:

```
cn=manisha, o=company, c= us
```

cannot be mapped to Oracle Virtual Directory

```
cn=manisha, o=iPlanet, o=company, c=us
```

- You must run the tool at least once to convert each DN value.

Note: If the configuration branch is not on the same directory server as the policy branch, you must run the tool twice for each DN value.

- The tool does not support SSL.
- Loading of the DSML version of the schema is not automated.
- If you are upgrading an existing Oracle Access Manager installation, before re-running system setup you must manually remove the DBProfile branch from the directory being integrated.

Note: Be sure to remove old DBProfiles manually before system setup; after setup the new DBProfiles are created automatically

- The tool does not support Active Directory Services Interface (ADSI); you must use SSL instead, if you want a secure connection.

Oracle Access Manager-Oracle Virtual Directory Implementation Templates

Oracle provides adapter templates and script templates to assist you with quick setup of each directory and a database. When you configure an adapter, you can choose a template described later to complete schema mapping and special handling. Depending on the mapping criteria, these templates can be used as they are or they can provide a base for tailoring.

The provided templates are listed in [Table 10–9, "Contents of the DN Conversion Toolkit for Oracle Access Manager"](#) To fully understand what each template can achieve, see the following discussions:

- [Templates for Active Directory](#)
- [Templates for ADAM](#)
- [Templates for Sun Directory Server](#)
- [Templates for eDirectory](#)
- [Templates for eDirectory](#)
- [Schema Mapping Script Templates](#)

Note: ObjectClass Mapper templates are a plug-in with parameter-based user interface. ObjectClass mapper templates and script templates perform the same operations and produce the same results. Using the script may be preferable to some and provide greater freedom while using the mapper may be preferable to others. The flexibility is yours to choose.

For additional information, see "[Creating Data Store Adapters](#)" on page 10-44 and "[About DN Conversion Toolkit](#)" on page 10-70.

Templates for Active Directory

Oracle Access Manager provides three templates for use with Active Directory:

- [OblixADAdapterUsingMapper for Active Directory](#)
- [OblixADAdapterUsingScript for Active Directory](#)
- [OblixADSSLAdapterUsingMapper for Active Directory](#)

OblixADAdapterUsingMapper for Active Directory

This template defines an adapter that converts Active Directory user and group data to Oracle Virtual Directory inetorgperson and groupofuniquenames using the Object Mapper plug-in, which includes the following:

1. A mechanism to set the DN attributes with all attributes that have DN syntax from inetorgperson, groupofuniquenames, and Oracle Access Manager user auxiliary classes to ensure these DNs are stored in the native DN format.
2. A plug-in for Active Directory ranged attributes
Active Directory returns the entry as xxx bytes chunks. This plug-in concatenates all the chunks into a single result entry.
3. A plug-in for the ObjectClass Mapper, which provides a parameter-based user interface for object class and attribute mappings as described in [Table 10–10](#) is also included.

Table 10–10 *OblixADAdapterUsingMapper, ObjectClass Mapper Plug-in Parameters*

Parameter	Value	Comment
filterObjectClassOnModify	true	Assume Active Directory is configured as static auxiliary class.

Table 10–10 (Cont.) OblixADAdapterUsingMapper, ObjectClass Mapper Plug-in

Parameter	Value	Comment
addAttribute-group	samaccountname=%cn %	If the object class is group, add attribute samaccountname and set the value to be equal to cn. This is because Active Directory requires samaccountname for group.
addAttribute-group	groupType=4	If the object class is group, add attribute groupType and set the default value to 4. The value can be changed based on customer needs.
mapAttribute	uniqueMember=member	Map Oracle Virtual Directory attribute uniqueMember to Active Directory attribute member.
mapAttribute	owner=managedBy	Map Oracle Virtual Directory attribute owner to Active Directory attribute managedBy.
mapAttribute	uid=samaccountname	Map Oracle Virtual Directory attribute uid to Active Directory attribute samaccountName.
filterAttribute-group	see Also, businessCategory	If the object class is group, filter out these attributes. This is because Active Directory group does not support these three attributes.
mapObjectClass	groupofuniqueNames=group	Map Oracle Virtual Directory objectclass groupOfUniqueNames to Active Directory objectclass group.
mapObjectClass	inetorgperson=user	Map Oracle Virtual Directory objectclass inetorgperson to Active Directory objectclass user.
filterAttribute	(list of system attributes)	Filter out all the attributes in the list. Active Directory has a long list of system attributes that we don't want Oracle Access Manager to see.
directoryType	ActiveDirectory	The directory type.
activationAttribute	obuseraccountcontrol	The Oracle Access Manager attribute name that the Active Directory adapter should use to find for activation and deactivation. The Active Directory adapter then sets the native flag useraccountcontrol based on this.
activationValue	ACTIVATED	The activation value of obuseraccountcontrol.
deactivationValue	DEACTIVATED ObWfPendingActivate ObWfPendingDeactivate	The deactivation values of obuseraccountcontrol.

Table 10–10 (Cont.) OblixADAdapterUsingMapper, ObjectClass Mapper Plug-in

Parameter	Value	Comment
filterAuxiliaryClass	person, organizationalPerson OblixOrgPerson, oblixpersonpwdpolicy oblixadvancedgroup oblixgroup, oblixAuxLocation	The auxiliary classes to be filtered out. This is based on the assumption that the Active Directory is configured as static auxiliary class.

- A plug-in for the Active Directory password, which requires the use of the SSL connection to set or change the user password using the parameters in Table 53, is included.

Note: If the current adapter is using the open connection, this plug-in redirects password set/change to an adapter configured with an SSL connection.

Table 10–11 Active Directory Password Plug-in Parameters and Values

parameter	value	comment
adapter	AD SSL Adapter	Redirect to the adapter defined in template OblixADSSLAdapterUsingMapper.
mapPassword	(not set. Default is true)	Map password attribute from userPassword to unicodePwd.

OblixADAdapterUsingScript for Active Directory

This template (OblixADAdapterUsingScript) achieves exactly the same result as described earlier, and includes the same items described in 1, 2, and 4 of "[OblixADAdapterUsingMapper for Active Directory](#)" on page 10-74.

The only difference when using the OblixADAdapterUsingScript is item 3, which in this case will be:

- A plug-in script written in Python, defined in the mapping script template OblixADMMapping, with the parameters in [Table 10–10, "OblixADAdapterUsingMapper, ObjectClass Mapper Plug-in Parameters"](#) accomplishes everything stated for the ObjectClass Mapper in item 3 of "[OblixADAdapterUsingMapper for Active Directory](#)" on page 10-74.

OblixADSSLAdapterUsingMapper for Active Directory

This template defines an adapter that connects to the Active Directory through SSL. This is for the redirected adapter identified in item 4 of the preceding discussions. See:

- [OblixADAdapterUsingScript for Active Directory](#)
- [OblixADAdapterUsingScript for Active Directory](#)

Templates for ADAM

Three templates are provided for ADAM:

- [OblixADAMAdapterUsingMapper](#) for ADAM
- [OblixADAMAdapterUsingScript](#) for ADAM
- [OblixADAMSSLAdapterUsingMapper](#) for ADAM

OblixADAMAdapterUsingMapper for ADAM

This template (OblixADAMAdapterUsingMapper) defines an adapter that converts ADAM user and group data to Oracle Virtual Directory inetorgperson and groupofuniquenames using the Object Mapper plug-in, which includes the following:

1. A mechanism to set the DN attributes with all attributes that have DN syntax from inetorgperson, groupofuniquenames and Oracle Access Manager user auxiliary classes to ensure these DNs are stored in the native DN format.
2. A plug-in for the Active Directory Ranged Attributes.
ADAM also returns the entry as xxx bytes chunks. This plug-in concatenates all the chunks to a single-result entry.
3. A plug-in for the ObjectClass Mapper, which provides a parameter-based user interface for object class and attribute mappings, as described in [Table 10–12](#) is also included.

Table 10–12 *OblixADAMAdapterUsingMapper, ObjectClass Mapper Parameters and Values*

Parameter	Value	Comment
filterObjectClassOnModify	(not set. Default is false)	Assume dynamic auxiliary class. Set it to true if ADAM is configured as static auxiliary class.
addAttribute-group	groupType=4	If the object class is group, add attribute groupType and set the default value to 4. The value can be changed based on customer needs.
mapAttribute	uniqueMember=member	Map Oracle Virtual Directory attribute uniqueMember to ADAM attribute member.
mapAttribute	owner=managedBy	Map Oracle Virtual Directory attribute owner to ADAM attribute managedBy.
mapAttribute	uid=samaccountname	Map Oracle Virtual Directory attribute uid to ADAM attribute samaccountName.
filterAttribute-group	seeAlso,businessCategory,o	If the object class is group, filter out these attributes. This is because Active Directory group does not support these three attributes.
mapObjectClass	groupofuniquenames=group	Map Oracle Virtual Directory objectclass groupOfUniqueNames to Active Directory objectclass group.
mapObjectClass	inetorgperson=user	Map Oracle Virtual Directory objectclass inetorgperson to ADAM objectclass user.
filterAttribute	(list of system attributes)	Filter out all the attributes in the list. ADAM has a long list of system attributes that we don't want Oracle Access Manager to see.
directoryType	ADAM	The directory type.

Table 10–12 (Cont.) OblixADAMAdapterUsingMapper, ObjectClass Mapper Parameters and Values

Parameter	Value	Comment
activationAttribute	obuseraccountcontrol	The Oracle Access Manager attribute name that the ADAM adapter should use to find for activation and deactivation. The ADAM adapter then sets the native flag useraccountcontrol based on this.
activationValue	ACTIVATED	The activation value of obuseraccountcontrol.
deactivationValue	DEACTIVATED ObWfPendingActivate ObWfPendingDeactivate	The deactivation values of obuseraccountcontrol
filterAuxiliaryClass	(not set)	The auxiliary classes to be filtered out. This is based on the assumption that the ADAM is configured as dynamic auxiliary class. Give auxiliary class names for this parameter if ADAM is configured as static auxiliary class.

4. A plug-in for the Active Directory password (ADAM requires the use of the SSL connection to set or change the user password using the parameters in Table 55), is included.

Note: If the current adapter is using the open connection, this plug-in redirects password set/change to an adapter configured with an SSL connection.

Table 10–13 OblixADAMAdapterUsingMapper, ActiveDirectory Password Parameters

Parameter	Value	Comment
adapter	ADAM SSL Adapter	Redirect to the adapter defined in template OblixADAMSSLAdapterUsingMapper.
mapPassword	false	Do no map password attribute because ADAM uses attribute userPassword.

OblixADAMAdapterUsingScript for ADAM

This template (OblixADAMAdapterUsingScript) achieves exactly the same result as described earlier, and includes the same items described in 1, 2, and 4 of "OblixADAMAdapterUsingMapper for ADAM" on page 10-77.

The only difference when using the OblixADAMAdapterUsingScript is item 3, which in this case will be:

3. A plug-in script written in Python, defined in template OblixADAMMapping, with the parameters in Table 10–12, "OblixADAMAdapterUsingMapper, ObjectClass Mapper Parameters and Values" accomplishes everything stated for the ObjectClass Mapper in item 3 in "OblixADAMSSLAdapterUsingMapper for ADAM" on page 10-79.

OblixADAMSSLAdapterUsingMapper for ADAM

This template defines an adapter that connects to the ADAM directory through SSL. It is for the redirected adapter identified in item 4 of the preceding discussions. See:

- [OblixADAMAdapterUsingMapper for ADAM](#)
- [OblixADAMAdapterUsingScript for ADAM](#)

Templates for Sun Directory Server

Two templates are provided for the Sun Directory Server:

- [OblixSunOneAdapterUsingMapper for SunOne](#)
- [OblixSunOneAdapterUsingScript for SunOne](#)

OblixSunOneAdapterUsingMapper for SunOne

This template defines an adapter that converts Sun Directory Server (formerly SunOne) inetorgperson and groupofuniquenames to Oracle Virtual Directory inetorgperson and groupofuniquenames using the Object Mapper plug-in, which includes the following:

1. A mechanism to set the DN attributes with all attributes that have DN syntax from inetorgperson, groupofuniquenames and Oracle Access Manager user auxiliary classes. This is to ensure these DNs are stored in the native DN format.
2. A plug-in (ObjectClass Mapper), which provides a parameter based user interface for object class and attribute mappings as shown in [Table 10–14](#) is also included.

Table 10–14 *OblixSunOneAdapterUsingMapper, ObjectClass Mapper Parameters*

Parameter	Value	Comment
directoryType	SunOne	The directory type.
activationAttribute	obuseraccountcontrol	The Oracle Access Manager attribute name that the SunOne adapter should use to find for activation and deactivation. The SunOne adapter then sets the native flag nsaccountlock based on this.
activationValue	ACTIVATED	The activation value of obuseraccountcontrol.
deactivationValue	DEACTIVATED, ObWfPendingActivate, ObWfPendingDeactivate	The deactivation values of obuseraccountcontrol.

OblixSunOneAdapterUsingScript for SunOne

This template (OblixSunOneAdapterUsingScript) achieves exactly the same result as described earlier, and includes the same item described in 1 of "[OblixSunOneAdapterUsingMapper for SunOne](#)" on page 10-79.

The only difference when using the OblixSunOneAdapterUsingScript is item 2, which in this case will be:

2. A script written in Python, defined in template OblixSunOneMapping, with the parameters in [Table 10–14](#), "[OblixSunOneAdapterUsingMapper, ObjectClass Mapper Parameters](#)". This table accomplishes everything stated for the ObjectClass Mapper in item 2 of "[OblixSunOneAdapterUsingMapper for SunOne](#)" on page 10-79.

Templates for eDirectory

Two templates are provided for eDirectory:

- [OblixeDirectoryAdapterUsingMapper for eDirectory](#)
- [OblixeDirectoryAdapterUsingScript for eDirectory](#)

OblixeDirectoryAdapterUsingMapper for eDirectory

This template defines an adapter that converts eDirectory inetorgperson and groupofuniquenames to Oracle Virtual Directory inetorgperson and groupofuniquenames using the Object Mapper plug-in, which includes the following:

1. A mechanism to set the DN attributes with all attributes that have DN syntax from inetorgperson, groupofuniquenames and Oracle Access Manager user auxiliary classes. This is to ensure these DNs are stored in the native DN format.
2. A plug-in (ObjectClass Mapper), which provides a parameter based user interface for object class and attribute mappings as shown in [Table 10-15](#).

Table 10-15 *OblixeDirectoryAdapterUsingMapper for eDirectory*

Parameter	Value	Comment
directoryType	SunOne	The directory type.
activationAttribute	obuseraccountcontrol	The Oracle Access Manager attribute name that the eDirectory adapter should use to find for activation and deactivation. The eDirectory adapter then sets the native flag logindisabled.
activationValue	ACTIVATED	The activation value of obuseraccountcontrol.
deactivationValue	DEACTIVATED, ObWfPendingActivate, ObWfPendingDeactivate	The deactivation values of obuseraccountcontrol.

OblixeDirectoryAdapterUsingScript for eDirectory

This template (OblixeDirectoryAdapterUsingScript) achieves exactly the same result as stated earlier using OblixeDirectoryAdapterUsingMapper.

The only difference when using the OblixeDirectoryAdapterUsingScript is item 2, which in this case will be:

2. A script written in Python, defined in template OblixeDirectoryMapping, with the parameters in "[OblixeDirectoryAdapterUsingMapper for eDirectory](#)" on page 10-80 accomplishes everything stated for the ObjectClass Mapper in item 2 of "[OblixeDirectoryAdapterUsingMapper for eDirectory](#)" on page 10-80.

Database Template: OblixDBAdapterUsingScript

This template defines an adapter for a database. It does not include specific mapping but does call the OblixDBMapping script. The script, defined in the template OblixDBMapping, is written in Python and filters out the unnecessary mention of objectclass during the LDAP operation.

Schema Mapping Script Templates

The following mapping script templates are used by the adapter templates described earlier. These sample mappings achieve the same configuration as those produced by

the Object Class Mapper plug-in in the adapter template. Mapping scripts are more flexible and can produce a fine level of adjustment not available through the plug-in. These mapping script templates provide a script alternative to accomplishing the schema mapping and special handling:

OblixADMMapping: This mapping template performs the following:

- Converts the Active Directory user and group to inetorgperson and groupofuniquenames, respectively
- Sets the native flag when a user is activated or deactivated
- Handles the static auxiliary objectclass
- Sets grouptype to 4
- Sets the useraccountname so that it is identical to cn.

OblixADAMMapping: This mapping template performs the following:

- Converts Active Directory user and group to inetorgperson/groupofuniquenames, respectively
- Sets the native flag when a user is activated or deactivated
- Handles the static auxiliary object class
- Sets grouptype to 4.

OblixeDirectoryMapping: Sets the native flag when a user is activated or deactivated

OblixSunOneMapping: Sets the native flag when a user is activated or deactivated

Tips

The following section provides miscellaneous information to guide your implementation with Oracle Virtual Directory. See also "[Database Connectivity Tips](#)" on page 10-82.

Mapping DN: The mapped DN is the logical DN in Oracle Virtual Directory. However, there is no physical node for the mapped DN.

If the application (Identity System for example) needs to search the logical DN to detect that DN's existence or to retrieve its attributes, the entry needs to be added manually (using the `ldp.exe` utility, for example). For example, if the mapping DN is `o=virtual company`, the corresponding entry needs to be created through `ldp.exe` so that:

- `objectclass: organization`
- `o: virtual_company`

where *organization* is xxx, and *virtual_company* is xxx.

Reference DN in Configuration and Policy Data: The reference DN such as a UID used in policy data, is in its logical form. This means that it is stored as the DN of Oracle Virtual Directory, not the native directory. As a result, once the Oracle Virtual Directory namespace mapping is completed, that mapping should not be changed. Changing the namespace mapping will impact the reference DN's stored in the Oracle Access Manager configuration and policy data.

Schema Mapping: When mapping an attribute from logical to native, be sensitive to the syntax and whether it is multi-valued or single-valued.

- For Oracle Virtual Directory to directory mapping, the syntax should be kept the same except for minor adjustments of string syntaxes.
- For Oracle Virtual Directory to database mapping, use [Table 10–16, "Oracle Virtual Directory to Database Mapping"](#) as a guideline.

Table 10–16 Oracle Virtual Directory to Database Mapping

LDAP Attribute Syntax	MS SQL Data Type
1.3.6.1.4.1.1466.115.121.1.5 DESC 'Binary'	binary
1.3.6.1.4.1.1466.115.121.1.6 DESC 'Bit String'	binary
1.3.6.1.4.1.1466.115.121.1.7 DESC 'Boolean'	varchar
1.3.6.1.4.1.1466.115.121.1.8 DESC 'Certificate'binary	binary
1.3.6.1.4.1.1466.115.121.1.9 DESC 'Certificate List'	binary
1.3.6.1.4.1.1466.115.121.1.10 DESC 'Certificate Pair'	binary
1.3.6.1.4.1.1466.115.121.1.11 DESC 'Country String'	varchar
1.3.6.1.4.1.1466.115.121.1.12 DESC 'DN'	varchar
1.3.6.1.4.1.1466.115.121.1.15 DESC 'Directory String'	varchar
1.3.6.1.4.1.1466.115.121.1.22 DESC 'Facsimile Telephone Number'	varchar
1.3.6.1.4.1.1466.115.121.1.23 DESC 'Fax'	tvchar
1.3.6.1.4.1.1466.115.121.1.24 DESC 'Generalized Time'	timestamp
1.3.6.1.4.1.1466.115.121.1.26 DESC 'IA5 String'	binary
1.3.6.1.4.1.1466.115.121.1.27 DESC 'INTEGER'	Numeric / int
1.3.6.1.4.1.1466.115.121.1.28 DESC 'JPEG'	binary
1.3.6.1.4.1.1466.115.121.1.33 DESC 'MHS OR Address'	varchar
1.3.6.1.4.1.1466.115.121.1.36 DESC 'Numeric String'	varchar
1.3.6.1.4.1.1466.115.121.1.38 DESC 'OID'	varchar
1.3.6.1.4.1.1466.115.121.1.39 DESC 'Other Mailbox'	varchar
1.3.6.1.4.1.1466.115.121.1.41 DESC 'Postal Address'	varchar
1.3.6.1.4.1.1466.115.121.1.44 DESC 'Printable String'	varchar
1.3.6.1.4.1.1466.115.121.1.50 DESC 'Telephone Number'	varchar
1.3.6.1.4.1.1466.115.121.1.53 DESC 'UTC Time'	timestamp

Database Connectivity Tips

Following are several database connectivity considerations:

- Entry Name Formation
- Multi-Table Writes
- Multi-Value Attributes
- Searches
- Writes
- Cascading Deletes

Entry Name Formation: All database fields that will be used as part of an entry's name (with the exception of the "base" part of the entry's name) must be contained in the rows that are mapped and returned to Oracle Virtual Directory through the database adapter.

For example, if a hierarchy is being created in which user objects will contain both a common name (cn) and an organizational unit (ou) in their name (cn=Joe User,ou=Marketing), both cn and ou must be part of the entry being created.

In pure LDAP, the ou attribute would not be required as it is part of the parent entry. Since databases are not hierarchical, this is the only reasonable way to support this functionality without requiring considerable new metadata be created and managed to define hierarchy.

Multi-Table Writes: Multi-table writes are not possible directly to a single database adapter. This is not a design limitation of Oracle Virtual Directory, but rather a practical database limitation. For example, views in most databases cannot be updated directly when they present multiple tables.

Oracle Virtual Directory gets around this limitation through the use of its own Join View implementation. By creating multiple database adapters (perhaps one for each table) and defining the relationship between them, it is possible to have Oracle Virtual Directory manage writes to entries that are constructed through multiple tables. Further information on creating Join Views can be found in Oracle Virtual Directory Product Manual.

Multi-Value Attributes: Databases typically do not allow for multiple values for a single field within a single table row. There are exceptions where an array type is supported, but these data types tend to be relatively limited. Some users put multiple values into a row by separating data (such as account flags) within a field using delimiters such as commas or pipes (|).

Traditional database design states that fields that would have multiple values should be normalized into an additional table. Databases that are part of a data warehouse may take a different approach in which every permutation of every field is placed into a denormalized table.

Oracle Virtual Directory can generally support either model. Oracle suggests using the "permutations in a denormalized table" method only as an absolute last resort. The normalized, secondary table approach probably won't return consistently accurate results for Oracle Access Manager searches.

For more information about multi-valued attributes, see:

- [Oracle Access Manager-Oracle Virtual Directory Implementation Templates](#)
- [Multi-Value Attribute Problems](#) on page E-10

Searches: Searches are supported to normalized or denormalized tables without doing anything other than configuring a database-level join as necessary.

The one consideration that should be kept in mind on searches is that due to the most popular use cases, the current design of the database adapter is that if a multi-value attribute is searched to only return the value that matched the search as part of the entry. This facilitates high performance large group searches. Oracle expects to make this configurable (to support doing a subselect to return all values) in a future version. An additional search can be performed in mapping to retrieve all attributes in the mean time.

Writes: Writes to normalized tables must be performed through a Join View that splits out attributes to each table based on the design of the database. This is required since

while there are general guidelines for database design, every customer's database is different.

Most customers that use existing and important tables also use stored procedures as part of controlling updates to those tables. These are akin to API calls and are proprietary to each database in the way they are constructed and called, while the calls themselves are proprietary to the customer. Oracle supports stored procedures through the use of its plug-in system.

Oracle Virtual Directory can manage direct writes to denormalized tables that have each value for the field that will be used in the entry associated with the field used as the RDN for the entry. Add, Modify, and Delete are all supported.

Within the modify operation, it should be noted that the way the modify->replace works is to remove existing attribute values and add new ones. This translates into a SQL delete and a SQL insert rather than a complex set of SQL inserts, updates, and deletes. The potential issue here is with customers that have normalized tables in which the insert or delete will trigger other actions within the database. In such a plugin would need to be constructed that would handle the modify->replace operation.

Most customers do not run into this as they either are not using direct SQL access for changes, are using multiple values only for read, or are only using modify->add and modify->remove directly. For example, customers solving issues with big groups are storing groups in databases through Oracle Virtual Directory. Most group membership changes are adds and removes rather than replaces.

Cascading Deletes: Of the issues mentioned in writes, the biggest thing to watch for in an existing database where Oracle Virtual Directory will be handling database writes directly is a customer database's use of cascading deletes. With cascading deletes it is possible that a modify->replace as documented in the previous section would trigger deletes outside of the table being directly affected.

This said, if the trigger is based on the normalized table, this is not an issue as when modifying the single-valued normalized table Oracle Virtual Directory will do an SQL update rather than a delete-insert sequence. This should eliminate the issue mentioned.

If in doubt about what will happen with a particular customer situation with a potentially dangerous setup involving Cascading Deletes, contact Oracle. You can also turn up the debug level on Oracle Virtual Directory and point it to a test database to see the SQL being generated by Oracle Virtual Directory for any sequence of LDAP operations.

Troubleshooting Implementations with Oracle Virtual Directory

For information, see ["Issues with Oracle Virtual Directory Implementations"](#) on page E-10.

Installing the SNMP Agent

This chapter describes how to install the Simple Network Management Protocol (SNMP) Agent that enables you to monitor the activity of different components on your network. See:

- [About the SNMP Agent and Installation](#)
- [SNMP Agent Installation Considerations](#)
- [SNMP Installation Prerequisites Checklist](#)
- [SNMP Installation Prerequisites Checklist](#)

For details about configuring the SNMP agent after installation, see the *Oracle Access Manager Identity and Common Administration Guide*.

About the SNMP Agent and Installation

Oracle Access Manager provides data that can be used by SNMP and a Network Management System (NMS), which enables you to monitor the status and activity of the Identity and Access Servers.

The SNMP Agent is an optional component. If installed, the SNMP Agent accesses information about the Identity or Access Server resident on the same server host on which the agent was installed. The installation process for the SNMP Agent is similar to other Oracle Access Manager components. The following shows the installation directory:

Default on Windows: \Program Files\OracleAccessManager_SnmpAgent\snmp

Default on Unix: /opt/OracleAccessManager_SnmpAgent\snmp

In This Guide: \SNMP_install_dir\snmp

SNMP Agent Installation Considerations

The SNMP Agent must be installed on the same machine as the Oracle Access Manager server that it is going to service: Identity Server or Access Server. The Oracle Access Manager SNMP Agent should run as the same user as the Identity Server or Access Server.

You need a community name for SNMP data in your network management station (NMS) host. You also need to configure trap destinations for Oracle Access Manager SNMP traps in your NMS host.

The SNMP Agent needs to be owned by a dedicated user. On Unix, only root or the dedicated user may be able to start the agent service. Most of the time the agent is run as "root" or "nobody". On Unix, you also enter a group to which the user belongs.

SNMP Installation Prerequisites Checklist

Table 11-1 provides a checklist of items that must be completed before you install the SNMP Agent with Oracle Access Manager.

Table 11-1 WebGate Prerequisites Checklist

Checklist	SNMP Prerequisites
	Review and complete all prerequisites and requirements that apply to your environment, as described in Part I, "Installation Planning and Prerequisites"
	Complete all activities in Part II, "Identity System Installation and Setup" .
	Complete all activities in Part III, "Access System Installation and Setup"
	Create a Community Name for SNMP data in your network management station (NMS) host.
	Configure trap destinations for SNMP traps in your NMS host.

Installing the Oracle Access Manager SNMP Agent

The following discussions explain how to install the SNMP agent.

To install the SNMP Agent includes

1. ["Starting the Installation"](#) on page 11-2
2. ["Installing the Oracle Access Manager SNMP Agent"](#) on page 11-2
3. ["Specifying SNMP Agent Configuration Details"](#) on page 11-3
4. ["Finishing the Installation"](#) on page 11-4

Starting the Installation

The installation is similar to other Oracle Access Manager components with differences specific to the SNMP Agent.

To install the SNMP Agent

1. Log in as a user with Administrator privileges.
2. Locate and launch the component installer in a temporary directory created when you downloaded the software.

For example:

- **GUI Mode**

Windows: Oracle_Access_Manager10_1_4_0_1_win32_Snmp_Agent.exe

- **Console Mode**

Unix: ./ Oracle_Access_Manager10_1_4_0_1_sparc-s2_Snmp_Agent

The Welcome dialog appears.

3. Click Next to dismiss the Welcome screen.

4. Respond to the administrator-rights question based upon your platform. For example:
5. Specify the installation directory, then click Next. For example:

\SnmAgent

A summary of the installation directory and required disk space appears.

6. Record the installation directory and click Next.

The SNMP Agent is installed, which may take a few seconds. On Windows systems, a screen appears informing you that the Microsoft Managed Interfaces are being configured.

- **Windows:** Enter the following information to distinguish this SNMP Agent in the Windows Service window, then click Next. For example:
- **Windows service name:** A unique name for this SNMP Agent.
For example, if you provide the name SNMP1014, the name in the Service window will appear as Oracle Access Manager SNMP Agent (SNMP1014).
- **Account name:** DomainName\UserName for this SNMP Agent (the default is LocalSystem).
- **Password:** The password for this account.

Next, you are asked to define specific details for this SNMP Agent.

Specifying SNMP Agent Configuration Details

During this sequence, you will enter the port and community information for this SNMP Agent.

To specify SNMP Agent details

1. Enter the SNMP Agent TCP port that Oracle Access Manager will use to publish SNMP statistics. For example:

6012

This is the same port number that you specify when enabling the SNMP agent from the specific Identity or Access Server. Oracle Access Manager components will communicate with this port to publish their statistics to your Manager Station.

Next, you are asked for the UDP port and the Community Name defined in your Network Management System, which you will use to query this SNMP Agent. These should be same as those used by your Network Manager Station to query this SNMP Agent.

2. Enter the SNMP Agent UDP port and community name, then click next.

For example:

- **SNMP Agent UDP port:** *161*
- **Community Name:** *Your Community*

Now you are asked to enter the network monitor station name and trap port from your Network Management System, which the SNMP Agent should use when sending SNMP traps. The trap port is the port that will receive SNMP traps for Oracle Access Manager.

3. Enter the following information, then click Next.

For example:

- **Manager Station name:** *Your_Station_Name*
- **Trap port:** *162*

You now need to specify whether you want to configure another Manager Station for this SNMP Agent.

4. Click Yes to indicate that you want to configure another Manager Station now (otherwise, click No), then click Next.
 - If Yes, you will be asked to repeat step 3 for the new station, then you will be asked if you want to configure another station.
 - If No, you may manually configure another Manager Station using the following tool later: *SNMP_install_dir\snmp\tools\setup\setup_agent*.

A confirmation dialog appears.

Finishing the Installation

The installation concludes as other component installations have.

To finish the installation

1. Review the ReadMe information, then click Next.

A summary screen appears.

2. Write the details of this installation on the preparation worksheets, if you have not yet done so, then click Next to finish the installation.

You are ready to configure the SNMP Agent, as described in the *Oracle Access Manager Identity and Common Administration Guide*.

Installing Language Packs Independently

This chapter describes how to add an optional Language Pack after installing and setting up Oracle Access Manager components (or after upgrading from an earlier release). Topics include:

- [About Language Packs and Installation](#)
- [Language Pack Installation Considerations](#)
- [Language Pack Prerequisites Checklist](#)
- [Installing the Language Pack Independently](#)
- [Installed Files](#)
- [Confirming Language Status](#)

For an overview of languages, see the *Oracle Access Manager Introduction* and [Chapter 3, "About Multi-Language Environments"](#).

About Language Packs and Installation

Oracle provides the capability to localize Oracle Access Manager applications to display static data such as error messages and display names for tabs, panels, and attributes to users in their native language. The English language is always available, which requires no special installation or configuration. In addition, you can install Oracle-provided Language Packs and select a default Administrator language for Oracle Access Manager

Note: Oracle Access Manager supports Latin1 and UTF-8 data, including multibyte languages such as Chinese, Japanese, and so on. Contact Oracle for information about specific Language Packs.

For each language that Oracle supports, one Language Pack installer is provided for the Identity System and one is provided for the Access System. Language Packs can be installed together with Oracle Access Manager components, as described in other chapters. However you may also install a Language Pack independently, after Oracle Access Manager installation and setup, as discussed in this chapter.

As discussed in [Chapter 3, "About Multi-Language Environments"](#) the installation directory you specify for the Language Pack must match the installation directory of the component with which it is to operate. For example:

```
\IdentityServer_install_dir
```

```
\WebPass_install_dir
```

```
\PolicyManager_install_dir
```

```
\AccessServer_install_dir
```

```
\WebGate_install_dir
```

In addition, if you install a Language Pack on Identity System components you must install the same language Pack on all Access System components.

During installation, a `\langTag` directory is created in the file system under the main Oracle Access Manager component's installation directory. For example, `Component_install_dir\identity\access\oblix\lang\langTag`, where `langTag` represents specific language, such as English (en-us) or French (fr-fr). See "Installed Files" on page 12-5 for an example.

A language entry is created for each installed language under the Oblix node in the LDAP directory as follows: `obid=langTag, configDN`, where `configDN` is the configuration DN in the directory.

The `obnls.xml` configuration file is updated for each component in `\Component_install_dir\identity\access\oblix\config\obnls.xml`. Installed languages and entries in `obnls.xml` must match for each component. In the following sample `obnls.xml` file, installed languages include English, Arabic, Czech, Japanese, and simplified Chinese:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <ParamsCtlg xmlns="http://www.oblix.com" CtlgName="obnls.xml">

- <CompoundList xmlns="http://www.oblix.com" ListName="">
- <SimpleList>
  <NameValPair ParamName="default" Value="en-us" />
</SimpleList>
- <ValList xmlns="http://www.oblix.com" ListName="languages">

  <ValListMember Value="en-us" />
  <ValListMember Value="ar-ar" />
  <ValListMember Value="cs-cs" />
    <ValListMember Value="ja-jp" />
  <ValListMember Value="zh-CN" />
</ValList>
- <ValNameList xmlns="http://www.oblix.com" ListName="en-us">
  <NameValPair ParamName="sortRulesFile" />
  <NameValPair ParamName="dirPath" Value="en-us" />
</ValNameList>
- <ValNameList xmlns="http://www.oblix.com" ListName="ar-ar">
  <NameValPair ParamName="sortRulesFile" />
  <NameValPair ParamName="dirPath" Value="ar-ar" />
</ValNameList>
- <ValNameList xmlns="http://www.oblix.com" ListName="cs-cs">
  <NameValPair ParamName="sortRulesFile" />
  <NameValPair ParamName="dirPath" Value="cs-cs" />
</ValNameList>
- <ValNameList xmlns="http://www.oblix.com" ListName="ja-jp">
  <NameValPair ParamName="sortRulesFile" />
  <NameValPair ParamName="dirPath" Value="ja-jp" />
</ValNameList>
- <ValNameList xmlns="http://www.oblix.com" ListName="zh-CN">
  <NameValPair ParamName="sortRulesFile" />
  <NameValPair ParamName="dirPath" Value="zh-CN" />
  </ValNameList>
- <!-- List of locales that require bidi support
-->
- <ValList xmlns="http://www.oblix.com"
```

```

ListName="bidiLanguages">
  <ValListMember Value="he" />
  <ValListMember Value="ar" />
  <ValListMember Value="iw" />
</ValList>
</CompoundList>
</ParamsCtlg>

```

Task overview: Installing a Language Pack independently

1. Run the Identity System Language Pack installer, as described in ["Installing the Language Pack Independently"](#) on page 12-4, on each machine hosting an installed (or upgraded) Identity Server and an installed WebPass.
2. Confirm that the languages you installed are enabled, as described in ["Confirming Language Status"](#) on page 12-5.
3. Run the Access System Language Pack installer, as described in ["Installing the Language Pack Independently"](#) on page 12-4 on each machine hosting an installed (or upgraded) Policy Manager, Access Server, and WebGate.

Even though there are no user interfaces for the Access Server and WebGate, you need to install the same Language Packs for these components as you do for others.

4. Confirm that the languages you installed are enabled, as described in ["Confirming Language Status"](#) on page 12-5.

Language Pack Installation Considerations

Installing additional languages to enable multi-language functionality in Oracle Access Manager may be done either during or after installation of each component.

If you install a Policy Manager and WebGate in the same directory, you may then install a Language Pack in the same directory and both components will use it. This means that you do not need to install a Language Pack for the Policy Manager then repeat the process for a WebGate that resides in the same directory.

Note: Do not install the Policy Manager, then install a Language Pack, then install a WebGate in the same directory. Otherwise, the appropriate language entries will not appear in the WebGate obnls.xml file.

On Unix systems, you must ensure that the Language Pack has execute permissions before launching the installer. For example:

```

chmod +x "Oracle_Access_Manager10_1_4_0_1_FR_sparc-s2_LP_Identity_System"
chmod +x "Oracle_Access_Manager10_1_4_0_1_FR_sparc-s2_LP_Access_System"

```

If you prefer to install the Language Pack silently while installing each Oracle Access Manager component, the Language Pack installer must reside in the same temporary directory as the component installer. See [Chapter 3, "About Multi-Language Environments"](#) and individual installation chapters in this guide for more information.

Language Pack Prerequisites Checklist

Before you begin installing the Language Pack independently, check the tasks in "Language Pack Prerequisites Checklist" on page 12-4. It ensure they have been completed. Failure to complete prerequisites may adversely affect your Oracle Access Manager installation.

Table 12–1 Language Pack Installation Prerequisites Checklist

Checklist	Language Pack Installation Prerequisites
	Review and complete all prerequisites and requirements that apply to your environment, as described in Part I, "Installation Planning and Prerequisites"
	Complete all activities in Part II, "Identity System Installation and Setup" .
	Complete all activities in Part III, "Access System Installation and Setup"

To install a Language Pack at the same time as the Oracle Access Manager component, move any Language Pack installation packages into the same directory as the component installation package and refer to the appropriate chapter in this guide.

Installing the Language Pack Independently

This procedure walks you through adding a Language Pack independently, after Oracle Access Manager component installation and setup.

To perform independent Language Pack installation

1. Log in as a user with Administrator privileges.
2. Locate and launch the installation package for the desired Language Pack and component and launch the installer.

For example:

- **GUI Method, Windows**

Oracle_Access_Manager10_1_4_0_1_win32_langTag_Identity_System.exe
or

Oracle_Access_Manager10_1_4_0_1_win32_langTag_Access_System.exe

- **Console Method, UNIX.**

/ Oracle_Access_Manager10_1_4_0_1_sparc-s2_langTag_Identity_System
or

/ Oracle_Access_Manager10_1_4_0_1_sparc-s2_langTag_Access_System

where *langTag* refers to a specific language tag, such as FR (French).

The Welcome screen appears.

3. Click Next to dismiss the Welcome screen.
4. Respond to the question about administrator rights based on your platform:
5. Change the destination directory to match the main component for which this is being installed, then click Next.

For example:

```
\IdentityServer_install_dir
```

You are informed that the Language Pack is being installed, which may take a few seconds. ReadMe information appears next.

6. Review the ReadMe information, then click Next to dismiss it.
A summary screen appears.
7. Click Finish to complete this installation.
8. Restart the service for which you just installed the Language Pack.
9. Repeat the Language Pack installation for:
 - All WebPass components, using the Identity System Language Pack
 - All Access System components, using the Access System Language Pack
 - Policy Managers
 - Access Servers
 - WebGates

Installed Files

As discussed in [Chapter 3, "About Multi-Language Environments"](#), the `\lang` directory and the `\lang\en-us` and `\lang\shared` subdirectories are included with all installations. When you install additional languages, a `\langTag` subdirectory is created under `\lang`. It includes the same type of content as `\en-us`, only localized.

Confirming Language Status

Use the following procedure to confirm which languages are installed and enabled within Oracle Access Manager.

To confirm which languages are enabled

1. Navigate to the Identity System Console, and log in as usual.

`http://hostname:port/identity/oblix/`

where *hostname* refers to machine that hosts the Web server; *port* refers to the HTTP port number of the WebPass Web server instance; `/identity/oblix` connects to the Identity System Console.

2. Click Identity System Console, select System Configuration, then click Server Settings.
3. Click the Multi-Language link at the bottom of the page.
The Manage Multiple Languages page appears showing which languages are currently installed and which are enabled.
4. Click the box beside the languages you want to enable, then click the Enable button.
5. Refresh the browser screen or reopen the browser.

After installation, you must enable all the languages that you want to use, then configure the User, Group, or Org Manager applications to use the installed languages by entering display names (at the Object Class level) for attributes, tabs, and panels using the Identity System Console.

6. See the following documentation for additional information:

- [Uninstalling Oracle Access Manager Components](#)
- [Language Issues](#)
- *Oracle Access Manager Identity and Common Administration Guide* for more information about setting language preferences and localizing Oracle Access Manager information

About Installing Audit-to-Database Components

The Oracle Access Manager auditing feature collects and presents data pertaining to policy and profile settings, system events, and usage patterns. Oracle Access Manager can generate two types of audit reports:

- **Static:** These reports are derived from policy and profile information that is stored on the Oracle Access Manager directory server.
- **Dynamic:** These reports are derived from Access System and Identity System events that are collected from the servers in your system.

At the most detailed level, dynamic audit reports reveal when a system event was triggered and who triggered it. At a higher level, these reports can reveal component load levels, resource request patterns, system intrusion attempts, and overall system performance.

In addition to auditing, Oracle Access Manager supports logging, SNMP monitoring, and other reporting features.

You can record all dynamic audit reports and some static audit reports to disk file, to a relational database, or both. Some static reports can also be displayed in limited form through the graphical user interface.

Displaying audit reports on-screen or sending audit output to disk files does not require the installation of special components. If you intend to display audit reports in this way, you must complete configuration instructions in the *Oracle Access Manager Identity and Common Administration Guide*.

Auditing to a database is restricted to certain Oracle Access Manager system configurations and requires the installation of special components in addition to setup using the System Console. For instructions, see the *Oracle Access Manager Identity and Common Administration Guide*.



About the Software Developer Kit

The Access Manager Software Developer's Kit (SDK) allows developers to enhance the Oracle Access Manager access management capabilities. The Access Manager SDK consists of libraries, build instructions, and examples that you (or Oblix) can use to build a custom AccessGate for Web and non-Web resources.

An AccessGate uses an Access Server to control attempts to access a Web site. An AccessGate is similar to the WebGate access client provided by Oracle Access Manager. The WebGate client acts as the interface between individual Web servers and the Access Server. The WebGate intercepts requests from users requesting Web resources and authorizes them through the Access Server.

AccessGates allow you to extend authorization and authentication rules to other resources in addition to URLs and to control user interaction with applications outside of Oracle Access Manager. This provides you with centralized policy information that applies to Web and non-Web resources.

The Access Manager SDK enables you to create an interface that can be built into commercially available application servers such as BEA's WebLogic, IBM's WebSphere, iPlanet's Application Server, or another application that can access the Access Server. The Access Manager API can integrate with Java and C/C++ applications.

The use of the SDK is optional and of interest only to developers. Therefore, information about installing the SDK is located in the *Oracle Access Manager Developer Guide*.

Certain functions in the Identity System require the Oracle Access Manager Software Developer Kit (SDK). By default, the SDK is installed in a subdirectory under `\IdentityServer_install_dir\identity`. Following Identity System set up, you must manually configure the SDK for the Identity System to enable required functions, as described in the *Oracle Access Manager Identity and Common Administration Guide*.



Part V

Replication

This part discusses replication and silent mode options as well as cloning and synchronizing installed components. In addition, uninstalling components is discussed.

Part VI contains the following chapter:

- [Chapter 15, "Replicating Components"](#)

Replicating Components

Instead of using the command line or the installation GUI to install an Oracle Access Manager component, you can automate the installation process by replicating the configuration of one installed component to another. You do this by installing from an options file or by cloning an installed component. You can also partially replicate a component by synchronizing two installed components.

This chapter describes installation using an options file, cloning, and synchronization. It covers the following topics:

- [About the Silent Mode Options File](#)
- [Running the Silent Mode Options File](#)
- [Editing the Silent Mode Options File](#)
- [Silent Mode Parameters](#)
- [Uninstalling a Component Installed With Silent Mode](#)
- [Cloning and Synchronizing Installed Components](#)
- [Uninstalling a Cloned Component](#)

About the Silent Mode Options File

In addition to installing Oracle Access Manager from a GUI or the console, you can perform an automated installation using a file that contains installation parameters and values. This is called installing in *silent mode*. Silent mode permits installation without user intervention.

Note: Silent mode is intended for new Oracle Access Manager installations only, not for migrations or upgrades. For details about ADAM and silent installation, see "[Oracle Access Manager Silent Mode Installation Parameters](#)" on page B-16.

You perform silent mode installations using an options file. When you install a Oracle Access Manager component, the installation program automatically creates a file named `install_options.txt`. This file is written to the installation directory for the component. The general path is:

```
/Component_install_dir/identity | access/oblix/config/install_options.txt
```

`Component_install_dir` is the top-level directory in the path and `identity | access` represents the suffix for the respective Oracle Access Manager component. For example:

/OracleAccessManager/identity/oblix/config/install_options.txt

Your installation session is recorded in the installation options file. This file contains information about the prompts you received and the values that you supplied during installation. You can use this file as a template for future installations, changing parameter values as needed.

You need to edit the file if you re-entered any values during installation. The entire installation session is recorded in this file, so you may need to delete information if you input data several times for the same option. You also need to edit this file to change parameter values for the new installation. For Identity Server and WebPass, you at least need to specify a unique ID for the new component. Passwords entered during installation are not stored for security reasons.

Additional Uses of the Silent Mode Options File

The silent mode options file can also be used to provide default values for an interactive installation. This is useful if you want to provide default values for installing multiple instances of a Oracle Access Manager component. To provide default values for an installation, follow the instructions in this chapter with the following exceptions:

- Remove any parameters and values from the options file that have no defaults, such as password values.
- Invoke the installation program without the `-silent` option described next.

Running the Silent Mode Options File

The procedure to run the silent mode options file follows.

Note: Silent mode is intended for new installations only, not for migrations or upgrades.

To install new components in silent mode

1. Make a copy of the original options file if you have not already done so.
2. Run installation from the command prompt using the following options:

```
-options path_to_install_options.txt -silent
```

where *path_to_install_options.txt* is the location of the silent mode options file. You must include the file name in the path. The file name does not have to be `install_options.txt`.

Note: To suppress the installation dialog box, add the `-is:silent` option to this command.

Selecting an Installation Directory on HP-UX and AIX

To direct an installation to a directory with sufficient space, you can use the `-is:tempdir` path parameter. The path must be an absolute path, to a file system with sufficient space.

Inputting Installation Passwords

You must supply a password at the command line or edit the silent mode options file and store the password there. If you do not supply a password, the installation will fail. Here is an example of entering the password using the command line:

```
installer -is:silent -silent -options path_to_install_options.txt -W oblixDSinfoBean.dsPassword=Your_Password
```

where *path_to_install_options.txt* is the location of the silent mode options file.

Editing the Silent Mode Options File

You can find the options file in the following location:

```
/Component_install_dir/identity|access/oblix/config/install_options.txt
```

where *Component_install_dir* is the top-level directory in the path and *identity|access* represents the component type.

You need to copy the options file and edit the copy to match your environment, using the following guidelines:

- Parameters and values are case-sensitive
- All values should be enclosed in quotes

See the following examples:

- [Sample Options Files](#)
- [Sample Access Server Options Files](#)

Sample Options Files

An example of a Identity Server options file is shown in [Example 15-5](#).

Note: By default, the password field is commented out and a password is not provided when the silent mode options file is first created. Edit the password field if you want to insert a password. Delete the "#" and enter the correct password.

Sample Access Server Options Files

Several examples are presented here:

- [Sample: Same Directory Server](#)
- [Sample: Separate directory servers](#)
- [Sample: Separate directory servers with SSL enabled for user data](#)
- [Sample: Separate directory servers with SSL enabled for policies](#)

Sample: Same Directory Server

An example of an Access Server options file is shown in [Example 15-1](#). In this example, the configuration and policy data are stored in the same directory server.

Example 15-1 Access Server Options File, Same Directory Server

Log file for this installation is located at C:\DOC\AMIT\LOCAL\Temp\aaa.log

```

-P aaa.installLocation="C:\OracleAccessManager\oblix\access"
-W securityModeBean.securityModeChoices="open"
# The following are recommended to be entered as command line arguments.
-W oblixDSInfoBean.dsType="NCSP10"
-W oblixDSInfoBean.dsMode="open"
-W oblixDSInfoBean.dsHostMachine="marinello"
-W oblixDSInfoBean.dsPortNumber="999"
-W oblixDSInfoBean.dsBindDN="cn=administrator"
# The following are recommended to be entered as command line arguments.
-W oblixDSInfoBean.dsPassword="mypassword"
-W policyDataInWhichDSBean.askPolicyDataInWhichDS="OBLIX"
W aaaInfoBean.accessServerID="aaa"
-W aaaInfoBean.policyDataConfigDN="o=company,c=us"
-W aaaInfoBean.policyDSBase="o=company,c=us"

```

Sample: Separate directory servers

In this example of an Access Server options file, the configuration and policy data are stored in separate directory servers. See [Example 15–2](#).

Example 15–2 Access Server Options File, Configuration and Policy Data in Separate Directories

```

# Log file for this installation is located at C:\DOC\AMIT\LOCAL\Temp\aaa.log
-P aaa.installLocation="C:\OracleAccessManager\oblix\access"
-W securityModeBean.securityModeChoices="open"
# The following are recommended to be entered as command line arguments.
-W oblixDSInfoBean.dsType="NCSP10"
-W oblixDSInfoBean.dsMode="open"
-W oblixDSInfoBean.dsHostMachine="marinello"
-W oblixDSInfoBean.dsPortNumber="999"
-W oblixDSInfoBean.dsBindDN="cn=administrator"
# The following are recommended to be entered as command line arguments.
-W oblixDSInfoBean.dsPassword="mypassword"
-W policyDataInWhichDSBean.askPolicyDataInWhichDS="POLICY"
-W policyDSInfoBean.dsMode="open"
-W policyDSInfoBean.dsHostMachine="marinello"
-W policyDSInfoBean.dsPortNumber="999"
-W policyDSInfoBean.dsBindDN="cn=administrator"
# The following are recommended to be entered as command line arguments.
-W policyDSInfoBean.dsPassword="mypassword"
-W aaaInfoBean.accessServerID="aaa"
-W aaaInfoBean.policyDataConfigDN="o=company,c=us"
-W aaaInfoBean.policyDSBase="o=company,c=us"

```

Sample: Separate directory servers with SSL enabled for user data

In this example of an Access Server options file, [Example 15–3](#), the configuration and policy data are stored in separate directory servers and the user directory server is operating in SSL mode.

Example 15–3 Access Server Options with Separate Directory Servers with SSL-Enabled

```

# Log file for this installation is located at C:\DOC\AMIT\LOCAL\Temp\aaa.log
-P aaa.installLocation="C:\OracleAccessManager\oblix\access"
-W securityModeBean.securityModeChoices="open"
# The following are recommended to be entered as command line arguments
-W oblixDSInfoBean.dsType="NCSP10"
-W oblixDSInfoBean.dsMode="open"
-W oblixDSInfoBean.dsHostMachine="marinello"

```

```

-W oblixDSInfoBean.dsPortNumber="999"
-W oblixDSInfoBean.dsBindDN="cn=administrator"
# The following are recommended to be entered as command line arguments.
-W oblixDSInfoBean.dsPassword="mypassword"
-W policyDataInWhichDSBean.askPolicyDataInWhichDS="POLICY"
-W policyDSInfoBean.dsMode="open"
-W policyDSInfoBean.dsHostMachine="marinello"
-W oblixDSInfoBean.dsPortNumber="999"
-W policyDSInfoBean.dsBindDN="cn=administrator"
# The following are recommended to be entered as command line arguments.
-W policyDSInfoBean.dsPassword="mypassword"
-W aaaInfoBean.accessServerID="aaa"
-W aaaInfoBean.policyDataConfigDN="o=company,c=us"
-W aaaInfoBean.policyDSBase=o=company,c=us"
-W userDSSSLCertPath.sslCertPath="C:\Cert\ca.cert"

```

Sample: Separate directory servers with SSL enabled for policies

In this example of an Access Server options file, [Example 15-4](#), the configuration and policy data are stored in separate directory servers and the policy directory server is operating in SSL mode.

Example 15-4 Access Server Options, Separate Configuration and Policy Data, SSL-Enabled

```

# Log file for this installation is located at C:\DOC\AMIT\LOCAL\Temp\aaa.log
-P aaa.installLocation="C:\OracleAccessManager\oblix\access"
-W securityModeBean.securityModeChoices="open"
# The following are recommended to be entered as command line arguments
-W oblixDSInfoBean.dsType="NCSP10"
-W oblixDSInfoBean.dsMode="open"
-W oblixDSInfoBean.dsHostMachine="marinello"
-W oblixDSInfoBean.dsPortNumber="999"
-W oblixDSInfoBean.dsBindDN="cn=administrator"
# The following are recommended to be entered as command line arguments.
-W oblixDSInfoBean.dsPassword="mypassword"
-W policyDataInWhichDSBean.askPolicyDataInWhichDS="POLICY"
-W policyDSInfoBean.dsMode="ssl"
-W policyDSInfoBean.dsHostMachine="marinello"
-W policyDSInfoBean.dsPortNumber="333"
-W policyDSInfoBean.dsBindDN="cn=administrator"
# The following are recommended to be entered as command line arguments.
-W policyDSInfoBean.dsPassword="mypassword"
-W userDSSSLCertPath.sslCertPath="C:\Cert\ca.cert"
-W aaaInfoBean.accessServerID="aaa"
-W aaaInfoBean.policyDataConfigDN="o=company,c=us"
-W aaaInfoBean.policyDSBase=o=company,c=us"

```

Sample: Identity Server Installation using Active Directory

In this example of a Identity Server options file, [Example 15-5](#), the installation is being done on Active Directory.

Example 15-5 Identity Server Installation Options using Active Directory

```

# Log file for this installation is located at
C:\DOCUME~1\ADMINI~1\Temp\OracleAccessManager.log
-P ois.installLocation="D:\test\adsi\ois\identity"
-W securityModeBean.securityModeChoices="open"
-W oisInfoBean.hostName="test001"
-W oisInfoBean.serverID="test002"

```

```
-W oisInfoBean.portNumber="9002"
-W askFirstIdentityServer.askFirstIdentityServerField="n"
-W askSSLSetup.askSSLSetupField="No"
-W askADSI.isADSI="yes"
-W askUseImplicitBind.useImplicitBind="yes"
-W askNTServiceName.netServiceNameField="testcoreid"
-W askNTServiceAccount.ntServiceUserAccount=". \Administrator"
# The following is recommended to be entered as a command line argument
# -W askNTServiceAccount.netServiceUserPassword=<your password>
```

Silent Mode Parameters

The following discussions describe options you may edit in the silent installation options file for each component. Anything shown in *italics* is a value you supply for a parameter. You must supply a value for each parameter in the file. Enclose all values in double quotes.

For details on installation prompts and their values, refer to the other chapters in this installation guide. For example, see [Chapter 4, "Installing the Identity Server"](#) on page 4-1 for information on Identity Server installation prompts and values.

The following sections are sequenced in the recommended order for installing Oracle Access Manager components. The parameters are listed in the same order that they appear in the installation GUI.

Note: When installing a component, you may not need to supply every parameter. You need only supply values for parameters that apply to your installation.

Identity Server Parameters

[Table 15-1](#) describes silent installation parameters for the Identity Server.

Table 15-1 *Silent Installation Parameters for the Identity Server*

Identity Server Parameters and Descriptions	Possible Values
-P ois.installLocation: The installation directory. The default directory is "C:\COREid" on Windows and "/coreid" on UNIX.	<i>"installation directory"</i>
-W userInfoBean.user: UNIX only. The user ID that the product will be running as.	<i>"user ID"</i>
-W userInfoBean.group: UNIX only. The group that corresponds to the userInfoBean.user.	<i>"group id"</i>
-W localePanel.defaultLang: Required when extra languages are to be installed with the main installation.	<i>"en-us"</i>
-W localePanel.installLanguages: Required when extra languages are to be installed with the main installation.	<i>"en-us;fr-fr"</i>
-W securityModeBean.securityModeChoices: Security mode for the Identity Server. A value of "open" means no security is used, a value of "simple" means encryption is used, and a value of "cert" means you are running your own CA.	<i>"open", "simple", "cert"</i>
-W oisInfoBean.hostName: Host name where Identity Server is installed.	<i>"ip address" or "hostname"</i>
-W oisInfoBean.serverID: Identity Server ID. This is a unique ID that you create.	<i>"server id"</i>

Table 15–1 (Cont.) Silent Installation Parameters for the Identity Server

Identity Server Parameters and Descriptions	Possible Values
-W oisInfoBean.portNumber: Port number of the Identity Server. This port number cannot be used by another instance on the same machine.	<i>"port number"</i>
-W askFirstIdentityServer.askFirstIdentityServerField: This parameter specifies whether this is the first Identity Server being installed. The value "y" means yes, this is the first Identity Server installed, "n" means no.	<i>"y" or "n"</i>
-W askSSLSetup.askSSLSetupField: This parameter specifies whether to set up SSL between the Identity Server and the directory server.	<i>"Yes" or "No"</i>
-W askSSLSetup.askUserSSLSetupField: This parameter specifies whether to set up SSL between the Identity Server and the directory server containing user data.	<i>"Yes" or "No"</i>
-W askSSLSetup.askOblisSSLSetupField: This parameter specifies whether to set up SSL between the Identity Server and the directory server containing Oracle Access Manager configuration data	<i>"Yes" or "No"</i>
-W askUserSSLCertPath.sslCertPath: The absolute path to the SSL certificate. Use only if "askSSLSetup.askUserSSLSetupField" = "Yes".	<i>"absolute path including the file name"</i>
-W askOblisSSLCertPath.sslCertPath: The absolute path to the SSL certificate. Use only if "askSSLSetup.askOblisSSLSetupField" = "Yes".	<i>"absolute path including the file name"</i>
-W simpleModeBean.passphrase: This parameter is used if you are using the Simple transport security mode. This is a pass phrase allowing the Identity Server to communicate with the WebPass. Use only if "securityModeBean.securityModeChoices" = "simple".	<i>"passphrase"</i>
-W simpleModeBean.passphraseVerify: This parameter is used if you are using the Simple transport security mode. This parameter verifies that the pass phrase matches that of simpleModeBean.passphrase. Use only if securityModeBean.securityModeChoices = "simple".	<i>"passphrase"</i>
-W certModeBean.passphrase: This parameter is used if you are using the Cert transport security mode. This is a pass phrase allowing the Identity Server to communicate with the WebPass. Use only if securityModeBean.securityModeChoices = "cert".	<i>"passphrase"</i>
-W certModeBean.passphraseVerify: This parameter is used if you are using the Cert transport security mode. This parameter verifies that the pass phrase matches that of certModeBean.passphrase. Use only if securityModeBean.securityModeChoices = "cert".	<i>"passphrase"</i>
-W installOrRequestCertBean.installOrRequest: Determines whether to install or request a certificate to be used to configure the Identity System. Used if your security mode is set to "cert". If you already have a certificate, use "install". If you want Oracle Access Manager to request a certificate, use "request".	<i>"request" or "install"</i>
-W certReqInfoBean.countryName: Country name. This is a two-letter country code that is valid for use in a DN. It is part of the information used to request a certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	<i>"country code"</i>
-W certReqInfoBean.stateOrProvinceName: State or province name. This is a two-letter state or province code. It is part of the information used to request a certificate. Use only if installOrRequestCertBean.installOrRequest = "request"	<i>"state or province code"</i>

Table 15–1 (Cont.) Silent Installation Parameters for the Identity Server

Identity Server Parameters and Descriptions	Possible Values
<p>-W certReqInfoBean.localityName: Locality name. This is usually the name of a geographic region. It is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code></p>	<p><i>"locality name"</i></p>
<p>-W certReqInfoBean.organizationName: Organization name. This is usually the name of an organization. It is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code>.</p>	<p><i>"organization name"</i></p>
<p>-W certReqInfoBean.organizationalUnitName: Organizational unit name. This is usually the name of a department. It is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code>.</p>	<p><i>"organization unit name"</i></p>
<p>-W certReqInfoBean.commonName: Common name. This is usually the name of a person or entity. It is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code>.</p>	<p><i>"name"</i></p>
<p>-W certReqInfoBean.emailAddress: Email address. This is usually a valid email address. This is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code>.</p>	<p><i>"email address"</i></p>
<p>-W readyToInstallCertBean.readyToInstallField: If you requested that Oracle Access Manager request a certificate, this verifies that the certificate is ready for installation. Only use if <code>installOrRequestCertBean.installOrRequest = "request"</code>. Oracle recommends that you use a value of "No" for silent mode. It is unlikely that you can take the request generated by Oracle Access Manager and receive the certificate faster than the Oracle Access Manager installation script can run from one step to the next.</p>	<p><i>"Yes" or "No"</i></p>
<p>-W copyCertificatesInputBean.certFile: A certificate is composed of three files: a certificate file, a key file, and a chain file. This parameter specifies the absolute path, including the file name for the certificate file (for example: <code>ois_cert.pem</code>). Use if: <code>installOrRequestCertBean.installOrRequest = "install"</code> or if <code>installOrRequestCertBean.installOrRequest = "request"</code> and <code>readyToInstallCertBean.readyToInstallField = "Yes"</code>.</p>	<p><i>"absolute path including the file name"</i></p>
<p>-W copyCertificatesInputBean.keyFile: A certificate is composed of three files: a certificate file, a key file, and a chain file. This parameter specifies the absolute path including the file name for the key file (for example: <code>ois_key.pem</code>). Use if: <code>installOrRequestCertBean.installOrRequest = "install"</code> or if <code>installOrRequestCertBean.installOrRequest = "request"</code> and <code>readyToInstallCertBean.readyToInstallField = "Yes"</code>.</p>	<p><i>"absolute path including the file name"</i></p>
<p>-W copyCertificatesInputBean.chainFile: A certificate is composed of three files: a certificate file, a key file, and a chain file. This parameter specifies the absolute path including the file name for the chain file (for example: <code>ois_chain.pem</code>). Use if: <code>installOrRequestCertBean.installOrRequest = "install"</code> or if <code>installOrRequestCertBean.installOrRequest = "request"</code> and <code>readyToInstallCertBean.readyToInstallField = "Yes"</code>.</p>	<p><i>"absolute path including the file name"</i></p>

Table 15–1 (Cont.) Silent Installation Parameters for the Identity Server

Identity Server Parameters and Descriptions	Possible Values
<p>-W updateDSInfo.updateDSInfoChoice: Determines whether to automatically update the configuration and user schemas. Used only if askFirstIdentityServer.askFirstIdentityServeField= "y".</p> <p>"YesOneDS" performs an automatic update. Configuration and User directory server are the same.</p> <p>"YesTwoDS" performs an automatic update. Configuration and User directory servers are separate.</p> <p>"No" does not perform an automatic update.</p>	<p>"YesOneDS", "YesTwoDS", "No"</p>
<p>-W AutoUpdateInput.AutoUpdateInputChoice: Determines whether to automatically update the schema when configuration data is in the same directory where user data is stored.</p>	<p>"Yes" or "No"</p>
<p>-W OblixDSAutoUpdateInput.AutoUpdateInputChoice: Determines whether to automatically update the schema when configuration data is in a different directory from user data.</p>	<p>"Yes" or "No"</p>
<p>-W dsTypeInput.dsType: Use this parameter if Oracle Access Manager is automatically updating the Configuration and User schemas (that is, if updateDSInfo.updateDSInfoChoice = "YesOneDS" or "YesTwoDS"). User directory server Types are:</p> <p>1 - Sun Directory Server 5.x 2 - NDS 3 - Active Directory 4 - ADSI (Schema will be uploaded using LDAP) 5 - Active Directory on Windows Server 2003 6 - ADSI on Windows Server 2003 7 - Active Directory Application Mode (On Windows 2003 Only) 8 - Siemens DirX -- Not Supported in 10g (10.1.4.0.1) 9 - IBM Directory Server 10 - Data Anywhere 11 - Oracle Internet Directory</p> <p>Note: Data Anywhere may be used with user data only and requires integration with Oracle Virtual Directory Server (VDS), as described in Chapter 10, "Setting Up Oracle Access Manager with Oracle Virtual Directory". The LDAP directory branches containing configuration (and policy) data must reside on one or more directory servers other than the one hosting VDS or user data.</p> <p>Also Note: When the directory server type for user data differs from the directory server type for configuration data, use the following to specify the directory server type for configuration data: -W dsTypeInput1.dsType=#.</p>	<p>"1", "2", "3", "4", "5", "6", "7", "9", "10", "11"</p>
<p>-W dsTypeInput1.dsType: See Also Note, in the preceding description.</p>	
<p>-W dsUserDynAuxClassInput.dynamicAuxiliary: Set this parameter to "y" if you want to support dynamic auxiliary classes with Active Directory. Use only if you have set -W dsTypeInput.dsType to "5" or "7".</p>	<p>"y" or "n"</p>
<p>-W dsInfoInput.dsName: For most directory types, this is the User directory server host name. For Active Directory, use the Schema Master host name. Use only if updateDSInfo.updateDSInfoChoice = "YesOneDS" or "YesTwoDS".</p>	<p>"ip address" or "hostname"</p>

Table 15–1 (Cont.) Silent Installation Parameters for the Identity Server

Identity Server Parameters and Descriptions	Possible Values
<p>-W dsInfoInput.dsName: For most directory types, this is the User directory server host name. For Active Directory, use the Schema Master host name. Use only if updateDSInfo.updateDSInfoChoice = "YesOneDS" or "YesTwoDS".</p>	<p>"ip address" or "hostname"</p>
<p>-W dsInfoInput.dsPortNumber: For most directory types, this is the User directory server port number. For Active Directory, use the Schema Master port number. Use only if updateDSInfo.updateDSInfoChoice = "YesOneDS" or "YesTwoDS".</p>	<p>"port number"</p>
<p>-W dsInfoInput.bindDN: For most directory types, this is the DN used to authenticate to the User directory server. For Active Directory, use the Schema Master bind DN. Use only if updateDSInfo.updateDSInfoChoice = "YesOneDS" or "YesTwoDS". Enter this value using valid DN syntax, for example, "cn=User Directory, o=Oblix".</p>	<p>"bind DN"</p>
<p>-W dsInfoInput.password: For most directory types, this is the User directory server password. For Active Directory, use the Schema Master password. Use only if updateDSInfo.updateDSInfoChoice = "YesOneDS" or "YesTwoDS". See the note in "Running the Silent Mode Options File" on page 15-2 regarding secure password input.</p>	<p>"password"</p>
<p>-W OblixdsInfoInput.dsName: Configuration directory server name. Use only if configuration and User directory servers are separate and you are not using NDS or Active Directory, that is: updateDSInfo.updateDSInfoChoice = "YesTwoDS" and dsTypeInput1.dsType does not equal to "2" or "3".</p>	
<p>-W OblixdsInfoInput.dsPortNumber: Configuration directory server port number. Use only if configuration and User directory servers are separate and you are not using NDS or Active Directory, that is: updateDSInfo.updateDSInfoChoice = "YesTwoDS" and dsTypeInput1.dsType does not equal "2" or "3".</p>	<p>"port number"</p>
<p>-W OblixdsInfoInput.bindDN: DN used to authenticate to the Configuration directory server. Use only if configuration and user data directory servers are separate and you are not using NDS or Active Directory, that is: updateDSInfo.updateDSInfoChoice = "YesTwoDS" and dsTypeInput1.dsType does not equal to "2" or "3". Enter this value using valid DN syntax, for example: "cn=Configuration Directory, o=Oblix".</p>	<p>"bind DN"</p>
<p>-W OblixdsInfoInput.password: Configuration directory server password. Use only if configuration and User directory servers are separate and you are not using NDS or Active Directory, that is: updateDSInfo.updateDSInfoChoice = "YesTwoDS" and dsTypeInput1.dsType does not equal "2" or "3".</p>	<p>"password"</p>
<p>-W askNTServiceName.ntServiceNameField: Windows only. A service name for the Identity Server. This name will appear in the services control panel.</p>	<p>"name"</p>
<p>-W askADSI.isADSI: Confirms if you are using Active Directory with ADSI.</p>	<p>"yes", "no"</p>
<p>-W askADSISSL.isADSISSL: Confirms if you are running Active Directory with ADSI using SSL.</p>	<p>"yes", "no"</p>

Table 15–1 (Cont.) Silent Installation Parameters for the Identity Server

Identity Server Parameters and Descriptions	Possible Values
-W askSeparateADDomain.isSeparateDomain: Specifies if the machine where you are installing this Identity Server instance is in a different forest from the target Active Directory Forest that Oracle Access Manager is configured to use.	"yes", "no"
-W askUseImplicitBind.useImplicitBind: If the installation machine is in the same domain, do you want to use the Service account credentials to access Active Directory? A "yes" sets the parameter useImplicitBind in the adsi_params.xml file.	"yes", "no"
-W askNTServiceAccount.ntServiceUserAccount: If you set the value of askUseImplicitBind to "yes," this is the account that the service runs as, for example, ".\Administrator".	"account ID"
-W askNTServiceAccount.ntServiceUserPassword: If you set the value of ask UseImplicitBind to "yes", this is the service account password. Oracle recommends that you supply this value at the command line.	"password"

WebPass Parameters

Table 15–2 describes silent installation parameters for WebPass.

Table 15–2 Silent Installation Parameters for WebPass

WebPass Parameter and Description	Possible Values
-P webpass.installLocation: Installation directory. The default directory is "C:\COREid\WebComponent" on Windows and "/coreid/webcomponent" on UNIX.	"installation directory"
-W userInfoBean.user: UNIX only. The user ID that the product will be running as.	"user ID"
-W userInfoBean.group: UNIX only. The group that corresponds to the userInfoBean.user.	"group id"
-W localePanel.defaultLang: Required when extra languages are to be installed with the main installation.	"en-us"
-W localePanel.installLanguages: Required when extra languages are to be installed with the main installation.	"en-us;fr-fr"
-W securityModeBean.securityModeChoices: The security mode for the Identity Server. A value of "open" means no security is used, a value of "simple" means encryption is used, and a value of "cert" means you are running your own CA.	"open", "simple", "cert"
-W webpassInfoBean.hostName: Host name of the Identity Server.	"ip address" or "hostname"
-W webpassInfoBean.webpassID: WebPass ID. This is an unique ID you specify during installation.	"ID"
-W webpassInfoBean.portNumber: Port number of the Identity Server.	"port number"
-W simpleModeBean.passphrase: Pass phrase allowing the Identity Server to communicate with the WebPass. Use only if securityModeBean.securityModeChoices = "simple".	"passphrase"

Table 15–2 (Cont.) Silent Installation Parameters for WebPass

WebPass Parameter and Description	Possible Values
<p>-W simpleModeBean.passphraseVerify: Pass phrase allowing the Identity Server to communicate with the WebPass. This parameter is used to verify that the pass phrase matches that of certModeBean.passphrase. Use only if securityModeBean.securityModeChoices = "simple".</p>	<p>"passphrase"</p>
<p>-W certModeBean.passphrase: Pass phrase allowing the Identity Server to communicate with the WebPass. Use only if securityModeBean.securityModeChoices = "cert".</p>	<p>"passphrase"</p>
<p>-W certModeBean.passphraseVerify: Pass phrase allowing the Identity Server to communicate with the WebPass. This parameter verifies that the pass phrase matches that of certModeBean.passphrase. Use only if securityModeBean.securityModeChoices = "cert".</p>	<p>"passphrase"</p>
<p>-W installOrRequestCertBean.installOrRequest: Determines whether to install or request a certificate that is used to configure the Identity System. Use if your security mode is set to "cert". If you already have requested a certificate, choose "install". If you want Oracle Access Manager to request a certificate that can be submitted to the CA, choose "request".</p>	<p>"install" or "request"</p>
<p>-W certReqInfoBean.countryName: Country name. This is a two-letter country code that is valid for use in a DN. It is part of the information that Oracle Access Manager uses to request a certificate. Use this parameter if you have opted to have Oracle Access Manager request a certificate, that is, if installOrRequestCertBean.installOrRequest = "request".</p>	<p>"country code"</p>
<p>-W certReqInfoBean.stateOrProvinceName: State or province name. This is a two-letter state or province code that is valid for use in a DN. It is part of the information that Oracle Access Manager uses to request certificate. Use if you have opted to have Oracle Access Manager request a certificate, that is, if installOrRequestCertBean.installOrRequest = "request".</p>	<p>"state or province code"</p>
<p>-W certReqInfoBean.localityName: Locality name. Part of the information that Oracle Access Manager uses to request a certificate. Use if you have opted to have Oracle Access Manager request a certificate, that is, if installOrRequestCertBean.installOrRequest = "request".</p>	<p>"locality name"</p>
<p>-W certReqInfoBean.organizationName: Organization name. This is usually the name of an organization. It is part of the information that Oracle Access Manager uses to request a certificate. Use if you have opted to have Oracle Access Manager request a certificate, that is, if installOrRequestCertBean.installOrRequest = "request".</p>	<p>"organization name"</p>
<p>-W certReqInfoBean.organizationalUnitName: Organizational unit name. This is usually the name of a department. It is part of the information that Oracle Access Manager uses to request a certificate. Use if you have opted to have Oracle Access Manager request a certificate, that is, if installOrRequestCertBean.installOrRequest = "request".</p>	<p>"organization unit name"</p>
<p>-W certReqInfoBean.commonName: Common name. This is usually the name of a person or entity. This is part of the information that Oracle Access Manager uses to request a certificate. Use if you have opted to have Oracle Access Manager request a certificate, that is, if installOrRequestCertBean.installOrRequest = "request".</p>	<p>"name"</p>

Table 15–2 (Cont.) Silent Installation Parameters for WebPass

WebPass Parameter and Description	Possible Values
-W certReqInfoBean.emailAddress: Email address. This is usually a valid email address. This is part of the information that Oracle Access Manager uses to request a certificate. Use if you have opted to have Oracle Access Manager request a certificate, that is, if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	<i>"email address"</i>
-W readyToInstallCertBean.readyToInstallField: If you requested that Oracle Access Manager request a certificate, this parameter asks if the certificate is ready for installation. Only use if <code>installOrRequestCertBean.installOrRequest = "request"</code> . Oracle recommends you do not use "Yes" for silent mode. It is unlikely that you can take the request generated by Oracle Access Manager and receive the certificates faster than the Oracle Access Manager installation can run from one step to the next.	"Yes" or "No"
-W copyCertificatesInputBean.certFile: A certificate is composed of three files: a certificate file, a key file, and a chain file. This parameter specifies the absolute path including the file name for the certificate file (for example: <code>ois_cert.pem</code>). Use if: <code>installOrRequestCertBean.installOrRequest = "install"</code> or if <code>installOrRequestCertBean.installOrRequest = "request"</code> and <code>readyToInstallCertBean.readyToInstallField = "Yes"</code> .	<i>"absolute path including the file name"</i>
-W copyCertificatesInputBean.keyFile: A certificate is composed of three files: a certificate file, a key file, and a chain file. This parameter specifies the absolute path including the file name for the key file (for example: <code>ois_key.pem</code>). Use if: <code>installOrRequestCertBean.installOrRequest = "install"</code> or if <code>installOrRequestCertBean.installOrRequest = "request"</code> and <code>readyToInstallCertBean.readyToInstallField = "Yes"</code> .	<i>"absolute path including the file name"</i>
-W copyCertificatesInputBean.chainFile: A certificate is composed of three files: a certificate file, a key file, and a chain file. This parameter specifies the absolute path including the file name for the chain file (for example: <code>ois_chain.pem</code>). Use if: <code>installOrRequestCertBean.installOrRequest = "install"</code> or if <code>installOrRequestCertBean.installOrRequest = "request"</code> and <code>readyToInstallCertBean.readyToInstallField = "Yes"</code> .	<i>"absolute path including the file name"</i>
-W askAutoUpdateWSBean.askAutoUpdateWSField: Determines whether to update the Web server configuration automatically.	"Yes" or "No"
-W askConfFilePathBean.askConfFilePathField: For NSAPI, this is the absolute path of the Web server configuration directory containing <code>obj.conf</code> (for example: <code>/export/Sun/servers/https-oblix/config</code>). For Apache/Apache SSL, this is the absolute path of <code>httpd.conf</code> in your Web server configuration directory (for example: <code>/export/apache/conf/httpd.conf</code>). Use only for Apache, Apache SSL, and NSAPI Web servers and if <code>askAutoUpdateWSBean.askAutoUpdateWSField = "Yes"</code> .	<i>"absolute path (including the file name for Apache)"</i>
-W askLaunchBrowserBean.launchBrowser: Determines whether to launch a browser to display instructions to manually update the Web server configuration. Use only if installing on UNIX and <code>askAutoUpdateWSBean.askAutoUpdateWSField = "No"</code> .	"Yes" or "No"

Policy Manager Parameters

Table 15–3 describes silent installation parameters for the Policy Manager.

Table 15–3 Silent Installation Parameters for Policy Manager

Policy Manager Parameter and Description	Possible Values
<p>-P manager.installLocation: Installation directory. The default directory is "C:\COREid" on Windows and "/coreid" on UNIX.</p>	<p>"installation directory"</p>
<p>-W userInfoBean.user: UNIX only. The user ID that the product will be running as. This can be any valid user ID.</p>	<p>"user ID"</p>
<p>-W userInfoBean.group: UNIX only. The group that corresponds to the userInfoBean.user.</p>	<p>"group name"</p>
<p>-W localePanel.defaultLang: Required when extra languages are to be installed with the main installation.</p>	<p>"en-us"</p>
<p>-W localePanel.installLanguages: Required when extra languages are to be installed with the main installation.</p>	<p>"en-us;fr-fr"</p>
<p>-W updateDSInfo.updateDSInfoChoice: This parameter determines whether Oracle Access Manager updates the policy schema automatically. Use this parameter if the Policy directory server is the same as the configuration data directory server, but different from the User directory server.</p>	<p>"Yes" or "No"</p>
<p>-W dsTypeInput.dsType: If the Policy directory server is the same as the configuration server, but different from the User directory server updateDSInfo.updateDSInfoChoice = "Yes", you need to specify the Policy directory server type:</p> <p>1 - Sun Directory Server 5.x 2 - NDS 3 - Active Directory 5 - Active Directory on Windows Server 2003 7 - Active Directory Application Mode (On Windows 2003 Only) 8 - Siemens DirX -- Not Supported in 10g (10.1.4.0.1) 9 - IBM Directory Server 10 - Oracle Internet Directory</p>	<p>"1", "2", "3", "4", "5", "6", "7", "9", "10"</p>
<p>-W dsInfoInput.dsName: Policy directory server name. Use this parameter if the Policy directory server is the same as the configuration data server, but different from the User directory server, and you are not using NDS or Active Directory.</p> <p>updateDSInfo.updateDSInfoChoice = "Yes" and dsTypeInput.dsType does not equal "2" or "3".</p>	<p>"ip address " or "hostname"</p>
<p>-W dsInfoInput.dsPortNumber: Policy directory server port number. Use this parameter if the Policy directory server is the same as the configuration data directory server, but different from the User directory server, and you are not using NDS or Active Directory</p> <p>updateDSInfo.updateDSInfoChoice = "Yes" and dsTypeInput.dsType does not equal "2" or "3".</p>	<p>"port"</p>
<p>-W dsInfoInput.bindDN: DN used to authenticate to the Policy directory server. Use this parameter if the Policy directory server is the same as the configuration data directory server, but different from the User directory server, and you are not using NDS or Active Directory:</p> <p>updateDSInfo.updateDSInfoChoice = "Yes" and dsTypeInput.dsType does not equal "2" or "3".</p> <p>Use conventional DN syntax for this entry, for example: "cn=Policy Directory, o=Obliv".</p>	<p>"bind DN"</p>

Table 15–3 (Cont.) Silent Installation Parameters for Policy Manager

Policy Manager Parameter and Description	Possible Values
-W dsInfoInput.password : Policy directory server password. Use this parameter if the Policy directory server is the same as the configuration data directory server, but different from the User directory server, and you are not using NDS or Active Directory: updateDSInfo.updateDSInfoChoice = "Yes" and dsTypeInfo.dsType does not equal "2" or "3".	"password"
-W dsInfoInput.dsSSLConnect : Determines whether the Policy directory server uses an SSL connection. Use this parameter if the Policy directory server is the same as the configuration data directory server, but different from the User directory server, and you are not using NDS or Active Directory: updateDSInfo.updateDSInfoChoice = "Yes" and dsTypeInfo.dsType does not equal "2" or "3".	"Yes" or "No"
-W askSSLCertPath.askSSLCertificatePathField : The absolute path to the SSL certificate. Use this parameter if the Policy directory server is the same as the configuration data directory server, but different from the User directory server, and you are not using NDS or Active Directory: updateDSInfo.updateDSInfoChoice = "Yes" and dsTypeInfo.dsType does not equal "2" or "3" and dsInfoInput.dsSSLConnect = "Yes".	"absolute path including the file name"
-W askAutoUpdateWSBean.askAutoUpdateWSField : Determines whether to update the Web server configuration automatically.	"Yes" or "No"
-W askConfFilePathBean.askConfFilePathField : For NSAPI, this is the absolute path of the Web server config directory containing obj.conf (for example: /export/Sun/servers/https-oblix/config). For Apache and Apache SSL, this is the absolute path of httpd.conf in your Web server config directory (for example: /export/apache/conf/httpd.conf). Use only for Apache, Apache SSL, and NSAPI Web servers and if askAutoUpdateWSBean.askAutoUpdateWSField = "Yes".	"absolute path (including the file name for Apache)"
-W askLaunchBrowserBean.launchBrowser : Determines whether to launch a browser that displays instructions to manually update the Web server configuration. Use only on UNIX and only if askAutoUpdateWSBean.askAutoUpdateWSField = "No".	"Yes" or "No"
-W askADSI.isADSI : Confirms if you are running Active Directory with ADSI.	"yes", "no"
-W askADSISSL.isADSISSL : Confirms if you are running Active Directory with ADSI using SSL.	"yes", "no"

Access Server Parameters

Table 15–4 describes silent installation parameters for the Access Server.

Table 15–4 Silent Installation Parameters for Access Server

Access Server Parameter and Description	Possible Values
-P aaa.installLocation : The installation directory. The default directory is "C:\COREid" on Windows and "/coreid" on UNIX.	"installation directory"
-W userInfoBean.user : UNIX only. The user ID that the product will be running as.	"user ID"
-W userInfoBean.group : UNIX only. The group that corresponds to the userInfoBean.user.	"group name"

Table 15–4 (Cont.) Silent Installation Parameters for Access Server

Access Server Parameter and Description	Possible Values
-W localePanel.defaultLang: Required when extra languages are to be installed with the main installation.	"en-us"
-W localePanel.installLanguages: Required when extra languages are to be installed with the main installation.	"en-us;fr-fr"
-W securityModeBean.securityModeChoices: The security mode for the Access Server. A value of "open" means no security is used, a value of "simple" means encryption is used, and a value of "cert" means you are running your own CA.	"open", "simple", or "cert"
-W userDSSSLCerPath.sslCertPath: The absolute path to the SSL certificate. Use only if the user directory is in SSL mode.	<i>"absolute path including the file name"</i>
-W oblixDSInfoBean.dsHostMachine: Configuration directory server host machine.	<i>"ip address "</i> or <i>"hostname"</i>
-W oblixDSInfoBean.dsPortNumber: Configuration directory server port number.	<i>"port number"</i>
-W oblixDSInfoBean.dsBindDN: DN used to authenticate to the Configuration directory server.	<i>"bind DN"</i>
-W oblixDSInfoBean.dsPassword: Configuration directory server password.	<i>"password"</i>
-W oblixDSInfoBean.dsMode: Configuration directory server's mode (open or ssl).	<i>"open" or "ssl"</i>
-W oblixDSInfoBean.dsType: The Configuration directory server type: NS5 - Sun Directory Server 5.x NOVELL - NDS MSAD - Microsoft Active Directory MSAD_ADSI - Microsoft Active Directory with ADSI MSADAM - Active Directory Application Mode DIRX - Siemens DirX -- Not Supported in 10g (10.1.4.0.1) IBMSWAY - IBM Directory Server? Oracle Internet Directory	<i>"NS5", "NOVELL", "MSAD", "MSAD_ADSI" "MSADAM" "IBMSWAY" "OID"</i>
-W oblixDSSSLCerPath.sslCertPath: The absolute path to the ssl certificate. Use only if oblixDSInfoBean.dsMode = "ssl".	<i>"absolute path including the file name"</i>
-W policyDataInWhichDSBean.askPolicyDataInWhichDS: Determines whether the Policy directory server is the same as the User or Configuration directory server. The value "OBLIX" means that the Policy and Configuration directory server are the same. The value "POLICY" means that the Policy directory server is different from that of User and Configuration directory server.	<i>"OBLIX" or "POLICY"</i>
-W policyDSInfoBean.dsHostMachine: The Policy directory server host machine. Use only if the Policy directory server is the same as the Configuration server, but different from the User directory server, policyDataInWhichDSBean.askPolicyDataInWhichDS = "OBLIX".	<i>"ip address "</i> or <i>"hostname"</i>
-W policyDSInfoBean.dsPortNumber: The Policy directory server port number. Use only if the Policy directory server is the same as the Configuration server, but different from the User directory server, policyDataInWhichDSBean.askPolicyDataInWhichDS = "OBLIX".	<i>"port number"</i>

Table 15–4 (Cont.) Silent Installation Parameters for Access Server

Access Server Parameter and Description	Possible Values
<p>-W policyDSInfoBean.dsBindDN: The DN used to authenticate to the Policy directory server. Use only if the Policy directory server is different from that of User and Configuration directory server policyDataInWhichDSBean.askPolicyDataInWhichDS = "POLICY". Use conventional DN syntax for this entry. Example: "cn=Policy Directory, o=Oblix".</p>	<p>"bind DN"</p>
<p>-W policyDSInfoBean.dsPassword: Policy directory server password. Use only if the Policy directory server is different from that of User and Configuration directory server policyDataInWhichDSBean.askPolicyDataInWhichDS = "POLICY".</p>	<p>"password"</p>
<p>-W policyDSInfoBean.dsMode: Policy directory server mode (open or ssl). Use only if the Policy directory server is different from that of User and Configuration directory server policyDataInWhichDSBean.askPolicyDataInWhichDS = "POLICY".</p>	<p>"open" or "ssl"</p>
<p>-W policyDSSSLCertPath.sslCertPath: The absolute path to the ssl certificate. Use only if the Policy directory server is the same as the Configuration server, but different from the User directory server, policyDataInWhichDSBean.askPolicyDataInWhichDS = "OBLIX".</p>	<p>"absolute path including the file name"</p>
<p>-W aaaInfoBean.accessServerID: The ID of the Access Server registered in the Access System Console. Supply the value you entered at the Access System Console for this Access Server.</p>	<p>"value"</p>
<p>-W aaaInfoBean.policyDataConfigDN: The configuration DN for the policy data. Use conventional DN syntax for this entry. Example: "cn=Policy Data, o=Oblix".</p>	<p>"DN"</p>
<p>-W aaaInfoBean.policyDSBase: The policy base, which is the node in the Policy directory under which Configuration stores its policy-related data. Example: "cn=Policy Data, o=Oblix".</p>	<p>"DN"</p>
<p>-W simpleModeInfoBean.passphrase: The pass phrase. Use only if securityModeBean.securityModeChoices = "simple".</p>	<p>"passphrase"</p>
<p>-W simpleModeInfoBean.passphraseVerify: The pass phrase again, used for verification. This should be the same as simpleModeInfoBean.passphrase. Use only if securityModeBean.securityModeChoices = "simple".</p>	<p>"passphrase"</p>
<p>-W simpleModeInfoBean.storePassPhraseinFile: Determines whether the pass phrase is stored in a file. If stored in a file, the Access Server can be started without a user or a script providing the pass phrase when the Access Server starts up.</p>	<p>"true" or "false"</p>
<p>-W certModeInfoBean.passphrase: Pass phrase. Use only if securityModeBean.securityModeChoices = "cert".</p>	<p>"passphrase"</p>
<p>-W certModeInfoBean.passphraseVerify: Pass phrase again, used for verification. This should be the same as certModeInfoBean.passphrase. Use only if securityModeBean.securityModeChoices = "cert".</p>	<p>"passphrase"</p>
<p>-W certModeInfoBean.storePassPhraseinFile: Determines whether the password or pass phrase is stored in a file. If stored in a file, the Access Server can be started without a user providing the pass phrase when the Access Server starts up.</p>	<p>"true" or "false"</p>
<p>-W installOrRequestCertBean.installOrRequest: Determines whether to install or request a certificate that is used to configure the Access System. Used if your security mode is set to "cert". If you already have a certificate, use "install." If you want Oracle Access Manager to request a request for a certificate, use "request".</p>	<p>"request" or "install"</p>

Table 15–4 (Cont.) Silent Installation Parameters for Access Server

Access Server Parameter and Description	Possible Values
-W certReqInfoBean.countryName: Country name. This is usually a two-letter country code that is valid for use in DNs. This is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	<i>"country code"</i>
-W certReqInfoBean.stateOrProvinceName: State or province name. This is usually a two-letter state or province code that this valid for use in DNs. This is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	<i>"state or province code"</i>
-W certReqInfoBean.localityName: Locality name. This is usually the name of a geographic region. This is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	<i>"locality name"</i>
-W certReqInfoBean.organizationName: Organization name. This is usually the name of an organization. It is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	<i>"organization name"</i>
-W certReqInfoBean.organizationalUnitName: Organization unit name. This is usually the name of a department. It is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	<i>"organization name"</i>
-W certReqInfoBean.commonName: Common name. This is usually the name of a person or another entity. This is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	<i>"name"</i>
-W certReqInfoBean.emailAddress: Email address. This is usually a valid email address. This is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	<i>"email address"</i>
-W readyToInstallCertBean.readyToInstallField: If you requested that Oracle Access Manager request a certificate, this verifies that the certificate is ready for installation. Only use if <code>installOrRequestCertBean.installOrRequest = "request"</code> . Oracle recommends you do not use "Yes" for silent mode. You probably cannot take the request generated by Oracle Access Manager and receive the certificates faster than the Oracle Access Manager installation can run from one step in the installation to the next.	<i>"Yes" or "No"</i>
-W copyCertificatesInputBean.certFile: The absolute path including the file name for the certificate file (for example: <code>aaa_cert.pem</code>). Use if: <code>installOrRequestCertBean.installOrRequest = "install"</code> or if <code>installOrRequestCertBean.installOrRequest = "request"</code> and <code>readyToInstallCertBean.readyToInstallField = "Yes"</code> .	<i>"absolute path including the file name"</i>
-W copyCertificatesInputBean.keyFile: The absolute path including the file name for the key file (example: <code>aaa_key.pem</code>). Use if: <code>installOrRequestCertBean.installOrRequest = "install"</code> or if <code>installOrRequestCertBean.installOrRequest = "request"</code> and <code>readyToInstallCertBean.readyToInstallField = "Yes"</code> .	<i>"absolute path including the file name"</i>
-W copyCertificatesInputBean.chainFile: The absolute path including the file name for the chain file (example: <code>aaa_chain.pem</code>). Use if: <code>installOrRequestCertBean.installOrRequest = "install"</code> or if <code>installOrRequestCertBean.installOrRequest = "request"</code> and <code>readyToInstallCertBean.readyToInstallField = "Yes"</code> .	<i>"absolute path including the file name"</i>

Table 15–4 (Cont.) Silent Installation Parameters for Access Server

Access Server Parameter and Description	Possible Values
-W askSeparateDomain.isSeparateDomain: Specifies if the machine where you are installing this Identity Server instance is in a different forest from the target Active Directory Forest that Oracle Access Manager is configured to use.	"yes", "no"
-W askUseImplicitBind.useImplicitBind: If the installation machine is in the same domain, do you want to use the Service account credentials to access Active Directory? A "yes" sets the parameter useImplicitBind in the adsi_params.xml file.	"yes", "no"
-W askNTServiceAccount.ntServiceUserAccount: If you set the value of askUseImplicitBind to "yes," this is the account that the service runs as, for example, ".\Administrator".	"account ID"
-W askNTServiceAccount.ntServiceUserPassword: If you set the value of askUseImplicitBind to "yes", this is the service account password. Oracle recommends that you supply this value at the command line.	"password"

WebGate Parameters

Table 15–5 describes silent installation parameters for WebGate.

Table 15–5 Silent Installation Parameters for WebGate

WebGate Parameter and Description	Possible Values
-P webgate.installLocation: Installation directory. The default directory is "C:\COREid\WebComponent" on Windows and "/coreid/WebComponent" on UNIX.	"installation directory"
-W userInfoBean.user: UNIX only. The user ID that the product will be running as.	"user ID"
-W userInfoBean.group: UNIX only. The group that corresponds to the userInfoBean.user.	"group id"
-W localePanel.defaultLang: Required when extra languages are to be installed with the main installation.	"en-us"
-W localePanel.installLanguages: Required when extra languages are to be installed with the main installation.	"en-us;fr-fr"
-W securityModeBean.securityModeChoices: Security mode for WebGate. A value of "open" means no security is used, a value of "simple" means encryption is used, and a value of "cert" means you are running your own CA.	"open", "simple", "cert"
-W openModeBean.serverID: Access Server ID. Use the value you supplied at the Access System Console before installation. Use only if securityModeBean.securityModeChoices = "open".	"server id"
-W openModeBean.hostName: Access Server host name. Use only if securityModeBean.securityModeChoices = "open".	"ip address" or "hostname"
-W openModeBean.webgateID: WebGate ID. Use the ID that you entered in the Access System Console before running the installation. Use only if securityModeBean.securityModeChoices = "open".	"value"
-W openModeBean.portNumber: Access Server port number. Use only if securityModeBean.securityModeChoices = "open".	"port number"
-W openModeBean.password: WebGate password (optional). Use only if securityModeBean.securityModeChoices = "open".	"password"

Table 15–5 (Cont.) Silent Installation Parameters for WebGate

WebGate Parameter and Description	Possible Values
-W simpleModeBean.serverID : Access Server ID. Use the value you supplied at the Access System Console before installation. Use only if securityModeBean.securityModeChoices = "simple".	<i>"value"</i>
-W simpleModeBean.hostName : Access Server host name. Use only if securityModeBean.securityModeChoices = "simple".	<i>"ip address "</i> or <i>"hostname"</i>
-W simpleModeBean.webgateID : The WebGate ID. Use the value you supplied at the Access System Console. Use only if securityModeBean.securityModeChoices = "simple".	<i>"value"</i>
-W simpleModeBean.portNumber : The Access Server port number. Use only if securityModeBean.securityModeChoices = "simple".	<i>"port number"</i>
-W simpleModeBean.password : The WebGate password (optional). Use only if securityModeBean.securityModeChoices = "simple".	<i>"password"</i>
-W simpleModeBean.passphrase : The pass phrase. Use only if securityModeBean.securityModeChoices = "simple".	<i>"passphrase"</i>
-W simpleModeBean.passphraseVerify : The pass phrase again, used for verification and should be the same as simpleModeInfoBean.passphrase. Use only if securityModeBean.securityModeChoices = "simple".	<i>"passphrase"</i>
-W certModeBean.serverID : The Access Server ID. Use the value you supplied at the Access System Console before installation. Use only if securityModeBean.securityModeChoices = "cert".	<i>"value"</i>
-W certModeBean.hostName : The Access Server host name use only if securityModeBean.securityModeChoices = "cert".	<i>"ip address "</i> or <i>"hostname"</i>
-W certModeBean.webgateID : The WebGate ID (optional). Use the value you supplied at the Access System Console. Use only if securityModeBean.securityModeChoices = "cert".	<i>"value"</i>
-W certModeBean.portNumber : The Access Server port number. Use only if securityModeBean.securityModeChoices = "cert".	<i>"port number"</i>
-W certModeBean.password : The WebGate password. Use only if securityModeBean.securityModeChoices = "cert".	<i>"password"</i>
-W certModeBean.passphrase : The pass phrase. Use only if securityModeBean.securityModeChoices = "cert".	<i>"passphrase"</i>
-W askAutoUpdateWSBean.askAutoUpdateWSField : Determines whether to perform an automatic update of the Web server configuration.	<i>"Yes"</i> or <i>"No"</i>
-W askConfFilePathBean.askConfFilePathField : For NSAPI, this is the absolute path of the Web server configuration directory containing the obj.conf (for example: /export/Planet/servers/https-oblix/config). For Apache/Apache SSL, this is the absolute path of httpd.conf in your Web server config directory (for example: /export/apache/conf/httpd.conf). Use only for Apache, Apache SSL, and NSAPI Web servers and if askAutoUpdateWSBean.askAutoUpdateWSField = "Yes".	<i>"absolute path (including the file name for Apache)"</i>
-W certModeBean.passphraseVerify : The pass phrase again, used for verification and should be the same as certModeInfoBean.passphrase. Use only if securityModeBean.securityModeChoices = "cert".	<i>"passphrase"</i>

Table 15–5 (Cont.) Silent Installation Parameters for WebGate

WebGate Parameter and Description	Possible Values
-W installOrRequestCertBean.installOrRequest: Determines whether to install or request a certificate to be used to configure the Access System. Used if your security mode is set to "cert". If you already have a certificate, use "install". If you want Oracle Access Manager to request and request a certificate, use "request".	"request" or "install"
-W certReqInfoBean.countryName: Country name. This is a two-letter country code that is valid for use in a DN. It is part of the information used to request a certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	"country code"
-W certReqInfoBean.stateOrProvinceName: State or province name. This is a two-letter code that is valid for use in a DN. It is part of the information used to request a certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	"state or province code"
-W certReqInfoBean.localityName: Locality name. This is usually a geographic region. It is part of the information used to request a certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	"locality name"
-W certReqInfoBean.organizationName: Organization name. This is usually the name of the organization. It is part of the information used to request certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	"organization name"
-W certReqInfoBean.organizationalUnitName: Organization unit name. This is usually a department name. It is part of the information used to request certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	"organization unit name"
-W certReqInfoBean.commonName: Common name. This is usually a person's or entity's name. It is part of the information used to request certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	"common name"
-W certReqInfoBean.emailAddress: Email address. This is usually a valid email address. This is part of the information used to request certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	"email address"
-W readyToInstallCertBean.readyToInstallField: If you requested that Oracle Access Manager request a certificate, this verifies that the certificate is ready for installation. Only used if installOrRequestCertBean.installOrRequest = "request". Oracle recommends that you use a value of "No" for silent mode. It is unlikely that you can take the request generated by Oracle Access Manager and receive the certificate faster than the Oracle Access Manager installation can run from one step to the next.	"Yes" or "No"
-W copyCertificatesInputBean.certFile: A certificate is composed of three files: a certificate file, a key file, and a chain file. This parameter specifies the absolute path including the file name for the certificate file (for example: aaa_cert.pem). Use if: installOrRequestCertBean.installOrRequest = "install" or if installOrRequestCertBean.installOrRequest = "request" and readyToInstallCertBean.readyToInstallField = "Yes".	"absolute path including the file name"
-W copyCertificatesInputBean.keyFile: A certificate is composed of three files: a certificate file, a key file, and a chain file. This parameter specifies the absolute path including the file name for the key file (for example: aaa_key.pem). Use if: installOrRequestCertBean.installOrRequest = "install" or if installOrRequestCertBean.installOrRequest = "request" and readyToInstallCertBean.readyToInstallField = "Yes".	"absolute path including the file name"

Table 15–5 (Cont.) Silent Installation Parameters for WebGate

WebGate Parameter and Description	Possible Values
-W copyCertificatesInputBean.chainFile: A certificate is composed of three files: a certificate file, a key file, and a chain file. This parameter specifies the absolute path including the file name for the chain file (for example: aaa_chain.pem). Use if: installOrRequestCertBean.installOrRequest = "install" or if installOrRequestCertBean.installOrRequest = "request" and readyToInstallCertBean.readyToInstallField = "Yes."	"absolute path including the file name"

Access Manager SDK Parameters

Table 15–6 describes silent installation parameters for the Access Manager SDK.

Table 15–6 Silent Installation Parameters for the SDK

SDK Parameter and Description	Possible Values
-P sdk.installLocation: The installation directory. The default directory is "C:\COREid" on Windows and "/coreid" on UNIX.	"installation directory"
-W userInfoBean.user: UNIX only. The user ID that the product will be running as.	"user ID"
-W userInfoBean.group: UNIX only. The group that corresponds to the userInfoBean.user.	"group id"

BEA WebLogic SSPI Parameters

Table 15–7 describes silent installation parameters for BEA WebLogic SSPI.

Table 15–7 Silent Installation Parameters for BEA WebLogic SSPI

BEA SSPI Parameters and Description	Possible Value
-P bea.installLocation: The installation directory. The default directory is "C:\COREid" on Windows and "/coreid" on UNIX.	"installation directory"
-W localePanel.defaultLang: Required when extra languages are to be installed with the main installation.	"en-us"
-W localePanel.installLanguages: Required when extra languages are to be installed with the main installation	"en-us;fr-fr"
-W sspiConfigLevel.ConfigMode: Configuration options. Typical option will require minimal inputs. Advanced option enables overriding of all defaults.	"typical;advanced"
-W verifyUserBean.verifyUserBeanField: Determines whether the user installing the product is the same one that the product should be running as. If the value is No, the installation exits.	"Yes" or "No"
-W sspiAdv1.authResType: Resource type used by Oracle Access Manager Security Provider in the policy to authenticate users in Weblogic. wl_authen	wl_authen
-W sspiAdv1.authRes: Resource name used by Oracle Access Manager Security Provider in the policy to authenticate users in Weblogic.	/Authen/Basic
-W sspiAdv1.authResOp: Resource operation used by Oracle Access Manager Security Provider in the policy to authenticate users in Weblogic. LOGIN	LOGIN

Table 15–7 (Cont.) Silent Installation Parameters for BEA WebLogic SSPI

BEA SSPI Parameters and Description	Possible Value
-W sspiAdv1.authAnonymousRes: Resource name used for anonymous access by Oracle Access Manager Security Provider in the policy to authenticate users in Weblogic. / Authen/Anonymous	Authen/Anonymous
-W sspiAdv1.authUID: LoginId--parameter used in credential_ mapping plugin of authentication. userid	userid
-W sspiAdv1.authPass: Password parameter used in validate password of authentication scheme. Password	Password
-W sspiAdv1.authnActionType: Action Type (action is configured to get the loginId from ObSSOCookie). WL_REALM	WL_REALM
-W sspiAdv1.authnActionName: Action Name (action is configured to get the loginId from ObSSOCookie). uid	uid
-W sspiAdv1.obDummyUser: Dummy username used by form login for doing SSO when there is no webgate on proxy HTTP server. Obdummyuser	Obdummyuser
-W sspiAdv2.webAppResourceTypes: Weblogic resource types used for Web applications (comma separated) <url>,<web>	url>,<web>
-W sspiAdv2.roleResType: Resource type used by Oracle Access Manager Security Provider in the policy to get roles for a user.	wl_authen
-W sspiAdv2.roleRes: Resource name used by Oracle Access Manager Security Provider in the policy to get roles for a user.	/Authen/Roles
-W sspiAdv2.roleResOp: Resource operation used by Oracle Access Manager Security Provider in the policy to get roles for a user.	LOGIN
-W sspiAdv2.rolesCacheTTL: TTL(time to live) of elements in roles cache.	60
-W sspiAdv2.rolesCacheCleanupSchedule: Time to delete expired elements of cache (in seconds).	60
-W sspiAdv2.roleActionType: Action Type in authorization rule to get roles.	WL_REALM
-W sspiAdv3.notProtectedAction: Default access to resources not protected by Oracle Access Manager.	allow;deny;abstain
-W sspiAdv3.abstainMapsTo: Map the authorization result ABSTAIN to (allow,deny).	allow;deny
-W sspiAdv3.debug: Set debugging (This should be set to Off for production systems).	1 - On 2 - Off
-W securityModeBean.securityModeChoices: The security mode for BEA SSPI. A value of "open" means no security is used, a value of "simple" means encryption is used, and a value of "cert" means you are running your own CA.	"open", "simple", "cert"
-W openModeBean.serverID: Access Server ID. Use the value you supplied at the Access System Console before installation. Use only if securityModeBean.securityModeChoices = "open". Example: "AccessServer1".	"server ID"
-W openModeBean.hostName: Host name where Access Server is installed. Use only if securityModeBean.securityModeChoices = "open".	"ip_address" or "hostname"

Table 15–7 (Cont.) Silent Installation Parameters for BEA WebLogic SSPI

BEA SSPI Parameters and Description	Possible Value
<p>-W openModeBean.accessGateID: Access Gate ID. Use only if securityModeBean.securityModeChoices = "open". Example: "WeblogicRealm1".</p>	"value"
<p>-W openModeBean.portNumber: Port number of the Access Server. Use only if securityModeBean.securityModeChoices = "open".</p>	"port_number"
<p>#-W openModeBean.password: Password for Access Gate, if one is set. Use only if securityModeBean.securityModeChoices = "open".</p>	"password"
<p>-W simpleModeBean.serverID: Access Server ID. Use the value you supplied at the Access System Console before installation. Use only if securityModeBean.securityModeChoices = "simple".</p>	"server ID"
<p>-W simpleModeBean.hostName: Host name where Access Server is installed. Use only if securityModeBean.securityModeChoices = "simple".</p>	"ip address" or "hostname"
<p>-W simpleModeBean.accessGateID: Access Gate ID. This value has to match the one you specified at the Access System Console. Use only if securityModeBean.securityModeChoices = "simple".</p>	"value"
<p>-W simpleModeBean.portNumber --Port number of the Access Server. Use only if securityModeBean.securityModeChoices = "simple".</p>	"port number"
<p>#-W simpleModeBean.password: Password for Access Gate, if one is set. Use only if securityModeBean.securityModeChoices = "simple".</p>	"password"
<p>#-W simpleModeBean.passphrase: Pass phrase for the Access Gate to communicate with the Access Server. Use only if securityModeBean.securityModeChoices = "simple".</p>	"passphrase"
<p>#-W simpleModeBean.passphraseVerify: Pass phrase for the Access Gate to communicate with the Access Server. This parameter verifies that the pass phrase matches that of securityModeBean.passphrase. Use only if securityModeBean.securityModeChoices = "simple".</p>	"passphrase"
<p>-W certModeBean.serverID: Access Server ID. Use the value you supplied at the Access System Console before installation. This value has to match the one you specified at the Access System Console. Use only if securityModeBean.securityModeChoices = "cert".</p>	"value"
<p>-W certModeBean.hostname: Host name where Access Server is installed. Use only if securityModeBean.securityModeChoices = "cert".</p>	"ip address" or "hostname"
<p>-W certModeBean.accessGateID: Access Gate ID. This value has to match the one you specified at the Access System Console. Use only if securityModeBean.securityModeChoices = "cert".</p>	"value"
<p>-W certModeBean.portNumber: Port number of the Access Server. Use only if securityModeBean.securityModeChoices = "cert".</p>	"port number"

Table 15–7 (Cont.) Silent Installation Parameters for BEA WebLogic SSPI

BEA SSPI Parameters and Description	Possible Value
-W certModeBean.password : Password for Access Gate, if one is set. Use only if securityModeBean.securityModeChoices = "cert". "password"	"port number"
-W certModeBean.passphrase : Pass phrase allowing the Access Gate to communicate with the Access Server. Use only if securityModeBean.securityModeChoices = "cert".	"passphrase"
-W certModeBean.passphraseVerify : Pass phrase allowing the Access Gate to communicate with the Access Server. Use only if securityModeBean.securityModeChoices = "cert".	"passphrase"
-W installOrRequestCertBean.installOrRequest : Determines whether to install or request a certificate to be used to configure the Access System. Used if your security mode is set to "cert". If you already have a certificate, choose "install". If you want Oracle Access Manager to request and request a certificate, choose "request".	"install", "request"
-W certReqInfoBean.countryName : Country name. This is usually a two-letter country code that is valid for use in DNs. It is part of the information used to request a certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	"country code"
-W certReqInfoBean.stateOrProvinceName : State or province name. This is usually a two-letter state or province code that is valid for use in a DN. It is part of the information used to request a certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	"state or province code"
-W certReqInfoBean.localityName : Locality name. This is usually the name of a geographic region. This is part of the information used to request a certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	"locality name"
-W certReqInfoBean.organizationName : Organization name. This is usually the name of an organization. It is part of the information used to request a certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	"organization name"
-W certReqInfoBean.organizationalUnitName : Organization unit name. This is usually the name of a department. It is part of the information used to request a certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	"organization unit name"
-W certReqInfoBean.commonName : Common name. This is usually the name of a person or an entity. It is part of the information used to request a certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	"name"
-W certReqInfoBean.emailAddress : Email address. This is usually a valid email address. This is part of the information used to request a certificate. Use only if installOrRequestCertBean.installOrRequest = "request".	"email address"
-W readyToInstallCertBean.readyToInstallField : If you requested that Oracle Access Manager request a certificate, this verifies that the certificate is ready for installation. This is only used if installOrRequestCertBean.installOrRequest = "request". Oracle recommends that you do not use "Yes" for silent mode. You probably cannot take the request generated by Oracle Access Manager and receive the certificates faster than the Oracle Access Manager installation can run from one step in the installation to the next.	"Yes" or "No"

Table 15–7 (Cont.) Silent Installation Parameters for BEA WebLogic SSPI

BEA SSPI Parameters and Description	Possible Value
-W copyCertificatesInputBean.certFile : The absolute path including the file name for the certificate file (for example: aaa_cert.pem). Use if: installOrRequestCertBean.installOrRequest = "install" or if installOrRequestCertBean.installOrRequest = "request" and readyToInstallCertBean.readyToInstallField = "Yes".	"absolute path including the file name"
-W copyCertificatesInputBean.keyFile : The absolute path including the file name for the key file (for example: aaa_key.pem). Use if: installOrRequestCertBean.installOrRequest = "install" or if installOrRequestCertBean.installOrRequest = "request" and readyToInstallCertBean.readyToInstallField = "Yes".	"absolute path including the file name"
-W copyCertificatesInputBean.chainFile : The absolute path including the file name to the chain file (for example: aaa_chain.pem). Use if: installOrRequestCertBean.installOrRequest = "install" or if installOrRequestCertBean.installOrRequest = "request" and readyToInstallCertBean.readyToInstallField = "Yes".	"absolute path including the file name"

WAS Registry Parameters

Table 15–8 provides silent installation parameters for the WAS registry.

Table 15–8 Silent Installation Parameters for WAS Registry

WAS Registry Parameter—Description	Possible Values
-P was_registry.installLocation : The installation directory. The default directory is "C:\Program Files\COREid" on windows and "/opt/coreid" on UNIX.	"installation directory"
-W verifyUserBean.verifyUserBeanField : Determines whether the user installing the product is the same one that the product should be running as. If the value is No, the installation exists.	"Yes" or "No"
-W wasConfig.WPHostName : Hostname of Webpass	"machine name"
-W wasConfig.WPPortNumber : Port Number of webpass	"port number"
-W wasConfig.WPisProtected : Is webpass protected by webgate.	"true" or "false"
-W wasWebPassConfig.cookieDomain : Cookie domain set for WebGate, for example "company.com"	"domain name"
-W wasWebPassConfig.cookiePath : Cookie path set for WebGate, for example, "/"	"path"
-W wasDSConfig.WPSSL : Determines whether the Oracle Access Manager connector for websphere requires WebPass to connect to it in SSL mode (transmitting data using https).	"true" or "false"
-W wasDSConfig.UserAttr : User attribute.	"uid"
-W wasDSConfig.UserSearchAttr : User search attribute.	"cn"
-W wasDSConfig.GroupSearchAttr : Group search attribute	"cn"
-W wasWSClassesDir.classesDir : Full Path of the WebSphere classes directory.	"path"

Table 15–8 (Cont.) Silent Installation Parameters for WAS Registry

WAS Registry Parameter—Description	Possible Values
-W configPortalInput.isPortalTobeUsed: Oracle Access Manager WebSphere connector requires certain files to be copied to WebSphere Application directory for websphere portal server integration. This parameter asks if portal server needs to be integrated.	"true" or "false"
-W wasInfoBean.wasInstallDir: If -W configPortalInput.isPortalTobeUsed = "true" then enter WebSphere Application directory path.	<i>\$WebSphere_install_dir / AppServer</i>
-W securityModeBean.securityModeChoices: AccessGate mode configuration. A value of "open" means no security is required, a value of "simple" means encryption is used, and a value of "cert" means you are running your own CA.	"open", "simple" or "cert"
-W openModeBean.serverID: Access server ID. Use value you specified at the Access System Console before installation. Use only if securityModeBean.securityModeChoices = "open ".	"server id"
-W openModeBean.hostname: Machine name where Access Server is installed. Use only if securityModeBean.securityModeChoices = "open ".	"ip_addr" or "machine_name"
-W openModeBean.accessGateID: AccessGate ID. Use only if securityModeBean.securityModeChoices = "open "	"AccessGate ID"
-W openModeBean.portNumber: Port number of the access server.	"port number"
#-W openModeBean.password: Password for AccessGate if one is set. Use only if securityModeBean.securityModeChoices = "open".	"password"
-W simpleModeBean.serverID: Access Server ID. Use value you specified at the Access System Console before installation. Use only if securityModeBean.securityModeChoices = "simple "	"Access Server ID"
-W simpleModeBean.hostname: Host name where access server is installed. Use only if securityModeBean.securityModeChoices = "simple "	"ip_addr" or "machine_name"
-W simpleModeBean.accessGateID: AccessGate ID. Use only if securityModeBean.securityModeChoices = "simple "	"AccessGate ID"
-W simpleModeBean.portNumber: Port number of the access server. Use only if securityModeBean.securityModeChoices = "simple "	"port number"
#-W simpleModeBean.password: Password for access gate. Use only if securityModeBean.securityModeChoices = "simple "	"password"
#-W simpleModeBean.passphrase: Pass phrase for the access gate to communicate with the Access Server. Use only if securityModeBean.securityModeChoices = "simple "	"passphrase"
#-W simpleModeBean.passphraseVerify: Pass phrase for the access gate to communicate with the access server. This parameter verifies that the pass phrase matches the simpleModeBean.passphrase. Use only if securityModeBean.securityModeChoices = "simple "	"passphrase"
-W certModeBean.serverID: Access Server ID. Use the value you supplied at the Access System Console before installation. This value has to match the one you specified at the Access System Console. Use only if securityModeBean.securityModeChoices = "cert".	"server id"
-W certModeBean.hostname: Host name where Access Server is installed. Use only if securityModeBean.securityModeChoices = "cert".	"ip address" "ip addr" or "machine name"

Table 15–8 (Cont.) Silent Installation Parameters for WAS Registry

WAS Registry Parameter—Description	Possible Values
-W certModeBean.accessGateID : AccessGate ID. This value has to match the one you specified at the Access System Console. Use only if <code>securityModeBean.securityModeChoices = "cert"</code> .	"AccessGate ID"
-W certModeBean.portNumber : Port number of the Access Server. Use only if <code>securityModeBean.securityModeChoices = "cert"</code> .	"port number"
#-W certModeBean.password : Password for Access Gate, if one is set. Use only if <code>securityModeBean.securityModeChoices = "cert"</code> .	"password"
#-W certModeBean.passphrase : Pass phrase allowing the AccessGate to communicate with the Access Server. Use only if <code>securityModeBean.securityModeChoices = "cert"</code> .	"passphrase"
#-W certModeBean.passphraseVerify : Pass phrase allowing the Access Gate to communicate with the Access Server. This parameter verifies that the pass phrase matches that of <code>certModeBean.passphrase</code> . Use only if <code>securityModeBean.securityModeChoices = "cert"</code> .	"passphrase"
-W installOrRequestCertBean.installOrRequest : Determines whether to install or request a certificate to be used to configure the Access System. Used if your security mode is set to "cert". If you already have a certificate, choose "install". If you want Oracle Access Manager to request and request a certificate, choose "request".	"install", "request"
-W certReqInfoBean.countryName : Country name. This is usually a two-letter country code that is valid for use in DNs. It is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	"country code"
-W certReqInfoBean.stateOrProvinceName : State or province name. This is usually a two-letter state or province code that is valid for use in a DN. It is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	"state or province code"
-W certReqInfoBean.localityName : Locality name. This is usually the name of a geographic region. This is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	"locality name"
-W certReqInfoBean.organizationName : Organization name. This is usually the name of an organization. It is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	"organization name"
-W certReqInfoBean.organizationalUnitName : Organization unit name. This is usually the name of a department. It is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	"organization unit name"
-W certReqInfoBean.commonName : Common name. This is usually the name of a person or an entity. It is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	"name"
-W certReqInfoBean.emailAddress : Email address. This is usually a valid email address. This is part of the information used to request a certificate. Use only if <code>installOrRequestCertBean.installOrRequest = "request"</code> .	"email address"

Table 15–8 (Cont.) Silent Installation Parameters for WAS Registry

WAS Registry Parameter—Description	Possible Values
-W readyToInstallCertBean.readyToInstallField: If you requested that Oracle Access Manager request a certificate, this verifies that the certificate is ready for installation. This is only used if <code>installOrRequestCertBean.installOrRequest = "request"</code> . Oracle recommends that you do not use "Yes" for silent mode. You probably cannot take the request generated by Oracle Access Manager and receive the certificates faster than the Oracle Access Manager installation can run from one step in the installation to the next.	"Yes" or "No"
-W copyCertificatesInputBean.certFile: The absolute path including the file name for the certificate file (for example: <code>aaa_cert.pem</code>). Use if: <code>installOrRequestCertBean.installOrRequest = "install"</code> or if <code>installOrRequestCertBean.installOrRequest = "request"</code> and <code>readyToInstallCertBean.readyToInstallField = "Yes"</code> .	"absolute path including the file name"
-W copyCertificatesInputBean.keyFile: The absolute path including the file name for the key file (for example: <code>aaa_key.pem</code>). Use if: <code>installOrRequestCertBean.installOrRequest = "install"</code> or if <code>installOrRequestCertBean.installOrRequest = "request"</code> and <code>readyToInstallCertBean.readyToInstallField = "Yes"</code> .	"absolute path including the file name"
-W copyCertificatesInputBean.chainFile: The absolute path including the file name to the chain file (for example: <code>aaa_chain.pem</code>). Use if: <code>installOrRequestCertBean.installOrRequest = "install"</code> or if <code>installOrRequestCertBean.installOrRequest = "request"</code> and <code>readyToInstallCertBean.readyToInstallField = "Yes"</code> .	"absolute path including the file name"
-W localePanel.defaultLang: Required when extra languages are to be installed with the main installation.	"en-us"
-W localePanel.installLanguages: Required when extra languages are to be installed with the main installation.	"en-us;fr-fr"

Uninstalling a Component Installed With Silent Mode

The method to uninstall a component that was installed using silent mode depends upon your platform.

On Windows: Run:

```
Component_install_dir\oblix\_uninstcomponent\uninstaller.exe -silent
```

On Solaris: Run:

```
Component_install_dir/oblix/_uninstcomponent/uninstaller.bin -silent
```

where `Component_install_dir` refers to the installation directory where the component is installed (in your path name `\identity` refers to the Identity System and `\access` refers to the Access System).

To remove components installed using GUI or Console method, see [Chapter 20, "Removing Oracle Access Manager"](#).

Cloning and Synchronizing Installed Components

Rather than using the command line or the installation GUI to install a component, you can automatically install a component by *cloning* the configuration of an already-installed component.

Cloning: Creates a mirrored copy of a component. That is, cloning creates a copy of a component on a local or remote system using an already-installed component as a template. Once a Identity Server or Access Server is cloned, you can:

- Start the cloned server on the remote system
- Reconfigure a cloned server after it has been started
- Partially replicate a configuration by synchronizing two installed components.

On Windows and UNIX, in the directory `oblix/tools/np_sync`, you can use the command `np_sync` to clone a component. The `np_sync` tool is described in "[Syntax and Options for np_sync](#)" on page 15-30.

Synchronizing: Allows you to harmonize two installations of the same Oracle Access Manager component when one is more up-to-date than the other. Synchronization can be used to upgrade or repair installations on similar platforms. To synchronize two components, use the `-sync` or `-sync-all` command-line options for the `np_sync` tool.

Note: For the Web server plug-ins WebPass, Policy Manager, and WebGate, the Web server configuration files are not updated using `np_sync`. This must be done either automatically during installation or manually afterward, as discussed earlier.

An Example of Using `np_sync`

Once a component has been installed and configured, a command such as:

```
np_sync -clone test2.oblix.com /export/home1/np7test2
```

clones the current machine to the system `test2.oblix.com` in the directory `/export/home1/np7test2`.

Syntax and Options for `np_sync`

The basic syntax for `np_sync` is as follows:

```
./np_sync -mode [-opts] host destination_dir
```

where *mode* is one of the following, `sync`, `sync-all`, or `clone`. For example:

-sync: The `-sync` command only updates files that are machine independent. These are customization (text) files. This command can be used to upgrade or repair installations on similar platforms. For example, if you have an AIX system and a Solaris system, you should be able to synchronize them.

-sync-all: The `-sync-all` command includes binaries, shared libraries, executable files, and so on as well as customization (text) files. This command can be used to upgrade or repair installations on similar platforms.

-clone: The `-clone` option copies the entire installation. This option also requires using the `-p port` and `-n servername` options described later.

where *-opts* is any combination from the following:

-u username: UNIX only. When you issue the `np_sync` command to connect to a remote system as a user other than the one you are logged in as, use the `-u username` option. This does not change the credentials that you use, but rather changes the user who executes the receiving end of the remote-copy command.

-rsync: UNIX only. Use the `rsync` command (`rdist` is used by default, see "[UNIX-Specific Notes](#)" on page 15-31).

-ssh: UNIX only. Use ssh, the secure shell that uses an encrypted connection, to transfer data when using rsync. When using the rsync command, you may use ssh instead of the standard UNIX remote shell connection (rsh or remsh).

-path rsyncpath: UNIX only. Look for rsync in rsyncpath on the remote system (when using rsync).

-d: Debug mode: do not copy, just indicate what will be updated.

-l sorter: Use sorter as the source directory. By default, the current Oracle Access Manager installation area (where this program is located) is used as the source.

-n servername: For cloning an Identity or Access Server, you must specify a new Oracle Access Manager server name. Use servername for the new server.

-p portnumber: For cloning a Identity or Access Server, you need to specify a port. Use portnumber for the new server port.

Windows Only

-F: Windows only. This option forces an installation (ignore sanity checks). You may use the -F flag to force an installation to take place, even if some normal checks fail. This can be useful for re-executing a cloning operation that failed midway.

-f: Windows only. The -f flag forces a copy, ignoring the file modification times on the remote system, and updates all relevant files.

-r: Windows only. This option reboots the remote host after installation, if necessary. Use this option with cloning if system libraries need to be updated.

See also, "[Windows-Specific Notes](#)" on page 15-31.

UNIX-Specific Notes

On UNIX, remote copy permissions must be enabled using .rhosts.

The exact list of files to be copied for the -clone, -sync-all, and -sync command options is defined in the np_sync script. You may need to tune these files for special cases.

By default, UNIX uses the rdist command to update the remote system. Solaris assumes that rdist exists in /usr/ucb/ on the remote system, so it may not work on a different platform.

The rdist command is not usually shipped on Linux. You can use the rsync program on Linux. The rsync program is not usually shipped with Solaris, HP-UX, or AIX.

On UNIX, the remote system must grant permission for remote access, typically by using the .rhosts file of the remote system. The format of this file is any number of lines of the form *host username*, where *host* refers to the system that the copy is coming from, and *username* specifies the user who is permitted to issue the remote copy command.

Windows-Specific Notes

The Windows version of np_sync is an executable program (np_sync.exe).

The definitions for what files are transferred for the -clone, -sync, or -sync-all command options is defined in the patterns file in the np_sync subdirectory on Windows. You may need to tune the patterns files for special cases.

On Windows, the np_sync command automatically mounts the network drives necessary to complete cloning or synchronization. It unmounts the network drives when finished if the user does not interrupt the process.

When cloning on Windows, the np_sync tool also updates the system registry, updates any necessary system DLL files, and installs the appropriate entry in the system services. The np_sync command does not start or stop system services. Use the program NPServMgr.exe, in the directory oblix/tools/NPServMgr/ (on Windows only) to start, stop, add, or remove any Oracle Access Manager servers in the Windows system services.

Updating the registry and system services requires the local Windows user to have system administrator privileges on the remote system. To achieve this, use a network administrator login or assign administration privileges to the same user name and password on the remote system.

Note: If the system drive is on a partition other than C: the "-S" or "-R" flag needs to be used.

The np_sync command uses the default system directory C:\WINNT\system32. However, on Windows XP and Windows Server 2003 the system directory is C:\Windows\system32. When the local system or remote system operating system version is after Windows 2000, you need to use the following in the np_sync command:

"-S" flag for local system directory

"-R" flag for remote system directory

Uninstalling a Cloned Component

The following sections describe how to uninstall a cloned component on UNIX and on Windows:

- [Uninstalling a Cloned Component on UNIX](#)
- [Uninstalling a Cloned Component on Windows](#)

Note: To remove components installed using GUI or Console method, see [Chapter 20, "Removing Oracle Access Manager"](#).

Uninstalling a Cloned Component on UNIX

The procedure to uninstall on UNIX systems follows.

To uninstall on UNIX

1. If the component is WebPass, Policy Manager, or WebGate, delete the Oracle Access Manager-specific entries in their Web server's obj.conf file.
2. If the component runs a process (Identity Server, Access Server), stop the process.
3. Delete the component's directory.

Uninstalling a Cloned Component on Windows

You cannot uninstall a cloned component using InstallShield. On Windows, uninstallation requires removing registry entries. Installed services must be removed using a utility provided by Oracle.

Uninstalling Oracle Access Manager System

Two procedures follow.

To uninstall Identity and Access Server

1. Uninstall the Identity or AAA service using NPServMgr.exe located in the *Component_install_dir*\access\oblix\tools directory. Usage information is displayed by running NPServMgr.exe without any arguments.
2. Delete the registry entries associated with the component.
3. Delete the Identity or Access Server installation directory.

To uninstall WebPass, WebGate, and Policy Manager

1. Remove the Oracle Access Manager modifications from the Web server's obj.conf (NSAPI), or the Oracle Access Manager .dll's and virtual directories (ISAPI).
2. Stop the Web server instance that is hosting the component.
3. Delete the registry entries.
4. Delete the installation directory.

Part VI

Web Server Configuration

This part discusses Web server configuration information.

Part VII contains the following chapters:

- [Chapter 16, "Configuring the Apache v1.3 and Oracle HTTP Server Web Servers"](#)
- [Chapter 17, "Configuring Apache v2, IHS, and OHS Web Servers for Oracle Access Manager"](#)
- [Chapter 18, "Setting Up Lotus Domino Web Servers for WebGates"](#)

Configuring the Apache v1.3 and Oracle HTTP Server Web Servers

This chapter explains how to configure the Apache v1.3 Web server for Oracle Access Manager. 10g (10.1.4.0.1) Web components (WebPass, Policy Manager, WebGate) also provides support for Oracle HTTP Server (OHS) 10g R2 (10.1.2).

Note: The Oracle HTTP Server (OHS) implementation is based on open source Apache v1.3.

The following topics are included:

- [About OHS and Oracle Access Manager](#)
- [About Apache v1.3 and Oracle Access Manager](#)
- [Apache v1.3, Oracle HTTP Server \(OHS\), and Stronghold Requirements](#)
- [Apache v1.3, OHS, and IHS Web Server Support](#)
- [Downloading and Compiling the Base Apache Web Server](#)
- [Platform-Specific Compilation Options](#)
- [Platform Specific Run-Time Settings for AIX](#)
- [Installation Order for Oracle Access Manager Web Components](#)
- [Updating Web Server Configuration for Oracle Access Manager Web Components](#)
- [Tuning Apache 1.3 for Oracle Access Manager Web Components](#)
- [Setting OHS Client Certificates](#)
- [Tuning OHS for Oracle Access Manager Web Components](#)
- [Starting and Stopping the Web Server](#)

Note: See also [Chapter 17, "Configuring Apache v2, IHS, and OHS Web Servers for Oracle Access Manager"](#).

About OHS and Oracle Access Manager

OHS is an Internet Web server extension that identifies Oracle Access Manager Web components that communicate with the Oracle HTTP Server (OHS). Oracle Access Manager 10g (10.1.4.0.1) provides WebPass, Policy Manager, and WebGate components that can be installed on a standalone Oracle HTTP Server on Linux and Windows

platforms. You may use the OHS Web components for any application; however, there are some specific requirements you should be aware of:

- An OHS WebGate must be installed on the Oracle Application Server to enable integration with Oracle single sign-on as described in the *Oracle Access Manager Integration Guide*.
- OHS WebPass and OHS Access Manager (Policy Manager) components may be used with the Oracle Application Server. However, Apache WebPass and Apache Policy Manager are also supported for this application.

OHS 10g R2 (10.1.2) is based on Apache v1.3. See the *Oracle HTTP Server Administrator's Guide 10 g R2 (10.1.2)* for complete details about OHS, including:

- Differences between the Oracle implementation and the open source Apache product on which it is based
- Oracle HTTP Server directory structure, configuration files and syntax, modules, and directives
- Managing server processes, network connections, and security features

Be sure to familiarize yourself with the following OHS Web component caveats. See also "[Setting OHS Client Certificates](#)" on page 16-10.

OHS Web Component Caveats on Linux

When using Oracle Access Manager 10g (10.1.4.0.1) Web components for OHS, be sure to take these caveats into account for Linux:

- The `LD_ASSUME_KERNEL=2.4.19` environment variable must be set for Linux at runtime because the older Linux threading model is supported (*not* the native posix thread library (NPTL))
- Policy Manager communication to the directory server in SSL mode on Linux is *not* supported
- When installing Oracle Access Manager Web components for Apache or OHS on Linux, you are prompted to install as the same user under which the Web server is running. See the User and Group directive entries in the `httpd.conf` file.

OHS Web Component Caveats on Linux and Windows Platforms

When using Oracle Access Manager 10g (10.1.4.0.1) Web components for OHS on Windows and Linux platforms, the Perl module must be commented out in the `httpd.conf` for both Windows and Linux platforms.

About Apache v1.3 and Oracle Access Manager

This section explains how Apache's process-based architecture affects various Oracle Access Manager Web components:

- [Identity Server Accessed through WebPass](#)
- [Policy Manager](#)
- [WebGate](#)

Identity Server Accessed through WebPass

For Identity Servers communicating with WebPass for Apache v1:

- Each WebPass instance connects to the Identity Server.
- Each connection takes up system sources, and each connection has n file descriptors.
- Set the tuning parameters for Apache so that Apache does not need to start or stop processes too frequently. These settings would be similar with or without Oracle Access Manager.

Policy Manager

With Policy Managers for Apache v1:

- Each Web server process is an instance of the Policy Manager application.
- Each application maintains its own connections to the directory server. This may not directly affect performance. However, there may be a limit on the directory server side that you may want to consider when other directory server clients are involved.
- Multiple processes respond to a user's request (multiple HTTP events are triggered to build the frames).
- Latency of responses cannot be predicted.
- Fewer processes are better from a UI perspective, but this affects the number of concurrent users.

WebGate

With WebGate for Apache v1:

- There is no shared cache between processes.
- Each process maintains its own connections to the Access Server.
- Because each process has its own connection, you should limit the number of WebGate connections. This issue is partially affected by the performance of the systems running the Web servers and Access Servers.
- Reverse proxy capability may be enabled for Apache Web servers on Solaris and Linux platforms.

Example: Apache v1.3 Configuration for Unix Systems

Apache v1.3 is a process-based Web server. The Apache Web server creates a client process with the filters configured to process each request. As a result, each client process will have a WebGate included in it.

If the Apache Web server is configured to use 250 MaxClients (at peak load), there will be 250 WebGates in execution when all these processes execute. Each WebGate will use the connection configuration specified in the WebGate definition. For example, if the WebGate configuration has 4 connections to an Access Server, each of the client process will create 4 connections to the Access Server configured. So, at peak load, there would be $250 * 4 = 1000$ connections to the Access Server.

At the Access Server, each connection corresponds to a Message Thread, which reads client requests from the sockets. At peak load, it would have 1000 threads just to process requests from 1 WebGate with the above configuration. If there were multiple Apache v1.3 Web servers are configured to go to an Access Server, this number would be the sum of connections from each such Web server. The effect of this would be a large number of threads in the Access Server.

As the number of threads in a process increase, the overhead with respect to CPU utilization, memory utilization increases. Some operating systems would start thrashing if the number of threads is overwhelming, which negatively impacts Access Server performance.

The Apache v1.3 Web server creates and destroys client processes based on the request load. It load balances the requests among the child processes, As a result, each child process will have, at most, 1 request to process for which it requires only one connection to one of the Access Servers.

You could configure 2-3 Access Servers for a WebGate so that the load on the Access Servers from this WebGate is balanced. For example:

If the WebGate1 has 1 connection to Access Server1
1 connection to Access Server 2
1 connection to Access Server 3
Then MaxConnections=3

In the case above:

The 1st request from Web server to WebGate1 goes to Access Server1
2nd request from Web server to WebGate1 will go to Access Server2
3rd request from Web server to WebGate1 will go to Access Server3
4th request from Web server to WebGate1 will go to Access Server
and so on.

However, connection 1 to AccessServer2 and AccessServer3 may not be used if, during the life of the child process only, 1 request is sent to WebGate1 from the Web server.

To summarize, in a multi-process Web server such as Apache v1.3, you need only one connection to an Access Server.

In the case of Apache v2, the Web server can be configured to operate in 2 mpm modes: worker_mpm, pre-fork_mpm. Worker_mpm creates threads in the Web server for load balancing requests. In this case ThreadsPerChild and MaxClients are used to determine how many processes are spawned. The maximum number of active child processes is determined by the MaxClients directive divided by the ThreadsPerChild directive. Each process will have a webgate in it. In pre-fork mpm case, it behaves like Apache v1.3, creating 1 child process per request. For more information, see [Chapter 17, "Configuring Apache v2, IHS, and OHS Web Servers for Oracle Access Manager"](#).

Apache v1.3, Oracle HTTP Server (OHS), and Stronghold Requirements

Oracle Access Manager 10g (10.1.4.0.1) HTML pages use UTF-8 encoding. Apache-based Web servers allow administrators to specify a default character set for all HTML pages sent out using the AddDefaultCharset directive. This directive overrides any character specified by the application generating the HTML pages. If the AddDefaultCharset directive enables a character set other than UTF-8, Oracle Access Manager HTML pages are garbled.

Oracle recommends that you specify the AddDefaultCharset directive in the Web server configuration file (httpd.conf) as follows to ensure the correct display of Oracle Access Manager 10g (10.1.4.0.1) HTML pages:

```
AddDefaultCharset Off
```

See your Web server documentation for more information about this directive.

In addition, your system must also meet these requirements to implement an Apache v1.3, OHS, or Stronghold Web server:

- Dynamic Shared Object (DSO) support for WebGate and WebPass. On Apache, this means that `mod_so` must be enabled.

Note: DSO is required for all Oracle Access Manager plug-ins.

- Multi-threading is required for WebPass.
- When WebGate and WebPass are installed on the same Web server, DSO and multi-threading for WebPass are required.
- Building the Apache Web server requires access to the `gcc` and `make` commands in your path. Alternatively, you can use another ANSI-compliant C compiler.
- Reverse proxy capability may be enabled for Apache Web servers on Solaris and Linux platforms.
- When installing Oracle Access Manager Web components for Apache or OHS on Linux, you are prompted to install as the same user under which the Web server is running. See the User and Group directive entries in the `httpd.conf` file.

Apache v1.3, OHS, and IHS Web Server Support

Recent versions of Apache v1.3 contain important security fixes. It is strongly recommended that you use the most recent release of Apache 1.3. For details, see:

<http://apache.org>

The base Apache 1.3 Web server does not use SSL for browser connections (responding to `https://` requests). An add-on module for SSL support known as `mod_ssl` is available at:

<http://www.modssl.org>

The Oracle Access Manager plug-ins for base Apache servers are different from those for Apache with `mod_ssl` (also referred to as using EAPI) in following:

- Oracle Access Manager supports Apache with `mod_ssl` only.

Note: `openssl` is needed by `mod_ssl` when building Apache to support SSL. `openssl` should be part of the Apache server built with `mod_ssl`.

- No SSL-specific features of Oracle Access Manager operate with the version of Apache 1.3 known as Apache-SSL.

For complete details about OHS, see the *Oracle HTTP Server Administrator's Guide 10g R2 (10.1.2)*. For Oracle Access Manager 10g (10.1.4.0.1) support details, see the Certify tab on the following site:

<http://metalink.oracle.com>

To use Metalink

1. Navigate to <http://metalink.oracle.com>.
2. Log in to MetaLink as directed

3. Click the Certify tab.
4. Click View Certifications by Product.
5. Select the Application Server option and click Submit.
6. Choose Oracle Application Server and click Submit.

IHS (the IBM HTTP Server powered by Apache) is a variation of Apache 1.3. IHS uses a different implementation of SSL. Oracle Access Manager supports both the base IHS (non-SSL) and the IHS SSL plug-in interface.

For more information, see "[Preparing for Installation](#)" on page 2-1 and "[Downloading and Compiling the Base Apache Web Server](#)" on page 16-6.

Downloading and Compiling the Base Apache Web Server

This discussion applies only to Apache open source v1.3. You can download the latest version of Apache 1.3 from the Apache Web site:

<http://apache.org>

The SSL plug-in `mod_ssl` is available from:

<http://www.modssl.org>

These sites point you to other sites for any additional software needed by Apache or `mod_ssl` (such as `openssl`). Instructions for compiling the Apache Web server are included with the software distribution.

In order for the Apache Web server to support Oracle Access Manager plug-ins, the module `mod-so` must be compiled into the server binary.

To compile Apache or Apache with `mod_ssl` with `mod-so`:

1. Include the configuration option before compiling:

```
--enable-module=so
```
2. Ensure the configuration meets other Oracle Access Manager requirements and compile.

Apache Release Notes

The following URL contains information about the latest version of Apache and a link to pick up binary files for the Apache server:

<http://www.apache.org/dist/httpd/Announcement.html>

Other Useful Links

The following links provide information on building an Apache release and source code:

- For the latest support details, see *Oracle MetaLink* or Oracle Technology Network (OTN) at the following sites:

<http://metalink.oracle.com>.

If you do not find a solution for your problem, log a service request

<http://www.oracle.com/technology/deploy/security/index.html>

- *Apache source code*—<http://www.apache.org/dist/httpd>

- *Mod_SSL source code*—<http://www.modssl.org/source/>
- *OpenSSL source code*—<http://www.openssl.org/source/>
- *What is ApacheSSL*—http://www.apache-ssl.org/#What_is_Apache-SSL
- *Compiling and Installing Apache 1.3*—<http://httpd.apache.org/docs/install.html>
- *ApacheSSL build instructions for Win32*—<http://www.galatea.com/flashguides/apache-ssl-win32.xml>
- *How to build an Apache Unix release*—<http://httpd.apache.planetmirror.com/dev/how-to-release.html>
- *How to build a release of Apache for Windows*—<http://httpd.apache.planetmirror.com/dev/how-to-release-win32>

Platform-Specific Compilation Options

Some operating systems require additional options during configuration. Some options listed here may be redundant for some releases of Apache 1.3 but are necessary for other releases.

The following are environment settings for the operating system configuration command for your platform:

Solaris:

```
CFLAGS=-D_REENTRANT
LDFLAGS=-lpthreads
```

AIX:

```
CFLAGS=-D_REENTRANT
LDFLAGS=-lpthreads
```

HP-UX:

```
CFLAGS=-D_REENTRANT
LDFLAGS="-lc1 -lpthreads"
```

On HP-UX, you need to use PA-RISC1 compile options (the default). Do not use PA-RISC2 (64-bit) options. When using PA-RISC2, you will receive load errors such as "missing symbol," "bad magic number," or "share object is garbled." Similar errors may also appear under any operating system when loading Apache EAPI (mod_ssl) compiled modules into a plain Apache server.

Platform Specific Run-Time Settings for AIX

On AIX, you must set the environment variable AIXTHREAD_SCOPE to the value S (uppercase). Otherwise, there may be a segmentation fault when a worker process exits. This, however, does not affect the delivery of content, authentication, or authorization decisions by WebGate.

Also on AIX, you may wish to place the following directive in the httpd.conf file:

```
AcceptMutex fcntl
```

This directive is only supported in Apache 1.3.24 and later. It does not affect the delivery of content, authentication, or authorization decisions by WebGate. However, those familiar with the behavior of Apache on other platforms (through the `/server-status` URL) may prefer to use this setting.

Installation Order for Oracle Access Manager Web Components

Oracle Access Manager Web components (WebPass, Policy Manager, and WebGate), must be installed in a specific order. For example:

```
WebPass
Policy Manager
WebGate
```

For more information, see "[About the Installation Task](#)" on page 1-1.

Updating Web Server Configuration for Oracle Access Manager Web Components

During installation of Oracle Access Manager Web components (WebPass, Policy Manager, and WebGate), you can choose to update your Apache Web server configuration file (`httpd.conf`) manually or automatically. Oracle recommends that you update your Web server configuration file automatically. However, if you choose to update `httpd.conf` manually, instructions are provided.

When installing Oracle Access Manager Web components you are prompted for the location of the Web server configuration file. For Apache, enter the full path to `httpd.conf`. For example, the `httpd.conf` file may be found in the `Apache_install_dir/conf` directory.

If you later need to re-update the `httpd.conf` file (after Oracle Access Manager Web component installation and the initial Web server configuration update, refer to the `config.htm` file in `Component_install_dir/oblix/apps/common/docs/config.htm`, or use the `ManageHttpConf` program located in `Component_install_dir/oblix/tools/setup/InstallTools/ManageHttpConf`. Running `ManageHttpConf` without any options will print instructions on its use.

Tuning Apache 1.3 for Oracle Access Manager Web Components

Apache 1.3 uses a process model for serving multiple http requests at once. This is different from the single process (thread) model employed by other Web servers, which manage several requests simultaneously in one process. Each subordinate Apache worker-process responds to an incoming http request independently of every other worker-process.

Several parameters in the Apache server configuration file (`httpd.conf`) affect how an Apache server decides to create or destroy worker processes. The following affect the performance of the server:

- **MaxServers:** The number of simultaneous http requests that a system can handle depends on the maximum performance of the system.
- **Performance Tuning:** Performance tuning for a system should be done using an http load generating tool such as the `ab` program supplied with Apache.
- **MaxSpareServers:** Sets the desired maximum number of idle child server processes. An idle process is one which is not handling a request. If there are more

than `MaxSpareServers` idle, then the parent process will kill off the excess processes.

To preserve as much state as possible in the server, set the `MaxSpareServers` to a high value. Setting this value to the maximum of 255 keeps all Apache worker-processes available indefinitely, but it does not provide an opportunity for worker-process recycling during low-load periods.

- **MaxClients:** Sets the limit on the number of simultaneous requests that can be supported; more than this number of child server processes will not be created. Any connection attempts over the `MaxClients` limit will normally be queued, up to a number based on the `ListenBacklog` directive. Once a child process is freed at the end of a different request, the connection is then serviced.
- **MaxClientRequests:** Apache provides a safety mechanism to prevent a worker process from slowly acquiring too many system resources to be efficient. Setting `MaxClientRequests` to a value greater than zero limits the number of requests that a worker process can respond to, after which that process exits, to be replaced by a new, fresh worker process as soon as the need arises. This safety mechanism is not unreasonable, but the start-up delay for Oracle Access Manager Web components (also known as plug-ins) is noticeable at the Web browser.

If you use this parameter, set it high enough for end users to rarely notice the startup delay. Oracle Access Manager plug-ins are designed to run under Web servers without this safety mechanism.

- **MinSpareServers:** Sets the desired minimum number of idle child server processes. An idle process is one that is not handling a request. If there are fewer than `MinSpareServers` idle, then the parent process creates new children at a maximum rate of 1 every second. Use this with the Policy Manager.

Note: Setting this directive to some value m ensures that you will always have at least $n + m$ httpd processes running when you have n active client requests.

Because of the fact that Oracle Access Manager plug-in initialization is deferred until the first request, using a high value for the `MinSpareServers` parameter provides minimal advantage. However, it is useful to keep this parameter as high as possible. For dedicated Web server systems, this should pose no great burden.

- **StartServer:** As with the `MinSpareServers` parameter, the advantage of the `StartServers` parameter is limited by the delayed initialization of the Oracle Access Manager plug-ins.

Appropriate values for the preceding parameters depend on the expected load and the performance class of the systems involved, including the Access Server and LDAP server.

Apache servers on very high performance systems with high expected loads may be recompiled with a larger limit on the number of worker processes. These systems may see a greater performance impact on the `StartServers` and `MinSpareServers` parameters for dealing with sudden load spikes.

You may need to adjust operating system limits for the Access Server for proper operation. In particular, the maximum number of file descriptors available for any one Access Server may need to be increased beyond the default value. Configuring more than one connection between each Apache-based WebGate and an Access Server may quickly exceed this limit.

Policy Manager Tuning Factors

Policy Manager performance may be impacted by both Apache and Policy Manager configuration parameters. The following factors should be considered when tuning the Policy Manager for Apache:

- The idle child processes ensure that a new incoming request is serviced immediately. The more spare child processes, the faster the ramp up.
- Each child process opens separate connections to the directory server. The more child processes you have, the more directory server connections you have.

Assuming that each user is using one browser, there are four to five simultaneous requests to the Web server for images and js and HTML from the browser. Assuming that there are four simultaneous users, the total number of simultaneous requests to the Web server is $4 * 5 = 20$.

Given these factors, Oracle recommends the following to maintain a balance between how fast a new user is serviced and the number of connections to the directory server:

- `MaxClients = 25`
- `MinSpareServers = 4`
- `MaxSpareServers = 5`

Note: The Policy Manager does not open connections on Web Server startup. Instead, the Policy Manager creates connections on the first request.

To help compensate for any delay when the Policy Manager creates connections, the Policy Manager may be configured such that all directory server connections for all directory server profiles are set to 1. In this case, the Apache configuration may be as follows:

- `MinSpareServers = 1`
- `MaxSpareServers = 2`
- `MaxServers = 2`

In the preceding case, the Policy Manager responds in a reasonable time with some delay on the initial request.

Setting OHS Client Certificates

When using the `cert_decode` and `credential_mapping` plug-ins, you must ensure that the Access System Client Certificate authentication scheme works properly with SSL-enabled OHS by adding `+EarlierEnvVars` and `+ExportCertData` to the existing SSL options in the OHS Web server configuration file. For example:

credential_mapping:

```
obMappingBase="o=company,c=us",obMappingFilter=
" (&(objectclass=InetOrgPerson)(mail=%certSubject.E%)) "
```

ssl.conf must include:

```
SSLOptions +StdEnvVars +ExportCertData +EarlierEnvVars
```

To add ssl options

1. Locate and open the OHS Web server configuration file with a text editor. For example:

```
$ORACLE_HOME/Apache/Apache/conf/ssl.conf
```

2. In the ssl.conf file, add the following information to existing SSL options. For example:

```
SSLOptions +StdEnvVars +ExportCertData +EarlierEnvVars
```

3. Save the file and restart the Web server.

Tuning OHS for Oracle Access Manager Web Components

After installing the Oracle Access Manager Web component for OHS, you need to complete the following steps:

Note: As mentioned earlier, before installing Oracle Access Manager Web components for OHS, in the httpd.conf file change user and group to match the user that is installing the component.

To tune OHS for Oracle Access Manager Web components

1. Shut down opmn.
2. In the opmn.xml file, adjust items as follows:

```
<ias-component id="HTTP_Server">
<process-type id="HTTP_Server" module-id="OHS2">
  <environment>
    <variable id="LD_ASSUME_KERNEL" value="2.4.19"/>
  </environment>
  <module-data>
    <category id="start-parameters">
      <data id="start-mode" value="ssl-disabled"/>
    </category>
  </module-data>
  <process-set id="HTTP_Server" numprocs="1"/>
</process-type>
</ias-component>
```

3. In httpd.conf file on the Policy Manager, comment-out the following lines:

```
#LoadModule perl_module modules/mod_perl.so
#LoadModule php4_module modules/mod_php4.so
```

Starting and Stopping the Web Server

The following discussions provide information specific to running the Apache server on Unix and Windows:

- [Starting and Stopping OHS Web Servers](#)
- [Starting and Stopping Apache on Unix](#)
- [Starting and Stopping Apache on Windows](#)

Starting and Stopping OHS Web Servers

Starting and stopping an OHS Web server is the same procedure for both v1.3 and v2, on all platforms.

To start the OHS Web server

1. Locate and change to the following directory:

```
$ORACLE_HOME\opmn\bin\
```

2. From the command line, enter the following command:

```
opmnctl/startproc process-type=HTTP_Server
```

To stop the OHS Web server

1. Locate and change to the following directory:

```
$ORACLE_HOME\opmn\bin\
```

2. From the command line, enter the following command:

```
opmnctl/stopproc process-type=HTTP_Server
```

Starting and Stopping Apache on Unix

Typically, you perform a single step to start or stop the Apache Web server, as discussed in the following procedures:

Stopping Apache Web Server on Unix

To stop your Apache Web server on Unix:

1. Locate the *Apache_install_dir*/bin directory.
2. From the command line, use the `apachectl stop` command to stop the server:

```
./apachectl stop
```

Starting and stopping the Apache Web Server on Unix

To start and stop the Apache Web server on Unix:

1. Locate the *Apache_install_dir*/bin directory.
2. From the command line, use the `apachectl` command, as follows, to stop and restart the server:

```
./apachectl start
```

Starting the Server in SSL Mode

To start the server in SSL mode:

1. Locate the *Apache_install_dir*/bin directory.
2. From the command line, use the `apachectl startssl` command to start the server in SSL mode:

```
./apachectl startssl
```

Starting and Stopping Apache on Windows

The manner in which you start the apache Web server depends on the manner in which it is running: as a Windows service or as an application. See the procedures that follow.

To stop the Web server running as an application

1. in the Windows command line, hold down the Control key and then type the letter c.

To stop the Web server running as a Windows service

1. In the Windows Service window, locate the Web server service name.
2. Click the Stop icon.

To start the Web server running as a Windows service

1. Locate the *Apache_install_dir/bin* directory.
2. Enter the following command on the command line:

```
apache.exe -k start
```

To start the Web server running as an application

1. In the Windows Service window, locate the Web server service name.
2. Click the Start icon.

Configuring Apache v2, IHS, and OHS Web Servers for Oracle Access Manager

Oracle Access Manager provides Web components (Policy Manager, WebPass, and WebGate) for Apache v2 Web Servers, as well as for the IBM HTTP Server (IHS) powered by Apache v2 and Oracle HTTP Server (OHS).

This chapter provides details about configuring the three Web server types, and includes:

- [About OHS and Oracle Access Manager](#)
- [About Oracle Access Manager with Apache and IHS v2 Web Components](#)
- [About Apache v2 Architecture and Oracle Access Manager](#)
- [Compatibility and Platform Support](#)
- [Requirements for OHS/IHS/Apache v2 Web Servers](#)
- [Preparing Your Web Server](#)
- [Activating Reverse Proxy](#)
- [Installing Oracle Access Manager Web components](#)
- [Manually Updating a Web Server Configuration for Oracle Access Manager](#)
- [Verifying httpd.conf Updates for Oracle Access Manager Web Components](#)
- [Tuning OHS for Oracle Access Manager Web Components](#)
- [Starting and Stopping OHS Web Servers](#)
- [Tuning Apache/IHS v2 for Oracle Access Manager Web Components](#)
- [Tips and Troubleshooting](#)
- [Helpful Information](#)

About OHS and Oracle Access Manager

OHS2 is an Internet Web server extension that identifies Oracle Access Manager Web components that communicate with the Oracle HTTP Server (OHS) based on open source Apache v2. Oracle Access Manager 10g (10.1.4.0.1) provides WebPass, Policy Manager (formerly known as the Access Manager), and WebGate components that can be installed on a standalone Oracle HTTP Server on Linux and Windows platforms:

Linux: Oracle_Access_Manager10_1_4_0_1_linux_OHS2_WebPass

Linux: Oracle_Access_Manager10_1_4_0_1_linux_OHS2_Access_Manager

Linux: Oracle_Access_Manager10_1_4_0_1_linux_OHS2_WebGate

Windows: Oracle_Access_Manager10_1_4_0_1_Win32_OHS2_WebPass
Windows: Oracle_Access_Manager10_1_4_0_1_Win32_OHS2_Access_Manager
Windows: Oracle_Access_Manager10_1_4_0_1_Win32_OHS2_WebGate

You may use OHS Web components for any application. However, in some cases you must use OHS Web components. For example:

- OHS WebGate must be installed on the Oracle Application Server to enable integration with Oracle single sign-on as described in the *Oracle Access Manager Integration Guide*. During installation, the WebGate is installed as a module on OHS.
- OHS WebPass and OHS Access Manager (for the Policy Manager) components may be used with the Oracle Application Server. However, Apache WebPass and Apache Policy Manager are also supported for this application.

OHS2 Web components are based on Apache v2. See the *Oracle HTTP Server Administrator's Guide 10g R2 (10.1.2)* for complete details about OHS, including:

- Differences between the Oracle implementation and the open source Apache product on which it is based
- Oracle HTTP Server directory structure, configuration files and syntax, modules, and directives
- Managing server processes, network connections, and security features

Be sure to familiarize yourself with OHS Web component requirements, as described in "[Preparing Your Web Server](#)" on page 17-8.

About Oracle Access Manager with Apache and IHS v2 Web Components

Oracle Access Manager also provides components for Apache v2 Web servers and the IBM HTTP Server in addition to the Oracle HTTP Server. The IBM HTTP Server (IHS) is a variation of Apache v2. Unless otherwise stated, the following information applies to all three:

- 10g (10.1.4.0.1) WebPass, Policy Manager, and WebGate for Apache v2.0.5.2
- 10g (10.1.4.0.1) WebGate for Apache v2.0.48, including reverse proxy if you choose to activate this capability.
- 10g (10.1.4.0.1) WebGate for the IBM HTTP Server (IHS) powered by Apache v2.0.47, including reverse proxy if you choose to activate this capability.

Note: For the latest support information, see details under the Certify tab on:

Each platform-specific installation package supports both plain and SSL-capable Apache modes. For example:

AIX: Oracle_Access_Manager10_1_4_0_1_power-aix_IHS2_WebGate
Linux: Oracle_Access_Manager10_1_4_0_1_linux_Apache2_Access_Manager
Linux: Oracle_Access_Manager10_1_4_0_1_linux_Apache2_WebGate
Linux: Oracle_Access_Manager10_1_4_0_1_linux_Apache2_WebPass
Solaris: Oracle_Access_Manager10_1_4_0_1_sparc-s2_Apache2_WebGate

Windows: Oracle_Access_Manager10_1_4_0_1_Win32_APACHE2_WebGate

Earlier Oracle Access Manager releases included separate platform-specific installation packages for plain versus SSL-capable modes. For example, two WebGate files were provided for each platform: the APACHE_WebGate, and the APACHESSL_WebGate.

There have been no functional changes to Oracle Access Manager components to support these Web servers. Oracle Access Manager authentication occurs through the WebGate using HTTP basic, form, or SSL client certificates. Authorization for Web resources by authenticated users, and simple and multi-domain SSO with other Web servers or applications, also occurs through the WebGate.

Important: Information in this chapter focuses on WebGate. However, it applies to Policy Manager and WebPass components equally.

About the Apache HTTP Server

The Apache HTTP Server is an open-source HTTP Web server project of the Apache Software Foundation. The project goal is to provide a secure, efficient and extensible server and HTTP services that meet current HTTP standards.

For more information, see "[About Apache v2 Architecture and Oracle Access Manager](#)" on page 17-4.

About the IBM HTTP Server

The IBM HTTP Server (IHS) is a variation of Apache v2. Portions of the IBM HTTP Server are based on software developed by The Apache Group. The IBM HTTP Server component also includes software developed by the OpenSSL Project and software developed by Eric Young.

Details about the Apache architecture and Oracle Access Manager, discussed in "[About Apache v2 Architecture and Oracle Access Manager](#)" on page 1-3 apply to IHS with the following exceptions:

- Previous versions of IHS required a separate IDS Client to use the mod_ibm_ldap module. With IHS powered by Apache v2.0.47, this is not a requirement.
- IHS v2.0.47 supports FIPS 140-2. FIPS support is disabled by default. To enable FIPS support, just add the SSLFIPSEnable directive to the httpd.conf file. Similarly, use SSLFIPSDisable directive to disable FIPS support.
- On AIX, ensure that the appropriate runtime library is installed before you install IHS v2.0.47.

For example on AIX 5.1, the xIC.rte 6.0 runtime library (for example: xIC.rte.6.0.0.0) must be installed before you install IHS v2.0.47. This library is required on AIX to install and use SSL with IHS v2. You can download this library from the following Web site:

<http://www-912.ibm.com/eserver/support/fixes/fcgui.jsp>

About the Apache and IBM HTTP Reverse Proxy Server

Typically, a reverse proxy is used in the following situations:

- To provide Internet users with access to a server behind a firewall

- To balance the load among several back-end servers, or to provide caching for a slower back-end server
- To bring several servers into the same URL space

The proxy_module implements a proxy/gateway for Apache and IHS powered by Apache. The client requires no special configuration; a reverse proxy appears like an ordinary Web server. The client makes requests as usual for content in the name-space of the reverse proxy. It is the reverse proxy that decides where those requests are sent. Content is returned as if the reverse proxy was the origin.

Important: The proxy_module can be used to implement a proxy capability for FTP, CONNECT (for SSL), HTTP/0.9, HTTP/1.0, and HTTP/1.1. However, only the reverse proxy capability is supported with the WebGate.

For more information, see "[Requirements for Apache v2 Web Servers](#)" on page 17-7.

About Apache v2 Architecture and Oracle Access Manager

The Apache v2 Web server provides a hybrid multi-threaded, multi-process architecture that is compatible with the thread-safe Oracle Access Manager libraries.

Important: Unless explicitly stated otherwise, all details in this discussion apply equally to Apache v2 and IHS v2 Web Servers and to Policy Manager, WebPass, and WebGate.

In addition to the standard set of modules, the Apache v2 Web server includes Multi-Process Modules (MPMs) to bind network ports on the machine and to accept and process requests. The appropriate MPM must be compiled into the server and activated before you install a Oracle Access Manager component for Apache or IHS v2:

- **On Windows:** mpm_winnt is the default MPM on Windows platforms. mpm_winnt can use native networking features rather than the POSIX layer used in Apache 1.3.
- **On Unix:** The prefork MPM is the default MPM for Apache v2 Web servers on Unix platforms. The prefork MPM implements a non-threaded, pre-forking Web server that handles requests in a manner similar to Apache v1.3.

Note: If you compile Apache on Unix with the mpm_worker_module, you need to modify the thread.c file from the Apache source for the Unix environment as described in See "[Tips and Troubleshooting](#)" on page 17-34.

- **On AIX:** The worker MPM is the default MPM for IHS v2 on the AIX platform. The worker MPM implements a hybrid multi-process, multi-threaded server. The most important directives used to control this MPM are ThreadsPerChild and MaxClients. For details, see "[Tuning Apache/IHS v2 for Oracle Access Manager Web Components](#)" on page 17-31.

The Apache v2 Web server includes an Apache Portable Runtime (APR) library that provides an interface to platform-specific implementations, assures API developers predictable if not identical behavior regardless of platform, and eliminates the need for

conditional compilation #ifdefs. Although backward compatibility is supported with the include/apu_compat.h file, using the Apache v2 APR is recommended.

For more information, see your Apache v2 documentation. See also, "[Tuning Apache/IHS v2 for Oracle Access Manager Web Components](#)" on page 17-31.

The Apache architecture affects Oracle Access Manager components in different ways, as discussed in the following sections. For additional information, see "[Compatibility and Platform Support](#)" on page 17-6.

For a WebPass installed with Apache v1.3 and v2

- Each WebPass instance connects to the Identity Server.
- Each connection takes up system sources, and each connection has n file descriptors.
- The tuning parameters for Apache must be set so that Apache does not need to start or stop processes too frequently. These settings would be similar with or without Oracle Access Manager.

For a Policy Manager installed with Apache v1.3 and v2

- Each Web server process is an instance of the Policy Manager application.
- Each application maintains its own connections to the directory server. This may not directly affect performance. However, there may be a limit on the directory server side that you may want to consider when other directory server clients are involved.
- Multiple processes respond to a user's request (multiple HTTP events are triggered to build the frames).
- Latency of responses cannot be predicted.
- Fewer processes are better from a UI perspective, but this affects the number of concurrent users.

For WebGates installed with IHS and Apache v1.3 and v2

- There is no shared cache between processes.
- Each process maintains its own connections to the Access Server. Therefore, you should limit the number of WebGate connections. This issue is partially affected by the performance of the systems running the Web servers and Access Servers.

Note: WebGates for Apache v2 (and derivatives) can be used in installations that contain other WebGates as well as WebPass and Policy Manager components for other Web servers.

Limitations of Apache and IHS v2 Web Servers

Due to limitations of the Apache v2 Web server, plug-ins configured for the Oracle Access Manager form-based authentication scheme do not pass variables when:

- The optional challenge parameter, passthrough:Yes, is included in the authentication scheme to pass login credentials through to a post-processing program.
- The form action is a CGI script that dumps all headers and variables passed to it and the method is called using the HTTP POST method.

For example:

```
<html>  
<form name="myloginform" action="/access/...cgi" method="post">
```

Compatibility and Platform Support

For the latest support information, see details under the Certify tab on:

Oracle Access Manager Web components for Apache v2, OHS2, and IHS v2 may be the only WebGates in your installation or may coexist with other WebGates. For more information, see "[Access System Guidelines](#)" on page 2-6.

Requirements for OHS/IHS/Apache v2 Web Servers

Oracle Access Manager 10g (10.1.4.0.1) HTML pages use UTF-8 encoding. Apache-based Web servers, including Apache, Oracle HTTP Server (OHS), and IBM HTTP Server (IHS) allow administrators to specify a default character set for all HTML pages sent out using the `AddDefaultCharset` directive. This directive overrides any character specified by the application generating the HTML pages. If the `AddDefaultCharset` directive enables a character set other than UTF-8, Oracle Access Manager HTML pages are garbled.

Oracle recommends that you specify the `AddDefaultCharset` directive in the Web server configuration file (`httpd.conf`) as follows to ensure the correct display of Oracle Access Manager 10g (10.1.4.0.1) HTML pages:

```
AddDefaultCharset Off
```

See your Web server documentation for more information about this directive.

The following topics provide additional details you should be aware of:

- [Preparing OHS Web Servers on Linux](#)
- [Preparing OHS Web Servers on Linux and Windows Platforms](#)
- [Setting OHS Client Certificates](#)
- [Requirements for IHS2 Web Servers](#)
- [Requirements for Apache and IHS v2 Reverse Proxy Servers](#)
- [Requirements for Apache v2 Web Servers](#)

Important: Unless explicitly stated otherwise, information here applies to WebPass, Policy Manager, and WebGate components equally.

Requirements for IHS2 Web Servers

This discussion identifies specific requirements for IHS v2 with Oracle Access Manager. With IHS v2, you do not compile any source code to get the binaries. However, the following requirements do apply to IHS v2 Web servers:

- For an SSL capable configuration on AIX, the `xLC.rte.6.0` runtime library is required.
- For an SSL capable configuration, the `GSKit7` is required and can be downloaded from <https://techsupport.services.ibm.com/server/aix.fdc>.

Requirements for Apache and IHS v2 Reverse Proxy Servers

As discussed earlier, the proxy_module implements a proxy/gateway. The client requires no special configuration. Although the proxy_module can be used to implement a proxy capability for FTP, CONNECT (for SSL), HTTP/0.9, HTTP/1.0, and HTTP/1.1, only the reverse proxy capability is supported with certain Oracle Access Manager Apache and IHS v2 WebGates. For details, see "[Compatibility and Platform Support](#)" on page 17-6.

For Apache Web Servers: To use reverse proxy functions with Oracle Access Manager, you need to include the proxy module in the configure command. For example:

```
--enable-proxy: Apache proxy module
--enable-proxy-connect: Apache proxy CONNECT module
--enable-proxy-ftp: Apache proxy FTP module
--enable-proxy-http: Apache proxy HTTP module
```

You also need to load mod_proxy and the mod_proxy_http module into the server dynamically. A reverse proxy is activated using the ProxyPass directive or the [P] flag to the RewriteRule directive.

For IHS Web Servers: After installing the IHS Web server, reverse proxy configurations must be completed in the httpd.conf file in the following directory:

IHS_install_dir/conf directory

For more information, see "[Activating Reverse Proxy](#)" on page 17-20.

Requirements for Apache v2 Web Servers

This discussion identifies specific requirements for Apache v2 with Oracle Access Manager. Additional information can be found in your Apache documentation:

PATH Variable: On Unix systems, your PATH variable must contain the gcc location before you compile Apache v2. However, the Sun C compiler location must not be in your PATH variable. On Windows systems, Apache can be built using either command-line tools or the Visual Studio IDE Workbench. The command-line build requires that the environment reflect the PATH, INCLUDE, LIB and other variables that can be configured with the vcvars32 batch file.

Multi-Process Module (MPM): With Apache v2, a default MPM is provided for each platform to bind network ports on the machine and to accept and process requests. Apache must have one, and only one, MPM in use at any time. If no MPM is selected during compilation, the default will be loaded into the Web server. You may activate the MPM during compilation.

mod_ssl: Oracle Access Manager supports Apache with or without SSL-capable communication. The base Apache Web server does not use SSL for browser connections and will not respond to HTTPS requests. For SSL-capable communication, Oracle Access Manager supports Apache with mod_ssl only. No SSL-specific Oracle Access Manager features operate with Apache-SSL.

mod_ssl relies on OpenSSL to provide the cryptography engine; mod_ssl provides an interface to the OpenSSL library. The OpenSSL library provides Strong Encryption using the Secure Sockets Layer and Transport Layer Security protocols.

With previous versions of Apache, the mod_ssl module had to be downloaded separately and compiled into the server. With Apache HTTP Server v2 module, mod_ssl comes as a loadable module that you can enable during configuration.

Multi-threading: Multi-threading is required for WebPass installations with Apache v1.3.27 or later. WebGates for Apache v2 can be used in Oracle Access Manager installations that contain WebPass, Policy Manager, and WebGates for Apache 1.3.27 or later Web servers.

Dynamic Shared Object (DSO): DSO support is required for all Oracle Access Manager plug-ins (WebGate and WebPass). Apache modules that extend basic core server functionality may be either statically compiled for permanent inclusion in the Apache binary, or dynamically compiled and stored separately to load at runtime without recompiling. With Apache v1.3, mod_so had to be compiled. With Apache v2 on Windows systems, mod_so is a Base module and always included. With Apache v2 on Unix, the loaded code typically comes from shared object files.

Note: Dynamically loaded Apache 1.3 modules cannot be used directly with Apache v2. Apache v1.3 modules must be modified to load dynamically or compile into Apache v2.

mod_perl: mod_perl embeds the Perl programming language in the Apache Web server. Without Perl, Apache v2 can still be built and installed; however, some support scripts written in Perl cannot be used.

Note: With Apache v.1.3.2x, some operating systems required additional options during configuration. However, to build Apache v2, there is no need to set any additional variables.

Preparing Your Web Server

The methods and steps to prepare your host machine for the Oracle Access Manager Web component installation depends upon the specific Web server and platform, as discussed in the following task overview.

To use reverse proxy functions with Oracle Access Manager, you need to include the proxy module in the configure command, as discussed in "[About the Apache and IBM HTTP Reverse Proxy Server](#)" on page 17-3. See also "[Activating Reverse Proxy](#)" on page 17-20.

Task overview: Preparing your Web server and installing Oracle Access Manager

1. Install the IHS v2 Web server or compile and install the Apache v2 Web server as discussed in:
 - [Preparing the IHS v2 Web Server](#)
 - [Preparing Apache and OHS Web Servers on Linux](#)
 - [Preparing OHS Web Servers on Linux](#)
 - [Preparing OHS Web Servers on Linux and Windows Platforms](#)
 - [Setting OHS Client Certificates](#)
 - [Preparing the Apache v2 Web Server on Unix](#)
 - [Preparing the Apache v2 SSL Web Server on AIX](#)
 - [Preparing the Apache v2 Web Server on Windows](#)

2. Activate reverse proxy capability if desired, as described in "[Activating Reverse Proxy](#)" on page 17-20.
3. Install Oracle Access Manager components, as described elsewhere in this guide.
4. Finish Web server configuration, as described in "[Verifying httpd.conf Updates for Oracle Access Manager Web Components](#)" on page 17-25.
5. Refer to "[Tuning Apache/IHS v2 for Oracle Access Manager Web Components](#)" on page 17-31.

Note: In all the procedures that follow, path name variables, modules, and options are examples provided only to illustrate the steps. Your environment will vary. Refer to your Web server documentation for additional details.

Preparing the IHS v2 Web Server

To prepare your IHS v2 Web server to accept and use the WebGate for IHS v2, you need to complete one or more of the following procedures, depending on your environment and requirements:

- [Preparing the Host for IHS v2 Installation](#)
- [Installing the IBM HTTP Server v2](#)
- [Setting Up SSL-Capability](#)
- [Starting a Secure Virtual Host](#)
- [Activating Reverse Proxy](#)

When you have completed the appropriate procedures, you are ready to install the WebGate for IHS v2.

Preparing the Host for IHS v2 Installation

You need to complete this procedure to set up the host machine before you install the IHS Web server. For additional information, see "[Requirements for IHS2 Web Servers](#)" on page 17-6 and "[Requirements for Apache v2 Web Servers](#)" on page 17-7.

This example illustrates installation on AIX 5.1. Your environment may vary.

To prepare for IHS v2 installation

1. On the host machine, download and install the IBM Developer Kit, Java Technology Edition version 1.4 from the following site:

<http://www.ibm.com/java/jdk>

The IBM Developer Kit ships with the WebSphere Application Server or can be downloaded from this site.

2. On the host machine, download and install the xIC.rte 6.0 runtime for AIX 5.1, which is required by the GSKit7 runtime executable from the following site:

<https://techsupport.services.ibm.com/server/aix.fdc>

3. On the host machine, create a new directory in which you will uncompress the IBM HTTP Server install image.
4. On the host machine, download the IBM HTTP Server install image from the following Web site:

<http://www-306.ibm.com/software/webservers/httpservers/>

5. On the host machine, uncompress the install image in your new directory.

For example:

```
tar -xf IHS.tar
```

A listing of the following files appears, based on your operating system:

```
gskit.sh
setup.jar
gskta.rte (a GSKit runtime executable for AIX)
```

You are ready to begin the installation, as described next.

6. Proceed to "[Installing the IBM HTTP Server v2](#)" on page 17-10.

Installing the IBM HTTP Server v2

The procedure that follows walks you through a typical IBM HTTP Web server installation. Alternatively, you may choose to perform a silent installation. In this case, you use silent.res file with the `java -jar setup.jar -silent -options silent.res` command. You can customize silent install options by editing the silent.res text file. All options are set to true by default. To disable an option, set its value to false.

To install the IBM HTTP Web server powered by Apache v2

1. Set your path to point to the Java Technology Edition version 1.4 installed on your machine in the previous example. For example:

```
export PATH=$PATH:/usr/java14/java/bin
```

2. From to the directory where you uncompress the install image, type the following command:

```
java -jar setup.jar
```

3. Choose the language in which to run the installation.

The Welcome to the InstallShield Wizard for the IBM HTTP Server appears.

4. Click Next to dismiss the Welcome screen.
5. Specify the directory name. For example:

```
AIX: /usr/IBMIHS/
```

6. Click Next to continue.

Options appear for a typical, custom, or developer installation. When you choose a typical installation, a list will appear with everything included and the size of the image. If you choose a custom installation, a list of components appears and you can clear the box next to the any components you do not want to install.

7. Select the type of installation you would like to perform, then click Next. For example:

```
Typical
```

The following message appears. You can click Cancel to stop the installation.

```
Installing IBM HTTP Server. Please wait.
```

The next message also appears. You can click Cancel to stop the inventory update.

Updating the inventory.

8. Click Finish to complete your installation.
9. Stop then start the IHS server using the apachectl commands, as follows:

For example:

```
IHS2_install_dir/bin
./apachectl stop
./apachectl start
```

where *IHS2_install_dir* is the directory where you installed the IHS v2 Web server.

You may configure the IHS v2 Web server in the following modes either before or after installing the WebGate for IHS v2:

- [Setting Up SSL-Capability](#)
- [Starting a Secure Virtual Host](#)
- [Activating Reverse Proxy](#)
- [Installing Oracle Access Manager Web components](#)

Setting Up SSL-Capability

If you need to setup SSL-capability, use the following procedure either before or after installing the WebGate for IHS v2.

To setup SSL for IHS v2 using the default configuration file

1. Locate and open the following file:

```
IHS2_install_dir/conf/httpd.conf
```

2. Specify the SSLEnable directive to enable SSL.
3. Specify a Keyfile directive and any SSL directives you want to enable.
4. Stop then start the IHS server, as follows. For example:

```
IHS2_install_dir/bin
./apachectl stop
./apachectl start
```

where *IHS2_install_dir* is the directory where you installed the IHS v2 Web server.

5. Continue with the following procedures:
 - [Starting a Secure Virtual Host](#)
 - [Activating Reverse Proxy](#)
 - [Installing Oracle Access Manager Web components](#)

Starting a Secure Virtual Host

If you need to start a secure virtual host, use the following procedure either before or after installing the WebGate for IHS v2.

To start an IHS v2 secure virtual host

1. Locate and open the following file:

```
IHS2_install_dir/conf/httpd.conf
```

where `IHS2_install_dir` is the directory where you installed the IHS v2 Web server.

2. Specify the `SSLEnable` directive in the virtual host stanza of the configuration file, to enable SSL for a virtual host.

You can specify any directive, with the exception of the cache directives, inside a virtual host.

3. Specify a `Keyfile` directive and any SSL directives you want to enable for that particular virtual host.
4. Load the `mod_ibm_ssl.so` using the `LoadModule` directive in the conf file.
5. Stop then start the IHS virtual host, as follows. For example:

```
IHS2_install_dir/bin
./apachectl stop
./apachectl start
```

Note: The start and stop instructions for an SSL implementation are the same as non-SSL-capable implementations.

6. Continue with the following procedures:
 - [Activating Reverse Proxy](#)
 - [Installing Oracle Access Manager Web components](#)

Preparing Apache and OHS Web Servers on Linux

When installing Oracle Access Manager Web components for Apache or OHS on Linux, you are prompted to install as the same user under which the Web server is running. See the `User` and `Group` directive entries in the `httpd.conf` file.

When installing Oracle Access Manager Web components for vendor-bundled Apache v2 on Red Hat Enterprise Linux 4, ensure that all Oracle Access Manager Web components are installed for Web server user & group (default: `apache`). See also "[Tuning Apache/IHS v2 for Oracle Access Manager Web Components](#)" on page 17-31.

Preparing OHS Web Servers on Linux

When using Oracle Access Manager 10g (10.1.4.0.1) Web components for OHS, be sure to take these caveats into account for Linux:

- The `LD_ASSUME_KERNEL=2.4.19` environment variable must be set for Linux at runtime because the older Linux threading model is supported (*not* the native posix thread library (NPTL)).
- Policy Manager communication to the directory server in SSL mode on Linux is *not* supported.

Preparing OHS Web Servers on Linux and Windows Platforms

When using Oracle Access Manager 10g (10.1.4.0.1) Web components for OHS on Windows and Linux platforms, both the Perl module and the PHP module must be commented out in the `httpd.conf` for both Windows and Linux platforms.

Note: When installing Oracle Access Manager Web components OHS on Linux, you are prompted to install as the same user under which the Web server is running. See the User and Group directive entries in the httpd.conf file.

Setting OHS Client Certificates

When using the cert_decode and credential_mapping plug-ins, you must ensure that the Access System Client Certificate authentication scheme works properly with SSL-enabled OHS by adding +EarlierEnvVars and +ExportCertData to the existing SSL options in the OHS Web server configuration file. For example:

credential_mapping:

```
obMappingBase="o=company,c=us",obMappingFilter=
"(&(objectclass=InetOrgPerson)(mail=%certSubject.E%))"
```

ssl.conf must include:

```
SSLOptions +StdEnvVars +ExportCertData +EarlierEnvVars
```

To add ssl options

1. Locate and open the OHS Web server configuration file with a text editor. For example:

```
$ORACLE_HOME/ohs/conf/ssl.conf
```

2. In the ssl.conf file, add the following information to existing SSL options. For example:

```
SSLOptions +StdEnvVars +ExportCertData +EarlierEnvVars
```

3. Save the file and restart the Web server.

Preparing the Apache v2 Web Server on Unix

This discussion provides an overview and steps to prepare the Apache v2 HTTP Web server for Oracle Access Manager on Unix platforms, including Solaris, Unix, Linux, and AIX. See also ["Preparing the Apache v2 SSL Web Server on AIX"](#) on page 17-17

Apache v2 can be configured, built, and installed plain or as SSL-capable. After downloading and extracting Apache source files, you use a script (configure script on Unix and the makefile.win make script for Windows) to compile the source tree for your environment.

Note: Basic requirements are the same regardless of your platform. However, the remainder of this discussion and the procedures that follow focus on Unix platforms. For more information, see also ["Preparing the Apache v2 SSL Web Server on AIX"](#) on page 17-17.

When you configure Apache v2 on Unix platforms, you specify the installation directory path name using the `-prefix=` option with the `./configure` command. During configuration you enable the modules that are appropriate for your environment. For example, `mod_so` is included in the server automatically when dynamic modules are included in the compilation. However, you can ensure the server is capable of loading DSOs by including the `-enable-so` option with the configure

command. If you have multiple Perl interpreters installed, you can include the `-with-perl` option to ensure the correct interpreter is selected during configuration.

In the `configure` command, you can also include the options to enable `mod_ssl`, and to activate an MPM. After configuration, you can verify which MPM was chosen using `./httpd -l` to list every module that is compiled into the server.

When you finish configuring Apache, you build the various parts that form the Apache package using the `make` command then install the package under the installation directory you specified with the `-prefix=` option during configuration.

For steps and examples, see the following procedures and your Apache documentation:

- [To prepare plain Apache v2 for Unix](#)
- [To prepare SSL-capable Apache v2 on Unix](#)
- [To prepare Apache v2 for Windows](#)
- [Activating Reverse Proxy](#)

In the procedures that follow, path name variables, modules, and options are examples provided only to illustrate the steps. Your environment will vary. Refer to your Web server documentation for additional details. There is no difference in the build procedure between Apache v2.0.48 and v2.0.52.

To prepare plain Apache v2 for Unix

1. Confirm that your environment meets Apache requirements for the appropriate compiler and build tools, as described in Apache documentation located at:

<http://httpd.apache.org/docs-2.0/install.html#requirements>

Note: There are no known restrictions with regard to supported compiler versions for Apache v2 and Oracle Access Manager plug-ins. See the Apache documentation.

2. Download a complete, unmodified version of the Apache HTTP Server v2, as described in the Apache documentation. For example:

<http://httpd.apache.org/download.cgi>

Note: Be sure to download Perl, if needed.

3. Extract (uncompress, then `untar`) source files from the tarball, as described in the Apache documentation. For example:

```
gzip -d httpd-2_0_48.tar.gz
tar -xvf httpd-2_0_48.tar
```

You can use the following step as an example of configuring the Apache source tree. If you compile Apache on Unix with the `mpm_worker_module`, see "[Tips and Troubleshooting](#)" on page 17-34.

Note: To use reverse proxy functions with Oracle Access Manager, you need to include the proxy module in the configure command, as discussed in "[About the Apache and IBM HTTP Reverse Proxy Server](#)" on page 17-3.

4. Configure the Apache source tree and enable or activate the desired modules using details in the Apache documentation. For example:

```
cd apache_source_dir
./configure --prefix=apache_install_dir --enable-so \ --with-mpm='prefork'
--with-perl=perl_interpreter_path \ --with-port=non_ssl_port_number
```

where *apache_source_dir* refers to the directory where you extracted Apache and *apache_install_dir* refers to the directory where you want to install Apache.

5. Compile the Apache package you configured using the make command. For example:

```
make
```

6. Install the Apache package in the configured directory path that you specified earlier using the --prefix= option. For example:

```
make install
```

7. Customize the installation using instructions in the Apache documentation.

For example, you may need to tune the httpd.conf to set basic values for:

```
ServerName
User/owner of the WebServer
Group
```

Note: To view the complete list of values, use the command:
./configure --help.

8. Stop then restart the Apache Web server to test the installation using commands in the *apache_install_dir/bin* directory. For example:

```
./apachectl stop
./apachectl start
```

9. Continue with appropriate tasks for your environment, as follows:

- [To prepare SSL-capable Apache v2 on Unix](#)
- [Preparing the Apache v2 Web Server on Unix](#)
- [Activating Reverse Proxy](#)
- [Installing Oracle Access Manager Web components](#)

The following procedure outlines how to prepare an SSL-capable Apache v2 Web server on Unix. The Apache mod_ssl is loadable; however, this installation requires the Open Source toolkit for SSL/TLS. Again, be sure to download Perl, if needed. If AIX is the platform you are using, be sure to see "[Preparing the Apache v2 SSL Web Server on AIX](#)" on page 17-17 for additional information.

To prepare SSL-capable Apache v2 on Unix

1. Confirm that your environment meets Apache requirements for the appropriate compiler and build tools, as described in Apache documentation located at:

<http://httpd.apache.org/docs-2.0/install.html>

2. Download a complete, unmodified version of the Apache HTTP Server v2 and Open Source, as described in the Apache documentation.

<http://httpd.apache.org/download.cgi>

<http://www.openssl.org/>

3. Extract (uncompress, then untar) source files from the tarballs, as described in the Apache documentation. For example:

```
gzip -d httpd-2_0_48.tar.gz
tar -xvf httpd-2_0_48.tar
gzip -d openssl-0_9_6f.tar.gz
tar -xvf openssl-0_9_6f.tar
```

4. Configure the OpenSSL source tree, as described in Apache documentation. For example:

```
cd openssl_source_dir
./configure -fPIC --prefix=openssl_install_dir
```

where *openssl_source_dir* refers to the directory where you extracted OpenSSL and *openssl_install_dir* refers to the directory where you want to install the configured OpenSSL package.

5. Compile the OpenSSL package in the installation directory you configured using the make command with the --prefix= option. For example:

```
make
```

6. Issue the make test command to complete any sanity testing of OpenSSL and check the correct version of the tools required. For example:

```
make test
```

7. Install the OpenSSL package in the configured directory path that you specified earlier using the --prefix= option. For example:

```
make install
```

8. Configure the Apache source tree and enable or activate desired modules, as described in your Apache documentation. For example:

```
cd apache_source_dir ./configure --prefix=apache_install_dir
--enable-so \ --with-mpm='prefork' --with-perl=perl_interpreter_path \
--with-port=non_ssl_port --enable-ssl \ --with-ssl=openssl_install_dir
```

where *apache_source_dir* refers to the directory where you extracted Apache; *apache_install_dir* refers to the directory where you want to install Apache; and *openssl_install_dir* refers to the directory where you installed the configured OpenSSL package.

9. Compile using the make command to build the Apache SSL-capable package in the installation directory you configured using the --prefix= option. For example:

```
make install
```

10. Install the Apache SSL-capable package in the configured directory path that you specified earlier using the `--prefix=` option. For example:

```
make install
```

You must explicitly make certificates for the Apache v2 server to enable SSL using the `openssl` tool located at `openssl_install_dir/bin/`. The `make certificate` command does not work with Apache v2.

11. Make certificates using the OpenSSL tool in the `openssl_install_dir/bin` directory, as described in your OpenSSL documentation and remember that "Common Name" is the fully qualified host name.
12. Customize the installation using instructions in the Apache documentation:

- Tune the `httpd.conf` to set basic values for:

```
ServerName
User/owner of the WebServer
`Group
```

- Tune the `ssl.conf` to set basic values for:

```
Listen 7000
<VirtualHost _default_:7000>
ServerName ps0733.persistent.co.in:7000
SSLCertificateFile /home/qa/software/ws/apache/
apache-2.0.48_ssl_7000/conf/ssl.crt/server.crt
SSLCertificateKeyFile /home/qa/software/ws/apache?
apache-2.0.48_ssl_7000/conf/ssl.key/server.key
```

13. Stop then restart the Apache Web server to test the installation using commands in the `apache_install_dir/bin` directory. For example:

```
./apachectl stop
./apachectl startssl
```

14. Continue with the appropriate procedures as follows:

- [Activating Reverse Proxy](#)
- [Installing Oracle Access Manager Web components](#)

Preparing the Apache v2 SSL Web Server on AIX

While building the Apache v2 SSL Web server, the symbols from the OpenSSL Library `libssl.a` are exported into the `httpd` executable in Apache. The symbols needed by Oracle Access Manager from the OpenSSL library are:

- `SSL_get_peer_certificate()`
- `i2d_X509()`

During linking and binding on the AIX platform, any unused or unreferenced symbols are deleted. Therefore, the two symbols required by Oracle Access Manager are missing from the `httpd` executable.

You need to use `openssl-0.9.7d` to compile on AIX (`openssl-0.9.7e` does not compile on AIX). The rest of the steps are the same as on UNIX `openssl-0.9.7d`.

Client Cert Authentication: If you are using Client Cert Authentication on the AIX platform, be sure to use AIX 5.2 Maintenance Level 4 with the following hot fix applied for dlsym problem on AIX:

<http://www-1.ibm.com/support/docview.wss?uid=isg1IY63366>

To prepare the AIX platform for Apache v2

1. Ensure that your AIX platform meets the system requirements for Oracle Access Manager, as described in "Preparing to Install Oracle Access Manager" on page 39.
2. See details in "Preparing the Apache v2 Web Server on Unix" on page 17-13 and when building the Apache v2 Web server:
 - Use openssl-0.9.7d to compile the Web server for AIX.
 - Use the make command in the following manner:

```
make MFLAGS=EXTRA_LDFLAGS=' -Wl, -bE:OpenSSL_Symbols.exp '
```

where OpenSSL_Symbols.exp is the file containing the two required symbols. The symbol must be exported using the export file only, as shown.

Note: Do not export the symbol on AIX with the following methods:
 -bnog : To suppress garbage collection of symbols
 -bexpal: To export all symbols
 -uSymbolName: To export a particular symbol

Preparing the Apache v2 Web Server on Windows

Following are some details about how installing and configuring Apache v2 on Windows differs from Apache v2 on Unix. For more information, see your Apache documentation.

During Installation: Apache will configure files in the \conf subdirectory to reflect the chosen installation directory. If any configuration files in this directory already exist, a new copy of the corresponding file will be written with the extension .ORIG. For example, \conf\httpd.conf.ORIG.

After Installation: Apache is configured using the files in the \conf subdirectory. These are the same files used to configure the Unix version. However, there are a few differences.

You must edit the configuration files in the \conf subdirectory to customize Apache for your environment. These files will be configured during the installation; Apache is ready to run from the installation directory, with the documents server from the subdirectory htdocs. There are many options you should set before starting to use Apache. For example, Apache listens on port 80 unless you change the Listen directive in the configuration files or install Apache only for the current user.

Multi-Threading: Apache for Windows is multi-threaded, which means that it does not use a separate process for each request as Apache does on Unix. Instead there are usually only two Apache processes running: a parent process, and a child which handles the requests. Within the child process each request is handled by a separate thread.

Unix-Style Names: Apache uses Unix-style names internally. The directives that accept filenames as arguments must use Windows filenames instead of Unix filenames. However, you must use forward slashes, not back slashes. Drive letters may be used. However, if a drive letter is omitted, the drive with the Apache executable is assumed.

LoadModule Directive: Apache for Windows includes the ability to load modules at runtime without recompiling the server. If Apache is compiled normally, it will install a number of optional modules in the \Apache2\modules directory. To activate these or other modules, you must use the LoadModule directive. For example, to activate the status module, use the following (in addition to the status-activating directives in access.conf):

```
LoadModule status_module modules/mod_status.so
```

On Unix, the loaded code typically comes from shared object files (.so extension), on Windows this may be either the .so or .dll extension.

Process Management Directives: These directives are also different for Apache on Windows.

Error Logging: During Apache startup, any errors are logged into the Windows event log, which provides a backup to the error.log file. For more information, see your Apache documentation.

Apache Service Monitor: Apache comes with an Apache Service Monitor utility. With it you can see and manage the state of all installed Apache services on any machine on your network. To manage an Apache service with the monitor, you must first install the service. Apache may be run as a service on Windows. For details, see your Apache documentation.

Starting, Restarting, Shutting Down: Running Apache as a service is the recommended method. An Apache service is typically started, restarted, and shut down using the Apache Service Monitor and commands like NET START Apache2 and NET STOP Apache2. You may also use standard Windows service management.

You may work with Apache from the command line using the apache command. Apache will execute and remain running until it is stopped by pressing Control-C. You may also run Apache from the Start Menu during installation.

Note: Pressing Control-C may not allow Apache to end any current operations and clean up gracefully.

Apache Services Accounts: By default, all Apache services are registered to run as the system user (the LocalSystem account). The LocalSystem account has no network privileges through any Windows-secured mechanism. However, the LocalSystem account has wide privileges locally. For details about creating a separate account to run one or more Apache services, see your Apache documentation.

To prepare Apache v2 for Windows

1. Confirm that your environment meets Apache requirements, as described in Apache documentation located at:

<http://httpd.apache.org/docs-2.0/install.html>

For Windows installations a list of HTTP and FTP mirrors from which you can download Apache v2 is provided online.

When you complete the next step, be sure to download the version of Apache for Windows with the .msi extension.

2. Download a complete, unmodified version of the Apache HTTP Server v2 (and OpenSSL), as described in the Apache documentation. For example:

<http://httpd.apache.org/download.cgi>

<http://www.openssl.org/>

3. Install Apache v2 (run the .msi file you downloaded and supply requested information), using your Apache documentation as a guide.
4. Locate the .default.conf file, verify new settings, then update your existing configuration file if needed.
5. Start Apache, either in a console window or as a service.
6. Launch a browser and enter the following URL to connect to the server and access the default page. For example:

`http://localhost/`

A welcome page and a link to the Apache manual should appear. If not, look in the error.log file in the logs subdirectory.

Once your basic installation is working, you need to configure it properly by editing the files in the \conf subdirectory.

7. Configure the Apache installation for your environment, using the Apache documentation as a guide.
8. Test your customized environment.
9. Continue with the following as needed:
 - [Activating Reverse Proxy](#)
 - [Installing Oracle Access Manager Web components](#)

Activating Reverse Proxy

The WebGates for Apache v2 and IHS v2 powered by Apache support reverse proxy capability, if you choose to activate this capability. The procedures to implement reverse proxy capability differ, depending on your environment:

- [To activate reverse proxy capability for Apache v2 Web servers](#)
- [To activate reverse proxy capability for IHS v2 Web servers](#)

Activating Reverse Proxy For Apache v2 Web Servers

For reverse proxy functions with Oracle Access Manager, you need to include the Apache proxy module in the configure command for the Web server. You also need to load mod_proxy and the mod_proxy_http module into the server dynamically. A reverse proxy is activated using the ProxyPass directive or the [P] flag to the RewriteRule directive.

Reverse proxy capability is activated using the ProxyPass directive or the [P] flag to the RewriteRule directive. It is not necessary to turn ProxyRequests on to configure a reverse proxy. Access control is less critical when using a reverse proxy (ProxyPass directive with ProxyRequests Off), because clients can contact only the hosts that you have specifically configured. You can control access to your proxy using the <Proxy> control block.

To activate reverse proxy capability for Apache v2 Web servers

1. Review "[About the Apache and IBM HTTP Reverse Proxy Server](#)" on page 17-3.
2. Include the Apache proxy module in the configure command for the Web server, if needed.

For example:

```
--enable-proxy
--enable-proxy-connect
--enable-proxy-ftp
--enable-proxy-http
```

See the Apache documentation for more information.

3. Use the ProxyPass directive or the [P] flag to the RewriteRule directive to activate a reverse proxy, as follows:

```
Reverse Proxy
ProxyRequests Off
<Proxy *>
    Order deny,allow
    Allow from all
</Proxy>
ProxyPass /foo http://foo.example.com/bar
ProxyPassReverse /foo http://foo.example.com/bar
```

4. Control access to your proxy using the <Proxy> control block as follows:

```
<Proxy *>
    Order Deny,Allow
    Deny from all
    Allow from 192.168.0
</Proxy>
```

5. Complete "[Installing Oracle Access Manager Web components](#)" on page 17-23, if you haven't yet done so.

Activating Reverse Proxy For IHS v2 Web Servers

Use the following procedure after installing the Web server.

To activate reverse proxy capability for IHS v2 Web servers

1. Review "[About the Apache and IBM HTTP Reverse Proxy Server](#)" on page 17-3
2. Install the IHS v2 Web server, as described in "[Preparing the IHS v2 Web Server](#)" on page 17-9.
3. Load the modules by including these lines (uncommented) in the Dynamic Shared Object section of the httpd.conf file in:

IHS_install_dir/conf/httpd.conf

```
LoadModule access_module modules/mod_access.so
LoadModule auth_module modules/mod_auth.so
LoadModule auth_dbm_module modules/mod_auth_dbm.so
LoadModule include_module modules/mod_include.so
LoadModule log_config_module modules/mod_log_config.so
LoadModule env_module modules/mod_env.so
LoadModule unique_id_module modules/mod_unique_id.so
LoadModule setenvif_module modules/mod_setenvif.so
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_connect_module modules/mod_proxy_connect.so
LoadModule proxy_ftp_module modules/mod_proxy_ftp.so
LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule mime_module modules/mod_mime.so
LoadModule dav_module modules/mod_dav.so
LoadModule autoindex_module modules/mod_autoindex.so
```

```

LoadModule asis_module modules/mod_asis.so
LoadModule info_module modules/mod_info.so
LoadModule cgid_module modules/mod_cgid.so
LoadModule dav_fs_module modules/mod_dav_fs.so
LoadModule vhost_alias_module modules/mod_vhost_alias.so
LoadModule dir_module modules/mod_dir.so
LoadModule imap_module modules/mod_imap.so
LoadModule actions_module modules/mod_actions.so
LoadModule userdir_module modules/mod_userdir.so
LoadModule alias_module modules/mod_alias.so
LoadModule rewrite_module modules/mod_rewrite.so

```

4. Directives Under the IfModule mod_proxy.c Tag--Use the information and the following examples to ensure that:

- Allow and Deny conditions are appropriately commented.

For example:

```

<Proxy *>
    Order deny, allow
#    Deny from all
    Allow from all
#    Allow from .domain.com
</Proxy>

```

- URLs to be protected are mentioned in both the ProxyPass and the ProxyPassReverse directives.

For example:

```

<IfModule mod_proxy.c>
ProxyRequests Off
ProxyPass /testproxy http://bedford: 8809/testrev/
ProxyPassReverse /testproxy http://bedford: 8809/testrev/
ProxyPass /test2 http://bedford: 8809/testrev/
ProxyPassReverse /test2 http://bedford: 8809/testrev/

```

5. Restart the Web server after any modifications to the httpd.conf file.
6. **Testing:** To access the proxy URL, access `http://<proxy_
host>:80/testproxy/`

Note:

While testing, make sure the URLs have a trailing forward slash. Sometimes resources cannot be accessed without the forward slash at the end.

7. Enabling SSL on Reverse Proxy Server: Use the documentation on the IHS default page.

For example, sample SSL settings in the DSO section of the httpd.conf file load the `ibm_ssl_module` as:

```

LoadModule ibm_ssl_module modules/mod_ibm_ssl.so

```

8. Include the following directives in your httpd.conf file:

```

SSLEnable
    Keyfile /opt/IBMIHS/bin/key.kdb
    SSLClientAuth none

```

SSLProxyEngine on

9. Restart server.
10. Access the Web server URL and confirm that the browser is presented with a certificate.

Note: You can switch back to open mode for the Web server simply by commenting out the preceding directives and restarting the server.

11. **key.kdb:** To generate the key.kdb, use the ikeyman utility (preferably in GUI mode) provided in the *IHS_install_dir/bin* directory.

Note: The ikeyman utility uses the gsk7bas utility. However, you need to apply fix pack PQ83048 on gsk7bas.

12. Complete "Installing Oracle Access Manager Web components" on page 17-23, if you haven't yet done so.

Installing Oracle Access Manager Web components

As discussed earlier, Oracle provides one installation package for each Web component for each platform, which handles both plain and SSL-capable installations. All types of Oracle Access Manager Web components may be used within your deployment. See the certify tab at <https://metalink.oracle.com>.

Before you begin installing Oracle Access Manager Web components, confirm that you have completed all the following tasks.

- Complete all activities in "Preparing Your Web Server" on page 17-8
- Complete activities in "Activating Reverse Proxy" on page 17-20, as appropriate for your environment.
- Complete all activities in Chapter 2, "Preparing for Installation".
- Review "Task overview: Installing Oracle Access Manager".

For more information, see individual chapters in this guide. Failure to complete all prerequisites may adversely affect your installation.

Installing Oracle Access Manager components is similar for all platforms and Web servers. Oracle recommends that you automatically update your Web server configuration for Oracle Access Manager when asked during component installation.

Task overview: Installing Oracle Access Manager

1. Before installing Oracle Access Manager, complete all activities in "Installing Oracle Access Manager Web components" on page 17-23.
2. **Identity Server:** Locate the appropriate package for your platform, then install the Identity Server and confirm that you have a working installation, as described in Chapter 4, "Installing the Identity Server".
3. **WebPass:** Locate the appropriate installation package for your platform and complete activities in:
 - [Manually Updating a Web Server Configuration for Oracle Access Manager](#)

- [Verifying httpd.conf Updates for Oracle Access Manager Web Components](#)
- [Tuning Apache/IHS v2 for Oracle Access Manager Web Components](#)
- 4. **Policy Manager:** Locate the appropriate installation package for your platform and complete activities in:
 - [Manually Updating a Web Server Configuration for Oracle Access Manager](#), if needed
 - [Verifying httpd.conf Updates for Oracle Access Manager Web Components](#)
 - [Tuning Apache/IHS v2 for Oracle Access Manager Web Components](#)
- 5. **Access Server:** Locate the appropriate installation package for your platform and install the Access Server, as described in [Chapter 8, "Installing the Access Server"](#).
- 6. **WebGate:** Locate the appropriate installation package for your platform and complete activities in:
 - [Chapter 9, "Installing the WebGate"](#)
 - [Manually Updating a Web Server Configuration for Oracle Access Manager](#), if needed
 - [Verifying httpd.conf Updates for Oracle Access Manager Web Components](#)
 - [Tuning Apache/IHS v2 for Oracle Access Manager Web Components](#)
- 7. After installing Oracle Access Manager, you can complete the following activities:
 - Configure Oracle Access Manager, as described in the *Oracle Access Manager Identity and Common Administration Guide*.
 - Customize Oracle Access Manager, as described in the *Oracle Access Manager Customization Guide*.
 - Perform integrations, as described in the *Oracle Access Manager Integration Guide*.

Manually Updating a Web Server Configuration for Oracle Access Manager

Oracle recommends that you automatically update your Web server configuration for Oracle Access Manager. During Oracle Access Manager installation you are asked if you want to automatically update your Web server installation. If you selected No, you must do this manually.

Note: If the manual configuration process was launched during Oracle Access Manager installation, you can skip Launch your Web browser, and open the following file, if needed in the following procedure, which shows the steps for a WebGate Web server.

To manually configure your Web server for Oracle Access Manager

1. Launch your Web browser, and open the following file, if needed. For example:
`WebGate_install_dir/access/oblix/lang/langTag/docs/config.htm`
where `/WebGate_install_dir` is the directory where you installed the WebGate.
2. Select the Apache v2 or IHS v2 Web server link.

3. Follow all instructions that appear and make a back up copy of any file that you are required to modify during Web server set up, so it is available if you need to start over.

Some setups launch a new browser window or require you to launch a Command window to input information, so ensure that you return to and complete all original setup instructions to enable your Web server to recognize the appropriate Oracle Access Manager files.

Note: If you accidentally closed the window, return to step1 and click the appropriate link again.

4. Continue with "[Verifying httpd.conf Updates for Oracle Access Manager Web Components](#)" on page 17-25.

Verifying httpd.conf Updates for Oracle Access Manager Web Components

It is a good idea to complete the following procedures to ensure that the Apache or IHS v2 httpd.conf file includes Web server configuration updates for Oracle Access Manager. For details, see:

- [Verifying WebPass Details](#)
- [Verifying Policy Manager Details](#)
- [Verifying WebGate Details](#)
- [Verifying Language Encoding](#)

To update httpd.conf for reverse proxy on IHS Web servers, see "[Activating Reverse Proxy For IHS v2 Web Servers](#)" on page 17-21. To customize httpd.conf for your Web server, see your Web server documentation.

Verifying WebPass Details

The example that follows shows the WebPass section in the httpd.conf file following an update for Oracle Access Manager. Specific details will vary, depending on your environment. This example is provided only to illustrate the type of changes you will see in httpd.conf for Oracle Access Manager.

To verify WebPass entries in httpd.conf

1. Locate the updated httpd.conf file on the machine hosting the WebPass.
2. Open the httpd.conf file and ensure that the section that loads the WebPass in your platform is present. For example:

```
# Note: Copy the following lines only if they do not already exist in your
httpd.conf
##*** BEGIN Oblix NetPoint WebPass Specific ***
include "/home/netpoint/703/wp/identity/oblix/.apacheconfig"
    LoadFile "/home/netpoint/703/wp/identity/oblix/lib/libgcc_s.so.1"
    LoadFile "/home/netpoint/703/wp/identity/oblix/lib/libstdc++.so.5"
<IfModule mod_ssl.c>
    LoadModule OBWebPass_Module "/home/netpoint/703/wp/identity/oblix/apps/
webpass/bin/libwebpassssl.so"
</IfModule>
<IfModule !mod_ssl.c>
```

```
        LoadModule OBWebPass_Module "/home/netpoint/703/wp/identity/oblix/apps/
webpass/bin/libwebpass.so"
</IfModule>
obwebpassinstalldir "/home/netpoint/703/wp/identity"
    Alias /identity/oblix "/home/netpoint/703/wp/identity/oblix/"
<Directory "/home/netpoint/703/wp/identity/oblix/">
    DirectoryIndex index.htm index.html
</Directory>
<Location /identity/oblix/apps/asynch/bin/asynch.cgi>
    SetHandler asynch
</Location>
<Location /identity/oblix/apps/common/bin/common.cgi>
    SetHandler common
</Location>
<Location /identity/oblix/apps/corpcdir/bin/corpcdir.cgi>
    SetHandler corpcdir
</Location>
<Location /identity/oblix/apps/admin/bin/corpcdir_admin.cgi>
    SetHandler corpcdiradmin
</Location>
<Location /identity/oblix/apps/admin/bin/front_page_admin.cgi>
    SetHandler front_pageadmin
</Location>
<Location /identity/oblix/apps/admin/bin/genconfig.cgi>
    SetHandler genconfig
</Location>
<Location /identity/oblix/apps/groupservcenter/bin/groupservcenter.cgi>
    SetHandler groupservcenter
</Location>
<Location /identity/oblix/apps/admin/bin/groupservcenter_admin.cgi>
    SetHandler groupservcenteradmin
</Location>
<Location /identity/oblix/apps/help/bin/help.cgi>
    SetHandler help
</Location>
<Location /identity/oblix/apps/lost_pwd_mgmt/bin/lost_pwd_mgmt.cgi>
    SetHandler lost_pwd_mgmt
</Location>
<Location /identity/oblix/apps/objservcenter/bin/objservcenter.cgi>
    SetHandler objservcenter
</Location>
<Location /identity/oblix/apps/admin/bin/objservcenter_admin.cgi>
    SetHandler objservcenteradmin
</Location>
<Location /identity/oblix/apps/querybuilder/bin/querybuilder.cgi>
    SetHandler querybuilder
</Location>
<Location /identity/oblix/apps/selector/bin/selector.cgi>
    SetHandler selector
</Location>
<Location /identity/oblix/apps/admin/bin/servcenter_admin.cgi>
    SetHandler servcenteradmin
</Location>
<Location /identity/oblix/apps/admin/bin/setup_admin.cgi>
    SetHandler setupadmin
</Location>
<Location /identity/oblix/apps/admin/bin/sysmgmt.cgi>
    SetHandler sysmgmt
</Location>
<Location /identity/oblix/apps/userservcenter/bin/userservcenter.cgi>
```

```

        SetHandler userservcenter
    </Location>
<Location /identity/oblix/apps/admin/bin/wrsc_admin.cgi>
    SetHandler wrscadmin
</Location>
##**** END Oblix NetPoint WebPass Specific ****

```

Verifying Policy Manager Details

The example that follows shows the Policy Manager section in the httpd.conf file following an update for Oracle Access Manager. Specific details will vary, depending on your environment. This example is provided only to illustrate the type of changes you will see in httpd.conf for Oracle Access Manager.

To verify Policy Manager entries in httpd.conf

1. Locate the updated httpd.conf file on the machine hosting the Policy Manager.
2. Open the httpd.conf file and ensure that the section that loads the Policy Manager in your platform is present. For example:

```

**** BEGIN Oblix NetPoint Access Manager Specific ****
LoadFile "/home/netpoint/703/wp/access/oblix/lib/libgcc_s.so.1"
LoadFile "/home/netpoint/703/wp/access/oblix/lib/libstdc++.so.5"
LoadFile "/home/netpoint/703/wp/access/oblix/lib/libobnspr4.so"
LoadFile "/home/netpoint/703/wp/access/oblix/lib/libobplc4.so"
LoadFile "/home/netpoint/703/wp/access/oblix/lib/libobplds4.so"
LoadFile "/home/netpoint/703/wp/access/oblix/lib/libobsoftokn3.so"
LoadFile "/home/netpoint/703/wp/access/oblix/lib/libobnss3.so"
LoadFile "/home/netpoint/703/wp/access/oblix/lib/libobssl3.so"
LoadFile "/home/netpoint/703/wp/access/oblix/lib/libobldap50.so"
LoadFile "/home/netpoint/703/wp/access/oblix/lib/libobprldap50.so"
LoadFile "/home/netpoint/703/wp/access/oblix/lib/libobssldap50.so"
Alias /access/oblix "/home/netpoint/703/wp/access/oblix/"
<IfModule mod_ssl.c>
    LoadModule OBAccessManager "/home/netpoint/703/wp/access/oblix/lib/
webpluginssl.so"
</IfModule>
<IfModule !mod_ssl.c>
    LoadModule OBAccessManager "/home/netpoint/703/wp/access/oblix/lib/
webplugins.so"
</IfModule>
obinstalldir "/home/netpoint/703/wp/access"
<Location /access/oblix/apps/front_page/bin/front_page.cgi>
    SetHandler obfrontpage
</Location>
<Location /access/oblix/apps/common/bin/common.cgi>
    SetHandler obcommon
</Location>
<Location /access/oblix/apps/admin/bin/genconfig.cgi>
    SetHandler obgenconfig
</Location>
<Location /access/oblix/apps/admin/bin/sysmgmt.cgi>
    SetHandler obsysmgmt
</Location>
<Location /access/oblix/apps/admin/bin/setup_admin.cgi>
    SetHandler obsetupadmin
</Location>
<Location /access/oblix/apps/admin/bin/front_page_admin.cgi>
    SetHandler obfrontpageadmin

```

```
</Location>
<Location /access/oblix/apps/admin/bin/wrsc_admin.cgi>
    SetHandler obwrscadmin
</Location>
<Location /access/oblix/apps/help/bin/help.cgi>
    SetHandler obhelp
</Location>
<Location /access/oblix/apps/policyservcenter/bin/policyservcenter.cgi>
    SetHandler obpolicyservcenter
</Location>
**** END Oblix NetPoint Access Manager Specific ****
```

Verifying WebGate Details

The example that follows shows the WebGate section in the httpd.conf file. The details will vary, depending on your environment. This example is provided only to illustrate the type of changes you will see in httpd.conf.

To verify the WebGate section in httpd.conf

1. Locate the updated httpd.conf file on the machine hosting the WebGate.
2. Open the httpd.conf file and ensure that the section that loads the WebGate in your platform is present.

For example:

On Windows

```
**** BEGIN Oblix NetPoint WebGate Specific ****
<IfModule mod_ssl.c>
LoadModule obWebgateModule "WebGate_install_
dir\access\oblix\apps\webgate\bin\webgatessl.d ll"
    WebGateInstallDir "WebGate_install_dir"
    WebGateMode PEER
</IfModule>
<IfModule !mod_ssl.c>
LoadModule obWebgateModule "WebGate_install_
dir\access\oblix\apps\webgate\bin\webgate.dll"
    WebGateInstallDir "WebGate_install_dir"
    WebGateMode PEER
</IfModule>
<Location "\oberr.cgi">
SetHandler obwebgateerr
</Location>
<LocationMatch "/*">
    AuthType Oblix
    require valid-user
</LocationMatch>
**** END Oblix NetPoint WebGate Specific ****
```

On Unix

```
**** BEGIN Oblix NetPoint WebGate Specific ****
LoadFile "/home/qa/netpoint/703/c1-copy/wg/access/oblix/lib/libgcc_s.so.1"
LoadFile "/home/qa/netpoint/703/c1-copy/wg/access/oblix/lib/libstdc++.so.5"
<IfModule mod_ssl.c>
    LoadModule obWebgateModule "/home/qa/netpoint/703/c1-copy/wg/access/oblix/
apps/webgate/bin/webgatessl.so"
</IfModule>
<IfModule !mod_ssl.c>
```

```

    LoadModule obWebgateModule "/home/qa/netpoint/703/c1-copy/wg/access/oblix/
apps/webgate/bin/webgate.so"
</IfModule>
WebGateInstalldir "/home/qa/netpoint/703/c1-copy/wg/access"
WebGateMode PEER
<Location /access/oblix/apps/webgate/bin/webgate.cgi>
SetHandler obwebgateerr
</Location>
<Location "/oberr.cgi">
SetHandler obwebgateerr
</Location>
<LocationMatch "/*">
AuthType Oblix
require valid-user
</LocationMatch>
**** END Oblix NetPoint WebGate Specific ****

```

Notes for Unix

When running Apache v2 on HP-UX, do not use nobody for User or Group, because shared memory may not work. Instead, use your login name as User Name with a group Group as "Oblix" (or "www" as User Name and "others" as Group Name). On HP-UX, "www" is equivalent to "nobody" on Solaris.

When running Apache v2 on HPUX 11.11, ensure that the AcceptMutex directive in the Apache httpd.conf file is set to "fcntl". If the directive is not present, add it to the httpd.conf file (AcceptMutex fcntl). For more information, see http://issues.apache.org/bugzilla/show_bug.cgi?id=22484).

For more information about Unix implementations, see "[Tips and Troubleshooting](#)" on page 17-34

For IHS on AIX

```

**** BEGIN Oblix NetPoint WebGate Specific ****
LoadModule obWebgateModule DR/oblix/apps/webgate/bin/webgate.so
WebGateInstalldir DR
WebGateMode PEER
<Location "/oberr.cgi">
    SetHandler obwebgateerr
</Location>
<LocationMatch "/*">
    AuthType Oblix
    require valid-user
</LocationMatch>
**** END Oblix NetPoint WebGate Specific ****

```

3. Use the `chmod -r username:groupname directory/file` to change the User Name and Group Name of a directory or a file

When you do this, you need to change the User and Group parameters in the httpd.conf file accordingly.

4. See "[Tuning Apache/IHS v2 for Oracle Access Manager Web Components](#)" on page 17-31 for more information and complete any additional steps needed to finish the Oracle Access Manager implementation for Apache v2.

Important: You use the following procedure only if you need to clear the httpd.conf file of WebGate-related changes, then complete the Apache v2 Web server configuration for the WebGate anew.

To start httpd.conf updates anew

1. Restore the original httpd.conf file to remove any Oracle Access Manager entries that are present.
2. Update the httpd.conf file for Oracle Access Manager using one of the following methods:
 - **Either** open the file `Component_install_dir/access/oblix/lang/LangTag/docs/config.htm` and perform a manual configuration, as described in ["Manually Updating a Web Server Configuration for Oracle Access Manager"](#) on page 17-24.
 - **Or** launch the ManageHttpConf program in `Component_install_dir/access/oblix/tools/setup/InstallTools/ManageHttpConf` without any options to print instructions on its use.

Note: If the ManageHttpConf program is run with WebGate entries already present in the httpd.conf file, an error message will be printed and the httpd.conf file will not be updated.

3. Complete activities in ["Tuning Apache/IHS v2 for Oracle Access Manager Web Components"](#) on page 17-31, then you are ready to:
 - Configure Oracle Access Manager, as described in the *Oracle Access Manager Identity and Common Administration Guide*.
 - Customize Oracle Access Manager, as described in the *Oracle Access Manager Customization Guide*.
 - Perform integrations, as described in the *Oracle Access Manager Integration Guide*.

Verifying Language Encoding

As mentioned earlier, Oracle Access Manager HTML pages use UTF-8 encoding. Apache-based Web servers allow administrators to specify a default character set for all HTML pages sent out using the `AddDefaultCharset` directive, which overrides any character specified by the application generating the HTML pages. If the `AddDefaultCharset` directive enables a character set other than UTF-8, Oracle Access Manager HTML pages are garbled.

To ensure proper language encoding

1. Open the httpd.conf file.
2. Locate the `AddDefaultCharset` directive.
3. Complete one of the following activities to ensure that proper encoding of Oracle Access Manager HTML pages:
 - Either set the `AddDefaultCharset` directive to `Off`.
 - Or Comment out the `AddDefaultCharset` directive.
4. Save the httpd.conf file and restart the Web server.

Tuning OHS for Oracle Access Manager Web Components

After installing the Oracle Access Manager Web component for OHS, you need to complete the steps that follow.

Note: As mentioned earlier, before installing Oracle Access Manager Web components for OHS, in the httpd.conf file change user and group to match the user that is installing the component.

To tune OHS for Oracle Access Manager Web components

1. Shut down opmn.
2. In the opmn.xml file, adjust items as follows:

```
<ias-component id="HTTP_Server">
<process-type id="HTTP_Server" module-id="OHS2">
  <environment>
    <variable id="LD_ASSUME_KERNEL" value="2.4.19"/>
  </environment>
  <module-data>
    <category id="start-parameters">
      <data id="start-mode" value="ssl-disabled"/>
    </category>
  </module-data>
  <process-set id="HTTP_Server" numprocs="1"/>
</process-type>
</ias-component>
```

3. In httpd.conf file on the Policy Manager, comment-out the following lines:

```
#LoadModule perl_module modules/mod_perl.so
#LoadModule php4_module modules/mod_php4.so
```

Starting and Stopping OHS Web Servers

Starting and stopping an OHS Web server is the same procedure for both v1.3 and v2, on all platforms.

To start the OHS Web server

1. Locate and change to the following directory:

```
$ORACLE_HOME\opmn\bin\
```

2. From the command line, enter the following command:

```
opmnctl/startproc process-type=HTTP_Server
```

To stop the OHS Web server

1. Locate and change to the following directory:

```
$ORACLE_HOME\opmn\bin\
```

2. From the command line, enter the following command:

```
opmnctl/stopproc process-type=HTTP_Server
```

Tuning Apache/IHS v2 for Oracle Access Manager Web Components

Unless explicitly stated, information here applies to both Apache and IHS v2, and to Policy Manager, WebPass, and WebGate components (also known as plug-ins). For details about OHS, see the *Oracle HTTP Server Administrator's Guide 10g R2 (10.1.2)*.

Apache v2 bundled with the Red Hat Enterprise Linux 4: SELinux (Security-enhanced Linux) is an automatically enabled implementation of a mandatory access-control mechanism. As described in your Red Hat documentation, SELinux policies provide access to certain pre-defined system directories such as `/etc/httpd/conf`, `/usr/sbin/apachectl`, and `/var/log/` (to name a few) for system daemons.

When Oracle Access Manager Web components are installed with the bundled Apache Web server, certain policies must be added to allow Apache processes to access Oracle Access Manager installation files.

The bundled Apache Web server runs as user "apache" with a security context defined as `context=user_u:system_r:unconfined_t`. As a result, when Oracle Access Manager Web components are installed in any of the user folders, the Apache Web server will not start.

The `$SELINUX_SRC` variable represents the SELinux policy source directory. The default value is `/etc/selinux/targeted/src/policy`. However, your environment may vary. Be sure to consult your system administrator for the actual value for your system.

To add Oracle Access Manager policies to Apache bundled with Red Hat Enterprise Linux 4

1. After installing each Oracle Access Manager Web component, log in as the 'root' user.
2. Ensure that all Oracle Access Manager Web components are installed for Web server user & group (default: apache).
3. Create an `oracle_access_manager.te` policy file in the `$SELINUX_SRC/domains/programs/directory` and add the following rules:

```
type oracle_access_manager_t, file_type, sysadmfile;
allow httpd_t oracle_access_manager_t:file { rw_file_perms create rename
link unlink setattr execute };
allow httpd_t oracle_access_manager_t:dir { rw_dir_perms create append
rename link unlink setattr };
```

4. Create an `oracle_access_manager.fc` file context in the directory `$SELINUX_SRC/file_contexts/program`, then register the Oracle Access Manager Web component installation directory (without identity or access suffix). For example:

```
Oracle_Access_Manager_install_dir(/.*)?
system_u:object_r:oracle_access_manager_t
```

Note: When the WebGate is installed in a separate directory from the Access Manager, be sure to register the WebGate installation directory separately.

5. Compile and deploy the policy files as follows:

```
cd $SELINUX_SRC
run make load
Label Oracle Access Manager files
run restorecon -R Oracle_Access_Manager_install_dir (without the identity or
access suffix)
```

Apache v2 Directives: Apache 1.3 uses a process model for serving multiple HTTP requests at once. This differs from the single process (thread) model employed by other Web servers, which manage several requests simultaneously in one process. For more information about Apache v1.3 Web Servers and Oracle Access Manager, see [Chapter 16, "Configuring the Apache v1.3 and Oracle HTTP Server Web Servers"](#).

Note: Only the prefork MPM in Apache v2 uses the same process model for serving HTTP requests as Apache v1.3. For all other MPMs, Apache v2 uses a hybrid process-thread model.

Several directives in the Apache v2 Web server configuration file (`httpd.conf`) affect how the Apache Web server decides to create or destroy worker processes. The following parameters affect the performance of the Apache v2 Web server:

- **ThreadsPerChild:** This directive sets the number of threads created by each child process. The child creates these threads at startup and never creates more.
 - If you are using an MPM like `mpm_winnt`, where there is only one child process, this number should be high enough to handle the entire load of the server.
 - If you are using an MPM like `mpm_worker`, where there are multiple child processes, the total number of threads should be high enough to handle the common load on the server.
- **MinSpareThreads:** This value is only used with `mpm_worker`. Since Oracle Access Manager plug-in initialization is deferred until the first request, there is minimal advantage of keeping high value for this directive. However, it is useful to keep this parameter as high as possible.
- **MaxSpareThreads:** This value is only used with `mpm_worker`. The value for `MaxSpareThreads` must be greater than or equal to the sum of `MinSpareThreads` and `ThreadsPerChild` or the Apache HTTP Server automatically corrects it.

Recommendation: Keep the value high. For a dedicated server this will not be a problem.
- **MaxSpareServers:** With Apache v2, this is used only with the prefork MPM model. To preserve as much state as possible in the server, set the `MaxSpareServers` to a high value. Setting this value to the maximum of 255 keeps all Apache worker-processes available indefinitely, but it does not provide an opportunity for worker-process recycling during low-load periods.
- **MinSpareServers:** With Apache v2, this is used only with the prefork MPM model. Since Oracle Access Manager plug-in initialization is deferred until the first request, using a high value for the `MinSpareServers` parameter provides minimal advantage. However, it is useful to keep this parameter as high as possible. For dedicated Web server systems, this should pose no great burden.
- **MaxClients:** With IHS v2 and the worker MPM, `MaxClients` restricts the total number of threads that will be available to serve clients. For hybrid MPMs, the default value is 16 (`ServerLimit`) multiplied by a value of 25 (`ThreadsPerChild`). To increase `MaxClients` to a value that requires more than 16 processes, you must also raise `ServerLimit`.

Appropriate values for the preceding parameters depend on the expected load and the performance class of the systems involved, including the Access Server and LDAP server.

Apache servers on very high performance systems with high expected loads may be recompiled with a larger limit on the number of worker processes. These systems may see a greater performance impact on the StartServers and MinSpareServers parameters for dealing with sudden load spikes.

You may need to adjust operating system limits for the Access Server for proper operation. In particular, the maximum number of file descriptors available for any one Access Server may need to be increased beyond the default value. Configuring more than one connection between each Apache-based WebGate and an Access Server may quickly exceed this limit.

For additional information, see your Apache documentation.

Tips and Troubleshooting

The following additional information may prove helpful:

- [Apache v2 on HP-UX](#)
- [Apache v2 Bundled with Red Hat Enterprise Linux 4](#)
- [Apache v2 on Unix with the mpm_worker_module](#)

Apache v2 on HP-UX

When running Apache v2 on HP-UX, do not use nobody for User or Group, because shared memory may not work. Instead, use your login name as User Name with a group Group as "Obliv" (or "www" as User Name and "others" as Group Name). On HP-UX, "www" is equivalent to "nobody" on Solaris.

When running Apache v2 on HP-UX 11.11, ensure that the AcceptMutex directive in the Apache httpd.conf file is set to "fcntl". If the directive is not present, add it to the httpd.conf file (AcceptMutex fcntl). For more information, see:

http://issues.apache.org/bugzilla/show_bug.cgi?id=22484

Apache v2 Bundled with Red Hat Enterprise Linux 4

After installing a WebPass or WebGate on vendor-bundled Apache, the Web server may give the following error upon startup:

```
Error: Cannot load libgcc_s.so.1 library - Permission denied.
```

Solution: Change the SELinux policy rules for Oracle Access Manager Web components as described in "[Tuning Apache/IHS v2 for Oracle Access Manager Web Components](#)" on page 17-31.

Apache v2 on Unix with the mpm_worker_module

If you compile Apache on Unix with the mpm_worker_module, you need to modify the thread.c file from the Apache source for the Unix environment.

To modify the thread.c file for the Unix environment

1. Locate the thread.c file. For example:

```
APACHE 2.0.52 source/src/lib/apr/threadproc/unix/thread.c
```

2. Locate the function named apr_threadattr_create(apr_threadattr_t **new, apr_pool_t *pool) in the following code segment:

`**new, apr_pool_t *pool)` in the following code segment:

```

1-----> apr_status_t stat;
2
3-----> (*new) = (apr_threadattr_t *)apr_palloc(pool, sizeof(apr_threadattr_
t));
4-----> (*new)->attr = (pthread_attr_t *)apr_palloc(pool, sizeof(pthread_attr_
t));
5
6-----> if ((*new) == NULL || (*new)->attr == NULL) {
7----->             return APR_ENOMEM;
8-----> }
9
10-----> (*new)->pool = pool;
11-----> stat = pthread_attr_init((*new)->attr);
12
13-----> if (stat == 0) {
14----->             return APR_SUCCESS;
15-----> }
16-----> #ifdef PTHREAD_SETS_ERRNO
17-----> stat = errno;
18-----> #endif
19
20-----> return stat;
21

```

3. Add the following code before line 13 shown earlier.

```

int stacksize = 1 << 20;
pthread_attr_setstacksize(&(*new)->attr, stacksize);

```

4. Run `configure`, `make`, and `make install` to set up the Apache Web server with the `mpm_worker_module`.

Helpful Information

Consult the following manual for more information about the Oracle HTTP Server (OHS):

Oracle HTTP Server Administrator's Guide 10g R2 (10.1.2)

The following URLs provide information about building an Apache release and source code:

Apache v2 documentation:

<http://httpd.apache.org/docs-2.0/>

Apache v2 source code:

<http://httpd.apache.org/download.cgi>

Mod-SSL documentation:

http://httpd.apache.org/docs-2.0/mod/mod_ssl.html

OpenSSL documentation:

<http://www.openssl.org/docs/>

OpenSSL source code:

<http://www.openssl.org/source/>

Compiling and Installing Apache v2:

<http://httpd.apache.org/docs-2.0/install.html#test>

IHS:

[http://www-306.ibm.com/software/webservers/htpservers/doc/v2047
/manual/readme.html](http://www-306.ibm.com/software/webservers/htpservers/doc/v2047/manual/readme.html)

Setting Up Lotus Domino Web Servers for WebGates

Before you can install the WebGate with a Domino Web server, you must have properly installed and set up the Domino Enterprise Server.

This chapter provides tips about installing and configuring Lotus Domino to operate with the WebGate. Topics include:

- [Installing the Domino Web Server](#)
- [Setting Up the First Domino Web Server](#)
- [Starting the Domino Web Server](#)
- [Enabling SSL \(Optional\)](#)
- [Installing a Domino Security \(DSAPI\) Filter](#)
- [Tips](#)

Note: The information here presumes that you are familiar with your operating system commands, Lotus Notes, and the Domino Web server.

Installing the Domino Web Server

The following information focuses on Solaris. However, with some modifications, these steps can be used as a guide for other Unix systems.

Note: You will need to register if this is the first time you download from lotus.com.

To download the Domino Web server on Unix

1. Download Lotus Domino from the following URL:

```
http://www-10.lotus.com/1dd/down.nsf
```

2. Untar the downloaded file to your staging area. For example:

```
gct@planetearth[/export/users2/gct/temp] 433 : ls C37UUNA.tar
gct@planetearth[/export/users2/gct/temp] 434 : tar xf C37UUNA.tar
gct@planetearth[/export/users2/gct/temp] 435 : ls C37UUNA.tar sol/
```

You need to install Domino as user "root". The installation script creates soft link, /opt/lotus, to link to your Lotus Domino installation directory.

To install the Domino Web server on Unix

1. Run the install script for the Domino Web server. For example:

```
gct@planetearth[/export/users2/gct/temp/sol] 441 : su root
Password:
root@planetearth[/export/users2/gct/temp/sol] 1 : ls
install* license.txt script.dat sets/ tools/
root@planetearth[/export/users2/gct/temp/sol] 2 :
root@planetearth[/export/users2/gct/temp/sol] 2 : ./install
=====
Domino Server Installation
=====
Welcome to the Domino Server Install Program.
Type h for help on how to use this program.
Press TAB to begin the installation.
-----
Type h for help
Type e to exit installation
Press TAB to continue to the next screen.
-----
```

You are asked to select the setup type.

2. Select Setup type. For example:
Select Setup type: [Domino Enterprise Server]
3. Complete the installation with the following considerations in mind. For example:
 - The default program directory is set to /opt/lotus. You may over write it to another directory. For example, /export/home/WWW/lotus.
 - The default data directory is set to /local/notesdata1. You may also over write this to something else. For example, /export/home/WWW/lotus/data1.
 - Over write Domino UNIX user to own data directory. The default user is set to notes. You may change it to a valid Unix user. For example, gct or root.
 - Over write "The UNIX user for this directory must be a member of this group". The default group is set to notes. You may change it to a valid Unix group name. For example: oblix.

Note: Be sure to put Domino data directory in your \$PATH before you proceed from here.

Setting Up the First Domino Web Server

After successfully installing, you must set up the first Domino server.

To set up first Domino server

1. Run /opt/lotus/bin/http httpsetup.
By default, Domino will use port 8081.
2. Ensure that port 8081 is not already in use.
3. Launch your browser and enter the URL that follows. For example:

`http://hostname:8081`

4. Follow instructions on the screen and keep the following in mind.
 - Check HTTP to get the Web server.
 - Ensure the designated administrator has a first and last name.
 - Keep passwords simple, and record them in a safe location. For example, oblixoblix.
5. Run all commands as the Unix user that you've configured for this Domino Web server.

WARNING: Do not run as root.

Starting the Domino Web Server

After successfully setting up the first Domino Web server, you must start it.

To start Domino server

1. Run `/opt/lotus/bin/server`.
2. Launch your browser and enter the following URL.
For example:
`http://hostname:80/names.nsf`
You will be prompted for login name and password.
3. Select Server-Server.
4. Select your intended server.
5. Select Edit Server.
6. Select Ports, select Internet Ports, then click Web.
7. Change the value for TCP/IP port number to your desired port number.
8. Click Save and Close to save all your changes.
9. Restart server `/opt/lotus/bin/server`.

Enabling SSL (Optional)

Enabling SSL is not mandatory for the WebGate. However, if you need to generate a keyring file (.kyr) and its corresponding stash file (.sth) from the Lotus Notes client on a Windows system to the Unix system, use the steps that follow.

To generate the keyring and stash files

1. Launch the Lotus Notes Client on your Windows system.
For example:
File, select Databases, then click Open
2. Select Server Certificate Admin.
3. Create the key ring file.
4. Create the certificate request.

5. Install the trusted root certificate into the key ring file.
6. Install the certificate into the key ring file.
7. Copy or ftp the newly created keyring file and stash file from the Windows system to your Unix machine.
8. Store both files in your Domino data directory.

To enable SSL

1. Launch your browser and enter the following URL.
For example:
`http://hostname:port/names.nsf`
You will be prompted for login name and password
2. Select Server-Server.
3. Select your intended server.
4. Select Edit Server.
5. Select Ports, select Internet Ports, then click Web.
6. In the SSL Key file name field, enter the absolute path to the keyring file.
7. Change the SSL Port number value to your desired port number.
8. Enable SSL port status.
9. Select Client Certificate "Yes" for Client Certificate authentication.
10. Click Save and Close to save all your changes.
11. Restart the Web server.

For example:

`/opt/lotus/bin/server`

Installing a Domino Security (DSAPI) Filter

The Domino security API filter, DSAPI, is an authentication method that enables you to register a DLL with the Domino Web server. In this case, the Web server calls the WebGate DLL to authenticate the user when a request for authentication occurs rather than using SSL or basic authentication.

Authentication within Domino is optional with the Oracle Access Manager DSAPI filter. You can implement certain aspects of authentication that the default Web server does not support.

Task overview: Completing WebGate and filter installation

1. Before you install the WebGate on a Domino Web server, complete all steps described earlier.
2. Complete the WebGate installation and Web server update as described in "[Installing the WebGate](#)" on page 9-1.
3. See "[Completing the WebGate Installation](#)" on page 18-5 and choose one of the two options discussed there.

Completing the WebGate Installation

To ensure the Domino Web Server can use the WebGate DLL, you need to edit the enter the name or names of the DLL/DLLs (DSAPI libraries) to be called for authentication in the DSAPI filter file names field of the HTTP tab under the Internet Protocols tab in the Server document.

Note: Relative paths will be based on the Domino executable directory. DSAPI filter libraries will be called to handle events in the order they appear in this list.

There are two ways to install the filter:

- Through a Web browser and names.nsf (option 1)
- Through a Lotus Notes workstation and the Address Book (option 2)

Option 1: To setup the DSAPI filter to access names.nsf

1. Go to the names.nsf URL and log in. For example:

`http://hostname:port/names.nsf`

2. Click the Server-Servers link.
A Java applet will be loaded.
3. Select a server from those listed.
4. Click the Edit Server link to go to Edit mode.
5. Click the Internet Protocols link.
By default, the HTTP tab is selected and information is displayed in Edit mode.
6. Look for DSAPI where it says "DSAPI filter file names:", then type in the absolute path to the libwebgate.so file.
7. Save your changes.
8. Restart the Domino http server task.

Option 2: To access the Address Book through Lotus Notes

1. Open Domino Name and Address book. For example, select:
File, Database, Open, then click Address Book
2. Switch to server view and open the server document.
3. Edit the server document.
4. Click the Internet Protocols tab.
By default, the HTTP tab is selected and information is displayed in Edit mode.
5. Look for DSAPI where it says "DSAPI filter file names:", then type in the absolute path to the libwebgate.so file.
6. Save your changes.
7. Restart the Domino http server task.

Tips

The following tips may be helpful in your installation:

Failure Authentication Event: For Domino Web servers, the redirection of a URL through Oracle Access Manager may not work if the authentication type is set as Basic Over LDAP and the URL to be redirected is mentioned as one of the following:

- Either a relative path present on the same Web server

- Or the Full path URL on the same Web server containing a machine name defined in the host identifier string combinations.

To overcome a failure authentication event, you must set the redirected URL with a machine name that is not defined under the host identifier group. For example, the IP address of the machine.

This problem does not occur with a form-based authentication type.

Header Variables: It may not be possible to pass header variables other than REMOTE_USER to WebGates installed on Lotus Notes Domino Web servers when using Client Certificate authentication scheme.

For example, header variables cannot be set on the one request where Client Certificate authentication occurs. However, all other requests do allow header variables to be set.

Part VII

Product Removal, Tips, and Troubleshooting

This part provides information about removing Oracle Access Manager 10g (10.1.4.0.1), tips, and troubleshooting tips.

Part VIII contains the following chapters:

- [Chapter 19, "Important Notes"](#)
- [Chapter 20, "Removing Oracle Access Manager"](#)

Important Notes

During Oracle Access Manager 10g (10.1.4.0.1) installation you may be directed to this appendix for some important notes. Topics include:

- [Enabling Java and JavaScript On The Client](#)
- [Changing MIME Type Settings](#)
- [Choosing a Unique ID for Each User](#)
- [Contacting Oracle](#)

Enabling Java and JavaScript On The Client

Oracle Access Manager components make extensive use of Java and JavaScript. To ensure that Oracle Access Manager works properly, both Java and JavaScript must be enabled in the browser.

To enable Java and JavaScript on the client

1. Enable Java in your browser using specific instructions from your browser vendor.
2. Enable JavaScript in your browser using specific instructions from your browser vendor.

Changing MIME Type Settings

Oracle Access Manager components allow users to publish and share files of any type of ASCII or binary files (.doc, .txt, .gif, and so on). For a user to view these files using appropriate products, the server must map each file to the corresponding MIME type. Oracle ships a set of MIME type mappings for most popular file formats in the following file:s:

```
IdentityServer_install_dir\identity\oblix\apps\admin\bin\mime_types.lst  
IdentityServer_install_dir\identity\oblix\apps\admin\bin\mime_types.xml
```

```
WebPass_install_dir\identity\oblix\apps\admin\bin\mime_types.lst  
WebPass_install_dir\identity\oblix\apps\admin\bin\mime_types.xml
```

The .xml version of the file is used by the Identity Server. The .lst version of the file is used by the WebPass Java applet. Both versions of the file must match. Both versions of the file must reside in the *IdentityServer_install_dir* and in the *WebPass_install_dir*.

Since most of the popular file types are already included in the mime_types.lst file, you should not need to modify the file. However, if you do need to modify mime_types.lst, be sure to use the instructions that follow.

To edit mime_types files

1. Locate the following file and open it with a text editor:

```
IdentityServer_install_dir\identity\oblix\apps\admin\bin\mime_types.lst
```

2. Add your new mapping (ensure that each line contains only one mapping and that your MIME type and extension are separated by a ':'), then save the file. For example:

```
image/gif:gif
```

3. Locate and open the following file with a text editor:

```
IdentityServer_install_dir\identity\oblix\apps\admin\bin\mime_types.xml
```

4. Add your new mapping (ensure that each line contains only one mapping and that your MIME type is expressed as shown), then save the file. For example:

```
<NameValuePair ParamName="image/gif" Value="gif" />
```

5. Copy the two files into your WebPass installation directory. For example:

From:

```
IdentityServer_install_dir\identity\oblix\apps\admin\bin\mime_types.lst  
IdentityServer_install_dir\identity\oblix\apps\admin\bin\mime_types.xml
```

To:

```
WebPass_install_dir\identity\oblix\apps\admin\bin\mime_types.lst  
WebPass_install_dir\identity\oblix\apps\admin\bin\mime_types.xml
```

Choosing a Unique ID for Each User

Each Oracle Access Manager user must have a unique identifier (separate from the login or account name). Typically, the unique identifier will be the Employee Number, or Social Security Number, or some other type of identifier.

There are no particular restrictions on the unique ID. It does not need to be a numeric value, for example. However, choose the unique ID carefully. Not only does the system require this ID to be embedded in the data files, it also expects all users to know their ID.

Contacting Oracle

There are several ways to get in touch with Oracle.

For information:

<http://www.oracle.com/corporate/contact/index.html>

For Support Services:

<http://www.oracle.com/support/contact.html>

For the latest support information, see details under the Certify tab on:

<http://www.metalink.oracle.com>

- Log in to MetaLink.
- Click the Certify tab.

- Click View Certifications by Product.
- Select the Application Server option and click Submit.
- Choose Oracle Application Server and click Submit.

Removing Oracle Access Manager

This chapter provides important information you need when removing Oracle Access Manager components. Topics include:

Note: Failure to complete all steps may adversely affect the removal and any subsequent installation. For details about removing cloned components or components installed in silent mode, see [Chapter 15, "Replicating Components"](#).

- [Uninstalling Oracle Access Manager Components](#)
- [Recycling an Identity Server Instance Name](#)

Uninstalling Oracle Access Manager Components

During Oracle Access Manager component installation, information is saved after certain operations. Until information is saved, you may return and restate details. However, after you are informed that a component is being installed, Oracle Access Manager files are added to the file system.

Note: If you cancel the installation process after receiving the message that a component is being installed and before completing all procedures, you must restore the system to its previous condition to remove Oracle Access Manager-related information.

There are several steps you need to complete to remove an Oracle Access Manager component, as outlined in the discussion that follows. Some changes made for Oracle Access Manager are not handled automatically and must be manually removed when the Uninstaller program finishes:

Language Packs: Each installed Language Pack must be removed individually using appropriate file in the component's uninstall directory: *Component_install_dir\identity\access_uninstComponentLP_langtag\uninstaller.exe*. For example, suppose you have an Identity Server and the WebPass installed with a Korean Language Pack. After uninstalling the Korean Language Pack on each component host, you must stop and restart both the Identity Server Service and the WebPass Web server instance. This will re-initialize corresponding components with the proper language support. Removing the Language Pack associated with the default Administrator language selected during installation is not supported.

WARNING: Do not remove (uninstall) the Language Pack associated with the default Administrator language selected during installation. If you accidentally remove the Language Pack associated with the default Administrator language selected during installation, see "[Language Issues](#)" on page E-15.

Schema and Data Changes: If Oracle Access Manager will be removed and reinstalled with the same directory instance, only the Oracle Access Manager configuration tree(s) need be deleted. In this case, there is no need to remove the Oracle Access Manager schema from the directory instance. When reinstalling the Identity Server, select "No" when asked if you want to update the schema (which is already present). Selecting "Yes" results in an error message "schema already exists".

If, however, you plan to remove and reinstall Oracle Access Manager a different directory instance (or not reinstall at all) then configuration data must be removed manually from the directory server and Oracle Access Manager schema extensions must also be removed using cleanup files provided for your directory server. You must remove data from the Identity Server and Policy Manager.

Depending on the type of directory server, you may have one or two cleanup files. For instance, schema extension cleanup files are provided for user data only for VDS. However schema extension cleanup files are provided for both user data and Oblix (configuration data) for NDS, IPlanet, and Oracle Internet Directory. Schema extension cleanup file names begin with an abbreviation that identifies the type of directory, followed by the type of data to be removed.

As an example, look for the files similar to the following in the Identity Server and Policy Manager installation directories:

- *DirectoryName_user_schema_delete.ldif*—Oracle Access Manager user data cleanup file for the specific named directory—removes user data that resides on a separate directory instance from configuration data
- *DirectoryName_oblix_schema_delete.ldif*—Oracle Access Manager configuration data cleanup file for the specific named directory—removes both user and configuration data when both reside on the same directory instance
- **OID_oblix_schema_index_delete.ldif**—Oracle Access Manager cleanup file for Oracle Internet Directory only—removes the Oracle Access Manager attribute index from Oracle Internet Directory before or after you use corresponding Oracle Access Manager data and schema cleanup files.
- **OID_user_index_delete.ldif**—Oracle Access Manager cleanup file for Oracle Internet Directory only when a separate instance is used to host user data.

Some directory vendors do not provide schema cleanup files. For instance, no such files are provided for ActiveDirectory, and Active Directory Application Mode (ADAM).

Note: If Oracle Access Manager will be removed and reinstalled with the same directory instance, only the Oracle Access Manager configuration tree must be deleted. In this case, there is no need to remove the Oracle Access Manager schema from the directory instance. When reinstalling the Identity Server, select "No" when asked if you want to update the schema (which is already present). Selecting "Yes" results in an error message "schema already exists".

For details about removing then reinstalling Oracle Access Manager with Oracle Internet Directory, see ["Reinstalling Oracle Access Manager with Oracle Internet Directory"](#) on page E-20.

Web Server Configuration Changes: Web server configuration changes that occur during installation must be manually reverted after uninstalling the Oracle Access Manager component (WebPass, Policy Manager, WebGate). For example, the ISAPI transfilter will be installed for IIS WebPass. However, when you uninstall WebPass this is not removed automatically. Also, the created Web service extension and the link to the identity directory will not be removed. This type of information must be removed manually. These are examples of information to remove, not a complete list. Further, you must remove any changes that you manually made to your Web server configuration file for the Oracle Access Manager component (WebPass, Policy Manager, WebGate) should be removed. For more information about what is added for each component, see [Part VI, "Web Server Configuration"](#).

To uninstall Oracle Access Manager components

1. Turn off the Identity or Access Server service (or WebPass, Policy Manager, WebGate Web server) for the component you will remove.

Note: If you don't turn off the Web server, uninstall may not succeed and the backup folder won't be removed. If this happens, you need to manually remove the backup folder.

2. **Language Packs:** Complete the following steps to remove one or more installed Language Packs (except the one selected as the default Administrator language (locale)):

- Locate the appropriate Language Pack file in the component's uninstall directory. For example:

```
Component_install_dir\uninstIdentityLP_fr-fr
\uninstaller.exe
```

- Run the Language Pack Uninstaller program to remove the files.
- Repeat this process to remove the same Language Pack from associated components.
- Stop and restart both the Identity Server Service and the WebPass Web server instance to re-initialize components with the proper language support.
- Repeat this process to remove each Language Pack (except the one selected as the default Administrator language (locale)). For example:

```
Component_install_dir\uninstIdentityLP_ja-jp
\uninstaller.exe
```

3. Complete the following steps to remove all Oracle Access Manager configuration data from the directory server instance, then remove Oracle Access Manager schema extensions from your directory server, if needed:

- Remove the Oracle Access Manager configuration tree from the directory server instance using instructions from your directory vendor.
- Locate the ldapmodify tool in the appropriate component directory. For example:

```
Component_install_dir\oblix\tools\ldap_tools
```

- **All Directories:** Using the ldapmodify tool, upload the appropriate schema cleanup files for your directory server from the following directory, then remove Oracle Access Manager schema extensions from your directory. For example:

```
Component_install_dir\oblix\data.ldap\common\  
\DirectoryName_*_schema_delete.ldif
```

where *Component_install_dir* refers to the installation directory for the specific Oracle Access Manager component (Identity Server or Policy Manager for example), and *DirectoryName_*_schema_delete.ldif* refers to the clean up file for your specific directory and data type.

- **Oracle Internet Directory:** After completing the preceding activity to remove Oracle Access Manager schema extensions from Oracle Internet Directory, use the ldapmodify tool to upload the Oracle Internet Directory attribute index cleanup file and remove the Oracle Access Manager attribute index. For example:

```
Component_install_dir\oblix\data.ldap\common\  
OID_oblix_schema_index_delete.ldif  
OID_user_index_delete.ldif (when a separate instance is used to host  
user data)
```

If you have only one instance of an Oracle Access Manager component, complete step 4 to remove it. If you have multiple instances of a component, see also step 5.

4. Locate and run the Uninstaller program for the specific component to remove Oracle Access Manager files. For example:

```
IdentityServer_install_dir\identity\_uninstIdentity\uninstaller.exe
```

```
WebPass_install_dir\identity\_uninstWebPass\uninstaller.exe
```

and so on.

Note: On Unix systems, use `uninstall.bin`

5. **Multiple Instances:** If you have multiple instances of a component and want to remove one or all of them, you must use a specific method for your platform:
 - **Windows:** The last component can be uninstalled from Add/Remove programs. Others can be uninstalled by running the uninstall program from the `\identity` or `\access \uninstComponent` directory.
 - **Unix:** You must always run `uninstall.bin`.
6. Remove Oracle Access Manager-related updates to your Web server configuration. For information about specific Web servers, see [Part VI, "Web Server Configuration"](#).
7. Restart the Web server, if needed.
8. Remove the component installation directory if it remains, especially if you plan to reinstall the product.

Recycling an Identity Server Instance Name

Under certain circumstances, you may want to reuse an existing Identity Server name. For example, you may want to use an existing Identity Server name if you are rolling

Oracle Access Manager out from a test environment to a production environment or if you need to remove an Identity Server for some reason.

If you do not delete the original Identity Server name from the System Console, a login following the set up of a new instance may result in the message "*Application has not been set up*". Special steps must be taken to ensure you can set up the application and login when recycling an Identity Server name.

The steps that follow presume that you have another Identity Server and WebPass setup within the same installation.

To recycle an Identity Server instance name

1. Delete the Identity Server name in the directory server under:

Oblix, Policies, WebResrcDB, *name*

2. Re-run Identity Server setup, as described in [Chapter 6, "Setting Up the Identity System"](#).

3. Go to the Identity System Console, delete the inoperable Identity Server instance. For example:

Identity System Console, System Configuration, Identity Server

name, Delete

4. From the List all Identity Servers page, re-create the instance using the same ID, as described in the *Oracle Access Manager Identity and Common Administration Guide*. For example:

Add

Name

Host name

Port

Transport Security

Part VIII

Appendixes

This part provides details about installing with third-party components, performing infrequent tasks such as adding directory certificates or changing directory server hosts after installation, as well as troubleshooting tips.

Part IV contains the following appendixes:

- [Appendix A, "Installing Oracle Access Manager with Active Directory"](#)
- [Appendix B, "Installing Oracle Access Manager with ADAM"](#)
- [Appendix C, "Adding Directory Certificates After Oracle Access Manager Installation"](#)
- [Appendix D, "Changing Directory Server Hosts"](#)
- [Appendix E, "Troubleshooting Installation Issues"](#)

Installing Oracle Access Manager with Active Directory

This chapter summarizes prerequisites, installation, and set up for Oracle Access Manager with Active Directory. The following topics are included:

- [About Active Directory](#)
- [About Oracle Access Manager and Active Directory](#)
- [About Oracle Access Manager and Active Directory Forests](#)
- [Installation and Setup Considerations for Active Directory](#)
- [Installing Oracle Access Manager with Active Directory](#)
- [Active Directory Tips and Troubleshooting](#)

Some introductory information is included for configurations that use both LDAP (the default), LDAP over SSL, and the optional Active Directory Services Interface (ASDI) as the communication protocol between Oracle Access Manager and Active Directory.

The *Oracle Access Manager Identity and Common Administration Guide* also includes details about:

- Configuring Oracle Access Manager for ADSI
- Configuring Oracle Access Manager for Active Directory using LDAP
- Deploying Oracle Access Manager with Active Directory and configuring Oracle Access Manager for specific Active Directory features
- Configuring Oracle Access Manager for .NET features

About Active Directory

This discussion provides a general overview of Active Directory in broad strokes. See also, "[About Oracle Access Manager and Active Directory](#)" on page A-2. Active Directory stores information about objects in one or more domains on a network and makes this information available to users and network administrators.

- An Active Directory domain defines an administrative boundary for a collection of objects that are relevant to a specific group of users on a network.
- A domain controller stores directory partitions, also known as "naming contexts", that correspond to the logically distributed segments of the Active Directory that are replicated as discrete units.

An Active Directory that supports multiple trees or domains is called an Active Directory forest. Active Directory uses a structured data store as the basis for a logical, hierarchical organization of directory information. The Active Directory includes:

- A schema that defines the classes of objects and attributes contained in the directory, the constraints and limits on instances of these objects, and the format of their names.
- A domain controller stores directory partitions, also known as "naming contexts", that correspond to the logically distributed segments of the Active Directory that are replicated as discrete units.

An Active Directory that supports multiple trees or domains is called an Active Directory forest. Active Directory uses a structured data store as the basis for a logical, hierarchical organization of directory information. The Active Directory includes:

- A schema that defines the classes of objects and attributes contained in the directory, the constraints and limits on instances of these objects, and the format of their names.
- A Global Catalog is a domain controller that stores a copy of all Active Directory objects in a forest that applications and clients can query to locate any object in a forest. This is no longer needed with Oracle Access Manager.
- A query and index mechanism, so that objects and their properties can be published and found by network users or applications.
- A replication service that synchronizes schema, configuration, application, and domain directory partitions between domain controllers and distributes directory data across a network.

Domain Controllers and Partitions

Every Active Directory server (domain controller) in an Active Directory forest participates in replication and contains a complete copy of all directory information for their domain. Any change to directory data is replicated to all domain controllers within the domain.

Every domain controller in an Active Directory forest stores three full, writable directory partitions. In Active Directory, a directory partition is a contiguous Active Directory subtree that is replicated as a unit to other domain controllers in the forest that contain a replica of the same subtree.

A single domain controller always holds at least three directory partitions:

- Schema: one per forest with class and attribute definitions for the directory
- Configuration: one per forest with replication topology and related metadata
- Domain: many in a forest with a subtree that contains the per-domain objects for one domain

About Oracle Access Manager and Active Directory

Oracle Access Manager supports Active Directory on Windows Server 2000 and Windows 2003 Server platforms. On a server running Windows Server 2003, Web Edition:

- You cannot install Active Directory on a server running Windows Server 2003, Web Edition.

- You can join the Windows Server 2003 Web Edition server to an Active Directory domain as a member server (that is not a domain controller) joined to a domain.

Oracle Access Manager supports storing user data on a separate directory server type from configuration and policy data. For more information, see ["Data Storage Requirements"](#) on page 2-21 and details about ["ADSI Option Considerations"](#) on page A-10. With Oracle Access Manager, use of the Global Catalog is not required.

Oracle Access Manager supports structural and auxiliary object classes. A structural object class can stand on its own and contains basic attributes required for use within Oracle Access Manager applications. A structural object class must be assigned when you create a tab within a Oracle Access Manager application. An auxiliary object class cannot stand alone because it contains supplementary attributes not necessarily found in a structural object class, for example, a billing address, challenge phrase, or a response to a challenge phrase. An auxiliary object class must be assigned to an entry that is based on an existing structural object class.

Oracle Access Manager supports both InetOrgperson and GroupofUniqueNames as standard Person and Group object classes, respectively, in addition to User and Group. Oracle Access Manager also supports both statically-linked and dynamically-linked auxiliary classes.

For additional information, see ["About Oracle Access Manager and Active Directory Forests"](#) on page A-4 and ["Installation and Setup Considerations for Active Directory"](#) on page A-7.

About Statically-Linked Auxiliary Classes

With Windows Server 2000, Active Directory supported only statically-linked auxiliary classes and provided support for statically linking auxiliary classes to another objectclass in the schema definition itself. A statically-linked object class is one that is included in the auxiliaryClass or systemAuxiliaryClass attribute of an object class's classSchema definition in the schema. A statically-linked object class is part of every instance of the class with which it is associated.

When designing the schema for implementation on Active Directory for statically-linked auxiliary classes, which is the default with Oracle Access Manager:

- Define oblixOrgPerson and oblixPersonPwdPlicy objects in your user (Person) object class.
- Define oblixGroup and oblixAdvancedGroup in your Group object class.

For details, see the procedure ["To modify your schema for statically-linked auxiliary classes"](#) on page A-18.

About Dynamically-Linked Auxiliary Classes

With a Windows 2003 Server, Active Directory provides support for dynamically linking auxiliary classes to individual objects (not just entire classes of objects). In this case, you add the name of the auxiliary objectclass to the values of an object's objectclass attribute for an entry. If there are mandatory attributes in the auxiliary class, then those need to be set at the same time as well.

Dynamic linking enables you to store additional attributes with an individual object without the forest-wide impact of extending the schema definition for an entire class. For example, an enterprise can use dynamic linking to attach a sales-specific auxiliary class to the user objects of its sales people, and other department-specific auxiliary classes to the user objects of employees in other departments.

Oracle Access Manager provides a static auxiliary schema, which specifies the associations between auxiliary classes and their corresponding structural object classes in the schema. When using static auxiliary classes, Oracle Access Manager will not update the objectclass attribute for auxiliary classes to be added/removed. For dynamic auxiliary support, there is no separate schema file as such and Oracle Access Manager will update the objectclass attribute with auxiliary class name as appropriate.

During Identity Server setup and Access System installation and setup, you will be asked if you want the target directory to support dynamically-linked auxiliary classes. The following overview identifies the tasks that will ensure dynamically-linked auxiliary classes are associated in Oracle Access Manager at runtime.

Oracle Access Manager supports both statically- and dynamically-linked auxiliary classes, but not both simultaneously.

WARNING: Dynamic auxiliary classes are supported when all domain controllers in the forest are running Windows Server 2003 and the forest functional mode is Windows Server 2003. A mixed environment, with 2003 domain controllers and earlier domain controllers, is not supported. Be sure to restart the Active Directory server after raising the forest level.

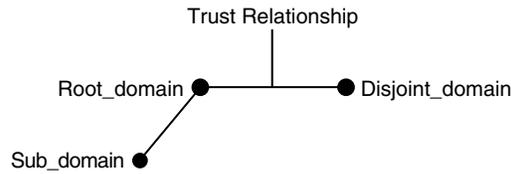
Task overview: Enabling dynamically-linked auxiliary classes

1. Before Oracle Access Manager installation, you must ensure that the Active Directory domain and forest functionality are operating at a Windows 2003 Server level, as described in the Microsoft documentation.
2. During Identity System installation and set up, specify dynamically-linked auxiliary classes as described in [Chapter 4, "Installing the Identity Server"](#) and [Chapter 6, "Setting Up the Identity System"](#).
3. During Policy Manager installation and set up, specify dynamically-linked auxiliary classes as described in [Chapter 7, "Installing the Policy Manager"](#).
4. During Access Server installation, specify dynamically-linked auxiliary classes as described in [Chapter 8, "Installing the Access Server"](#).
5. After installation and setup, configure Oracle Access Manager for dynamic auxiliary class support, as described in the *Oracle Access Manager Identity and Common Administration Guide*

About Oracle Access Manager and Active Directory Forests

In earlier versions of Oracle Access Manager, it was possible to install Oracle Access Manager within only one Active Directory forest. [Figure A-1](#) depicts a single Active Directory forest with three domains: Root_domain, Sub_domain, and Disjoint_domain.

Figure A-1 A Single Active Directory Forest with Three Domains



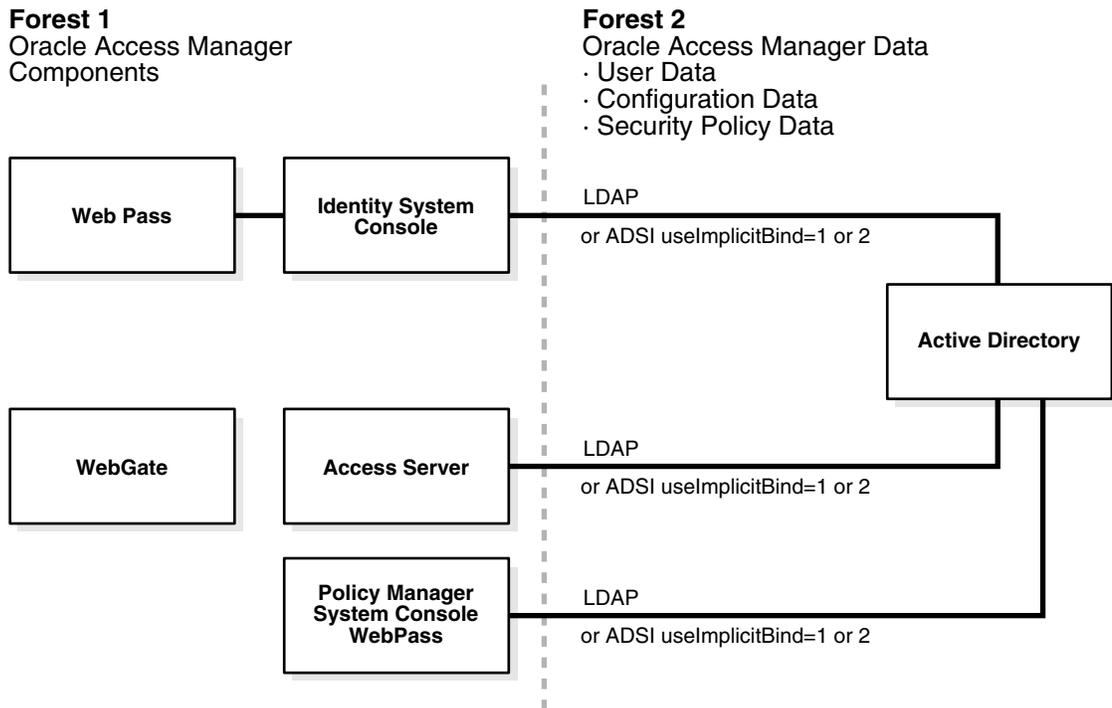
Today, Oracle Access Manager components may reside either inside the Active Directory forest with Oracle Access Manager user, configuration, and policy data, or outside the forest containing Oracle Access Manager data.

If you have installed Oracle Access Manager components inside a single Active Directory forest, you may use either of the following communication protocols between Oracle Access Manager and Active Directory:

- LDAP (the default)
- LDAP over SSL
- ADSI

Figure A-2 shows Oracle Access Manager components installed in one forest (Forest 1) with user, configuration, and policy data in another (Forest 2). This type of configuration is also known as having "Oracle Access Manager outside the forest". Using ADSI is optional.

Figure A-2 Oracle Access Manager Outside the Forest



In a two forest configuration a single domain controller in a forest, the schema master, is responsible for all changes to the schema directory partition. One domain controller per forest, the domain-naming master, is responsible for ensuring that domain names are unique in the forest and that cross-reference objects to external directories are maintained. For more information, see your Microsoft documentation.

A two forest configuration does not require setting up a trust relationship between the forest containing user, policy, and configuration data and the forest where the Oracle Access Manager servers are installed.

Your environment may include configuration and policy data in one forest and user data in another forest.

Oracle Access Manager and the Searchbase in a Parent-Child Domain

Active Directory domains can be organized into parent-child relationships to form a hierarchy. A parent domain is one that is directly superior in the hierarchy to one or more subordinate, or child, domains. A child domain may also be the parent of one or more child domains.

The searchbase in Oracle Access Manager defines the node in the directory information tree under which data is stored and the highest possible base for all searches. However, you may not locate an entry in a child domain. The default Oracle Access Manager directory server profile is created for only your Root_domain. You must set up directory profiles for the remaining domains in your installation, for example, Disjoint_domain and Sub_domain. For more information, see the *Oracle Access Manager Identity and Common Administration Guide*.

In the Access System, a sub-tree-level search is performed during authentication using the credential_mapping plug-in and during authorization using LDAP rules when the rule (for instance, LDAP URL) explicitly states the search is to occur at a sub-tree-level. Using the ObMyGroups action with an LDAP URL returns all groups to which the user belongs. [Table A-1](#) summarizes configurations that support a parent-child domain.

Table A-1 Configurations that Support a Parent-Child Domain

Function	Configurations for Parent-Child Domains
Authentication	<p>Use credential mapping to authenticate users against both the parent and child domain.</p> <p>The Oracle Access Manager credential_mapping plug-in can be used to obtain the user's DN. For an Active Directory forest, typical credential_mapping plug-in parameters are similar to those below:</p> <pre>credential_mapping?ObMappingBase="%domain%, ObMappingFilter="(&(objectclass=user) (samaccountname=%login%))", Obdomain="domain" accountname=%login%)"", Obdomain="domain"</pre>
Authorization	Use multiple LDAP URLs within an authorization rule.

Table A-1 (Cont.) Configurations that Support a Parent-Child Domain

Function	Configurations for Parent-Child Domains
ObMyGroups	<p>Use ObMyGroups with an LDAP URL. If the user belongs to groups from both parent and child domains, you must define separate header variables for groups from each domain.</p> <p>Using ObMyGroups with no URL will yield groups from the Access System searchbase only. If the searchbase is that of the parent domain, groups from the child domain will not be obtained at all.</p> <p>For example, suppose you have two domains and you want to obtain groups from both searchbases:</p> <pre>dc=goodwill,dc=oblix,dc=com and dc=dilbert,dc=goodwill,dc=oblix,dc=com</pre> <p>In this case, you must have two separate header variables, one for each domain.</p> <p>Return</p> <p>Type Name Return Attribute</p> <pre>headervar HTTP_PARENT_GROUP "obmygroups:ldap:///dc=goodwill,dc=oblix,dc=com??sub?(group_ type=role)"</pre> <pre>headervar HTTP_CHILD_GROUP "obmygroups:ldap:///dc=dilbert,dc=goodwill,dc=oblix,dc=com??sub?(g roup_type=role)"</pre> <p>Hence in HTTP_PARENT_GROUP: all the groups in "dc=goodwill,dc=oblix,dc=com" tree for which the logged-in user is a member and the group_type is "role" is returned.</p> <p>HTTP_CHILD_GROUP: all the groups in "dc=dilbert,dc=goodwill,dc=oblix,dc=com" tree for which the logged-in user is a member and the group_type is "role" is returned.</p>

Installation and Setup Considerations for Active Directory

An overview of specific considerations are summarized below for your review before you begin installing Oracle Access Manager with Active Directory:

- [Active Directory Schema Choices](#)
- [All Configurations](#)
- [ADSI Option Considerations](#)
- [LDAP Open Bind Considerations](#)
- [LDAP Over SSL Considerations](#)

Active Directory Schema Choices

There are several differences between the Active Directory 2000 and Active Directory 2003 schemas and how they apply to Oracle Access Manager:

Key Differences—The key differences between the Windows 2000 (ADSchema.ldif) and Windows 2003 schemas (dotnetschema.ldif) as it relates to Oracle Access Manager are the support of inetOrgPerson and groupofuniquenames. These two object classes exist in most other LDAP directories and were officially added to the Windows 2003 schema. The addition of the inetOrgPerson object class allows Oracle Access Manager to be configured using this object class without manually adding that object class as was required in Windows 2000.

Affect on Oracle Access Manager Schema Files—The main differences between the Oracle Access Manager Windows 2000 schema file (ADSchema.ldif) and the Windows 2003 schema file (ADdotnetschema.ldif) are outlined below:

- The following entries are in the ADSchema.ldif, but not in ADdotnetschema.ldif
dn: cn=groupOfUniqueNames,cn=schema,cn=configuration,<domain-dn>
dn: cn=uniquemember,cn=schema,cn=configuration,<domain-dn>
- The oblixgroupofuniquenames class definition is different between the two files, due to differences in how groupofuniquenames is defined in the 2000 schema file and how Microsoft implemented it in the 2003 schema.

Objectclass Differences Between the Two Schemas—The following shows how the objectclasses differ between the two schemas:

- **ADSchema.ldif:**

Must Contain/Required Attributes—There are no required attributes.

May Contain/Optional Attributes—

obuniquemember
businesscategory
obver

- **ADdotNetschema.ldif:**

Must Contain/Required Attributes

cn
businesscategory
obuniquemember
obver
description
o
ou
owner
seeAlso
uniqueMember

Determining which Schema to Load

The file named ADSchema.ldif is the Oracle Access Manager schema file for Windows 2000 and the file named ADdotnetschema.ldif is the Oracle Access Manager schema file for Windows 2003. Consider the following when deciding which schema to load in your environment:

ADdotnetschema.ldif—Install Oracle Access Manager with the .NETSchema (Windows 2003 Schema) when you have the Active Directory 2003 schema loaded whether you are running Windows 2000 or Windows 2003.

For example, some companies are preparing for an upgrade to Windows 2003 and have loaded the Windows 2003 schema in their existing Windows 2000 domain.

If this is the case, you should use the ADdotnetschema.ldif file when installing Oracle Access Manager.

ADschema.ldif—Load the Windows 2000 schema if you have the Windows 2000 Schema.

Note: If you don't load the schema files manually, the installer decides which schema file to use based on the answer you provide when asked whether you are installing on Windows 2003 or not. If you indicate that you are installing on Windows 2003, the installer uses the ADdotnetschema.ldif.

Determining the Schema Type—The easiest way to find out whether the environment has a Windows 2000 schema versus the Windows 2003 schema is to use the schema snapin and look for the new string syntax in the 2003 schema. For example:

- In the 2000 schema the string type used the Unicode format of attributesyntax 2.5.5.12.
- In the 2003 schema it changed to the new syntax of IA5 attributesyntax 2.5.5.5., omysyntax: 22.

All Configurations

The following is intended as an overview for all Oracle Access Manager installations with Active Directory.

Oracle recommends that you accept automatic schema updates during Identity Server and Access System installation and setup procedures to save time and eliminate errors. You can always make changes later. However, for manual schema updates you must use one or more of the following files during the setup process depending on your environment:

- For Windows 2003 and dynamically-linked auxiliary classes, you need only the file below:

```
\install_dir\identity | access\oblix\data\common\ADDotNetSchema.ldif
ldifde -i credentials -c "<domain-dn>" "your domain" -f ADDotNetSchema.ldif
```

- For Windows 2003 and statically-linked auxiliary classes, you need both of the files below:

```
\install_dir\identity | access\oblix\data\common\ADDotNetSchema.ldif
\install_dir\identity | access\oblix\data\common\ADAuxSchema.ldif
```

- For Windows 2000, you need only:

```
\install_dir\identity | access\oblix\data\common\ADSchema.ldif
```

It's a good idea to add the properties of the administrative user to members of:

Main Administrators
 Schema Administrators
 Group Policy Administrators
 Enterprise Administrators
 Domain Administrators
 Users, Administrators

The following guidelines apply to all Oracle Access Manager configurations with Active Directory.

Guidelines: Installing Oracle Access Manager with Active Directory

1. During Oracle Access Manager installation and setup, be sure to specify the version of Active Directory you are using, as well as responding appropriately to any related questions you are asked.
2. During Oracle Access Manager installation and setup, use the same configuration DN for the Identity Server and for the Policy Manager and Access Server.

Note: The login name for a multi domain forest is the display name from Access Server DB profile.

3. After installation and setup, you may create or change your authentication scheme or schemes for Active Directory as described in the *Oracle Access Manager Identity and Common Administration Guide*.
4. After installation and setup, you can expand a large dynamic group on Active Directory by adding the following to the globalparams.xml on the Identity Server:

```
<SimpleList>
  <NameValPair ParamName="maxForRangedMemberRetrieval "
  Value="1500" /
</SimpleList>
```

ADSI Option Considerations

The following is intended as an overview for configurations using ADSI. Using ADSI is optional.

The credentials for ADSI are used to bind to the entire forest. A forest can contain multiple Active Directory hosts. When user data and configuration data are stored on separate Active Directory hosts in separate forests, you cannot connect to these simultaneously using ADSI. For additional information, see ["ADSI Cannot Be Enabled for this DB Profile \(Active Directory\)"](#) on page E-4.

ADSI does *not* require specific host and port numbers for different domains in the forest. ADSI connects to Active directory hosts using an LDAP URL like the one shown below:

```
LDAP://domain.oracle.com/ou=oblix,dc=domain,dc=oblix,dc=com
```

When user data and configuration data are stored on separate Active Directory hosts in the same forest, you can connect to these using ADSI. The data will be searched and modified in respective Active Directory servers in the forest using the domain-naming context.

During installation you will be asked if configuration data will be stored in the user data directory. When user data and configuration data are stored on separate Active Directory hosts in the same forest and you are using ADSI for the user tree, be sure to indicate that configuration data will be stored in the user data directory. If you indicate that user data and configuration data are stored separately, you will not be allowed to connect to the configuration data directory server using ADSI and cannot create the DB profile for the configuration data by selecting ADSI from the Identity System Console page. Although you can connect to the configuration data directory server using LDAP.

- When using ADSI for the entire forest, the credentials of the Master Administrator should be that of an Enterprise Administrator with administrative privileges over the entire forest.

The same user tree credentials should be valid for the entire forest. If you decide to configure ADSI for the user tree and LDAP for the configuration/policy tree, you can change parameters in the globalparams file and define the appropriate profiles after setup is complete as described in the *Oracle Access Manager Identity and Common Administration Guide*.

- When Oracle Access Manager data and components are in the same domain, you must run the Identity Server in the context of a privileged administrative user who has change-password permissions after the Identity System is installed and set up.
- When storing user data on a separate Active Directory server from configuration and policy data in a separate forest, ADSI may be used for connecting to one or the other but not both.
 - **During Oracle Access Manager Installation**—Select Yes when asked if configuration data is stored with user data, select ADSI for the user data directory server (you won't be asked about ADSI for configuration data).
 - **During Oracle Access Manager Setup**—The directory type for the DB profiles will be indicated as follows:
 - User DB Profile**—Microsoft Active Directory and ASDI
 - Configuration DB Profile**—Microsoft Directory only (no ADSI)
- With Oracle Access Manager components in one domain and data in another, you may use either ADSI or LDAP between Oracle Access Manager components and Active Directory.

During installation and setup, Oracle Access Manager automatically updates certain parameters in the adsi_params.xml file on the Identity Server and the adsi_params.lst file on the Policy Manager. The path to these files is shown below:

`\IdentityServer_install_dir\identity\oblix\config\ads_i_params.xml`

`\AccessServer_install_dir\access\oblix\config\ads_i_params.xml`

Included in the files above is a useImplicitBind value for the user bind DN. [Table A-2](#) provides a summary of possible bind parameters.

Table A-2 Summary of Possible Bind Parameters

useImplicitBind Value	Definition	Description
0	Use the implicit credentials of the current process for the bind.	Single Active Directory Forest The default for the Identity Server in the adsi_params.xml file.
1	Use explicit credentials with the DN of the user for the bind.	Two Active Directory Forests Ensure that useImplicitBind value is set to 1 in the adsi_params.xml and .lst files.

Table A-2 (Cont.) Summary of Possible Bind Parameters

useImplicitBind Value	Definition	Description
2	Use userPrincipalName for the bind.	Two Active Directory Forests <ul style="list-style-type: none"> ■ The default for the Policy Manager in the adsi_params.lst file. ■ The preferred value when Oracle Access Manager components are in a different domain than Oracle Access Manager data. ■ The UPN should be specified in the adsiUPN parameter in the adsi_params.xml file.

If you are using ADSI with:

- **useImplicitBind=1**—You do not need to have your service login credentials set by the Identity Server if the value of useImplicitBind is 1. When Oracle Access Manager components reside in one forest and data in another, set useDNSPrefixedLDAPPaths=true.
- **implicitBind=0**—You do need to set the permissions of the IIS Anonymous user to a domain user if you are using implicitBind=0. The defaults are implicitBind=0, useDNSPrefixedLDAPPaths=false.

In this case, ADSI uses the context of the process to bind to the Active Directory server. By default the anonymous user (IWAM*) does not have rights on the directory server.

- **useDNSPrefixedLDAPPath**—You can prefix the domain name to LDAP strings using the useDNSPrefixedLDAPPath parameter in the adsi_params.xml and adsi_params.lst files. The default value is false.

Guidelines: Setting up ADSI

1. Before you install Oracle Access Manager, you need to set up Active Directory as described in the Microsoft documentation and ["Setting Up Your Environment"](#) on page A-14.
2. During Identity Server installation, enable ADSI as described in [Chapter 4, "Installing the Identity Server"](#) and ["Installing the Identity System"](#) on page A-16.
3. Before Identity System setup, complete steps in ["Setting Up ADSI \(Optional\)"](#) on page A-17.
4. During Identity System setup, specify ADSI as described in [Chapter 6, "Setting Up the Identity System"](#) and ["Setting Up the Identity System"](#) on page A-17.
5. During Policy Manager installation and set up, specify ADSI as described in [Chapter 7, "Installing the Policy Manager"](#) and ["Installing and Setting Up the Access System"](#) on page A-19.
6. During Access Server installation, enable ADSI as described in [Chapter 8, "Installing the Access Server"](#) and complete any additional steps in ["Setting Up ADSI on the Access Server \(Optional\)"](#) on page A-21 .

If you have Oracle Access Manager components in one forest and Oracle Access Manager data in another forest, as shown in Figure 18, before you set up the Identity System and Access System complete the task below.

7. After installation and setup, you should confirm the following parameters and values are set as shown below in the Oracle Access Manager `adsi_params` files. For example:

```
\IdentityServer_install_dir\identity\oblix\config\adsi_params.xml
\AccessServer_install_dir\access\oblix\config\adsi_params.xml

NameValPair ParamName="useDNSPrefixedLDAPPaths
Value="true"

\IdentityServer_install_dir\identity\oblix\config\adsi_params.xml

NameValPair ParamName="adsiCredential"
Value="cn=Administrator,cn=users,dc=goodwill,dc=oblix,dc=com"
```

For more information about the `adsi_params` files and parameters, see the *Oracle Access Manager Identity and Common Administration Guide*.

LDAP Open Bind Considerations

The following is intended as an overview.

LDAP open bind is the default (presumed) communication method between Oracle Access Manager and the directory server. If you are using an LDAP open bind between Oracle Access Manager components and Active Directory, you may complete some additional steps during Oracle Access Manager installation and set up.

Guidelines: Setting up an LDAP open bind

1. Before you install Oracle Access Manager, you need to set up Active Directory as described in the Microsoft documentation and "Setting Up Your Environment" on page 395.
2. During Identity System installation and setup, be sure to indicate that there is no SSL connection between Oracle Access Manager and Active Directory.
3. After Access Server installation, you are prompted to specify failover information and timeouts for LDAP and the port number.
 - Be sure to configure timeouts for the Access Server when it is installed against Active Directory, as described in the *Oracle Access Manager Identity and Common Administration Guide*.
 - Later, you may reconfigure failover information, as described in the *Oracle Access Manager Deployment Guide*.

LDAP Over SSL Considerations

The following is intended as an overview.

If you are using LDAP over SSL between Oracle Access Manager components and Active Directory, you will complete additional steps during Oracle Access Manager installation and set up.

Guidelines: Setting up SSL

1. Before you install Oracle Access Manager, ensure that you have a certificate on the machine, as described in "[Setting Up Your Environment](#)" on page A-14.

2. After installation and setup, you may reconfigure communication with the directory server, as described in the *Oracle Access Manager Identity and Common Administration Guide*.

If you have AD SI enabled and you reconfigure communication with the directory server, you must also edit the `adsi_params.xml` and `adsi_params.lst` files to reset the encryption parameter to false.

Installing Oracle Access Manager with Active Directory

With the above considerations in mind, you are ready to set up your Active Directory and install Oracle Access Manager. Following discussions explain all procedures and the order in which they must be completed.

Task overview: Installing Oracle Access Manager with Active Directory includes

1. ["Setting Up Your Environment"](#)
2. ["Installing the Identity System"](#)
3. ["Setting Up the Identity System"](#)
4. ["Validating Your Identity System Setup"](#)
5. ["Installing and Setting Up the Access System"](#)

Each of the items in the task overview above may include multiple procedures. When this is the case, additional task overviews will be provided to outline the order in which procedures must be completed.

Setting Up Your Environment

The topics below outline how to set up your Active Directory before installing Oracle Access Manager components.

Task overview: Setting up your environment includes

1. ["Setting Up Domain Controllers"](#) on page A-14
2. ["Installing the Certificate Server"](#) on page A-15
3. ["Retrieving the Certificate"](#) on page A-15

Setting Up Domain Controllers

Before you install Oracle Access Manager components, you must set up the domain controller.

WARNING: If you intend to enable dynamically-linked auxiliary classes, you must raise both the domain and the forest to a Windows 2003 Server level, as described in the Microsoft documentation.

To prepare the domain controllers

1. Set up and configure a domain controller for each machine in the Active Directory forest, using the instructions from Microsoft.
2. Specify the desired method for all auxiliary classes, either dynamically-linked or statically-linked auxiliary classes, using instructions from Microsoft.

Installing the Certificate Server

When you use LDAP over SSL, you must install the Microsoft CA Certificate Server and retrieve a certificate.

Note: If you are using LDAP (the default) or ADSI, skip this discussion.

You can install the certificate server on any machine in the Active Directory forest. However, Oracle recommends that you install it on the root domain of the Active Directory forest (for example, Root_domain). When enabled, all domain controllers will automatically request a certificate and support LDAP using SSL port 636.

To set up the Microsoft CA certificate server on other domain controllers

1. Set up a policy to enable other domain controllers to automatically request certificates, using the instructions from Microsoft.
2. Set up and configure the Microsoft CA certificate server, using your Microsoft documentation.

Retrieving the Certificate

After the certificate server is setup, you must retrieve the Microsoft CA certificate from the machine where the certificate server is installed and save it on the machine where you will install the Oracle Access Manager component. For example, on the Identity Server, Root_domain.

WARNING: A certificate is required on each machine on which SSL is enabled.

Task overview: Retrieving and setting up a certificate includes

1. ["To retrieve a certificate for the intended Identity Server"](#) on page A-15
2. ["To set up the certificate"](#) on page A-16

To retrieve a certificate for the intended Identity Server

1. On the machine where you will install the Identity Server, navigate to the machine where the Microsoft CA certificate server is installed. For example:

`http://Root_domain/certsrv/`

2. Select Retrieve the CA certificate or certificate revocation list.
3. Select Base64 encoded.
4. Double-click the Download CA certificate link.
5. Select Save this file on the machine where you will install the Identity Server.
6. Enter a directory and file name.
7. Record the full path to this file so you will have it when you install the Identity Server. For example:

`F:\OracleAccessManager\certnew.cer`

You are ready to install the Identity System. See also, ["To set up the certificate"](#) on page A-16.

To set up the certificate

1. Navigate to the certificate server. For example:

`http://Root_domain/certsrv/`

2. Download the certificate chain to a file and save the certificate.

When you import the file and store it on the local computer, as described next, IE imports the CA to the personal certificate store of the user who is currently logged in by default.

3. Import the file to a trusted root CA store on the local computer. For example,:

Select Internet Explorer, Tools, Internet options

Select Content, Certificates, Trusted Root Certificate, CertName, Import, Next

Click Browse, Filename, then click Next

Trusted Root Certification Authorities, Local Computer

Installing the Identity System

After the preliminary set up of Active Directory, you need to install the Identity Server and WebPass, the two main components of the Identity System.

Task overview: Installing the Identity System with Active Directory includes

1. ["Installing the Identity System"](#) on page A-16
2. ["Setting Up ADSI \(Optional\)"](#) on page A-17

Installing the Identity System

During installation and setup, you will be asked to respond to the same questions as those who install Oracle Access Manager with other directory servers. In addition, you will be asked to specify options for ADSI and dynamically-linked auxiliary classes.

To install the Identity Server

1. Review ["Installation and Setup Considerations for Active Directory"](#) on page A-7 and ["Data Storage Requirements"](#) on page 2-21.
2. Complete ["Setting Up Your Environment"](#) on page A-14, as needed.
3. Follow the instructions in [Chapter 4, "Installing the Identity Server"](#), and include specifications and preferences for your Active Directory environment.

The bind DN you specify can be any user as long as there are sufficient privileges to modify/read the attributes and access the schema, including changing password permissions if using LDAP/SSL. With ADSI, the Identity Server service credentials need to be appropriate.

4. To expand large dynamic groups on Active Directory, if desired, add the following to the `globalparams.xml` file, then restart the Identity Server:

```
\IdentityServer_install_dir\identity\oblix\apps\common\bin\globalparams.xml
<SimpleList >
<NameValPair ParamName="maxForRangedMemberRetrieval" Value="1500"/>
</SimpleList>
```

To complete Identity System installation

1. Follow the instructions under [Chapter 5, "Installing WebPass"](#).

2. Complete the appropriate task below, depending upon your environment:
 - Either set up ADSI, as described under ["Setting Up ADSI \(Optional\)"](#) on page A-17.
 - Or complete the Identity System set up, as described under ["Setting Up the Identity System"](#) on page A-17.

Setting Up ADSI (Optional)

If you want to use optional ADSI, you need to complete the steps below:

- Immediately after Identity System installation (Identity Server and WebPass)
- Before setting up the Identity System

By default, ADSI uses an implicit bind. This corresponds to the Windows 2000 Server and Windows Server 2003 service logon credentials. For more information, see, ["ADSI Option Considerations"](#) on page A-10 and the *Oracle Access Manager Identity and Common Administration Guide*.

To set up ADSI, before you set up the Identity System

1. Choose the proper bind mechanism for your environment in the `\IdentityServer_install_dir\identity\oblix\config\adsi_params.xml`. For example:
 - **ADSI with a Single Forest—**

```
<NameValPair ParamName="useImplicitBind"
Value="0" />
```
 - **ADSI when Oracle Access Manager and Data are in Different Forests—**

```
<NameValPair ParamName="useImplicitBind"
Value="1" />
<NameValPair ParamName="useDNSPrefixedLDAPPaths"
Value="true">
```
2. **When useImplicitBind=0—**Set the service logon credentials for the Identity Server to an administrative user in the domain with the same privileges as the user you designated as the root bind DN during Identity Server installation.
3. Set up the Identity System as described under ["Setting Up the Identity System"](#) next.

Setting Up the Identity System

After you install the Identity Server and WebPass components, you are ready to set up the Identity System.

Discussions below explain what to do before and during Identity System set up to ensure success. Several situations are covered:

- [Enabling Active Directory Attributes](#)

Note: If you do not want to enable specific Active Directory attributes, go directly to ["Setting up the Identity System"](#) on page 6-4.

- [Enabling Change-Password Permissions](#)
- [Setting Up the Identity System](#)

Enabling Active Directory Attributes

To enable specific Active Directory attributes, you need to complete the procedure below. For example, if you want to use the userPrincipalName as a login attribute and you want this attribute to be available during Identity System setup, complete the activities below before Identity System setup.

Note: If you do not want to enable specific Active Directory attributes, skip this task and go directly to ["Setting Up the Identity System"](#) on page A-18.

To enable Active Directory attributes

1. Locate the ad_exclude_attrs.xml and exclude_attrs-ad.xml files. For example:

```
\IdentityServer_install_dir\identity\oblix\data.ldap\common\ad_exclude_attrs.xml
```

```
\IdentityServer_install_dir\identity\oblix\data.ldap\common\exclude_attrs-ad.xml
```

2. Edit the files to make specific Active Directory attributes available within Identity System user profiles. For example:

```
<ValNameList ListName="userPrincipalName">  
<NameValPair ParameterName="appliesto" Value="None" />
```

To modify your schema for statically-linked auxiliary classes

1. Modify your schema to attach the oblixOrgPerson and oblixPersonPwdPlicy objects to your user object class and oblixGroup and oblixAdvancedGroup to your Group object class.
2. Perform a schema reload within the MMC Schema Manager Application after making the changes above and allow approximately fifteen minutes for the schema changes to be replicated to all domain controllers.

Enabling Change-Password Permissions

When Oracle Access Manager data and components are in the same forest, you must run the Identity Server in the context of a privileged administrative user who has change-password permissions.

Note: With LDAP over SSL, you do not need to set service credentials for the Identity Server.

Setting Up the Identity System

After completing the tasks above, you are ready to set up the Identity System for the Active Directory forest, using the Root_domain.

To set up the Identity System for an Active Directory forest

1. Navigate to the Identity System set up page:
`http://hostname:port/identity/oblix`
2. Click Identity System Console then click Setup to activate the process.
3. Follow the instructions in [Chapter 6, "Setting Up the Identity System"](#).

- Enable ADSI, if appropriate.
 - Check the Dynamic Auxiliary Object Class box, if appropriate.
4. When setup is complete, you can perform the tasks outlined below:
- Validate your Identity System setup, as described in "[Validating Your Identity System Setup](#)" on page A-19.
 - Define directory server profiles for remaining domains, if needed, as described in the *Oracle Access Manager Identity and Common Administration Guide*.

Note: If you are using ADSI, see the *Oracle Access Manager Identity and Common Administration Guide* for details about enabling ADSI for the Default Directory Profile and additional directory profiles.

- Set up disjoint searchbases for the disjoint domain, if needed, as described in the *Oracle Access Manager Identity and Common Administration Guide*.
- Install and setup the Access System, as described in "Installing and Setting Up the Access System" on page 401.

Validating Your Identity System Setup

It is a good idea to validate that your Identity System is set up and properly operating with Active Directory before you begin installing the Access System.

To validate your Identity System setup

1. Navigate to the Identity System login page.
`http://hostname:port/identity/oblix`
2. Verify that all domain names appear in the drop-down list on the login page.
3. Log in.
4. Navigate to the Configure Directory Options page and confirm that your disjoint searchbase is listed.

From the Identity System Console select System Admin, System Configuration, Directory Options

Note: If your disjoint searchbase is not listed, you can add it now. For more information, see the *Oracle Access Manager Identity and Common Administration Guide*.

After validating that your Identity System is working properly, you can install and set up the Access System.

Installing and Setting Up the Access System

Refer to topics below as you install the Access System with your Active Directory forest.

- [Preparing for Access System Installation](#)
- [Installing and Setting Up the Access System](#)
- [Setting Up ADSI on the Access Server \(Optional\)](#)

Preparing for Access System Installation

Be sure to verify that the steps below have been completed before you attempt to install the Access System.

To prepare for the Access System installation and setup

1. Review "[Installation and Setup Considerations for Active Directory](#)" on page A-7.
2. Install the Identity System, as discussed in "[Installing the Identity System](#)" on page A-16.
3. Set up the Identity System, as discussed in "[Setting Up the Identity System](#)" on page A-17.
4. Validate your Identity System, as discussed in "[Validating Your Identity System Setup](#)" on page A-19.

WARNING: To prepare the Policy Manager host for ADSI when the machine is not a domain controller and resides in the same forest with other Oracle Access Manager data, ensure that `useImplicitBind=1` or `2` and `useDNSPrefixedLDAPPaths=true`.

Installing and Setting Up the Access System

Use the steps below as a guide when you install and set up the Access System in this environment.

- [To install and set up the Policy Manager](#)
- [To install and set up the Access Server and WebGate3](#)

To install and set up the Policy Manager

1. Confirm that you have a certificate on each machine, if needed, for ADSI or LDAP over SSL before you begin Oracle Access Manager installation.

In the next step, use the same directory server details that you used for the Identity Server installation **only** if your configuration data, user data, and Oracle Access Manager policy data will reside on the same directory server. Also, ensure that the distinguished name specified as the bind DN has full permissions for the user and configuration branches of the directory information tree (DIT).

2. Install the Policy Manager, as described in "[Installing the Policy Manager](#)" on page 7-2.
 - Select Active Directory using ADSI, if appropriate.
 - Select dynamically-linked auxiliary classes, if appropriate.
3. Set up the Policy Manager as described in "[Setting Up the Policy Manager](#)" on page 7-9 and consider the following:
 - With ADSI, select Enable ADSI and enter the `userPrincipleName` as the bind DN (for example, `user@company.com`), then complete set up.
 - With LDAP open bind, see the *Oracle Access Manager Access Administration Guide* to complete Policy Manager set up.

WARNING: You complete the steps below only when Oracle Access Manager server components reside in one forest and data in another.

4. Verify that the useDNSPrefixedLDAPPaths value in the `\PolicyManager_install_dir\access\oblix\config\adsi_params.lst` file is set to true. For example:

```
<NameValPair
ParamName="useDNSPrefixedLDAPPaths"
Value="true" />
```

5. Verify the forceExplicitBindUsingDN and set its value to true in the `\PolicyManager_install_dir\access\oblix\apps\common\bin\globalparams.lst` file. For example:

```
forceExplicitBindUsingDN:true
```

6. Restart the Identity Server and WebPass Web server.

To install and set up the Access Server and WebGate

1. Confirm that you have a certificate on each machine, if needed, for ADSI or LDAP over SSL before you begin Oracle Access Manager installation.
2. Install the Access Server, as described in "Installing the Access Server" on page 187 and consider the following items:

- Select Active Directory using ADSI, if appropriate, enter the adsiCredential and adsiPassword when prompted, then complete "Setting Up ADSI on the Access Server (Optional)" on page 404 *before* restarting the Access Server.

adsiCredential and adsiPassword are required to generate an encrypted password that can be used when explicitly binding to Active Directory using ADSI. Because Oracle Access Manager does not include an encryption tool, you must enter values for adsiCredential and adsiPassword when asked.

Select dynamic auxiliary classes, if appropriate.

Note: You will be asked where the user data, configuration data, and policy data are stored and for configuration details for the directory server.

You need to complete step 3 when you have Active Directory 2000, because it does not support concurrent bind requests coming from different Oracle Access Manager threads on the same LDAP connection. For more information, see "[Active Directory Tips and Troubleshooting](#)" on page A-22.

3. **Active Directory 2000**—On the Access Server, open the `globalparams.lst` file and add a new flag called `exclusiveAuthnConnection` set to true to force Oracle Access Manager threads to use separate LDAP connections for bind requests being sent to the directory server.
4. **WebGate**—Install and configure the WebGate, as described in [Chapter 9, "Installing the WebGate"](#).

See the *Oracle Access Manager Identity and Common Administration Guide* for more information about authentication and authorization with Active Directory and configuring Oracle Access Manager for specific Active Directory features.

Setting Up ADSI on the Access Server (Optional)

If you choose to use ADSI, which is optional, you must set up ADSI on the Access Server:

- After installing the Access Server
- Before restarting the Access Server

To set up ADSI on the Access Server

1. Log in to the domain as an administrative user.
2. Set the service logon credentials for the Access Server to an administrative user in the domain.

This user must have the same privileges as the user that you provide in the Root DN during Policy Manager and Access Server installations.

3. Choose the proper bind mechanism in the `adsi_params.lst` file. For example, in a two forest configuration:

```
\AccessServer_install_dir\access\oblix\config\adsi_params.xml  
useImplicitBind Value="1"
```

By default, the Access Server `useImplicitBind` is set to 0 for a single-forest configuration. ADSI uses an implicit bind. This corresponds to Windows 2000 Server, or Windows Server 2003 service logon credentials. See the *Oracle Access Manager Identity and Common Administration Guide* for more information on ADSI bind mechanics.

You complete step 4 and step 5, below, as needed for your environment.

4. When you have Oracle Access Manager components installed outside the Active Directory forest, you need to verify the `useDNSPrefixedLDAPPaths` parameter value in the `adsi_params.lst` is set to `true`. For example:

```
useDNSPrefixedLDAPPaths Value="true"
```

5. When you have Oracle Access Manager components installed in one forest and data in another, set the `forceExplicitBindUsingDN` parameter value to `true` in the `globalparams.lst` file. For example:

```
\AccessServer_install_dir\access\oblix\apps\common\bin\globalparams.lst  
forceExplicitBindUsingDN Value="true"
```

6. See the *Oracle Access Manager Identity and Common Administration Guide* for more information about authentication and authorization with Active Directory and configuring Oracle Access Manager for Active Directory features.

Active Directory Tips and Troubleshooting

For information, see "[Active Directory Issues](#)" on page E-3.

Installing Oracle Access Manager with ADAM

Oracle Access Manager supports the Microsoft Active Directory Application Mode (ADAM) as a standalone directory server. This chapter includes the following discussions:

- [About Oracle Access Manager and ADAM](#)
- [ADAM and Active Directory Differences](#)
- [Support Requirements](#)
- [Installing Oracle Access Manager with ADAM](#)
- [Oracle Access Manager Silent Mode Installation Parameters](#)
- [Troubleshooting ADAM Issues](#)

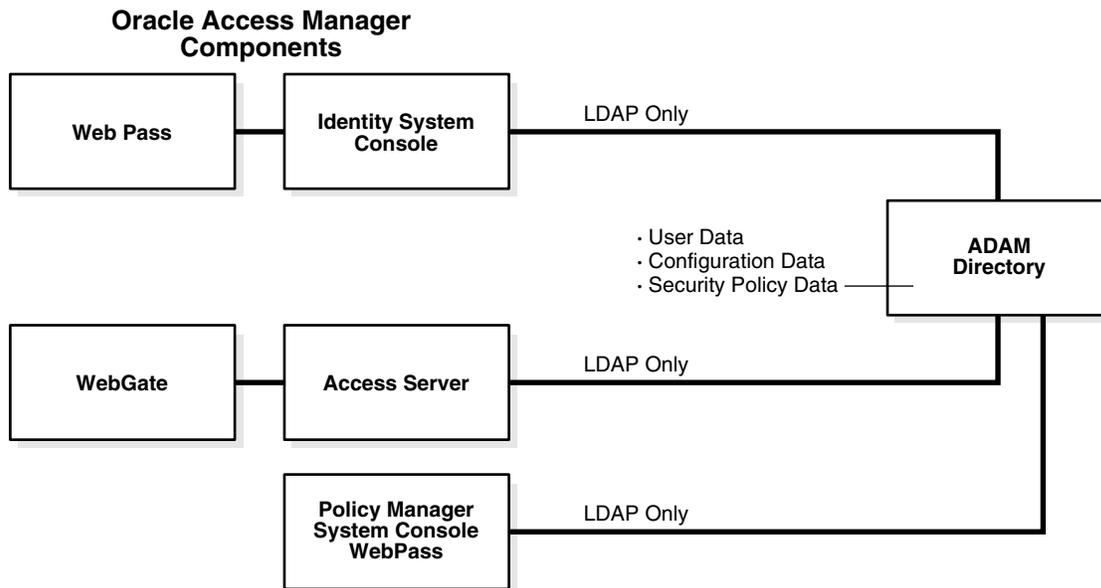
Upgrading to Oracle Access Manager 10g (10.1.4.0.1) is explained in the *Oracle Access Manager Upgrade Guide*.

About Oracle Access Manager and ADAM

This discussion introduces ADAM in general terms and provides details about using ADAM as a directory server for Oracle Access Manager. Differences between ADAM and Active Directory are also discussed.

Note: Oracle Access Manager supports storing user data on a separate type of directory server and storing Oracle Access Manager configuration and policy data on one or more instances of ADAM. This means that, for example, you may store user data on Active Directory and configuration and policy data on ADAM. Configuration and policy data must be stored on the same directory server type.

Whether you install Oracle Access Manager on a single machine with ADAM or in a distributed environment, as shown in [Figure B-1](#), Oracle Access Manager supports ADAM as a standalone directory server.

Figure B-1 ADAM as a Standalone Directory Server for Oracle Access Manager

ADAM uses the same storage management and the same programming model as the .NET Active Directory. In addition, ADAM provides a similar replication and administration model as Active Directory. However, ADAM is independent of Active Directory and Active Directory domains and forests. ADAM does not include Active Directory infrastructure features, does not include directory services for the Windows operating system, and does not require a domain controller. ADAM runs as an independent service, not an operating system service.

ADAM typically provides dedicated directory services for applications, including a data store and services to access the data store. For example, ADAM can provide an application-specific directory store for a small business unit. The information in ADAM may require specific local schema changes, may be relevant to only a small group of users, and may not require wide distribution.

During installation of each unique ADAM instance, you specify a name and port for the instance. The name ties the files, service, registry, and ports together. Ports may be configured for LDAP and SSL. An open LDAP port is required to extend the ADAM schema with Oracle Access Manager-related information. There have been no changes to the Oracle Access Manager schema for ADAM.

For more information, see:

- [ADAM Instances and Partitions](#)
- [The ADAM Schema](#)
- [The Oracle Access Manager Schema Extension for ADAM](#)
- [Windows Users and Security Principals](#)
- [Oracle Access Manager Directory Profiles](#)
- [Replication of an ADAM Instance](#)
- [ADSI with Oracle Access Manager and ADAM](#)
- [ADAM and APIs](#)

- [Authentication, Authorization, and Password Changes](#)

ADAM Instances and Partitions

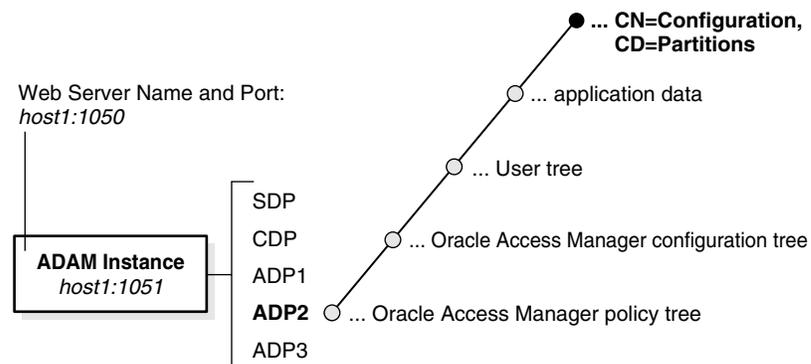
During installation of a unique ADAM instance, a schema directory partition (SDP) and configuration directory partition (CDP) are created. With ADAM, there is no domain partition. Each unique instance can be configured independently and may include multiple application directory partitions (ADPs) created either during ADAM installation and setup or later.

Important: Be sure to create your application directory partitions within ADAM before installing Oracle Access Manager. Oracle Access Manager does not create an ADP within ADAM.

ADAM ADPs are similar to Active Directory ADPs. Each ADAM ADP contains the data (objects) for an ADAM instance. However, ADAM ADPs cannot store a security principal (an account holder that is automatically assigned a security identifier (SID) to control access to resources). Applications and services can use ADAM ADPs to store application-specific data, which may contain highly volatile information with high replication requirements that could strain resources if stored in your Network Operating System (NOS) directory.

Originally Oracle Access Manager supported a single ADP within a single ADAM instance. As shown in [Figure B-2](#), an ADP can include application-specific data as well as the user tree, the Oracle Access Manager configuration tree, and the Oracle Access Manager policy tree.

Figure B-2 *Single ADAM Instance and Partitions*



Note: Oracle Access Manager supports multiple ADPs within a single instance and multiple instances. Oracle Access Manager will support a master instance and its replicas.

Also: Oracle Access Manager requires a node with the objectclass attribute value of organizationalUnit for the configuration and policy DN. When you create an ou, this is added by default.

Note: Oracle Access Manager supports multiple ADAM instances and multiple ADPs. Now, user data may be stored on a different directory server type. For example, Oracle Access Manager configuration and policy data may be stored on ADAM in a single ADAM ADP or instance or on different ADAM ADPs or instances while user data is stored on Active Directory.

The ADAM Schema

ADAM, like Active Directory on Windows Server 2003 platforms, supports dynamically-linked auxiliary classes. Oracle Access Manager supports both dynamically-linked and statically-linked auxiliary classes.

The ADAM schema contains definitions of the object classes that ADAM can access within a configuration set. The schema also includes definitions of the attributes that ADAM can access in an ADAM object. For more information about configuration sets, see "[Replication of an ADAM Instance](#)" on page B-7.

The ADAM schema is flexible. There are no namespace restrictions. ADAM can use X.500-style naming contexts (o=,c=) for various types of information (schema, sites, partitions, and services). Within the ADP, the user searchbase, configuration DN, and policy base may be the same (o=company,c=us) or may differ.

Note: Oracle Access Manager requires a node with the objectclass attribute value of organizationalUnit (ou) for the configuration and policy DNs. When you create an ou, this is added by default.

An example of *different* name spaces is shown:

Searchbase:o=company,c=us
Configuration DN:ou=config,o=company,c=us
Policybase:ou=policy,o=company,c=us

Note: When storing user data on a separate type of directory server and Oracle Access Manager configuration and policy data on one or more ADPs or one or more ADAM instances, different name spaces are required.

While the ADAM schema is similar to the Active Directory schema, the user object class is described differently in ADAM than in Active Directory. There is no security principal attached to ADAM. For example, *saMAccountName* is mandatory with Active Directory for user and group but does not exist in ADAM. However, *groupType* is still required.

The *groupType* attribute in the ADAM group object class "group" can have only the following values, which should be configured in the meta-attribute configuration applet (Identity System Console, Group Manager, Configure Tab, Modify Attributes) for the object class with a Display Type of radio button:

global - 2
domain local - 4
universal - 8
secure domain - -2147683644
secure global - -2147482646

In Active Directory, the password attribute is unicodePwd. The password attribute on ADAM is *userpassword*. The *uid* attribute is assigned the Semantic Type "Login" by default.

The Active Directory Application Mode (ADAM) schema is extensible using the *Ldifde.exe* command-line tool.

The Oracle Access Manager Schema Extension for ADAM

The Oracle Access Manager schema extension for ADAM must be loaded using a Windows Security Principal credential. At runtime, however, Oracle Access Manager communicates only with users within ADAM, not with security principals. For more information, see "[Windows Users and Security Principals](#)" on page B-6.

When you install Oracle Access Manager, you must manually update the ADAM schema. If the user data directory instance is separate from the configuration and policy data directory instance, you must manually upload the *ADAM_user_schema_add.ldif* file.

On the configuration data directory instance, you must manually upload the *ADAM_oblix_schema_add.ldif* file. When using static auxiliary classes, you must manually upload the *ADAMAuxSchema.ldif* file.

If the policy data directory instance is separate from the configuration data directory instance, you must manually upload the *ADAM_oblix_schema_add.ldif* file. When using static auxiliary classes, you must manually upload the *ADAMAuxSchema.ldif* file.

Oracle Access Manager supports both *InetOrgperson* and *GroupofUniqueNames* as standard Person and Group object classes, respectively, in addition to user and group. You may have an object class already in use and do not need to use a specific object class. Oracle Access Manager also supports both statically-linked and dynamically-linked auxiliary classes

The ADAM schema cannot be modified with a simple LDAP bind and must be modified using *Ldifde*, *not* *Ldapmodify*. Currently, *Ldifde* does not support binding to an SSL port on ADAM; therefore, the ADAM schema may be extended for Oracle Access Manager only on an open port. During Identity Server installation, you can specify an SSL connection and obtain the SSL certificate for ADAM, then specify an open port number for the schema update.

Note: With ADAM the schema update must be completed using an open port and a Windows security principal credential.

Oracle Access Manager provides the following schema files for Oracle Access Manager configuration and user directories. To update the schema manually, you must use the following files:

```
IdentityServer_install_dir\identity\oblix\data.ldap\common\  
ADAM_oblix_schema_add.ldif  
ADAM_user_schema_add.ldif
```

In addition, if you are using statically-linked auxiliary classes, you also need to run the *Ldifde* command with the following file:

```
IdentityServer_install_dir\identity\oblix\data.ldap\common\  
ADAMAuxSchema.ldif
```

A sample `ldifde` command to manually update the schema follows and is also described in [Table B-1](#). For more information, see your Microsoft documentation:

```
ldifde -k -b
"<user_distinguished_name>" "<domain_name>" "<user_password>"
-c "<GUID>" <ADAM_instance_ID> -i -f ADAM_oblix_schema_add -s
<ADAM_server_name> -t <port>
```

Table B-1 *ldifde* Command Description for ADAM

Option	Description
-k	This option ignores errors.
-b "<user_distinguished_name>" "<domain_name>" "<user_password>" For example: cn=administrator,o=oblix.com,c=us password	To extend the schema, the values represent: <ul style="list-style-type: none"> ▪ a Windows security principal user name ▪ domain name of the machine where ADAM is installed ▪ password
-c "<GUID>" <ADAM_instance_ID>	In this option, "<GUID>" should be retained as is, not replaced by any value; do include the quotes. <ADAM_instance_ID> should be substituted by the ADAM root DSE using tools like <code>ldp.exe</code> . When the initial connection is made, the root DSE is shown. For example, an ADAM root DSE value may be EC31B31B-19FC-4FD4-8590-3BD57D6A3E77.
-i -f <filename>	The -i option specifies the import option. The -f option identifies a file name; the value identifies the file you are importing. For example: ADAM_oblix_schema_add.ldif ADAMAuxSchema.ldif
-s <ADAM_server_name>	This value is the name of the machine where ADAM is installed.
-t <port >	This value is the port number on which this instance listens for the schema update (an open port is needed).

Windows Users and Security Principals

ADAM supports user credentials and uses Windows security principal credentials for authentication and access control. For example, the Windows security principal provides the rights to define users and replicate an instance of the ADAM directory store. However, Oracle Access Manager requires Windows security principal credentials only to update the ADAM schema.

Windows Security Principal for Schema Updates: When you install the Oracle Access Manager Identity Server with ADAM and update the schema, you must supply directory server details as follows:

- **Automatic Schema Updates:** Not available. You must manually update the schema.
- **Manual Schema Updates:** When you manually extend the schema, you must supply the Windows security principal name and password with the `ldifde` command as discussed in "[The Oracle Access Manager Schema Extension for ADAM](#)" on page B-5. For example:

```
-b "<user_distinguished_name>" "<domain_name>" "<user_password>"
```

Windows User within ADAM for Root (Bind) DN: At runtime, Oracle Access Manager communicates only with users within ADAM, not with a Windows security principal. During Identity System setup you must specify the Root (bind) DN for

ADAM and password for that user on the page where you specify Directory Server with User Data Configuration. This must be the name of a bindable user within ADAM with administrator privileges:

Root DN: Name of a *bindable* user within ADAM with administrator privileges

Root DN Password of the *bindable* user within ADAM with administrator privileges

You create a bindable user in ADAM by adding the ms-bindable-object auxiliary object class to the object class you are using for people objects, inetOrgPerson, for example.

The Oracle Access Manager Administrator must be a bindable user in ADAM with administrative privileges, *not* a Windows Security Principal.

Oracle Access Manager Directory Profiles

When you setup Oracle Access Manager, individual directory profiles are created for the Identity Server, Policy Manager, and Access Server, as usual. During Identity System setup, you specify the SSL-enabled port to properly configure the directory profile within Oracle Access Manager.

For details about configuring directory profiles after Oracle Access Manager installation, see the *Oracle Access Manager Identity and Common Administration Guide*.

Replication of an ADAM Instance

Replication of an ADAM instance creates a configuration set. All ADAM instances within a configuration set replicate a common schema partition and configuration partition, and can also replicate ADPs such as o=company,c=US. Only complete replicas are supported by Oracle Access Manager.

Oracle Access Manager will provide failover and load balancing between a master instance and its replicas; however, Oracle Access Manager does not support ADSI with ADAM. For more information, see "[ADSI with Oracle Access Manager and ADAM](#)" on page B-7.

Typically, multiple instances of ADAM may run concurrently on a single server, each with its own schema and configuration.

Note: You cannot replicate ADAM instances across the forest. In a production environment, ADAM instances within the same configuration set cannot reside on the same machine. See your Microsoft documentation for more information.

ADSI with Oracle Access Manager and ADAM

ADSI provides failover support in Active Directory environments and Oracle Access Manager supports ADSI with Active Directory. ADAM supports Active Directory Service Interfaces (ADSI). However, Oracle Access Manager does *not* support ADSI with ADAM.

With ADAM there is no domain controller, therefore, the native Oracle Access Manager directory server failover and connection management toolkits are recommended. For details about configuring failover and load balancing, see the *Oracle Access Manager Deployment Guide*.

ADAM and APIs

ADAM uses standard application programming interfaces (APIs) to access application data. These include Active Directory APIs, Lightweight Data Access Protocol, and System-Directory Services.

ADAM does *not* support the Messaging Application Programming Interface (API). Therefore, Microsoft Exchange cannot use ADAM. For more information, see your Microsoft documentation.

Authentication, Authorization, and Password Changes

Authentication and authorization processes should be managed in Oracle Access Manager, rather than in ADAM. This will avoid contentions between Oracle Access Manager "rules" and ADAM "rules" regarding authentication and authorization. Oracle Access Manager authentication and authorization processes are the same whether you are using ADAM or Active Directory. For more information, see the *Oracle Access Manager Identity and Common Administration Guide*.

With Oracle Access Manager, ADAM cannot use proxy objects that point to Active Directory. Users must be enabled within ADAM and must have a password.

With Active Directory and Oracle Access Manager, you can use native password management or Oracle Access Manager. Password management is available with ADAM. Both Active Directory and ADAM support changing passwords over a secure connection, either SSL or ADSI. However, Oracle Access Manager does not support ADSI with ADAM, therefore, SSL must be used for password changes with ADAM.

ADAM and Active Directory Differences

Differences between ADAM and Active Directory are summarized in the following table:

Table B-2 Differences between ADAM and Active Directory with Oracle Access Manager

Description
ADAM and Active Directory can operate concurrently in the same network; Oracle Access Manager supports the use of both independently and together.
The information in ADAM may require specific local schema changes, may be relevant to only a small group of users, and may not require wide distribution.
Applications and services can use ADAM ADPs to store application-specific data, which may contain highly volatile information with high replication requirements.
There is no security principal attached to ADAM. For example, saMAccountName is mandatory with Active Directory for user and group but does not exist in ADAM.
The password attribute on ADAM is userpassword.
Oracle Access Manager requires Windows security principal credentials to update the ADAM schema. At runtime, Oracle Access Manager communicates only with users within ADAM, not with security principals.
Both Active Directory and ADAM support ADSI. However, Oracle Access Manager supports ADSI only with Active Directory. An implicit bind is available only with Active Directory, not ADAM.
Both Active Directory and ADAM support changing passwords. Oracle Access Manager with ADAM requires an SSL-enabled port for password changes.

For more information about ADAM, see your Microsoft documentation.

Support Requirements

ADAM and Active Directory can operate concurrently within the same network. Oracle Access Manager supports the use of ADAM alone. In addition, Oracle Access Manager supports ADAM with Active Directory and ADAM with other directory server types for storing user data separately from configuration and policy data:

- All user data must be stored on the same directory server type.
- Configuration and policy data must be stored on the same type of directory server.
- Oracle Access Manager requires a node with the objectclass attribute value of organizationalUnit (ou) for the configuration and policy DNs.

For LDAP support with Oracle Access Manager 10g (10.1.4.0.1), see the Certify tab at <https://metalink.oracle.com>.

The schema update must be performed manually using the ldifde command. For more information, see "The Oracle Access Manager Schema Extension for ADAM" on page B-5.

Installing Oracle Access Manager with ADAM

The following tasks are included in the installation procedures:

Task overview: Installing Oracle Access Manager with ADAM

1. Prepare your environment, as described in [Part I, "Installation Planning and Prerequisites"](#).
2. Prepare ADAM, as described in "[Preparing ADAM for Oracle Access Manager](#)" on page B-9.

Note: Oracle Access Manager requires a node with the objectclass attribute value of organizationalUnit (ou) for the configuration and policy DNs.

3. Install and setup the Identity System, as described in "[Installing and Setting the Identity System with ADAM](#)" on page B-11.
4. Install and setup the Access System, if this is part of your environment, as described in "[Installing the Access System with ADAM](#)" on page B-14.
5. Complete any of the following activities after successful Oracle Access Manager installation and setup:
 - Add ADAM users to Oracle Access Manager, as described in the *Oracle Access Manager Identity and Common Administration Guide*.
 - Replicate an ADAM instance, as described in your Microsoft documentation.
 - Configure failover and load balancing, for the ADAM master and replicas, as described in the *Oracle Access Manager Deployment Guide*.

Preparing ADAM for Oracle Access Manager

The steps that follow outline how to prepare your ADAM instance and ADP so that Oracle Access Manager can manage authentication and authorization. You will install a unique ADAM instance, create an ADP and a top DN for user data, and create users in ADAM. Remember the following important points:

ADPs: You must create the application directory partition within ADAM. Oracle Access Manager does *not* create the ADP.

Administrators: At least one account should be designated as the ADAM instance administrator. An ADAM instance administrator should also be designated as a Master Administrator during Identity System setup. The Master Administrator must be a bindable user in ADAM with administrative privileges, *not* a Windows Security Principal.

Configuration and Policy DNs: Oracle Access Manager requires a node with the objectclass attribute value of organizationalUnit (ou) for the configuration and policy DNs.

ADSI: Oracle Access Manager does not support ADSI with ADAM. You will need to use native Oracle Access Manager directory server failover and connection management toolkits with ADAM.

Binding through Proxy Objects: Oracle Access Manager does *not* support binding through an ADAM proxy object.

Note: Failure to complete the following steps may result in an unsuccessful installation with Oracle Access Manager. For complete details about installing ADAM, setting up the instance, and other tasks, as well as details about tools such as ADAM ADSI Edit and Ldp.exe, see the Microsoft documentation that accompanies your ADAM download.

To install ADAM for Oracle Access Manager

1. Familiarize yourself with ADAM concepts, practices, and tools, as described in the Microsoft documentation that accompanies your ADAM download.
2. Install a unique ADAM instance on a machine with Windows Server 2003 by running ADAMSetup.exe from the ADAM installation directory.

The installation program will prompt you with the following screens:

- a. A unique instance
 - b. A valid Instance name (for example, OracleAccessManager)
 - c. Port number where you want this instance to run.
 - d. Creation of application directory partition. (Yes/No).
 - * Yes to create a new partition For example, o=company, c=us, (Default)
 - * No will refer it to an already existing partition.
 - e. Directory to install ADAM.
 - f. Service Account selection (Select an account you would like to use for further processing).
 - * Network Service Account (Default)
 - * Custom Account (Make sure this account is active)
 - g. Assign Permissions to the selected account
 - h. Import user LDIF file (for example, MS_User.ldf)
3. Be sure to:

- a. Specify an open LDAP port number to extend the ADAM schema for Oracle Access Manager, and an SSL-enabled port for password changes, authentication, and authorization with Oracle Access Manager.
 - b. Create an Application Directory Partition (naming context) to contain user data and Oracle Access Manager configuration and policy data by specifying any distinguished name that does not already exist within the instance--or to contain configuration and policy data while storing user data on another directory server type.
4. Start the ADAM instance.
For example: Start, Programs, ADAM, ADAM ADSI EDIT
 5. Right-click ADAM ADSI EDIT then select Connect to from the menu.
A user screen appears with the following options:
 - a. **Connection Name:** For example, OAM
 - b. **Host Name:** Local Host
 - c. **Port:** Port number of the instance you have created
 - d. **DN:** The bind DN
 - e. **Credentials**

Note: The ms-bindable-object should be added to ADAM. For more information, see "[Windows Users and Security Principals](#)" on page B-6.

6. Create and enable a bindable ADAM user account and use ADAM ADSI Edit to add the user you want to designate as the Master Administrator to the member attribute of the following:
`CN=Administrators,CN=Roles,CN=Configuration,CN={your GUID}`
7. Reset the user password.
8. Activate the user.
9. Manage directory partitions in the ADAM instance.
10. Manage ADAM configuration sets.
11. Ensure that your ADAM installation is operating properly before you continue.

Installing and Setting the Identity System with ADAM

Procedures in this discussion presume that you have completed all steps in "[Preparing ADAM for Oracle Access Manager](#)" on page B-9. Following are several important items to review before you begin:

Schema Update: The ADAM schema update for Oracle Access Manager must be completed using an open port. For more information, see "[The Oracle Access Manager Schema Extension for ADAM](#)" on page B-5.

The schema update must be completed with a Windows security principal credential. However, the root (bind) DN you specify during Identity System setup must be a user with an explicit physical location within ADAM. For more information, see "[Windows Users and Security Principals](#)" on page B-6.

Identity System Setup: During Identity System setup, an SSL-enabled connection should be specified for password changes. For more information, see "[Authentication, Authorization, and Password Changes](#)" on page B-8.

Administrators: The Master Administrator you designate during Identity System setup must be an ADAM user with administrative privileges, not a Windows Security Principal.

The steps that follow provide specific information for Oracle Access Manager and ADAM. For additional information about installing each Oracle Access Manager component, see chapters elsewhere in this guide.

To install the Identity Server and update the ADAM schema

1. Start the installation by selecting the Identity Server installation package that you downloaded.
2. Supply your installation directory, transport security mode, and Identity Server configuration details for Oracle Access Manager.
3. Select Yes to use SSL between the Identity Server and ADAM (required for password changes) then respond to all questions about certificates. Later you may specify an open port to extend the ADAM schema for Oracle Access Manager.
4. Supply the following details for ADAM:
 - a. **Directory Server Type:** Select Active Directory Application Mode from the Directory Server drop-down list to specify ADAM.
 - b. **Data Location:** Identify whether configuration data and user data are stored separately.
 - c. **Schema Update:** Review the instructions to manually update the schema when these appear.

You continue the installation process, then update the schema as described in steps 6 and 7.
5. Finish the installation and start the Identity Server, as described in the "[To install the Identity Server and update the ADAM schema](#)" on page B-12.
6. **Manual Schema Update Preparation:** Modify the following files to replace <guid> with {your GUID} before you run a manual schema update using step 7:
 - ADAM_oblix_schema_add.ldif
 - ADAM_oblix_user_schema_add.ldif
 - ADAMAuxSchema_add.ldif
7. **Manual Schema Updates:** Update the schema manually as a domain user if needed, using the appropriate file and ldifde command, then restart the Identity Server. For example:

```
IdentityServer_install_dir\identity\oblix\data.ldap\common\
```

```
ADAM_oblix_schema_add.ldif
ADAM_oblix_user_schema_add.ldif
```

```
ldifde -k -b <cn=admin, o=company, c=us password> -c "<GUID>"
<ADAM_instance_ID> -i -f ADAM_oblix_schema_add.ldif -s <ADAM_server_name>
-t <port>
```

```
ldifde -k -b <cn=admin, o=company, c=us password> -c "<GUID>"
<ADAM_instance_ID> -i -f ADAM_oblix_user_schema_add.ldif
```

```
-s ADAM_server_name -t <port>
```

Note: The Windows security principal name and domain in the preceding example are samples only. Your environment will differ.

After executing the preceding command, if you do *not* plan to use dynamic auxiliary classes:

- Use the `ldifde` command to import the Oracle Access Manager schema file `ADAMAuxSchema.ldif` for statically-linked auxiliary classes from the `IdentityServer_install_dir\identity\oblix\data.ldap\common` directory.
- Ensure that the object classes "oblixorgperson" and "oblixgroup" are explicitly attached as auxiliary classes to the Person and Group object classes, respectively.

Note: Be sure to restart the Identity Server after updating the schema manually.

The steps that follow summarize the information you need to supply when you install the WebPass and set up the Identity System with ADAM.

To install WebPass and set up the Identity System

1. Install the WebPass you downloaded, as described in "[To install WebPass and set up the Identity System](#)" on page B-13.
2. Start the Identity System setup, as described in the "[Installing and Setting the Identity System with ADAM](#)" on page B-11, then specify the following details for ADAM:
 - a. **Directory Server Type:** Select Microsoft Active Directory Application Mode when you specify a directory server type, *and* select Dynamic Auxiliary Object Class if appropriate for your environment.
 - b. **Location of Directory Server:** Specify the following for ADAM:
 - Port Number:** Specify the port to be used during runtime (SSL is required for password changes).
 - Root DN:** The name of a *bindable* user in ADAM with administrator privileges as the bind DN; do not specify a Windows security principal.
 - Root Password:** Password for the *bindable* user in ADAM
 - Directory Server Security Mode:** Specify SSL for password changes.
3. Finish setting up the Identity System, as usual.
4. Continue with any of the following activities when Identity System setup is finished:
 - Install the Access System with ADAM, if this is included in your environment, as described in "[Installing the Access System with ADAM](#)" on page B-14.
 - Add ADAM users to Oracle Access Manager, as described in the *Oracle Access Manager Identity and Common Administration Guide*.
 - Replicate an ADAM instance, as described in your Microsoft documentation.

- Configure failover and load balancing, for the ADAM master and replicas, as described in the *Oracle Access Manager Deployment Guide*.

Installing the Access System with ADAM

The Access System, which is optional, includes the Policy Manager, Access Server, and WebGate. The steps that follow provide specific details to install and setup the optional Access System with ADAM.

For details, see the following procedures:

- [To install the Policy Manager with ADAM](#)
- [To install the Access Server](#)
- [To set up the Policy Manager with ADAM](#)
- [To install the WebGate](#)

With ADAM, policy data may be stored with user and Oracle Access Manager configuration data. Alternatively, Oracle Access Manager supports separate ADAM instances for configuration, user and policy data.

To install the Policy Manager with ADAM

1. Locate and launch the Policy Manager installation and specify your installation directory, as described in "[To install the Policy Manager with ADAM](#)" on page B-14.
2. Select Microsoft Active Directory Application Mode when asked for the directory server type.
3. Select Yes if dynamically-linked auxiliary object classes are enabled in your environment, otherwise select No.
4. Manually update the schema after installation.

Note: Automatic schema updates are not supported.

5. Specify the directory server security mode: SSL-enabled is required for password change with ADAM.
6. Specify a transport security mode for the Access System, configure your Web server, and complete the Policy Manager installation, as described in the "[To install the Policy Manager with ADAM](#)" on page B-14.
7. Continue with the next procedure to set up the Policy Manager.

To set up the Policy Manager with ADAM

1. Start the Policy Manager setup process, as described in "[To install the Policy Manager with ADAM](#)" on page B-14, and specify the following details for ADAM:
 - a. **Directory Server Type:** Select Microsoft Active Directory Application Mode when you specify a directory server type, and select Dynamic Auxiliary Object Class if this is appropriate for your environment.
 - b. **Directory Server Details:** Specify the following for ADAM:

Port Number: Specify the port to be used during runtime (SSL is required for password changes).

Root (bind) DN: Specify the Root DN you provided when setting up the Identity Server; do not use a Windows security principal.

Password: Specify the password of the bind DN user.

Directory Server Security Mode: Specify SSL for password changes.

2. Specify the searchbase, configuration DN, and policy base for ADAM, see "[The ADAM Schema](#)" on page B-4 for details.
3. Complete the Policy Manager setup process, as described in "[To set up the Policy Manager with ADAM](#)" on page B-14.

Note: A warning may appear at the end of this setup instructing you to create the Anonymous user before enabling Oracle Access Manager policies.

4. Ensure that the OblixAnonymous user has been created within ADAM at the top of the searchbase you specified during Policy Manager setup.
5. Continue with the next procedure to install the Access Server.

To install the Access Server

1. Create an Access Server instance in the Access System Console, as described in "Creating an Access Server Instance" on page 191.
2. Locate and launch the Access Server installation package and specify your installation directory, as described in "[To install the Access Server](#)" on page B-15.
3. Select the transport security mode for the Access Server.
4. Specify ADAM details when asked.
 - **SSL:** SSL is required for password changes.
 - **Port Number:** The directory server port to be used during runtime.
 - **Bind (root) DN:** The Root DN you provided when setting up the Identity Server and Policy Manager; do *not* use a Windows security principal.
 - **Password:** The password for the bind DN.
 - **Directory Server Type:** Active Directory Application Mode.
5. Select Yes if dynamically-linked auxiliary classes are enabled in your environment, otherwise select No.
6. Provide the path to the directory server's certificate file.
7. Specify the Access Server ID and the configuration DN and policy base, which may be unique within the ADP. For example:

Access Server ID: *Access_Server_1014_A*

Configuration DN: *ou=config,o=company,c=us*

Policy base: *ou=policy,o=company,c=us*

Note: The preceding example presumes you are storing all data within a single ADAM instance and ADP.

8. Finish Access Server installation, as described in "Finishing the Access Server Installation" on page 197.
9. Continue with the next procedure to install the WebGate.

To install the WebGate

1. Create a WebGate instance in the Access System Console, as described in "[To install the WebGate](#)" on page B-16.
2. Associate the WebGate with the Access Server, as described in "Associating a WebGate and Access Server" on page 207.
3. Install the WebGate, as described in "[To install the WebGate](#)" on page B-16.
4. Complete any of the following activities when the Access System installation and setup is finished:
 - Add ADAM users to Oracle Access Manager, as described in the *Oracle Access Manager Identity and Common Administration Guide*.
 - Replicate an ADAM instance, as described in your Microsoft documentation.
 - Configure failover and load balancing, for the ADAM master and replicas, as described in the *Oracle Access Manager Deployment Guide*.

Oracle Access Manager Silent Mode Installation Parameters

Several parameter changes have been made in the Oracle Access Manager silent mode installer to support ADAM as a standalone directory server. For details, see:

- [Identity Server Silent Mode Installer for ADAM](#)
- [Policy Manager Silent Mode Installer for ADAM](#)
- [Access Server Silent Mode Installer for ADAM](#)

Note: The dynamic-auxiliary flag can be configured for ADAM in Oracle Access Manager as it is for Active Directory on Windows 2003.

Identity Server Silent Mode Installer for ADAM

The following changes have been made for ADAM in the Identity Server silent installer:

- **Windows User Name and Windows Domain:** Specify a Windows security principal name, Windows domain name, and password to update the ADAM schema; this will not be used as a bind DN.
- **Schema Update, Automatic:** Not supported for ADAM.
- **Schema Update, Manual:** To specify a manual schema update when installing the first Identity Server on Windows Server 2003, use:

```
-W updateDSInfo.updateDSInfoChoice="No"
```

Where:

`-W updateDSInfo.updateDSInfoChoice="No"` specifies a manual schema update.

- **Windows Domain Name for ADAM**—To specify a Windows domain name for ADAM use:

```
-W dsInfoInput.domainName="domainname.com
```

Where:

`-W dsInfoInput.domainName` specifies the Windows Domain Name for ADAM when `-W dsTypeInput.dsType=9` (ADAM).

"*domainname.com*" is the domain name in which the ADAM machine resides. If an incorrect domain name is given, the authentication to the directory will fail.

Note: This is a new parameter and does not alter or replace an existing silent installer parameter.

Policy Manager Silent Mode Installer for ADAM

To specify ADAM as the directory server type during Policy Manager installation use the following:

```
-W dsTypeInput.dsType="9"
```

Where:

`-W dsTypeInput.dsType` specifies the directory server type where policy data is stored.

The following types are supported:

- 2 - Sun 5.x
- 3 - NDS
- 5 - Active Directory
- 7 - Active Directory (Windows Server 2003)
- 9 - Active Directory Application Mode

Later select Dynamic Auxiliary Object Class if this is appropriate for your environment. For ADAM, the option `-W updateDSInfo.updateDSInfoChoice` is not applicable. Otherwise, use `-W updateDSInfo.updateDSInfoChoice = "Yes"` to specify the Policy directory server type.

Access Server Silent Mode Installer for ADAM

To specify ADAM as the directory server type during Access Server installation use:

```
-W oblixDSInfoBean.dsType="MSADAM"
```

Where:

`-W oblixDSInfoBean.dsType` specifies the Configuration directory server type. "MSADAM" stands for Microsoft Active Directory Application Mode

Troubleshooting ADAM Issues

For information, see "[ADAM Issues](#)" on page E-5.

Adding Directory Certificates After Oracle Access Manager Installation

This appendix provides the information you need to change your directory server communication mode to SSL-enabled or to add certificates to connect to multiple directory servers without uninstalling and reinstalling Oracle Access Manager. Topics include:

- [About Directory Certificates](#)
- [Prerequisites](#)
- [Creating a New Certificate Store](#)
- [Adding Certificates](#)
- [Changing the Directory Server Configuration](#)

About Directory Certificates

During installation of the Identity Server, Policy Manager, and Access Server, you specify a directory server communication mode, either open or SSL-enabled, as discussed in "[Securing Directory Server Communications](#)" on page 2-19. The certificate must be stored on the directory server before Oracle Access Manager installation. The certificate store format for LDAP SSL certificates is cert8.db in Oracle Access Manager 10g (10.1.4.0.1).

At times, you may want to enable SSL after Oracle Access Manager installation. For example, you may want change from an open communication mode to an SSL-enabled mode or you may want add directory certificates to connect to additional directory servers.

In such cases, you could either uninstall and reinstall Oracle Access Manager or use the steps that follow to create the cert8.db file needed by the Identity Server, Policy Manager, and Access Server.

Note: Oracle Access Manager 10g (10.1.4.0.1) works with both the cert7.db (upgraded environments) and cert8.db (new installations) certificate stores.

The default certificate store format and name has changed from cert7.db to cert8.db. When you upgrade to Oracle Access Manager 10g (10.1.4.0.1), you continue to use the old LDAP SSL certificate store (cert7.db). When you run the configureAAAServer, setup_ois, or setup_accessmanager utilities, the certificate store format and name is automatically modified to cert8.db.

Task overview: Enabling directory SSL after Oracle Access Manager installation

1. Complete all "Prerequisites" on page C-2.
2. Create a new certificate store, as described in "Creating a New Certificate Store" on page C-2.
3. Populate the new store, as described in "Adding Certificates" on page C-3.
4. Change the directory profile in Oracle Access Manager, as described in "Changing the Directory Server Configuration" on page C-4.
5. Repeat the preceding sequence on the Policy Manager and Access Server, as needed, or copy the store with the new certificates to the Policy Manager and Access Server, as needed.
6. See the *Oracle Access Manager Identity and Common Administration Guide* for details about transport security changes for Oracle Access Manager and the directory server using the appropriate utility for your platform: `start_setup_ois` (UNIX) or `setup_ois` (Windows).

Prerequisites

You need to have a copy of the Base 64 encoded root certificate from the certificate authority for all directory servers with whom the communication will be in SSL mode. This needs to be stored in the `cert8.db` store used by the Identity Server to establish the SSL connection with the directory server.

Note: If you have a `cert8.db` file in `\Component_install_dir\identity\oblix\config`, be sure to delete it before you start the procedures that follow.

With Active Directory, you need to enable SSL for all domain controllers and have a copy of the Microsoft CA Root Certificate available in Base64 encoded format.

Creating a New Certificate Store

The following procedure walks you through creating a new `cert8.db` certificate store. You can complete this task on the Identity Server, Policy Manager, and Access Server. Be sure to complete the prerequisites before you start.

Table C-1 lists the options you supply the command you provide to create the data store.

Table C-1 Options to Create the Data Store

Option	Description
<code>-d directory</code>	This option identifies the directory for the <code>cert8.db</code> store.
<code>-N</code>	This option creates a new certificate database.

To create the new certificate store

1. Obtain a copy of the Base64 encoded CA Root Certificate from your CA and store it on the machine hosting the installed Identity Server.
2. Locate the `certutil` utility in `IdentityServer_install_dir\identity\oblix\tools\certutil`.

3. In a command window, enter:

```
C:\IdentityServer_install_dir\identity\oblix\tools\certutil>certutil -d
c:\IdentityServer_install_dir\identity\oblix\config -N
```

You will be prompted for the cert8.db store password, which must be entered to encrypt this key and any future keys. The password must be at least 8 characters in length and must contain at least one non-alphabetic character.

4. Enter the cert8.db store password, then re-enter the password.

The cert8.db store is created on the Identity Server and ready to populate.

Adding Certificates

Once you have created the new cert8.db store, you need to add the CA Root Certificate. [Table C-2](#) lists the command options to complete this task.

Table C-2 Options to Add Certificates to the Data Store

Options	Description
-d directory	The value is the full path to the cert8.db store.
-A	This option adds a certificate to the store.
-a	This option indicates an ASCII encoded certificate.
-n	This option indicates the certificate nickname.
-t C,,	This option provides trust attributes, where C,, indicates the Trusted CA to Certs (only for SSL, and implies a valid CA).
-i CAROOT.cer	This option provides input, where CAROOT.cer is the name of your Base64 encoded CA root certificate.
-L	This option requests a list of certificates in the data store directory.

To add certificates to the data store

1. At the command prompt, enter the following to add certificates to the data store:

```
C:\OracleAccessManager\identity\oblix\tools\certutil>certutil -d
C:\OracleAccessManager\identity\oblix\config -A -a -n CAROOT -t C,, -i
CAROOT.cer
```

2. Verify that your certificate was added to the cert8.db store using the command to list the content of the cert8.db store directory.

For example:

```
C:\OracleAccessManager\identity\oblix\tools\certutil> certutil -d
C:\OracleAccessManager\identity\oblix\config -L
```

[Table C-3](#) shows sample results from the list command, which confirms that the certificate was added to the database with the Nickname of CAROOT:

Table C-3 Sample Results of the List Command

Certificate Name	Trust Attributes
CAROOT	C,,
Example.com Code Signing CA	,C
Example.com Individual CA	,C,

Table C-3 (Cont.) Sample Results of the List Command

Certificate Name	Trust Attributes
Example.com Server CA	CG,,

Changing the Directory Server Configuration

After adding certificates you need to complete the process for the directory server configuration within the Identity System Console.

To change the directory profile

1. Navigate to the Identity System Console, select System Configuration, then click Directory Options.
2. Under the Configure Profiles label, click Directory Server.
3. Select the appropriate Directory Server Security Mode*.

Note: When you change fields marked with an asterisk, *, you must repeat product setup. For more information about re-running Identity System setup, see the *Oracle Access Manager Identity and Common Administration Guide*.

4. Restart the Identity Server to have directory server changes take affect.
5. Verify that the Identity Server is running in SSL / Cert Mode by checking the process start-up message.

For example:

UNIX: A message is returned to the console saying the Process has started. The port number and communication mode are included in the message.

Windows: Look in the Event Viewer under Applications for the port number and communication mode.

6. Create and populate Policy Manager and Access Server stores, configure their directory profiles, then restart the Policy Manager Web server and Access Server.

Note: If directory server and CA details are the same for all Oracle Access Manager components that communicate with the directory, you can copy the Identity Server cert8.db store to the Policy Manager and Access Server. Be sure to complete all steps to finish and verify the configuration.

For more information about changing transport security modes after installation, see the *Oracle Access Manager Identity and Common Administration Guide*.

Changing Directory Server Hosts

The information here explains how to reconfigure Oracle Access Manager to recognize a new directory server host. Topics include:

- [About Changing Directory Server Hosts](#)
- [Minimizing Down Time](#)
- [Preparing the New Directory Server Instance](#)
- [Reconfiguring the Primary Identity Server](#)
- [Reconfiguring the Policy Manager](#)
- [Reconfiguring the Access Server](#)

About Changing Directory Server Hosts

After installing and setting up Oracle Access Manager, you may need to change the host machine for the directory server with which Oracle Access Manager communicates. If this occurs, you need to reconfigure Oracle Access Manager to recognize the new directory server host.

Task overview: Changing Directory Server hosts includes

1. [Minimizing Down Time](#)
2. [Preparing the New Directory Server Instance](#)
3. [Reconfiguring the Primary Identity Server](#)
4. [Reconfiguring the Policy Manager](#)
5. [Reconfiguring the Access Server](#)

Minimizing Down Time

When you reconfigure Oracle Access Manager to communicate with a new directory server instance (one that has moved to a different host), there will be some down time. You can minimize the downtime by configuring *failover* between Oracle Access Manager Web components and Access and Identity Servers.

Oracle Access Manager uses failover to provide uninterrupted service by redirecting requests to another server when the original request destination fails. Failover is accomplished by configuring a primary and secondary server and identifying specific parameters for the failover process. Oracle Access Manager Web components first attempt to connect to a primary server. If the primary server is unavailable, a connection attempt is made to a secondary server:

- WebPass requests are re-directed to secondary Identity Servers
- WebGate requests are re-directed to secondary Access Servers

Task overview: Minimizing down time includes

1. [Configuring Failover between an Identity Server and WebPass](#)
2. [Configuring Failover between an Access Server and WebGate](#)

Completing the preceding tasks ensures that users will enjoy uninterrupted service when you reconfigure the primary Identity and Access Servers for the new directory server instance.

For additional information on failover, see the *Oracle Access Manager Deployment Guide*.

Configuring Failover between an Identity Server and WebPass

Setting up a secondary Identity Server ensures that the WebPass fails-over to the secondary Identity Server if the primary Identity Server is stopped while you reconfigure this to communicate with the new directory server instance.

To configure failover between an Identity Server and WebPass

1. Confirm that you have a *second* Identity Server installed that meets the following requirements:
 - The second Identity Server must communicate with the existing directory server.
 - *The second Identity Server must be associated with the existing WebPass as a secondary server.*

Note: If your Oracle Access Manager installation does *not* include a second Identity Server that meets the preceding requirements, you need to install one that does. See "[About Installing Multiple Identity Servers](#)" on page 4-2.

2. Configure failover between the secondary Identity Server and WebPass:
 - a. From the Identity System Console select System Configuration, WebPass, *Name*, then click Modify.

See the *Oracle Access Manager Identity and Common Administration Guide* for more information about configuring a WebPass.
 - b. Complete the following information, then save your changes:
 - Failover Threshold:** Enter the required number of live connections from the Web component to its *primary* Access or Identity Server.
 - Identity Server Timeout Threshold:** Enter a Timeout Threshold to specify how long (in seconds) the Web component waits for a non-responsive server before it considers it unreachable and attempts to contact another.
 - Sleep For (seconds):** Enter the interval in seconds. After this interval, the WebGate verifies whether the number of valid connections equals the maximum number of connections configured
3. Configure relevant directory profiles to use all Identity Servers:
 - a. In the System Console, locate the list of LDAP Directory Server Profiles:

From the Identity System Console select System Configuration, then click Directory Options

The Configure Profiles page appears with Directory Server information as well as sections for Configure LDAP Directory Server Profiles and Configure RDBMS Profiles.

- b. Under the Configure LDAP Directory Server Profiles heading, select the name of the Identity Server profile:

Configure LDAP Directory Server Profiles

name

The Modify Directory Server Profile page.
 - c. In the Modify Directory Server Profile page, locate the "Used by" field and select "All Identity Servers".
 - d. Save the change.
4. Proceed with "[Configuring Failover between an Access Server and WebGate](#)" next.

Configuring Failover between an Access Server and WebGate

As with the Identity Server, setting up a secondary Access Server ensures that the WebGate fails-over to the secondary Access Server while the primary Access Server is stopped as you reconfigure this to communicate with the new directory server instance.

To configure failover between an Access Server and WebGate

1. Confirm that you have a *second* Access Server installed that meets the following requirements:
 - The second Access Server must communicate with the existing directory server containing Oracle Access Manager configuration and policy data.
 - The second Access Server must be associated with the existing WebGate as a *secondary* server.

Note: If your Oracle Access Manager installation does not include a second Access Server that meets the preceding requirements, you need to install one. See "[About Installing Multiple Access Servers](#)" on page 8-2.

2. Configure failover between the secondary Access Server and WebGate, as described in the *Oracle Access Manager Deployment Guide*:
 - a. From the Access System Console, select Access System Configuration, AccessGate Configuration, All, Go, then click the desired name.

The Details for the AccessGate page appears. See the *Oracle Access Manager Access Administration Guide* for more information about configuring a WebGate.
 - b. Click the Modify button, then fill in the following information and save your changes:

Failover Threshold: Enter the required number of live connections from the Web component to its primary Oracle Access Manager server.

Access Server Timeout Threshold: Enter a value to specify how long (in seconds) the Web component waits for a non-responsive Oracle Access Manager server before it considers it unreachable and attempts to contact another.

Sleep For (seconds): Enter the interval in seconds. After this interval, the WebGate verifies whether the number of valid connections equals the maximum number of connections configured

3. Configure the relevant directory server profiles to use all Access Servers:
 - a. In the Access System Console, locate the list of LDAP Directory Server Profiles:

Access System Console, System Configuration, Server Settings

The View Server Settings page appears with Directory Server information as well as sections for Configure LDAP Directory Server Profiles and Configure RDBMS Profiles.
 - b. Under the Configure LDAP Directory Server Profiles heading, select the name of the Access Server profile:

Configure LDAP Directory Server Profiles

name
 - c. On the Modify Directory Server Profile page, locate the "Used by" field and select the button beside "Access Servers", then select "All Servers" from the list. For example:

Used By	<input type="radio"/> All components
	<input type="radio"/> Identity Servers
	<input type="radio"/> Access servers
	<input type="radio"/> All Servers
4. Save the change.
5. Proceed with ["Preparing the New Directory Server Instance"](#) next.

Preparing the New Directory Server Instance

You must ensure that the new directory server instance is an exact replica of the directory server instance with which Oracle Access Manager communicates. This means that the schema, user data, Oracle Access Manager configuration data, and policy data must match. In addition:

- If any data is stored separately in the existing directory server instance, the new directory server instance must match this configuration.
- If the existing directory server uses SSL, the new directory server must have a certificate issued by the same Root CA as the one issued for the existing directory server.

As you prepare the new directory server instance, pay close attention to the following:

- If policy data is stored in a separate directory server than configuration data (o=oblix), you must also export (then import) an LDIF for policy data.
- If you are using NetPoint 6.5 or later, you must remove the entries under obcontainerId=DBAgents,<Configuration DN>... that are associated with the Policy Manager and Access Servers.

Note: When removing entries for DB agents, do *not* delete the container (obcontainerId=DBAgents).

To prepare the new directory server instance

1. Export the *original* Oracle Access Manager configuration tree (o=oblix) from the existing directory server instance to an LDIF file using the `ldapsearch` command that follows; repeat for policy data if this is stored separately. For example:

```
ldapsearch -h DS_hostname -p DS_port_number -b Configuration_DN (o=oblix...)
-D bind_dn -w password -s sub (objectClass=*) > Oblix_Data_original.ldif
```

where *DS_hostname* is the name of the machine hosting the new directory server instance (from which you export data); *DS_port_number* is the port on which the directory server is listening; *bind_dn* is the DN for Oracle Access Manager configuration data; *password* is the *password* for the bind DN; and *Oblix_Data_original.ldif* is the name of your configuration data ldif file.

2. Remove the entries for DB Agents *without* deleting the container (obcontainerId=DBAgents). For example:

```
obcontainerId=DBAgents,<Configuration DN>...
```

3. Import the *modified* LDIF you created to the new directory server instance using the `ldapmodify` command that follows. For example:

```
ldapmodify -h DS_hostname -p DS_port_number -D bind_dn -w password -a -f
Oblix_Data_modified.ldif
```

where *DS_hostname* is the name of the machine hosting the new directory server instance (to which you would import the data); *DS_port_number* is the port on which the directory server is listening; *bind_dn* is the DN for Oracle Access Manager configuration data; *password* is the password for the bind DN; and *Oblix_Data_modified.ldif* is the name of your configuration data ldif file.

4. Proceed to "[Reconfiguring the Primary Identity Server](#)" to add directory server profiles for the new directory server instance to the Identity System Console.

Reconfiguring the Primary Identity Server

The procedure that follows describes how to reconfigure the primary Identity Server, the one that communicates with the existing directory server instance, so that it communicates with the new directory server instance.

To configure the Identity Server to communicate with a new directory server instance

1. From the Identity System Console, select System Configuration, Directory Profiles, then click Directory Server.
2. On the Directory Server Configuration page, change the following information to reflect the new directory server instance, then save your changes:

Machine*—*new_hostname.domain.com*

Port Number*—*new_host_port*

When you change fields marked with an asterisk (*), you must manually re-run the Identity System setup.

3. Shut down all Identity Servers except the secondary server, if more than one are running.
4. On the only running Identity Server host, open the setup.xml file:
IdentityServer_install_dir\identity\oblix\config\setup.xml
5. Remove the status parameter (or change the status parameter value from "done" to "incomplete"), then save the file. For example:

```
<NameValPair ParamName="status" Value="incomplete"></NameValPair>
```

6. Restart this Identity Server.
7. From your Web browser, launch the Identity System Console.
A Setup page appears, like the one for the initial Identity System setup.
8. Click the Setup button and proceed through the setup process:
 - a. Specify *new* directory server instance information, as follows:
Host—The *new* user data directory server DNS hostname
Port Number—The *new* user data directory server port number

Note: If user data is stored separately from configuration data, a similar page appears where you can enter information for the configuration data directory. However, that sequence is *not* repeated here.

- b. Complete setup as described in [Chapter 6, "Setting Up the Identity System"](#).
9. Restart the Identity Servers, which should pick up the new information.
10. In the System Console, verify that a new database profile was created:
 - a. Navigate to the Directory Profiles page:
From the Identity System Console, select System Configuration, then click Directory Profiles
 - b. In the Configure Profiles page, select the name of the relevant profile under the heading Configure LDAP Directory Server Profiles.
 - c. In the Modify Directory Server Profile page, locate the name of the *new* Database Instance and confirm the new machine and port number.

Note: You can proceed with creating any additional DB profiles that you need. See the *Oracle Access Manager Identity and Common Administration Guide* for details.

11. Proceed with ["Reconfiguring the Policy Manager"](#) next.

Reconfiguring the Policy Manager

You need to reconfigure the Policy Manager to use the new directory server instance.

To reconfigure the Policy Manager for the new directory server instance

1. View server settings in the Access System Console, as follows:

From the Access System Console, select Access System Configuration, Server Settings, then click Directory Server

2. On the Directory Server Configuration page, change the following information to reflect the new directory server instance, then save your changes:

Machine*—*new_hostname.domain.com*

Port Number*—*new_host_port*

When you change fields marked with an asterisk (*), you must manually re-run the Policy Manager setup.

3. Shut down all but one Policy Manager Web server if there is more than one running.
4. On the only remaining running Policy Manager host and open the setup.lst file:


```
PolicyManager_install\dir\oblix\config\setup.lst
```
5. Remove the status parameter (or change the status parameter value from "done" to "incomplete"), and save the file. For example:

```
<NameValPair ParamName="status" Value="incomplete"></NameValPair>
```

6. Restart the Policy Manager Web server.
7. From your Web browser, launch the Access System Console.

You will see a Setup page similar to the one that appears during the initial Access System setup. You need to specify details about the directory servers where user data, configuration data, and policy data are stored and asked to provide information about the directory server for each type of data.

8. Initiate setup again and, when asked, specify the following:
 - If user data and configuration data are stored together, you are asked where policy data should be stored.
 - If the data is stored separately, you are asked to specify details for configuration data.

For more information about this, see "[Data Storage Requirements](#)" on page 2-21.

9. When asked, specify the new directory server instance information, as follows:
 - a. Specify *new* directory server instance information, as follows:

Host—The *new* directory server DNS hostname

Port Number—The *new* directory server port number

Note: Depending on how your data is stored, you may see an additional screen for policy data. However, that sequence is *not* repeated here.

- b. Complete setup as described in "[Setting Up the Policy Manager](#)" on page 7-9.
10. After completing setup, restart the other Policy Manager Web servers.

The other Policy Managers should pick up the new information.
11. Confirm the new Database Instance in the Access System Console, as follows:
 - a. View server settings in the Access System Console, as follows:

From the Access System Console select Access System Configuration, Server Settings.

- b. In the View Server Settings page, select the name of the relevant profile under the heading Configure LDAP Directory Server Profiles.
- c. In the Modify Directory Server Profile page, locate the name of the new Database Instance and confirm the *new* machine and port number.

Note: You can proceed with creating any additional DB profiles that you need. See the *Oracle Access Manager Identity and Common Administration Guide* for details.

12. Rerun Access Server setup, as described in "[Reconfiguring the Access Server](#)".

Reconfiguring the Access Server

After manually rerunning setup for the Policy Manager, you need to reconfigure the Access Server as indicated in the following discussion. For additional information on using the configureAAAServer tool, see *Oracle Access Manager Access Administration Guide*.

To reconfigure the Access Server for the new directory server instance

1. Locate the configureAAAServer tool. For example:

```
AccessServer_install_dir\access\oblix\tools\configureAAAServer
```
2. Use the command that follows with the configureAAAServer tool to set up the Access Server. For example:

```
configureAAAServer install -i AccessServer_install_dir/util/access
```
3. Specify new information for the host on which the new directory server instance resides.
4. Restart your Access Server.
5. Confirm the new Database Instance in the Access System Console, as follows:
 - a. View server settings in the Access System Console, as follows:
Access System Console, System Configuration, Server settings
 - b. On the View Server Settings page, select the name of the relevant profile under the heading Configure LDAP Directory Server Profiles.
 - c. On the Modify Directory Server Profile page, locate the name of the new Database Instance and confirm the *new* machine and port number.

You may see one more Database Profile created, in addition to the default, when the policy tree and the configuration tree are on the same directory server yet are using two different suffixes.

Note: You can proceed with creating any additional DB profiles that you need. See the *Oracle Access Manager Identity and Common Administration Guide* for details.

Troubleshooting Installation Issues

This chapter describes common troubleshooting issues and tips to resolve them. This chapter covers the following topics:

- [Browser Issues](#)
- [Directory Server Issues](#)
- [Identity System Issues](#)
- [IIS and Windows Issues](#)
- [Installation Issues](#)
- [Issues with Oracle Virtual Directory Implementations](#)
- [Language Issues](#)
- [Login Issues](#)
- [Policy Manager Issues](#)
- [Reinstalling Oracle Access Manager with Oracle Internet Directory](#)
- [Transport Security Mode Issues](#)
- [User Directory Issues](#)
- [Web Server Issues](#)
- [WebGate Issues](#)
- [Miscellaneous Issues](#)

Browser Issues

The following issues are browser specific:

- [Character Display Issues](#)
- [Microsoft Internet Explorer 6 with Sun VM v1.4.2_04](#)
- [Unable to Authenticate Resource on Internet Explorer](#)

Character Display Issues

When using Apache-based Web servers (including Apache, Oracle HTTP Server (OHS), and IBM HTTP Server (IHS)), Oracle Access Manager HTML pages may appear garbled.

Oracle Access Manager 10g (10.1.4.0.1) HTML pages use UTF-8 encoding. Apache-based Web servers allow administrators to specify a default character set for

HTML pages using the `AddDefaultCharset` directive. If the `AddDefaultCharset` directive enables a character set other than UTF-8, Oracle Access Manager HTML pages are garbled.

Oracle recommends that you disable the `AddDefaultCharset` directive in the Web server configuration file (`httpd.conf`) as follows to ensure the correct display of Oracle Access Manager 10g (10.1.4.0.1) HTML pages:

```
AddDefaultCharset Off
```

Microsoft Internet Explorer 6 with Sun VM v1.4.2_04

With this configuration, when you create a container limits policy and specify a user to receive notification for a containment limit event then click **Done** in the Person Selector you may notice that the **Save**, **Cancel**, and **Reset** buttons do not appear and the policy cannot be saved.

To work around this problem

1. Open the `oblixbaseparams.xml` file:

```
IdentityServer_install_dir\identity\oblix\apps\common\bin\oblixbaseparams.xml
```

2. Locate the section, `applet_customizations` and find the subsection for the applet where the problem is observed. For example, `containmentlimit_applet` subsection.
3. Adjust the width and height parameters suitably for this applet to resolve the issue. For example, modify the `applet_dimension_height` parameter to a value of 530.
4. Restart the Identity Server.
5. Open a new browser window and view the same applet.

Unable to Authenticate Resource on Internet Explorer

Previous releases of Oracle Access Manager supported only Latin-1 encoding. 10g (10.1.4.0.1), however, supports UTF-8 encoding. This problem should no longer occur.

Symptom: You receive Unable to Authenticate Resource errors on Internet Explorer.

Cause: Internet Explorer provides the advanced option to always convert UTF characters in URLs. Oracle Access Manager automatically does this as well. Having both enabled causes authentication errors.

Solution: Complete the steps that follow to remedy "Unable to Authenticate Resource" errors on IE.

To remedy this

1. Open the `globalparams.xml` file in a text editor. This file is stored in two locations, and both must be edited:
 - `\IdentityServer_install_dir\identity\oblix\apps\common\bin\globalparams.xml`
 - `\AccessServer_install_dir\access\oblix\apps\common\bin\globalparams.xml`
2. Set the `doUtfConversion` parameter to `NO` and save your changes.

Note: If you set this after importing the UTF-8 data, restart the Web server, close all browsers, and open a new browser window.

3. In Microsoft Internet Explorer, select **Internet Options** from the **Tools** menu.
4. In the Internet Options dialog box, select the **Advanced** tab.
5. Under Browsing, deselect the **Always send URLs as UTF-8** check box.

Directory Server Issues

The following discussions identify several directory server issues:

- [Active Directory Issues](#)
- [ADAM Issues](#)

See also "[Issues with Oracle Virtual Directory Implementations](#)" on page E-10.

Active Directory Issues

Active Directory 2000 does *not* support concurrent bind requests coming from different Oracle Access Manager threads on the same LDAP connection.

Oracle Access Manager servers are multi-threaded and maintain a pool multiple LDAP connections to the directory server. Several Oracle Access Manager threads may share the LDAP connection for efficient processing of requests. However, Active Directory 2000 does not support concurrent binds requests coming from different Oracle Access Manager threads on the same LDAP connection. This may cause spurious authentication failures.

To avoid this situation

1. On the Access Server, locate the `globalparams.lst` file and open this in an editor.
2. Add a new flag called `exclusiveAuthnConnection` and set it to `true`.

This forces Oracle Access Manager threads to use separate LDAP connections for bind requests being sent to the directory server.

For additional information, see:

- [Active Directory Search Halts](#)
- [ADSI Cannot Be Enabled for this DB Profile \(Active Directory\)](#)
- [Dynamically-Linked Auxiliary Classes for Active Directory](#)
- [Appendix A, "Installing Oracle Access Manager with Active Directory"](#)

Active Directory Search Halts

Symptom: 400 policy domains were created in Oracle Access Manager, each with 10 resources and 10 policies. The `limitAMPolicyDomainResourceDisplay` is set to `true` in the Policy Manager `globalparams.xml` file. When the **Search** icon is selected an error page appears stating "The following messages were produced by the product. Please contact your webmaster to fix the problem."

Cause: The number of policy domains exceeds the current limit.

Solution: Do not exceed 350 policy domains with Active Directory.

ADSI Cannot Be Enabled for this DB Profile (Active Directory)

Oracle Access Manager supports changing the user data DB profile between ADSI and LDAP using the Identity System Console. However, Oracle Access Manager does not support changing the configuration or policy DB profile between ADSI and LDAP using the System Console.

Symptom: Suppose you have user data stored in an Active Directory forest using LDAP and configuration and policy data stored in another Active Directory forest using ADSI. When you change the ADSI flag in the configuration data DB profile to LDAP using the Identity System Console and restart Oracle Access Manager servers and services, the ADSI flag remains enabled and the following message appears:

"ADSI can be enabled for either User or Configuration DB Profile if they are in a separate forest. ADSI Cannot be Enabled for this DB Profile."

Further, attempting to modify the user data DB profile to ADSI produces an error because Oracle Access Manager recognizes the DB profile for configuration and policy data as ADSI enabled.

Solution: Rerun the setup program to modify the DB profile for configuration and policy data.

Dynamically-Linked Auxiliary Classes for Active Directory

If you installed Active Directory with Windows Server 2003 and experience difficulty with dynamically-linked auxiliary classes, ensure that you have completed all items that follow.

See Also: For more information, see [Chapter A, "Installing Oracle Access Manager with Active Directory"](#) on page A-1.

To enable dynamically-linked auxiliary classes for Active Directory

1. Before Oracle Access Manager installation, you must ensure that the Active Directory domain and forest functionality are operating at a Windows 2003 Server level.

You must raise both the domain and the forest to a Windows 2003 Server level, as described in the Microsoft documentation.

2. During Identity System installation and set up, you must specify dynamically-linked auxiliary classes when asked.
3. During Policy Manager installation and set up, you must specify dynamically-linked auxiliary classes when asked.
4. During Access Server installation, you must specify dynamically-linked auxiliary classes when asked.
5. After setup, it's a good idea to verify that the `dynamicAuxiliary` flag is set to true in the following files:

- `\IdentityServer_install_dir\identity\oblix\data.ldap\common\ldapconfigdbparams.xml`
- `\PolicyManager_install_dir\access\oblix\data.ldap\common\ldapconfigdbparams.lxml`
- `\AccessServer_install_dir\access\oblix\data.ldap\common\ldapconfigdbparams.xml`
`NameValPair ParamName= "dynamicAuxiliary" Value= "true"`

Oracle Access Manager also sets a `dynamicAuxiliary` tag to true in the following file:

- `\IdentityServer_install_dir\identity\oblix\config\setup.xml`

Note: The directory is the best place to look for any static associations.

6. Configure Oracle Access Manager for dynamically-linked auxiliary class support, as described in the *Oracle Access Manager Identity and Common Administration Guide*.

ADAM Issues

- [ADAM: Cannot find the Config DN or Searchbase](#)
- [ADAM Directory Server Security](#)
- [ADAM Object Classes](#)
- [ADAM Password Changes](#)
- [ADAM Schema Updates](#)

For more information, see [Appendix B, "Installing Oracle Access Manager with ADAM"](#).

ADAM: Cannot find the Config DN or Searchbase

Please make sure the configuration DN or searchbase exist.

This error message may also indicate that you are not using `ous` for the configuration and policy DNs.

ADAM Directory Server Security

The ADAM schema must be updated through an open port. For details, see "[ADAM Schema Updates](#)" on page E-6.

Password changes can be made only through an SSL-enabled port. For details, see "[ADAM Password Changes](#)" on page E-6.

ADAM Object Classes

ADAM describes the user object class differently than Active Directory. `samaccountname`, which is required with Active Directory, does not exist in ADAM. `groupType` is still required with ADAM. Oracle Access Manager configures a `groupType` attribute when you automatically configure attributes during setup.

Keep the following in mind with manual schema updates when you don't plan to use dynamic auxiliary classes:

- Update the schema manually using `IdentityServer_install_dir\identity\oblix\data.ldap\common\ADAM_oblix_schema_add.ldif`, as described in "[Installing and Setting the Identity System with ADAM](#)" on page B-11.
- Use the `ldifde` command to import the Oracle Access Manager schema file `ADAMAuxSchema.ldif` from the `IdentityServer_install_dir\identity\oblix\data.ldap\common` directory.

- Ensure that the object classes "oblixorgperson" and "oblixgroup" are explicitly attached as auxiliary classes to the Person and Group object classes, respectively.

ADAM Password Changes

Password changes require SSL. If you have a problem changing a password, the directory server may have a native password policy that is not being honored.

When creating a user, ensure the user has a password. If you activate a user and the operation fails, the user may not have a password.

Users must be enabled within ADAM. If you search for a user in the Oracle Access Manager User Manager, and the user does not appear as the result of the search, check the msDS-UserAccountDisabled= user attribute in the object class to see if it is disabled or enabled.

ADAM Schema Updates

When you install Oracle Access Manager, you must manually update the ADAM schema.

If the user data directory instance is separate from the configuration and policy data directory instance, you must manually upload the ADAM_user_schema_add.ldif file.

On the configuration data directory instance, you must manually upload the ADAM_oblix_schema_add.ldif file. When using static auxiliary classes, you must manually upload the ADAMAuxSchema.ldif file.

If the policy data directory instance is separate from the configuration data directory instance, you must manually upload the ADAM_oblix_schema_add.ldif file. When using static auxiliary classes, you must manually upload the ADAMAuxSchema.ldif file.

Currently ldifde, which is used to extend the ADAM schema, does not support binding to an SSL port. If you are having trouble updating the schema, ensure that you specified the open ADAM port during Identity Server installation. You may still install certificates and specify SSL during Identity Server installation.

Identity System Issues

This discussion covers the following Identity Server issues that may arise:

- [Application Has Not Been Set Up](#)
- [Cannot Set Up Identity System](#)
- [Checking Access Server or Identity Server Availability](#)
- [Could Not Get Any DB Profile](#)
- [Identity Server Does Not Start](#)
- [Identity System Components May Fail](#)
- [IdentityXML Calls Fail After WebGate Install](#)
- [WebPass Identifier Not Available After Setup](#)

Application Has Not Been Set Up

Under certain circumstances, you may want to reuse an existing Identity Server name. For example, you may want to use an existing Identity Server name if you are rolling

Oracle Access Manager out from a test environment to a production environment or if you need to remove a Identity Server for some reason.

If you do not delete the original Identity Server name from the System Console, a login following set up of a new instance may result in the message "Application has not been set up". Special steps must be taken to ensure you can set up the application and login when recycling a Identity Server name. For more information, see "[Recycling an Identity Server Instance Name](#)" on page 20-4.

Cannot Set Up Identity System

When the Identity Server and WebPass are installed in Cert mode using certificates issued by a subordinate CA, you may see a blank page when you click the **Identity System Console** link to start Identity System setup. The event viewer may show a Oracle Access Manager error without specifying any cause.

When using certificates generated by a subordinate CA, the root CA's certificate must be present in the `xxx_chain.pem` along with the subordinate CA certificate. Both certificates must be present to ensure appropriate verification and successful Identity System setup.

See Also: For more information, see the information on transport security modes in the *Oracle Access Manager Identity and Common Administration Guide*.

Checking Access Server or Identity Server Availability

To see if Access Server is running, Telnet to it using the port it is listening to. The following is a Telnet session to an Access Server running on a server named `myserver.mycompany.com` running on Port 6021:

```
myserver% telnet myserver.mycompany.com 6021
Trying 192.168.5.18. . .
Connected to myserver.mycompany.com.
Escape character is '^]'.
^]
telnet> q
Connection closed.
myserver%
```

In the preceding example, the system's response:

```
Connected to myserver.mycompany.com.
Escape character is '^]'.

```

indicates that the Access Server accepted the Telnet request and is operational. If you cannot connect to the server on the port it was installed to listen on, there is a problem with the Access Server. Possible problems include:

- The connection is blocked by a firewall
- The server is not running

Check the firewall to see if the connection is open. Check the Access Server process to see if it is running. At the Access System server, you can use the `netstat` command to verify that the server is communicating through the ports you specified when you installed the Access Server.

Could Not Get Any DB Profile

Symptom: You receive a message "Could Not Get any DB Profile used by Identity2 during initialization of DBManager. Please verify that there exists enabled DB profile for Identity2".

Cause: This can occur when you:

- Install a second (or later) Identity Server
- Answer Yes during installation when asked if this is "the first Identity Server in the network for this LDAP directory server".
- Restart the Identity Servers and Web servers.

Solution: Uninstall the Identity Server, then reinstall the Identity Server and answer No when asked if this is "the first Identity Server in the network for this LDAP directory server".

Identity Server Does Not Start

Symptom: During Identity System setup you are asked to restart the Identity Server and Web servers. After a long wait, the browser returns a message, "The page cannot be displayed".

Cause: Your Web browser may time out waiting for a Identity Server response after specifying directory server details and automatically configuring user object classes, and Group object classes because the schema update may exceed the browser's timeout.

Solution: Wait for a minute or so and refresh the browser to continue.

If the Identity Server does not start, there are three items you can check that may cause the issue.

To troubleshoot the Identity Server

1. Ensure that the LDAP directory is clean. For example:
 - Is the configuration branch empty?
 - Does the configuration branch have the right data?
 - Does the configuration branch have data from a previous install with a different Identity Server entry?
2. Confirm that the following files are correct and in the correct folder:
 - `\IdentityServer_install_dir\identity\oblix\config\configinfo.xml`
 - `\IdentityServer_install_dir\identity\oblix\config\ois_server_config.xml`
 - `\IdentityServer_install_dir\identity\oblix\config\setup.xml`
3. Verify that the port chosen for the Identity Server is not already in use by another application.

Identity System Components May Fail

Symptom: Identity System components on RedHat Linux may fail with the new NPTL-based runtime libraries when "MAX_ROTATION_SIZE" is reduced to 10000 kb in the oblog_config.xml file.

Solution: Set the LD_ASSUME_KERNEL environment variable before starting the Web server and WebLogic and WebSphere components that integrate with the 10g

(10.1.4.0.1) Software Developer Kit and Oracle Access Manager Web components (WebPass, WebGate, and Access Manager). For example:

```
# export LD_ASSUME_KERNEL=2.4.19
```

This causes the Linux dynamic linker to use the old runtime libraries.

IdentityXML Calls Fail After WebGate Install

IdentityXML calls require authentication credentials. If there is no WebGate protecting WebPass, then the basic credential mechanism is used. This takes the form of username and password embedded in the SOAP request itself. However, when a WebGate is installed later, then the IdentityXML calls must be changed to use a SSO token-based authentication.

The IdentityXML calls need to be changed to first obtain an OBSSOCookie, and then pass that token into all the subsequent calls. An example of how to do this is shown in the *Oracle Access Manager Developer Guide*. Look for details on code examples of deployed IdentityXML functions, and the ObSSOCookie Example.

WebPass Identifier Not Available After Setup

The Identity Server identifier that you enter during installation must be unique and must differ from the WebPass identifier that you enter during WebPass installation.

If the WebPass identifier you enter during installation matches the Identity Server identifier entered during the Identity Server installation, the WebPass identifier is not created and will not be available in the Identity System Console after setup.

To reconfigure the WebPass

Use the following steps to reconfigure the WebPass:

1. Locate the `setup_webpass` utility. For example:

```
WebPass_install_dir\identity\oblix\tools\setup\setup_webpass.exe
```

where `WebPass_install_dir` is the directory where you installed the WebPass (`c:\OracleAccessManager\identity`, for instance).

2. Run the `setup_webpass` utility with the following options:

```
setup_webpass -i <WebPass_install_dir> [-q] [-n <WebPass ID>]
[-h <OIS hostname>] [-p <OIS port #>] [-s <open|simple|cert>]
[-P <simple|cert mode password>] [-c (request|install)]
[-W iis]
```

To change the WebPass password for simple/cert mode

Use the following steps to change the WebPass password for simple/cert mode:

1. Locate the `setup_webpass` utility. For example:

```
WebPass_install_dir\identity\oblix\tools\setup\setup_webpass.exe
```

2. Run the `setup_webpass` utility with the following options:

```
setup_webpass -i <WebPass_install_dir> -k
```

To reconfigure Webpass mode

Use the following steps to reconfigure Webpass mode:

1. Locate the `setup_webpass` utility. For example:

```
WebPass_install_dir\identity\oblix\tools\setup\setup_webpass.exe
```

2. Run the `setup_webpass` utility with the following options:

```
setup_webpass -i <WebPass_install_dir> -m
```

IIS and Windows Issues

The account that performs Oracle Access Manager installation must have administration privileges. The user account that is used to run the Identity Server and Access Server services must have the "Log on as a service" right, which can be set by selecting **Administrative Tools, Local Security Policy, Local Policies, User Rights Assignments, Log on as a service**.

On IIS 6 Web servers only, you must run the WWW service in IIS 5.0 isolation mode. This is required by the ISAPI postgate filter. During Oracle Access Manager installation, this is usually set automatically. If it is not, you must set it manually for the Default Web site.

When installing IIS WebGates, setting various permissions for the `/access` directory is required for IIS WebGates only when you are installing on a filesystem that supports NTFS. For example, suppose you install the ISAPI WebGate in Simple or Cert mode on a Windows 2000 machine running the FAT32 filesystem. The last installation panel provides instructions for manually setting various permissions that cannot be set on the FAT32 filesystem. In this case, these instructions may be ignored.

Issues with Oracle Virtual Directory Implementations

You should be aware of several conditions that might affect your implementation with Oracle Virtual Directory:

- [Directory Server Problems](#)
- [Multi-Value Attribute Problems](#)
- [Secondary Data Store Problems](#)
- [Unexpected Group Deletion Problem](#)

For more information, see [Chapter 10, "Setting Up Oracle Access Manager with Oracle Virtual Directory"](#).

Directory Server Problems

Active Directory or ADAM Search Problem: With Oracle Virtual Directory and Active Directory or ADAM directory servers, you cannot search with the "That Sounds Like" operator.

Cause: Active Directory or ADAM directory servers do not support the "That Sounds Like" search.

Workaround: Do not use the "That Sounds Like" search with Active Directory or ADAM directory servers.

Multi-Value Attribute Problems

You cannot modify multi-value attributes through a Change Attribute workflow.

Cause: The default Oracle Virtual Directory schema includes multi-valued attributes, as does the Sun Directory Server schema. The attribute syntax on Active Directory

sometimes may not match. For example, the mail address in Active Directory is single-valued but on a Sun Directory Server and Oracle Virtual Directory this is multi-valued.

For example, when Oracle Virtual Directory is communicating with Active Directory and a Sun directory server if you create a Change Attribute workflow and try to change a multi-valued attribute (such as an mail address) for a user on the Sun Directory Server, the attribute is changed but on Active Directory the commit step fails and the attribute does not get changed.

Workaround: Do not change multi-valued attributes.

Secondary Data Store Problems

1. **Sub-tree Search:** With a database split profile, you cannot derive an attribute from an attribute that is present in the secondary table.

Cause: Oracle Access Manager cannot perform a sub-tree search on an attribute from a secondary data store.

Suppose, for example, if you used the mapping template `CustomOracleDBMapping_mpy.xml` and defined a derived attribute for `InetOrgPerson` as follows:

- **Attribute Name:** MyAttr
- **Display Name:** MyAttr
- **Match Attribute:** employeenumber
- **Lookup Attribute:** employeenumber
- **Object class:** InetOrgPerson

When you search for a user (Rohit for example) and view his profile, you can see the value for the employeenumber attribute but the myAttr value is blank.

In the following example, there is a database and a split profile and the following adapter templates:

```
CustomOracleAdaptorSplitPrimary_adapter_template.xml
CustomOracleAdaptorSplitSecondary1-1_adapter_template.xml
CustomOracleAdaptorSplitSecondary1-M_adapter_template.xml
CustomAdapterJoinView_adapter_template.xml
```

Workaround: Do not configure attributes from a secondary data store

2. **Create User Workflow:** When defining a Create User workflow, Oracle Access Manager enables you to select attributes from the Oracle Virtual Directory secondary view. At run time, the user entries are created in the primary view; however, the workflow fails and these entries cannot be used by Oracle Access Manager.

Cause: Oracle Access Manager gets all the attributes from Oracle Virtual Directory and therefore has no knowledge about which attributes are obtained from the primary data store, rather than the secondary data store.

Workaround: Do not configure attributes from a secondary data store.

Unexpected Group Deletion Problem

When you set up Oracle Access Manager with a Oracle Virtual Directory virtual directory that federates at least one LDAP directory and at least one database table, then you try to remove a member from a group in the LDAP directory, the entire group is removed from that directory.

Cause: For performance reasons, Oracle Virtual Directory returns to Oracle Access Manager only the member you specify for deletion. By contrast, a "standard" LDAP directory server would return all the members in the group.

This non-standard Oracle Virtual Directory behavior has consequences when you try to use Identity System to delete a member from a group. Because a "standard" LDAP directory server returns all members of the group, Oracle Access Manager stills "sees" the rest of the members of the group after the one member has been deleted. But since Oracle Virtual Directory has returned to Oracle Access Manager only the single member of the group designated for deletion, Oracle Access Manager does not "see" any other group members after it deletes the returned member, so it assumes that the group is now empty and it deletes the group and all its members.

Important: This is generic to all DN attributes, not just uniqueMember of a group. The workaround must be applied to all DN attributes where multiple values are a possibility.

Work Around: See the customized file shown in "[Customized Mapping Script for Oracle Database](#)" on page 10-58 and pay attention to the following details to prevent the Identity System from writing dummy user to backend database:

```
Workaround to prevent COREid from writing dummy user to backend database
if haveAttributeValue('uniqueMember', 'cn=Dummy User'):
#removeAttributeValue('uniqueMember', 'cn=Dummy User')
if operation != 'modify':
removeAttributeValue('uniqueMember', 'cn=Dummy User')
else:
change = removeAttribute('uniqueMember')[0]
change.values.remove(DistinguishedName('cn=Dummy User'))
addEntryChange(change)
```

Installation Issues

The following issues arise during or immediately after installation:

- [Access Server Installation Halts](#)
- [CGI Programs Do not Run After Installation](#)
- [File Replace Warning When Installing on Windows](#)
- [Installation Fails with a "bad credentials error \(49\)"](#)
- [Installer Prompts to Replace DLL Files](#)
- [Performing Unix Installation in GUI Mode](#)
- [Quitting a Windows Installation](#)
- [Running as Non-Root User When Installing on AIX](#)
- [Specifying Installation Directories](#)

- [Testing Your Installation](#)
- [Unable to Leave Person Object Class Page](#)
- [WebGate Installation with Apache Web Server on AIX](#)

Access Server Installation Halts

Symptom: The Access Server installation halts with a message explaining that there is no DB profile for this server.

Solution: Perform these steps:

1. Navigate to the Access System Console from your browser by specifying the URL of the WebPass instance for the Policy Manager. For example:

```
http://hostname:port/access/oblix
```

where *hostname* refers to Web server host of the WebPass instance; *port* refers to the HTTP port number of the WebPass Web server instance; */access/oblix* targets the Access System Console.

2. Select the Access System Console link, then log in as a user with Master Administrator privileges.
3. Select the **Access System Configuration** tab, **Access Server Configurataion** in the left column, and **AccessServer_Link**.
4. Click the **Associate DB Profiles** button at the bottom of the details page.
5. Click the **AccessServer_default_user_profile** link at the bottom of the page.
6. Confirm that **AAA Servers** is checked, with either All Servers or the appropriate servers identified.
7. Confirm that the profile is enabled, at the bottom of the page.
8. Click **Save**.
9. Log out and continue the Access Server installation.

CGI Programs Do not Run After Installation

Symptom: Your Web server's CGI programs do not run after you install Oracle Access Manager.

Solution: Perform these steps:

1. In the `../https:server name/config directory obj.conf` file, add this line before the other Oracle Access Manager Init functions:

```
Init fn="Init-cgi" timeout=300 LateInit="yes"
```

Type the line exactly as shown.

2. Restart your Web server.

File Replace Warning When Installing on Windows

Symptom: When installing a Identity Server on a new machine, sometimes the installer attempts to replace the `winnt/system32/Msvcrt.dll` file with an updated one. Because this file is locked by Windows, you get "File is locked cannot replace" message.

Cause: The installer sometimes attempts to replace a file that is locked by the Windows operating system.

Solution: Click Restart in the warning box to replace the DLL.

Installation Fails with a "bad credentials error (49)"

Symptom: Identity Server installation in GUI mode may fail with a "bad credentials error (49)", though the credentials are valid.

Cause: Known problems with third-party Installshield's ISMP framework. If any inputs supplied during installation contain the character \$, the installer might interpret it unpredictably. For example, if the bind password supplied during the schema update for the first Identity Server is Admin\$\$, ISMP interprets this as Admin\$ while invoking the schema update tool and the update fails citing a "bad credentials error(49)".

Suggested Workaround: If this problem is observed during invocation of a particular tool, you may run that tool from the command line.

Note: Every Oracle Access Manager installer that uses the same password may also fail with a credential problem of some type.

Installer Prompts to Replace DLL Files

Symptom: During a subsequent Oracle Access Manager component installation on the same machine, or when installing a second instance of a component on the same machine, the user is prompted to replace one or more of the following DLL files even if the files had been updated during a previous installation:

- `messagedll.dll`
- `mt170mt.dll`

Cause: The preceding DLLs are not Oracle Access Manager DLLs. Because these DLLs do not contain version information, Oracle Access Manager uses a DLL's data stamp to evaluate whether the file needs to be replaced. Upon subsequent installations, the user is prompted to replace the file because the date stamp is older.

Solution: Click **OK** to replace the DLLs.

Performing Unix Installation in GUI Mode

Symptom: When starting a GUI installation on Unix, you may receive a warning regarding fonts and scroll bars.

Solution: These warnings can be ignored. They indicate a change in the appearance of the installation wizard GUI.

Quitting a Windows Installation

Symptom: If you terminate the Windows installation wizard abnormally (such as by hitting Control + C or terminating it from the Task Manager), the wizard is not able to properly clean up its files and leaves a large amount of data in the TEMP directory.

Solution: Delete these files manually.

Running as Non-Root User When Installing on AIX

To run Install Shield as a non-root user on AIX, set the environment variable as:

```
AIX_ISMP_SUPPORT=NONROOT
```

Specifying Installation Directories

Install Oracle Access Manager components in directories that have only standard alphanumeric characters in their path name. Be sure that all file and path names include only English language characters. In file and path names, no international characters are allowed.

Testing Your Installation

Once you finish installing Oracle Access Manager, test your installation.

To test Oracle Access Manager installation

Use the following steps to test an Oracle Access Manager installation:

1. Close all browsers.
2. Shut down your Access Server.
3. Try to open a page protected by Oracle Access Manager.
You should receive an Oracle Access Manager operation error indicating it was unable to authenticate your login.
4. Restart the Access Server and the Web server.
5. Attempt to connect to the same page as in Step 3. The page you specified should open.

Unable to Leave Person Object Class Page

Symptom: You cannot get past the Person object class page during installation and setup.

Cause: Your directory schema is probably invalid.

Solution: Review the changes you made to the directory schema to see if you have done them correctly.

WebGate Installation with Apache Web Server on AIX

Symptom: You install WebGate on an Apache Web server running on AIX in SSL mode. You make changes to `httpd.conf` file from the `sample.obj.conf` file. The Apache Web server fails to start after changing the `httpd.conf` file. You may receive this message:

"Name of server certificate chain file is hardcoded as `ca.cert` in the `httpd.conf` file."

Solution: Change the `Server-Certificate-Chain-filename` to match the actual name of the server certificate chain file for the user.

Language Issues

Following are solutions to problems you may experience:

- [Garbled Password Message](#)

- [Installing Additional Administrator Language Packs](#)
- [Installing Language Packs for Policy Manager and WebGate in Same Directory](#)
- [Removing the Default Administrator Language Pack](#)

Garbled Password Message

Problem: When running the installer using the Console method with some language packs, the "Enter Password" string does not display correctly. The prompt asking you to enter the LDAP password may be garbled.

Solution: The solution that works in most cases is to install all of the @ language support available on the computer where the Oracle Access Manager @ installation is being performed. Be sure all of the fonts that are required @ for the language are installed. Log in to the machine locally and choose the @ language to display on the login screen.

Installing Additional Administrator Language Packs

Symptom: If you install additional Administrator Language Packs for Access System components (after installing the same Language Packs for the Identity System), you may not be able to view new Administrator languages in the Policy Manager.

Solution: Use the procedure that follows to enable new Administrator languages.

To enable additional Administrator languages for the Access System

1. Install new Administrator languages for all Identity System components.
2. Install new Administrator languages for Access System components.
3. Use the Identity System Console to enable the Administrator languages as follows:
 - Disable the Administrator language, if previously enabled.
 - Enable the Administrator language (now installed on both the Identity and Access System components).
4. Restart the appropriate Oracle Access Manager server services (Identity and Access Server services for example) and Web component (WebPass, Policy Manager, and WebGate) Web servers (Apache, IIS, or Sun ONE for example).

Installing Language Packs for Policy Manager and WebGate in Same Directory

Symptom: WebGate is not using the installed Administrator language when the installation sequence is Policy Manager, Language Pack, then WebGate in the same directory.

Cause: If you install the Policy Manager then a Language Pack then a WebGate in the same directory, the language is installed for the Policy Manager only. In this case, both the Policy Manager and WebGate share a common obnls.xml file.

Soution: Install the same Language Pack for the WebGate.

Symptom: Policy Manager is not using the installed language when the installation sequence is Policy Manager, Language Pack, then WebGate in the same directory.

Cause: During WebGate installation you may be asked if you want to overwrite the obnls.xml in the Policy Manager installation directory. Selecting "Yes" replaces the Policy Manager's obnls.xml file (which contains additional Language Pack entries)

with a fresh `obnls.xml` file (which will not include listings for all installed languages). As a result, the Policy Manager will not be available in the additional Administrator language.

Solution: If you are asked during WebGate installation to overwrite `obnls.xml` in the Policy Manager installation directory, be sure to select "No". If you select "Yes", you must install all the same languages for the WebGate as you did for the Policy Manager.

Removing the Default Administrator Language Pack

Removing the Language Pack associated with the default Administrator language that was selected during installation is *not* supported.

If you accidentally uninstall the default Administrator language, you should be able to recover using the following procedure:

To restore the default Administrator language

1. Create one options file, (`options.txt`, for example) with the following content for the default Administrator language you removed:

```
-W ObPropBean.defaultLocale="ko-kr"
```

where the value of `ObPropBean.defaultLocale`, "ko-kr" (Korean) in the preceding example, is the locale of the default Administrator language selected for this installation.

2. Reinstall the default Administrator Language Pack for each component, as shown in the following example:

Identity System:

```
Oracle_Access_Manager10_1_4_0_1_KO_Win32_LP_Identity_System.exe
-options options.txt
```

Access System:

```
Oracle_Access_Manager10_1_4_0_1_KO_Win32_LP_Access_System.exe
-options options.txt
```

3. After reinstalling the default Administrator Language Pack for each component, restart the server services (Identity Server and Access Server) and Web servers OHS or IIS for WebPass, Policy Manager, and WebGate.

Login Issues

This discussion covers the following issues that may arise during login:

- [Identity Server Logged You In, Access System Logged You Out](#)
- [Windows 2000 Users Cannot Log in After Installation](#)
- [Receiving Repeated Login Prompts](#)
- [Unable to log in to Oracle Access Manager on IIS](#)
- [Restricting Access to Oracle Access Manager](#)

Identity Server Logged You In, Access System Logged You Out

The message "Identity Server Logged You In, Access System Logged You Out" could be triggered by one or more events. For example:

- You did not restart the Identity Servers after setting up Access System components.
- The Identity and Access Servers are running on different machines and the clock is set to a different time.
In this case, change the login slack parameter or synchronize system clocks.
- You have protected the Identity System with a policy domain but not the Access System, or vice versa.
Both systems must be protected.
- The shared secret needs to be regenerated.

To regenerate a shared secret

1. Delete the shared secret from the directory server.
2. Log in to the Access System Console.
3. Select Access System Configuration, Common Information Configuration, Shared Secret.
4. Generate a new shared secret.

Windows 2000 Users Cannot Log in After Installation

Symptom: Users are unable to log in after Oracle Access Manager installation.

Cause: When you import user data into Active Directory, all passwords are cleared. This is a security feature of Active Directory.

Solution: In the Active Directory User and Computer MMC, be sure the Change password on next login check box is not selected. Have your users change their passwords.

In the Policy Manager, enable access control for the Password attribute. This forces users to create a service ticket to change their passwords.

Receiving Repeated Login Prompts

Symptom: Users receive repeated login prompts.

Cause: This can occur when you install Oracle Access Manager on a Web server that has security policies enforced through a Web browser.

Solution: Enable Oracle Access Manager security and disable the browser's security.

Unable to log in to Oracle Access Manager on IIS

Symptom: Users may experience unpredictable behavior when they attempt to browse the /identity or /access directory or click the System Console and Policy Manager links (such as receiving File Download dialog boxes).

Cause: This can occur when you install Oracle Access Manager on a Web server that has security policies enforced through a Web browser. When the Oracle Virtual Directory has "Scripts only" permission, users are unable to log in to either the Access System Console or Policy Manager.

Solution: Change the Oracle virtual directory's permissions from "Scripts only" to "Scripts and Executables".

To change permissions for the Oracle virtual directory

1. Select the machine configured for Oracle Access Manager.
2. Expand the Default Web Site.
3. Right-click either **identity** or **access** (the virtual directory you created during Identity System or Access System install).
4. Select **Properties** and select **Scripts and Executables**.

Restricting Access to Oracle Access Manager

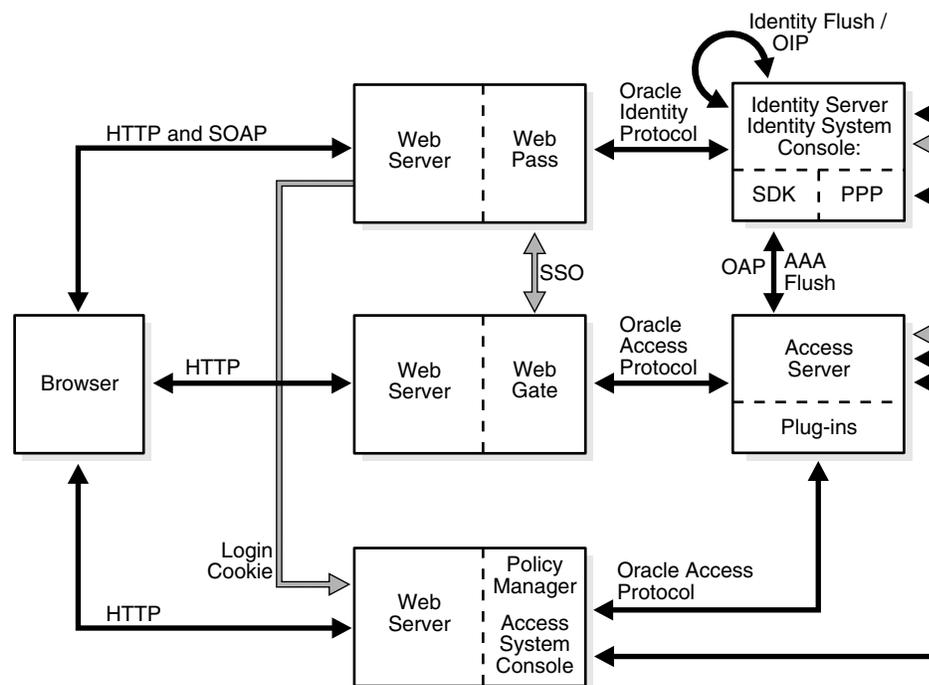
Symptom: After installation, while Oracle Access Manager is protecting access to your resources, access to Oracle Access Manager itself is still unrestricted.

Solution: Use the Policy Manager to restrict access to Oracle Access Manager.

Policy Manager Issues

You need to set the `TEMP` environment variable to point to a valid directory. Oracle recommends that you do this for the entire system, as shown in [Figure E-1](#). However, you can also set the `TEMP` variable for the IIS user.

Figure E-1 Setting the `TEMP` Environment Variable



Without this variable, the Policy Manager attempts to create temporary intermediate files in the root directory and if IIS doesn't have permission to create the files at that level, you may see an error in the following circumstances. For example:

- When setting up the Policy Manager
- When selecting either the **Policy Manager** or **Access System Console** link from the Access System Console main page

In this case, you may see an error message on setup pages warning about not being able to locate the TEMP directory.

Cannot Delete Policy Manager Policy Profile

Symptom: You receive the message, "You cannot delete the Directory Server Profile that accesses the Policy base." after uninstalling the optional Policy Manager and deleting the setup* and config* files in *PolicyManager_install_dir*. You may be able to delete the Policy Manager profile for user and configuration data from the Identity System Console, but not the profile for policy data.

Solution: After uninstalling the optional Policy Manager, you need to complete the steps that follow before you can remove remaining Policy Manager policy profiles.

To delete a leftover Policy Manager policy profile

To delete a leftover Policy Manager policy profile, use the following steps:

1. Uninstall the Policy Manager.
2. In the directory server, remove all oblix-related entries and be sure to delete the `obpolicybase` attribute from the top node. For example,

```
o=Oblix,o=oblixdata,c=uk
```
3. Restart the Identity Server.
4. In the Identity System Console, delete the Policy Manager policy profile.

See Also: See the following topics for more information about issues that may affect the Policy Manager:

- [Active Directory Search Halts](#)
- [Dynamically-Linked Auxiliary Classes for Active Directory](#)
- [Windows 2000 Users Cannot Log in After Installation](#)
- [Unable to log in to Oracle Access Manager on IIS](#)
- [Restricting Access to Oracle Access Manager](#)

Reinstalling Oracle Access Manager with Oracle Internet Directory

If Oracle Access Manager will be removed and reinstalled with the same directory instance, only the Oracle Access Manager configuration tree(s) need be deleted. In this case, there is no need to remove the Oracle Access Manager schema from the directory instance. When reinstalling the Identity Server, select "No" when asked if you want to update the schema (which is already present). Selecting "Yes" results in an error message "schema already exists".

You remove the Oracle Access Manager configuration tree from the directory server instance using tools and instructions from your directory vendor. For Oracle Internet Directory, for example, you may use the Oracle Internet Directory Administration Console. However, you cannot simply delete the parent object because there are dependencies and recursive deletes are not possible.

Oracle recommends that you do not remove the Oracle Access Manager schema from Oracle Internet Directory using the Console. Instead, Oracle recommends that you use the LDIF files in *Component_install_dir\identity\access\oblix\data ldap\common*. For example:

- **OID_oblix_schema_index_delete.ldif** : Oracle Access Manager attribute index cleanup file drops the Oracle Access Manager indexes before or after you clean up the schema.
- **OID_user_schema_delete.ldif**—Oracle Access Manager user data cleanup file for Oracle Internet Directory—removes user data that resides on a separate directory instance from configuration data
- **OID_oblix_schema_delete.ldif**—Oracle Access Manager configuration data cleanup file for Oracle Internet Directory—removes both user and configuration data when both reside on the same directory instance

When user data and configuration data reside in the same directory instance, only the `OID_oblix_schema_delete.ldif` needs to be used with the because it will also remove the user schema objects. However, when a separate directory instance hosts only user data the `OID_user_schema_delete.ldif` should be used. In either case, however, you must use the `OID_oblix_schema_delete.ldif` to remove the attribute index.

For steps, see [Chapter 20, "Removing Oracle Access Manager"](#).

Removal Issues

If a component installation terminates (or is terminated by you) after component files were extracted to the designated installation directory, you should run the Uninstaller for that component and then remove the installation directory before attempting to reinstall in the same location. If you simply delete the installation directory and attempt to reinstall the component in the same location, the `vpd.properties` file is left in an inconsistent state and reinstalling will not work.

For example, suppose you terminate a WebGate installation after component files were extracted, then you remove the installation directory manually rather than using the WebGate uninstaller. In this case, the extracted files are deleted but the `vpd.properties` file is not. This leaves the `vpd.properties` file in an inconsistent state that prevents successful installation.

The preferred method to avoid removal and reinstall issues is to run the component Uninstaller program, then remove the installation directory and then attempt to reinstall the component. However, you may manually remove the component installation directory (without running the Uninstaller) then back up and delete the `vpd.properties`, begin installing the component, then restore the `vpd.properties` file.

For more information, see [Chapter 20, "Removing Oracle Access Manager"](#).

Transport Security Mode Issues

Oracle Access Manager supports three different transport security modes: (Open, Simple, or Cert). While Open is initially easier to implement, it is not secure.

Rather than starting with open and changing later, Oracle recommends you install Oracle Access Manager with the desired transport security mode in place. Changing the transport security mode is outlined in the discussion that follows. For details, see the *Oracle Access Manager Identity and Common Administration Guide*.

To change transport security modes on the Identity System

1. For each Identity Server, run the Identity Server `certutil` program located at `IdentityServer_install_dir/identity/oblix/tools/certutil`.

You must configure the Identity Server before the WebPass. You do not need to use the same password and PEM keys for all Identity System components.

2. For each WebPass, run the WebPass `gencert` program located at `WebPass_install_dir/identity/oblix/tools/gencert`.

To change transport security modes on the Access System

1. For each Access Server, run the `configureAAAServer` program located at `AccessServer_install_dir/access/oblix/tools/configureAAAServer`.

You must configure the Access Server before the WebGate. Use the same password and PEM keys for all Access System components.
2. For each WebGate, run the `configureWebGate` program located at `WebGate_install_dir/access/oblix/tools/configureWebGate`.

User Directory Issues

The following issues pertain to directories.

Adding User to Replicated Directory

If you replicate your Sun directory server and make a change to a user, you are notified that the write will be delayed. However, you are not notified during new user creation.

New users appear in the Identity System pages that are configured against consumers the next time synchronization occurs between the supplier and its consumers.

Synchronization can either occur immediately or at scheduled times, depending on the replication agreement.

Data Corruption

Symptom: While using features in the System Console or in one of the Applications, Oracle Access Manager begins to display bug reports or error messages. This may occur because of corrupt data. Data corruption can be difficult to diagnose. Though data may appear to be valid as displayed in the directory interface tools, the actual data files may be corrupt. One way to verify this is to perform the same search using another tool such as `ldapsearch`. If the expected data is not returned, then the data is corrupt.

Solution: Consult with your directory vendor to determine the most appropriate solution. If possible, export the directory data to an LDIF and examine the LDIF for errors. If the data has obvious errors, correct these errors as appropriate and then import the corrected data.

Web Server Issues

The following issues with Web servers may arise:

- [Access Server Fails on an Apache Web Server](#)
- [Errors, Loss of Access, and Unpredictable Behavior](#)
- [PCLOSE Error When Starting Sun Web server](#)
- [Oracle HTTP Server \(OHS\) Fails to Start](#)
- [Removing and Reinstalling IIS DLLs](#)

Access Server Fails on an Apache Web Server

Symptom: You are running an Apache Web server, and an Access Server fails, displaying the following message:

```
libthread panic: cannot create new lwp
(PID: 9035 LWP 2). stacktrace:
ff3424cc
0
```

This symptom may be caused by the Apache Web server launching more instances of itself. This can happen when the server determines that more instances are needed to service the number of connections between one or more WebGates and the Access Server.

The additional instances create even more connections, which exceed the number of connections by the Access Server.

Solution: Reduce the number of `MinSpareServers`, `MaxSpareServers`, `StartServers`, and `MaxClients` parameters.

Go to the Access Server's configuration directory and open the `http.d` configuration file.

Recommended parameter settings:

- `MinSpareServers` 1
- `MaxSpareServers` 5
- `StartServers` 3
- `MaxClients` 5

Errors, Loss of Access, and Unpredictable Behavior

Symptom: If you installed Oracle Access Manager on Unix under a different user ID than you used to create your Web server instance, Oracle Access Manager can become unstable. Users may experience behavior such as:

- Random bug report pages
- Failure to write to log file errors
- Loss of access to Web pages

Solution: Change file permissions using the `chown` command. Change the Oracle Access Manager directory to the same user ID that you used to create your Web server instance.

Oracle HTTP Server (OHS) Fails to Start

After installing a WebPass, WebGate, or Policy Manager instance on an Oracle HTTP Server (OHS), the server does not start up. This occurs because Oracle Access Manager uses an older Linux threading model.

Solution: Comment out the `perl` module in the `httpd.conf` file, update the `LD_ASSUME_KERNEL` environment variable, and restart, as described in the following paragraphs.

To resolve the failure to start OHS

1. Comment out the `perl` module in the `httpd.conf` file in the following location:

For OHS v1.3: `OH$/Apache/Apache/conf/httpd.conf`

For OHS v2: `OH$/ohs/conf/httpd.conf`

2. To update the `LD_ASSUME_KERNEL` value, open the following file in a text editor:

```
OH$/opmn/conf/opm.xml
```

3. Find the following line:

```
<process-type id="HTTP_Server" module-id="OHS">
```

Add the following information under the line you found in the previous step:

```
<environment>
<variable id="LD_ASSUME_KERNEL" value="2.4.19" />
</environment>
```

4. Save this file.
5. Run the following commands to implement your changes:

```
opmnctl stopall
opmnctl startall
```

PCLOSE Error When Starting Sun Web server

Symptom: When attempting to start the Sun Web server, you get an error like the following:

```
Unable to start, PCLOSE
```

Solution: A number of problems can cause this error:

- A syntax error in your `obj.conf` file
- Leading spaces in your `obj.conf` file
- Installing Oracle Access Manager as a different user ID than what you used to create your Web server instance
- A carriage return at the end of the `obj.conf` file

Removing and Reinstalling IIS DLLs

When Oracle Access Manager is running with Microsoft's IIS Web server, you must manually uninstall and reinstall the following ISAPI filters when reinstalling Oracle Access Manager.

- `tranfilter.dll`
- `oblixlock.dll` (if you installed WebGate)
- `webgate.dll` (if you installed WebGate)

To remove and reinstall IIS DLLs

1. Uninstall Oracle Access Manager.
2. Manually uninstall the preceding DLLs.
3. Reinstall Oracle Access Manager.Active Directory.
4. Manually reinstall the DLLs.

Note: These filters can change depending on the version of IIS you are using. If these filters do not exist or there are others present, contact Oracle to determine if the filters that are present need to be removed.

WebGate Issues

The following issues may arise with WebGate:

- [Access Server and WebGate Naming](#)
- [Enabling WebGate Diagnostics](#)
- [Error Messages After Installing WebGate](#)
- [Installing WebGate and an Identity Server in Same Directory](#)
- [Receiving Access Server Down Errors](#)
- [WebGate Cannot Connect to Access Server](#)

Access Server and WebGate Naming

Access Servers and WebGates cannot be named using non-English keyboard characters.

Descriptions on the Modify AccessGate page of the Access System Console are case insensitive. For example, if you change capitalization of the description but do not alter any other details, you will see no change after the save. To work around this problem, add or alter other information so that the change is recognized.

To change capitalization in an AccessGate description

1. Navigate to the **Access System Console, Access System Configuration, AccessGate Configuration**.
2. Search for a particular AccessGate or just select the **Go** button to display a list of all AccessGates.
3. Double-click the link to the WebGate you want to change.
4. Click **Modify** at the bottom of the page.
5. Enter a new description with the capitalization you would like some new information. For example:

From: webgate

To: WebGate with IIS 6.0

Enabling WebGate Diagnostics

After WebGate installation and configuration, point your browser to the following URL for WebGate diagnostics:

```
http(s)://host:port/access/oblix/apps/webgate/bin/webgate.cgi?progid=1
```

Host and *port* are the WebGate's hostname and Web server instance port number. If the diagnostics page does not open, the WebGate was installed improperly.

Error Messages After Installing WebGate

Symptom: If you are running an Access Server with debugging enabled on a Solaris computer, then install a WebGate on a Windows server that uses that Access Server, you will probably see messages like the following in the Access Server's debug log:

```
Got a client!  
SSL handshake failed:  
error:140890B2:SSL routines:SSL3_GET_CLIENT_CERTIFICATE:no certificate returned
```

Solution: These messages are harmless and may be ignored.

Installing WebGate and an Identity Server in Same Directory

To provide maximum protection for the Identity System, install the WebGate and Identity System in the same directory.

Receiving Access Server Down Errors

Symptom: When attempting to connect, you receive errors indicating that your Access Server is down.

Solution: The clocks of computers hosting various Oracle Access Manager components must be synchronized to within 75 or fewer seconds of each other. If the clocks are out-of-sync by more than 75 seconds, installation will fail.

WebGate Cannot Connect to Access Server

Symptom: You get the following error when you attempt to start the Identity System:

```
Access Server error  
WebGate cannot connect to Access Server
```

Cause: When configuring a WebGate in the System Console, you must link each WebGate with at least one Access Server.

Solution: In Policy Manager, associate your WebGate with an Access Server. Then configure WebGate.

Miscellaneous Issues

The following are miscellaneous issues:

- [Unable to Flush the Cache](#)
- [Giving View Rights to the Master Administrator](#)
- [Idle Session Time, Maximum Cookie Session Time](#)
- [Loading the Directory in Secure Mode](#)
- [Peer Does Not Use Oracle Access Protocol](#)
- [Receiving Bug Report After Replication Attempt](#)
- [Search and Query Error Message \(Defect 4547\)](#)
- [Identity Server Logged You in but Access System Logged You Out](#)

Unable to Flush the Cache

Symptom: If the Policy Manager uses Simple or Cert transport security mode, flushing the cache from Policy Manager requires certificates. If your Policy Manager is protected by a WebGate that was installed in Simple or Cert mode, the certificates exist and there is no problem. However, if you did not install WebGate in Simple or Cert Mode, you cannot update Access System caches because the Policy Manager will not be able to communicate with the Access Servers.

Solution: For an installation that has no WebGate, use the GenCert tool to generate the certificates. This tool is stored at *IdentityServer_install_dir/identity/oblix/tools*, where *IdentityServer_install_dir* is your Identity Server installation directory.

To generate certificates for cache flushing, type

```
genCert install_dir
```

IdentityServer_install_dir is your Identity Server installation directory. You must be the user with the permissions to write files into the installation directory.

Giving View Rights to the Master Administrator

The Master Administrator is specified during Oracle Access Manager installation and setup. Even though this is the highest-ranking Master Administrator, this role does not have view rights for attributes until this right is specifically assigned to it.

The administrator must have permission to view the cn attribute (usually configured as Full Name) at the top level of the directory tree. Then the administrator can configure Access Control for attributes for others.

See Also: Refer to the *Oracle Access Manager Identity and Common Administration Guide* for instructions on completing this task

Idle Session Time, Maximum Cookie Session Time

Symptom: When using either Simple or Cert transport security mode, the user's browser caches the credentials and automatically resends them when a WebGate session times out. This causes the illusion that the timeout settings are not working when in fact a new authentication exchange is taking place without any action on the user's part.

Solution: Use form-based authentication. The browser does not cache form-based authentication information.

Loading the Directory in Secure Mode

Loading your directory can take much longer when SSL is activated. You can load your directory, then activate SSL on the Web server and directory server.

Peer Does Not Use Oracle Access Protocol

Symptom: When a non-Oracle Access Manager program that is set to an incorrect TCP port tries to communicate with your Access Server, you receive an error in your Access Server's debug output. Your Access Server's debug output displays the following error:

```
Peer does not use NetPoint Access Protocol. Connection dropped.
```

Other than the message in your Access Server's debug output, there probably is no impact to your Oracle Access Manager installation. However, the non-Oracle Access Manager peer attempting to communicate with the Access Server will probably fail.

Solution: Check your TCP port numbers. Something is connecting to the wrong thing.

Receiving Bug Report After Replication Attempt

Oracle Access Manager does not support replication with Sun by default.

Symptom: After making a write to a Sun consumer/slave, you get a bug report form.

Solution: To update the `enableLDAPReferral` parameter:

1. Open the `ldapconfigdbparams.xml` file in a text editor. This file is stored in two locations, and both must be edited:
 - *IdentityServer_install_*
`dir/identity/oblix/data/common/ldapconfigdbparams.xml`
 - *Access_Server_install_*
`dir/access/oblix/data/common/ldapconfigdbparams.xml`
2. Change the `enableLDAPReferral` parameter to true.
3. **Save** your changes.
4. Restart the consumer/slave's Web server.

Search and Query Error Message (Defect 4547)

Symptom: When performing a search or a query, you receive a "Bad request" message.

Cause: Your search or query string is too long for your browser. Browsers handle search and query strings as URLs. They generate an error if the string exceeds the maximum URL length.

Solution: Shorten the search or query string.

Identity Server Logged You in but Access System Logged You Out

Symptom: You receive the message "Identity Server logged you in but Access System (Policy Manager or System Console) logged you out." This occurs when the Access Domain policy is disabled and the Identity Domain policy is enabled when logging into the Policy Manager as a valid user.

Solution: For security reasons, both the Policy Manager (`/access`) policy and Identity Domain (`/identity`) policy must be enabled and protected when logging in to the Policy Manager.

To make FrontPage work correctly with Oracle Access Manager, the settings for IIS must be set up to allow Oracle Access Manager to do all of the authentication and authorization.

To allow Oracle Access Manager to do all authentication and authorization

1. The Web server needs to run as a user that has full control of all directories containing Web content.
2. Use the Web server MMC and click the **directory security** tab.
3. Click the anonymous user and authentication **Edit** button.

Make sure that only the **allow anonymous users** check box is checked. Un-check the other two (**basic authentication** and **nlm authentication**)

4. Add the Web server process user (such as IUSR_OBLIX) to the FrontPage admins by using the FrontPage server admin tools (these are different for every version of FrontPage).

Symbols

_territory, 3-3

A

About

- Access Server Installation, 8-1
 - Active Directory, A-1
 - ADAM, B-1
 - Adding New Access Servers to an Upgraded Environment, 8-2
 - Apache v1.3 and Oracle Access Manager, 16-2
 - Changing Directory Server Hosts, D-1
 - Directory Certificates, C-1
 - Identity Server Installation, 4-1
 - Installation, 1-1
 - Installing Audit-to-Database Components, 13-1
 - Installing in Multi-Language Environments, 3-1
 - Installing Multiple Access Servers, 8-2
 - Installing Multiple Identity Servers, 4-2
 - Installing Multiple Policy Managers, 7-2
 - Installing Multiple WebPass Instances, 5-2
 - Installing the Software Developer Kit, 14-1
 - Language Packs and Installation, 12-1
 - OHS and Oracle Access Manager, 16-1
 - Oracle Access Manager and Active Directory, A-2
 - Oracle Access Manager and Active Directory Forests, A-4
 - Oracle Access Manager Implementations with Oracle Virtual Directory, 10-2
 - Oracle Access Manager Object Classes, 6-7
 - Policy Manager Installation and Setup, 7-1
 - Setting Up the Identity System, 6-1
 - Silent Mode Options File, 15-1
 - SNMP Agent, 11-1
 - WebGate Installation, 9-1
 - WebPass Installation, 5-1
- Access Domain, 7-13
- formerly named NetPoint or COREid Access Manager Domain, xxii
- Access Management API
- now named Policy Manager API, xxii
- Access Manager
- now named Policy Manager, xxi
- Access Manager API, 14-1
- formerly named Access Server API, xxii
- Access Manager SDK, 14-1
- formerly named Access Server SDK, xxii
- Access Server
- Adding to an upgraded environment, 8-2
 - GUI Method, 8-5
 - guidelines, 2-7
 - installation, 8-1
 - prerequisites checklist, 8-3
 - Starting the Installation, 8-5
- Access Server API
- now named Access Manager API, xxii
- Access Server ID, 8-7
- Access Server SDK
- now named Access Manager SDK, xxii
- Access Server Timeout Threshold, D-4
- Access System
- Guidelines, 2-6
 - transport security, 9-6
- Access System Console, 9-2
- AccessGate, 9-1, 14-1
- creating, xix, 4
- Account name
- SNMP, 11-3
- Active Directory, 2-7, 2-23
- Installation and Setup Considerations, A-7
 - Issues, E-3
 - parent-child relationships, A-6
 - schema, B-4
 - schema update file, 1-6
- Active Directory Application Mode
- see ADAM, B-1
- ADAM, 2-23, 2-24, B-4
- about, B-1
 - administrators, B-10
 - ADPs, B-3, B-4, B-10
 - ADSI, B-10
 - differences with Active Directory, B-8
 - instance replication, B-7
 - instances, B-3, B-4
 - ldifde, B-6
 - manual schema update, 1-5
 - namespace, B-4
 - naming contexts, B-4
 - objectclass attribute value, B-10

- partitions, B-3
- preparation, B-9
- proxy object, B-8, B-10
- root DN, B-7
- schema, B-4, B-5
- schema update file, 1-7
- searchbase, B-4
- static auxiliary classes, B-5
 - Windows security principal, B-6
- ADAM_oblix_schema_add.ldif, B-5
- ADAM_user_schema_add.ldif, B-5
- ADAMAuxSchema.ldif, B-5
- Adapters
 - Oracle Virtual Directory, 14
- AddDefaultCharset directive, 17-6
- administrative rights
 - Oracle Access Manager, 2-4
- Administrator language, 3-1
 - Removal, 20-1, 20-3
- administrators
 - ADAM, B-10
- ADPs
 - ADAM, B-10
- ADSI, 2-7, A-10
 - ADAM, B-10
- AES encryption scheme, 2-8
- Aggregated Namespaces, 10-4, 10-8
- AIX
 - NTP, 2-3
- AL32UTF8, 3-4
- all directory servers, 2-20
- AM Service State
 - now named Policy Manager API Support Mode, xxii
- AMERICAN_AMERICA.US7ASCII, 3-3
- Anonymous
 - authentication scheme, 7-13
- Anonymous authentication scheme
 - formerly named NetPoint or COREid None, xxii
- Apache, 2-7, 2-8, 2-15
- Apache v1.3, 16-2
- Apache v1.3, OHS, and IHS
 - Web Server Support, 16-5
- Apache v2
 - Architecture, 17-4
 - Directives, 17-33
 - HTTP Server, 17-3
 - Limitations, 17-5
 - Portable Runtime library, 17-4
 - Web server, 17-1
 - Web server tips, 17-34
 - Web server troubleshooting, 17-34
- APACHE_WebGate, 17-3
- APACHESSL_WebGate, 17-3
- assigning a bind DN, 2-18
- Associating
 - WebGate and Access Server, 9-4
- attribute index cleanup, 20-2
- auditing, 13-1
- authentication, xviii, 4

- default schemes, xxii
- Authentication scheme
 - Anonymous, 7-13
 - Basic Over LDAP, 7-13
 - Client Certificate, 7-13
 - Oracle Access and Identity, 7-13
- authorization, xviii, 4
- Auto Configure
 - schema update, 1-6

B

- backward compatibility, 4-4, 8-3
- Base Apache v1 Web Server, 16-6
- Basic Over LDAP
 - Authentication scheme, 7-13
- Bind DN, 2-18
- Bind DN Values
 - Directory Servers, 4-11

C

- cancel
 - installation, 2-28
- Cert Mode, 2-13
- cert_authn.dll, 9-10
- cert8.db, 2-20
- certificate
 - Identity Server, 4-9
- Certificates
 - Generated by a Subordinate CA, 6-2
- changing MIME type settings, 19-1
- .charset, 3-3
- Checklist
 - Access Server prerequisites, 8-3
 - Identity Server Prerequisites, 4-4
 - Identity System setup, 6-4
 - Independent Installation of Language Packs, 12-4
 - installation preparation, 2-28
 - Language Pack installation, 12-4
 - Policy Manager, 7-2
 - SNMP Prerequisites, 11-2
 - WebGate installation prerequisites, 9-2
 - WebPass Prerequisites, 5-2
- choosing a unique ID for each user, 19-2
- Client Certificate
 - Authentication scheme, 7-13
- clones, 15-30
- Cloning
 - Installed Components, 1-8
- cloning, 15-29, 15-30
- Cloning installed components, 1-8
- Command-line tools, 3-2
- Community Name
 - SNMP, 11-3
- Completing
 - httpd.conf Updates, 9-12
 - Identity System Setup, 6-9
 - Policy Manager Setup, 7-14
 - WebGate Installation

- Domino, 18-5
- WebGate installation with IIS, 9-9
- component security, 2-5
- Configuration
 - SNMP, 11-3
- configuration data, 2-21, 2-22, 2-25
 - formerly named Oblix data, xxii
 - Removal, 20-3
- configuration details
 - Identity Server, 4-7
- Configuration DN, 2-21, 2-25, 6-6, 7-12, 8-7
- configuration tree
 - formerly named Oblix tree, xxii
 - Removal, 20-2
- configureAAAServer, D-8
- Configuring
 - Authentication Schemes
 - Policy Manager setup, 7-13
 - Failover between an Access Server and WebGate, D-3
 - Identity System attributes, 6-10
 - Master Administrators, 6-8
- Confirming
 - Language Status, 12-5
 - Object Class Changes, 6-8
 - Policy Manager Setup, 7-14
 - WebGate Installation, 9-13
- considerations
 - Novell, 6-10
 - SNMP agent, 11-1
- Console Method
 - Access Server, 8-5
 - WebGate, 9-5
 - WebPass, 5-3
- Console method, 4-5
- Console-based tools, 3-2
- Contacting Oracle, 19-2
- COREid
 - now named Oracle Access Manager, xxi
- COREid Access Manager Domain
 - now named Access Domain, xxii
- COREid Administrator
 - now named Master Administrator, xxii
- COREid Basic Over LDAP authentication
 - now named Oracle Access and Identity, xxii
- COREid for AD Forest Basic Over LDAP authentication
 - now named Oracle Access and Identity for AD Forest Basic over LDAP, xxii
- COREid Identity Domain
 - now named Identity Domain, xxii
- COREid None authentication
 - now named Anonymous authentication, xxii
- COREid System Console
 - now named Identity System Console, xxi
- COREID_NLS_LANG, 2-5, 3-3
 - Windows Systems, 3-4
- Creating
 - a WebGate Instance, 9-2
 - Access Server Instance in the System Console, 8-4

- Customer Schemas, 18

D

- Data Anywhere, 2-22, 10-1
 - schema update file, 1-7
- data storage
 - requirements, 2-21
- database profiles, 2-25
- DB profile, 2-20
- DB profiles, 2-22
- defining
 - Master Administrators, 6-8
- defining communication details
 - Identity Server, 4-8
- defining directory server details
 - Identity Server, 4-9
- Directory certificates, C-1
- directory security, 2-5
- directory server
 - bind DN values, 4-11
 - communication, 2-19
 - requirements, 2-17
- directory server hosts, D-1
- Directory Structure, 3-6
- disjoint namespace, 2-25
- Disjoint Searchbase
 - Oracle Virtual Directory, 10-5
- disk space requirements, 2-10
- Domino, 2-8
 - Tips, 18-6
 - Web server, 18-1
 - Web servers, 18-6
- Downloading and Compiling
 - Base Apache v1 Web Server, 16-6
- DSAPI filter, 18-4, 18-5
- dynamically-linked auxiliary classes, A-4

E

- Editing
 - Silent Mode Options File, 15-3
- EMailAdminsGroup, 6-4
- Embedded Virtual Data Source, 10-3
- Enabling
 - Java, 19-1
 - JavaScript, 19-1
- encoding, 4-3
- encryption schemes, 2-8
- English-only installation, 3-1
- environment variable, 2-5
- exclude_attrs-ad.xml, 2-24
- exclude_oblix_attrs.xml, 2-22
- exclude_user_attrs.xml, 2-22, 2-24

F

- failover, 2-7, D-2
- Failover Threshold, D-2, D-3
- Features
 - Oracle Virtual Directory, 10-4

- features
 - new, xxi
- Federated Data Stores, 10-4
- Federated Data stores, 10-4
- Federation, 10-3
- Files
 - DirectoryName_oblix_schema_delete.ldif, 20-2
 - DirectoryName_user_schema_delete.ldif, 20-2
 - OID_oblix_schema_index_delete.ldif, 20-2, 20-4
 - OID_user_index_delete.ldif, 20-2, 20-4
 - options.txt, E-17
- Finishing
 - WebGate Installation, 9-8
- firewall, 2-6, 2-7

G

- General Guidelines
 - Oracle Access Manager components, 2-4
 - Web servers, 2-16
- globalparams.xml, 2-23, 8-3
- GPS-based clocks, 2-4
- Group, 2-26
- Group Object Class, 2-26, 6-7
- Group Objects
 - Managed through Oracle Internet Directory, 6-3
- GroupofUniqueNames, 2-26, 6-7, B-5
- grouptype attribute, B-4
 - ADAM, B-4
- GUI Method, 4-5, 9-5
 - Access Server, 8-5
 - WebGate, 9-5
 - WebPass, 5-3
- Guidelines
 - Access Server, 2-7
 - Access System, 2-6
 - directory server communication, 2-19
 - Identity System, 2-6
 - Installing Oracle Access Manager with Active Directory, A-10
 - Policy Manager, 2-6
 - Setting up AD SI, A-12
 - Setting up an LDAP open bind for Active, A-13
 - WebGate, 2-8

I

- IBM Directory Server, 2-22
 - schema update file, 1-7
- IBM HTTP Server, 17-3
 - see also IHS, 2-15, 17-2
- Identity Domain, 7-13
 - formerly named COREid Identity Domain, xxii
 - formerly named NetPoint Identity Domain, xxii
- Identity Server
 - Adding to an upgraded environment, 4-3
 - configuration details, 4-7
 - define communication details, 4-8
 - defining directory server details, 4-9
 - install certificate, 4-9

- installation, 4-1
- Instance Name, 20-4
- prerequisites checklist, 4-4
- request certificate, 4-9
- schema extension, 4-10
- Identity Server Timeout Threshold, D-2
- Identity System
 - configuring, 0-xviii
 - guidelines, 2-6
 - IdentityXML, 0-xix, 1-4
 - landing page, 5-8
 - setup considerations, 6-2
 - setup Oracle Internet Directory, 6-3
 - setup Prerequisites Checklist, 6-4
- Identity System attribute configuration, 6-10
- Identity System Console, 5-8
 - formerly named COREid System Console, xxi
- Identity System Directory Server and Data Location Details, 6-5
- Identity System Setup
 - completing, 6-9
 - object class details, 6-6
- Identity Systemr
 - Language Pack installation, 4-6
- IdentityXML Calls, E-9
- IHS, 2-15, 16-6, 17-6
 - Limitations, 17-5
 - Web server, 17-1
 - Web serverr, 17-4
- IHS v2 Web servers, 2-8
- IIS, 2-8
 - SSL with WebGate, 9-9
 - WebGate, 9-7, 9-14
- IIS Virtual Web server, 2-9
- IIS Web server, 2-7
 - see also ISAPI, 2-15
- Implementation
 - Oracle Virtual Directory, 10-9
- implementation architecture
 - Oracle Virtual Directory, 10-11
- Important Notes, 19-1
- InetOrgPerson, 2-26, 6-7, 10-18, B-5, E-11
- Install Certificate
 - WebPass, 5-5
- install_options.txt, 15-2
- installation, xviii
 - cancel, 2-28
 - directories, 2-10
 - Identity Server, 4-1
 - multiple Identity Servers, 4-2
 - options, 1-5
 - preparation checklists, 2-28
 - Prerequisites, 2-1
- Installation Considerations
 - Language Packs, 12-3
- installionm
 - methods, 1-8
- Installing
 - Access Server, 8-5
 - additional Identity Servers on Windows, 4-12

- Domino Security (DSAPI) Filter, 18-4
- First Identity Server, 4-10
- from the GUI vs. Command Line, 1-8
- Identity Server certificate, 4-9
- Multiple WebGates, 9-2
- Oracle Access Managerr
 - introduction, 1-1
 - Policy Manager, 7-1
- Postgate ISAPI Filter for WebGate, 9-11
- postgate.dll on IIS for WebGate, 9-10
- SNMP Agent, 11-2
- WebGate, 9-1, 9-5
- WebPass, 5-2
- with Language Packs, 3-5

Integrating

- Oracle Virtual Directory Engine (VDE), 1-7

Internationalized data, 3-2

introduction, 1-8

iPlanet Web server

- see NSAPI, 2-15

ISA Proxy Servers, 2-9

ISAPI, 2-15

ISAPI Webgate filter, 9-11

J

- Java applet
 - WebPass, 19-1
- JDBC Driver, 14
- JNDI Driver, 14

K

- key3.db, 2-20

L

- LANG, 3-2
- LANG environment vairable, 2-5
- /langTag folder, 3-6
- language, 2-5, 3-3
 - directories, 3-7
 - precedence
 - command-line tools, 3-3
- Language Pack, 3-1
 - Installation Considerations, 12-3
 - Prerequisites Checklist, 12-4
 - Removal, 20-1, 20-3
 - WebPass, 5-3
- Language Packs, 2-5
 - Identity System, 4-6
- Latin-1, 4-3
- ldapmodify, 20-3
- ldapmodify tool, 1-6
- ldapmodify.exe, 17
- ldapreferentialintegrityparams.xml, 2-22
- ldif file, 1-6
- ldifde
 - for ADAM, B-6
- Ldifde.exe
 - and the Active Directory schema, 1-6

- and the ADAM schema, 1-7
- libgcc_s.so.1, 2-5, 2-6
- libstdc++.so.5, 2-5, 2-6
- Linux liibraries, 2-5
- load balancing, 2-7
- locale, 2-5
- Lotus Notes, 18-5, 18-6

M

- manual, 6-10
- Manual Schema Update, 1-6
- Manually Configuring
 - WebGate Web Server, 9-8
- Master Access Administrators, 6-9
- Master Administrator, 6-8
 - formerly named COREid Administrator, xxii
 - formerly named NetPoint Administrator, xxii
- Master Identity Administrators, 6-9
- MaxClients, 17-33
- MaxSpareServers, 17-33
- MaxSpareThreads, 17-33
- Metalink, 16-5
- Microsoft CA certificate, A-15
- MIME type mappings, 19-1
- mime_types.lst, 19-1
- mime_types.xml, 19-1
- MinSpareServers, 17-33
- MinSpareThreads, 17-33
- mod_ssl, 17-7
- MPM, 17-4, 17-33
- mpm_winnt, 17-4, 17-33
- mpm_worker_module, 17-4, 17-14, 17-34
- MPMs, 17-4
- Multi-Language Environments, 2-5, 3-1
- multiple Identity Servers, 4-2
- Multiple instances
 - Removal, 20-4
- Multiple User Data Directories, 6-2
- Multi-Process Modules
 - see also mpm, 17-4
- Multi-Table Database, 10-3

N

- name changes, xxi
- names, new, xxi
- namespace
 - ADAM, B-4
- Namespace Aggregation, 10-8
- native schema, 17
- native schemas, 15
- net start w3svc, 9-12
- net stop iisadmin, 9-12
- NetPoint
 - now named Oracle Access Manager, xxi
- NetPoint Access Manager Domain
 - now named Access Domain, xxii
- NetPoint Access Protocol
 - now named Oracle Access Protocol, xxii

- NetPoint Administrator
 - now named Master Administrator, xxii
- NetPoint Basic Over LDAP authentication
 - now named Oracle Access and Identity, xxii
- NetPoint for AD Forest Basic Over LDAP authentication
 - now named Oracle Access and Identity for AD Forest Basic over LDAP, xxii
- NetPoint Identity Domain
 - now named Identity Domain, xxii
- NetPoint Identity Protocol
 - now named Oracle Identity Protocol, xxii
- NetPoint None authentication
 - now named Anonymous authentication, xxii
- NetPoint SAML Services
 - now named Oracle Identity Federation, xxi
- NetScape Web server
 - see NSAPI, 2-15
- NETWORK account, 2-7
- Network Management System
 - see also NMS, 11-1
- Network Time Protocol, 2-3
 - see also NTP, 2-3
- NLS_LANG, 2-5, 3-3
- nlstrl, 3-6
- NMS, 11-1
- Novell
 - considerations, 6-10
- Novell Directory Server
 - schema update file, 1-7
- Novell eDirectory, 2-23, 2-24
- np_sync, 15-30
- NSAPI, 2-15
- NTP
 - HP-UX, 2-3
 - Unix, 2-3
 - Windows, 2-4
- ntp.conf, 2-3

O

- object classes, 6-7
- objectclass attribute value
 - ADAM, B-10
- Oblix data
 - now named configuration data, xxii
- Oblix tree
 - now named configuration tree, xxii
- oblixlock.dll, 9-10, 9-12
- oblixppcatalog.lst, 4-4
- obnls.xml configuration file, 3-6
- obpolicybase, E-20
- OctetString Virtual Directory Engine (VDE)
 - now named Oracle Virtual Directory, xxi
- OHS, 2-15, 16-1, 16-4
 - Web Component Caveats on Linux, 16-2
 - Web Component Caveats on Linux and Windows, 16-2
 - Web server, 17-1
- older WebGates, 2-8

- onfiguration data
 - Active Directory, 2-23
- Open Mode, 2-12
- Oracle Access and Identity
 - Authentication scheme, 7-13
- Oracle Access and Identity authentication
 - formerly named NetPoint or COREid Basic Over LDAP, xxii
- Oracle Access and Identity for AD Forest Basic over LDAP
 - formerly named NetPoint or COREid for AD Forest Basic Over LDAP, xxii
- Oracle Access Manager, E-15
 - administrative rights, 2-4
 - attribute index cleanup, 20-2
 - formerly NetPoint or COREid, xxi
 - Requirements, 2-4
- Oracle Access Manager Layer
 - Oracle Virtual Directory, 12
- Oracle Access Protocol
 - formerly named NetPoint Access Protocol, xxii
- Oracle Application Server 10g Release 2 (10.1.2)
 - also available as Oracle COREid 7.0.4, xxi
- Oracle COREid release 7.0.4
 - also available as part of Oracle Application Server 10g Release 2 (10.1.2), xxi
- Oracle HTTP Server
 - see also OHS, 2-15, 17-2
- Oracle HTTP Server (OHS), E-23
- Oracle Identity Federation, xxi
 - formerly SHAREid, xxi
- Oracle Identity Protocol
 - formerly named NetPoint Identity Protocol, xxii
- Oracle Internet Directory, 2-18, 2-22, 2-25, 2-26
 - group objects, 6-3
 - Identity System setup, 6-3
 - remove Oracle Access Manager schema extensions, 20-4
 - schema update file, 1-7
- Oracle Virtual Directory, 10-1, 10-5, 10-11
 - Adapters, 14
 - Disjoint Searchbases, 10-5
 - implementation, 10-9
 - implementation architecture, 10-11
 - Implementation Layers, 10-12
 - JDBC Driver, 14
 - JNDI Driver, 14
 - Oracle Access Manager Layer, 12
 - relational database, 10-9
 - schema extension, 15
 - System Layer, 12
 - Target Data Store Layer, 12
 - unified searchbase, 10-5, 10-6
- Oracle Virtual Directory Server
 - formerly OctetString Virtual Directory Engine (VDE), xxi
- orclGroup, 6-3
- organizationalPerson, 2-26

P

- pass phrase, 2-12
- Peoxwsuew
 - Somino
 - To install the Domino Web server on Unix, 18-2
- Person Object Class, 2-26, 6-7
- platform requirements, 2-26
- Policy Base, 7-12, 8-7
- policy data, 2-21, 2-22, 2-26
- policy domain
 - default, xxii
- Policy Manager
 - clock, 2-2
 - confirm setup, 7-14
 - formerly named Access Manager, xxi
 - installing, 7-1
 - Prerequisites Checklist, 7-2
 - setting up, 7-9
 - SSL-enabled communication, 2-7
- Policy Manager API, xxii
 - formerly named Access Management API, xxii
- Policy Manager API Support Mode
 - formerly named AM Service State, xxii
- Policy Manager Guidelines, 2-6
- policybase, 2-21, 2-26
- postgate filter, 9-11
- postgate.dll, 9-12
- prefork MPM, 17-4
- Preparation
 - synchronizing system clocks, 2-2
- Preparing
 - Access System Guidelines, 2-6
 - for Installation, 2-1
 - Identity System Guidelines, 2-6
 - Linux host machines, 2-5
- Preparing ADAM, B-9
- Prerequisites
 - installation, 2-1
- Prerequisites Checklist
 - Identity System setup, 6-4
 - Language Packs, 12-4
 - Policy Manager, 7-2
 - SNMP, 11-2
 - WebGate, 9-2
- private key, 2-12
- Procedure
 - Access Server
 - To add a new Access Server to an upgraded environment, 8-3
 - To create an Access Server instance, 8-4
 - To finish installation, 8-8
 - To reconfigure the Access Server for the new directory server instance, D-8
 - To specify a transport security mode, 8-6
 - To specify directory server details, 8-6
 - To start the installation, 8-5
 - Access System
 - To allow Oracle Access Manager to do all authentication and authorization, E-28
 - To change capitalization in an AccessGate description, E-25
 - To regenerate a shared secret, E-18
 - To restore default Administrator language, E-17
- Active Directory
 - To enable dynamically-linked auxiliary classes, E-4
- Directory Certificate
 - To change the directory profile, C-4
 - To create the new certificate store, C-2
- Domino
 - To generate the keyring and stash files, 18-3
 - To set up first Domino server, 18-2
 - To start Domino server, 18-3
- Failover
 - To configure failover between an Access Server and WebGate, D-3
 - To configure failover between an Identity Server and WebPass, D-2
- General
 - To enable Jave and JavaScript on the client, 19-1
- Identity Server
 - To add a new Identity Server to an upgraded environment, 4-3
 - To configure the Identity Server to communicate with a new directory server instance, D-5
 - To define communication details, 4-8
 - To finish the installation, 4-13
 - To identify this Identity Server, 4-7
 - To install, 4-6
 - To recycle an instance name, 20-5
 - To specify a transport security mode, 4-7
 - To specify Active Directory details on a Windows system, 4-12
 - To specify directory server details for the first Identity Server, 4-10
 - To start the installation in Console method, 4-5
 - To start the installation in GUI method, 4-5
 - To troubleshoot the Identity Server, E-8
- Identity System
 - To edit mime_types files, 19-2
- Identity System setup
 - To assign Master Administrators, 6-9
 - To associate a WebPass, 6-13
 - To complete setup, 6-10
 - To confirm object class changes, 6-8
 - To ensure NDS works, 6-11
 - To specify directory server details, 6-5
 - To specify Person and Group object class details, 6-7
 - To start, 6-5
- IIS
 - To verify the Web server configuration for Policy Manager, 7-9
- Installers
 - To store Oracle Access Manager installers for

- installation, 2-27
- Language
 - To enable additional Administrator languages (Access System), E-16
 - To prepare to install Language Packs in concert with Oracle Access Manager, 3-5
 - To set COREID_NLS_LANG, 3-4
 - To set NLS_LANG, 3-4
- Language Packs
 - To confirm which languages are enabled, 12-5
 - To perform independent installation, 12-4
- Linux
 - To install libgcc_s.so.1 and libstdc++.so.5 on Linux hosts, 2-6
- MetaLink
 - To use MetaLink, 16-5
- NDS
 - To change the order using the NDS Console1, 2-24
- OHS
 - To resolve the failure to start OHS, E-23
- Oracle Internet Directory
 - To tune Oracle Internet Directory for Oracle Access Manager, 4-13
- Policy Manager
 - To automatically update Web server configuration, 7-7
 - To avoid a IIS and Sun Web server conflicts, 7-2
 - To complete authentication scheme setup, 7-13
 - To complete setup, 7-14
 - To confirm setup, 7-15
 - To delete a leftover policy profile, E-20
 - To finish the installation, 7-8
 - To identify the location of policy data, 7-4
 - To manually configure your Web server, 7-8
 - To manually update Web server configuration, 7-8
 - To reconfigure the Policy Manager for the new directory server instance, D-6
 - To specify a transport security mode, 7-6
 - To specify directory server details during setup, 7-11
 - To specify directory server type and configuration details, 7-5
 - To specify existing directory server details, 7-5
 - To start set up, 7-10
 - To start the installation, 7-3
- Silent Mode
 - To install new components in silent mode, 15-2
- SNMP
 - To finish installation, 11-4
 - To specify SNMP Agent details, 11-3
 - To start installing, 11-2
- Transport Security
 - To change transport security modes on the Access System, E-22
 - To change transport security modes on the Identity System, E-21
 - To generate certificates for cache flushing, E-27
- Uninstall
 - To uninstall Oracle Access Manager components, 20-3
- Unix
 - To specify a temporary directory on Unix systems, 2-11
- Web server
 - To remove and re-install IIS DLLs, E-24
- WebGate
 - To add cert_authn.dll as an ISAPI filter, 9-9
 - To assign an Access Server to the WebGate, 9-4
 - To define a WebGate instance, 9-2
 - To enable SSL for IIS, 9-9
 - To finish installation, 9-8
 - To install the postgate ISAPI filter, 9-11
 - To manually configure a Web server, 9-8
 - To manually update your Web server configuration, 9-7
 - To order ISAPI filters, 9-10
 - To protect a Web site (not the default site), 9-12
 - To provide configuration details, 9-6
 - To reposition postgate IIS filter, 9-12
 - To set IIS 5.0 isolation on IIS 6, 9-11
 - To specify a transport security mode, 9-6
 - To start installation, 9-5
 - To update httpd.conf, 9-13
 - To update the Web server configuration, 9-7
- WebPass
 - To automatically update your Web server configuration, 5-5
 - To change the WebPass password for simple/cert mode, E-9
 - To confirm your installation, 5-8
 - To establish communications with the Identity Server, 5-8
 - To finish installation, 5-6
 - To manually update your Web server configuration, 5-6
 - To reconfigure the WebPass, E-9
 - To reconfigure Webpass mode, E-9
 - To specify a transport security mode, 5-4
 - To specify Web server configuration details, 5-4
 - To start the installation, 5-3
 - To update your Web server configuration, 5-7
 - To verify IIS Web server configuration, 5-7
 - To verify Web server in Simple or Cert mode, 5-7
- Protecting
 - When the default site is not setup, 9-12
- proxy object
 - ADAM, B-8, B-10
 - proxy_module, 17-4
 - public key, 2-12

R

- RC4 encryption scheme, 2-8
- RC6 encryption scheme, 2-8
- RDBMS databases, 10-5
- Recycling Identity Server Instance Name, 20-4
- Re-install
 - default Administrator Language, E-17
 - Oracle Access Manager, 20-2
- Relational Database
 - Oracle Virtual Directory, 10-9
- Removal
 - Administrator language, 20-2
 - attribute index, E-21
 - configuration data, 20-3, E-21
 - configuration tree, 20-2
 - Language Packs, 3-7, 20-1, 20-3
 - Multiple Instances, 20-4
 - Oracle Access Manager, 20-1
 - Schema and Data Changes, 20-2
 - schema extensions, 20-3
 - user data, E-21
 - Web server configuration, 20-3
 - Web server configuration changes, 20-4
- Replicating an Installed Oracle Access Manager
 - Component, 1-7
- Replicating Components, 15-1
- Replication
 - ADAM instance, B-7
 - Silent Mode, 1-8
- Request Certificate
 - Identity Server, 4-9
 - WebPass, 5-5
- Requirements
 - Oracle Access Manager, 2-4
- requirements
 - directory server, 2-17
- reverse proxy, 17-3
- root DN
 - ADAM, B-7

S

- schema
 - Active Directory, B-4
 - ADAM, B-4, B-5
- Schema Extension
 - ADAM, B-5
 - Identity Server, 4-10
 - Oracle Virtual Directory, 15
- schema extension
 - cleanup configuration data, 20-2
 - cleanup user data, 20-2
 - removal, 20-3
- Schema Mapping, 10-4, 10-8
- Schema update
 - manually configuring attributes, 1-6
- Schema update file
 - ADAM_user_schema_add.ldif, 1-7
 - ADAMAuxSchema.ldif, 1-7
 - ADAuxSchema.ldif, 1-6
 - ADdotNetSchema_add.ldif, 1-6
 - ADSchema.ldif, 1-6
 - ADUserSchema.ldif, 1-6
 - iPlanet_oblix_schema_add.ldif, 1-7
 - iPlanet_user_schema_add.ldif, 1-7
 - iPlanet5_oblix_index_add.ldif, 1-7
 - iPlanet5_user_index_add.ldif, 1-7
 - NDS_oblix_index_add.ldif, 1-7
 - NDS_oblix_schema_add.ldif, 1-7
 - NDS_user_index_add.ldif, 1-7
 - NDS_user_schema_add.ldif, 1-7
 - OID_oblix_schema_add.ldif, 1-7
 - OID_oblix_schema_delete.ldif, 1-7
 - OID_oblix_schema_index_add.ldif, 1-7
 - OID_user_index_add.ldif, 1-7
 - OID_user_schema_add.ldif, 1-7
 - OID_user_schema_delete.ldif, 1-7
 - V3.oblix.ibm_at.ldif, 1-7
 - V3.oblix.ibm_oc.ldif, 1-7
 - V3.user.ibm_at.ldif, 1-7
 - V3.user.ibm_oc.ldif, 1-7
 - VDE_user_schema_add.ldif, 1-7
- SDK, 4-2, 14-1
- Searchbase, 6-6
- searchbase, 2-21, 2-24, 7-12
 - ADAM, B-4
- secure request timestamp, 2-2
- SecureWay
 - see also IBM Directory Server, 2-22
- security principal
 - ADAM, B-4
- Separate Data Storage, 4-10
- Set up the Identity System, 6-1
- Setting, 3-4
 - Environment Variables for Command-Line Tools, 3-2
 - NLS_LANG
 - Unix Systems, 3-4
- Setting COREID_NLS_LANG
 - Unix Systems, 3-4
 - Windows Systems, 3-4
- Setting NLS_LANG
 - Windows Systems, 3-4
- Setting Up
 - Policy Manager, 7-9
- setup considerations
 - Identity System, 6-2
- SHAREid
 - now named Oracle Identity Federation, xxi
- Siemens DirX, 2-22
- silent mode options file, 15-2
- Silent Mode Replication, 1-8
- Simple Mode, 2-12
- Simple Network Management Protocol
 - see also SNMP, 11-1
- Single-Table Database, 10-3
- Sleep For (seconds), D-2, D-4
- SNMP, 11-1, 11-3
 - account name, 11-3
 - Agent Installation Considerations, 11-1

- Installation Prerequisites Checklist, 11-2
- Installing, 11-2
- Software Developer Kit
 - see also SDK, 4-2, 14-1
- Software Developer's Kit, 14-1
- Specifying
 - SNMP Configuration Details, 11-3
 - WebGate Configuration Details, 9-6
 - WebGate Transport Security Mode, 9-6
- Specifying Object Class Details, 6-6
- Split Profile, 10-3, 10-4, 10-7
- SSL, 2-13
- SSL-enabled communication
 - Policy Manager, 2-7
- static auxiliary classes
 - ADAM, B-5
- static auxiliary schema, A-4
- statically-linked auxiliary classes, A-3
- Stronghold Requirements, 16-4
- Sun directory server, 2-18, 2-22, 2-25
 - schema update file, 1-7
- Sun Web server
 - see NSAPI, 2-15
- Sun Web Servers
 - WebGate, 9-7
- super directory, 10-2, 10-8
- Synchronizing
 - installed components, 1-8
- synchronizing, 15-30
- Synchronizing installed components, 1-8
- Synchronizing system clocks, 2-2
- system clock requirements, 2-2

T

- Target Data Store Layer
 - Oracle Virtual Directory, 12
- Target Database Tables, 17
- Target Directory Schemas, 17
- Task overview
 - Adding an instance and installing the Access Server, 8-1
 - Adding an instance and installing WebGate, 9-1
 - Changing Directory Server hosts, D-1
 - Choosing your installation options, 1-5
 - Completing IIS WebGate installations, 9-9
 - Defining directory server communication security, 2-20
 - Enabling directory SSL after Oracle Access Manager installation, C-2
 - Enabling dynamically-linked auxiliary classes for Active Directory, A-4
 - Ensuring interaction with Oracle Internet Directory, 6-3
 - Installing a Language Pack independently, 12-3
 - Installing a WebPass, 5-2, 5-3
 - Installing additional Identity Servers, 4-2
 - Installing an Identity Server, 4-4
 - Installing multiple Access Servers, 8-2
 - Installing Oracle Access Manager, 1-3

- Installing Oracle Access Manager with Active Directory, A-14
- Installing Oracle Access Manager with ADAM, B-9
- Installing the Access Server includes, 8-5
- Installing the Policy Manager, 7-3
- Installing the WebGate includes, 9-5
- Preparing to install Oracle Access Manager, 2-1
- Preparing your directory server, 2-17
- Preparing your Web server, 2-14
- Setting up the Identity System, 6-4
- Setting up the Policy Manager, 7-9
- Setting up your environment for Active Directory, A-14

Terms

- Oracle Virtual Directory, 10-2

- Testing Your Installation, E-15

- ThreadsPerChild, 17-33

- timestamp

- secure request, 2-2

Tips

- Domino, 18-6

- tips

- Apache-based Web servers, 17-34

- transfilter.dll, 9-10

- transport security

- Access System, 9-6

- guidelines, 2-11

Troubleshooting

- Access Server Crashes on Apache, E-23

- Access Server Installation Halts, E-13

- Access Server Naming, E-25

- Active Directory or ADAM search problem, E-10

- Active Directory Search Halts, E-3

- ADAM Issues, E-5

- Adding User to Replicated Directory, E-22

- Administrator language, E-16

- ADSI Cannot Be Enabled for this DB Profile, E-4

- Application Has Not Been Set Up, E-6

- Authenticate Resource on Internet Explorer, E-2

- Browser Issues, E-1

- Cannot Delete Policy Manager Policy Profile, E-20

- Cannot Set Up Identity System, E-7

- CGI Programs Do not Run After Installation, E-13

- Character Display Issues, E-1

- Checking Access or Identity Server Availability, E-7

- Could Not Get DB Profile, E-8

- Data Corruption, E-22

- Directory Server Issues, E-3

- Dynamically-Linked Auxiliary Classes, E-4

- Enabling WebGate Diagnostics, E-25

- Error Messages After Installing WebGate, E-26

- Failure to write to log file errors, E-23

- File Replace Warning When Installing on Windows, E-13

- Garbled Password Message, E-16

- Identity Server Does Not Start, E-8

- Identity Server Logged You in but Access System Logged You Out, E-28
- Identity Server Logged You In, Access System Logged You Out, E-17
- Identity System Components May Crash, E-8
- Identity System Issues, E-6
- IdentityXML Calls Fail After WebGate Install, E-9
- Idle Session Time, E-27
- IIS and Windows Issues, E-10
- Installation Directory Names, E-15
- Installation Fails with a "bad credentials error (49)", E-14
- Installation Issues, E-12
- Installer Prompts to Replace DLL Files, E-14
- Installing Administrator Language Packs, E-16
- Installing WebGate and an Identity Server in Same Directory, E-26
- Installing WebGate with Apache on AIX, E-15
- Issues with Oracle Virtual Directory Implementations, E-10
- Language Issues, E-15
- Language Packs (Policy Manager and WebGate in Same Directory), E-16
- Loading the Directory in Secure Mode, E-27
- Login Issues, E-17
- Loss of Access, E-23
- Loss of access to Web pages, E-23
- Maximum Cookie Session Time, E-27
- Microsoft Internet Explorer 6 issues with Sun, E-2
- Multi-Value Attribute Problems with Oracle Virtual Directory, E-10
- Oracle HTTP Server (OHS) Fails to Start, E-23
- Oracle Virtual Directory
 - Secondary Data Store Problems, E-11
- PCLOSE Error Sun Web server, E-24
- Peer Does Not Use Oracle Access Protocol, E-27
- Policy Manager Issues, E-19
- Quitting a Windows Installation, E-14
- Random bug report pages, E-23
- Receiving Access Server Down Errors, E-26
- Receiving Bug Report After Replication Attempt, E-28
- Receiving Repeated Login Prompts, E-18
- Re-Installing IIS DLLs, E-24
- Re-installing Oracle Access Manager with Oracle Internet Directory, E-20
- Removal Issues, E-21
- Removing Default Administrator Language, E-17
- Restricting Access to Oracle Access Manager, E-19
- Running as Non-Root User on AIX, E-15
- schema cleanup, E-20
- Search and Query Error Message (Defect 4547), E-28
- TEMP Environment Variable, E-19
- Transport Security, E-21
- Unable to Flush the Cache, E-27
- Unable to Leave Person Object Class Page, E-15
- Unable to log in on IIS, E-18
- Unexpected Group Deletion Problem, E-12

- Unix Installation in GUI Mode, E-14
- User Directory Issues, E-22
- View Rights for Master Administrator, E-27
- Web Server Issues, E-22
- WebGate Cannot Connect to Access Server, E-26
- WebGate Issues, E-25
- WebGate Naming, E-25
- WebPass Identifier Not Available After Setup, E-9
- Windows 2000 Users Cannot Log in, E-18
- troubleshooting, E-1
- apache-based Web servers, 17-34
- Tuning
 - Oracle Internet Directory, 4-13

U

- UMAdminsGroup, 6-4
- Unable to Authenticate Resource on Internet Explorer, E-2
- Unified Searchbase
 - Oracle Virtual Directory, 10-5, 10-6
- Uninstaller, 20-4
- Uninstalling
 - Cloned Components, 15-32
 - see Removal, 20-1
- Unix WebGates, 2-8
- Updating
 - schema and attributes, 1-5
 - WebGate Web Server Configuration, 9-7
- Upgrading
 - from a earlier release, 1-8
- user data, 2-21, 2-22, 2-25
 - Active Directory, 2-23
 - directories, 6-2
- user ID, 19-2
- users
 - authentication of, xviii, 4
 - authorization of, xviii, 4
- UTF-8 encoding, 16-4

V

- Verifying
 - Policy Manager Permissions on IIS, 7-9
- virtual data
 - sources, 10-11
- Virtual Directory
 - schema, 17
- virtual directory, 10-2
 - schema, 10-4

W

- Web Server
 - Configuration Changes Removal, 20-3
- Web server
 - Apache v2, 17-1
 - configuration changes removal, 20-4

- IHS, 17-1, 17-4
- installation packages, 2-15
- OHS, 17-1
- Web Server Support, 16-5
- Web serverr
 - guidelines, 2-16
- WebGate, 9-5, 14-1
 - associating with Access Server, 9-4
 - clock, 2-3
 - Configuration Details, 9-6
 - Console method, 9-5
 - create instance, 9-2
 - Enabling SSL for IIS, 9-9
 - guidelines, 2-8
 - HTTP requests, 9-1
 - ID, 9-6
 - IIS, 9-7, 9-14
 - install, 9-1
 - installing, 9-5
 - Installing Postgate ISAPI Filter, 9-11
 - password, 9-6
 - Prerequisites Checklist, 9-2
 - Setting Up IIS Web Server Isolation Mode
 - IIS
 - WebGate, 9-11
 - Sun Web servers, 9-7
- webgate.dll, 9-10, 9-12
- WebGates
 - older, 2-8
- WebPass
 - Console method, 5-3
 - GUI method, 5-3
 - Install Certificate, 5-5
 - installing, 5-2
 - Java applet, 19-1
 - Language Pack, 5-3
 - prerequisites checklist, 5-2
 - Request Certificate, 5-5
 - system clock, 2-2
- Windows security principal
 - ADAM, B-6
- Windows service name, 11-3
 - SNMP, 11-3
- worker MPM, 17-4

X

- xlC.rte 6.0 runtime library, 17-3