

**Oracle® Authentication Services for Operating
Systems**

Administrator's Guide

10g (10.1.4.0.1-OAS4OS)

E12023-01

March 2008

Oracle Authentication Services for Operating Systems Administrator's Guide, 10g (10.1.4.0.1-OAS4OS)

E12023-01

Copyright © 2008, Oracle. All rights reserved.

Primary Author: Ellen Desmond

Contributing Author: Buddhika Kottahachchi

Contributors: Vasuki Ashok, Quan Dinh, Rui Konno, Karen Lee, David Lin, Daniel Shih, Olaf Stullich, Yoga Thyagarajan, Dai Vu, Forest Yin

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.



RSA and RC4 are trademarks of RSA Data Security. Portions of Oracle Internet Directory have been licensed by Oracle Corporation from RSA Data Security.

Contents

Preface	vii
Audience	vii
Documentation Accessibility	vii
Related Documents	viii
Conventions	viii
1 Product Overview	
Introduction to Oracle Internet Directory	1-1
Features of Oracle Authentication Services for Operating Systems	1-1
Components of Oracle Authentication Services for Operating Systems	1-2
How User Authentication Works With Oracle Internet Directory	1-2
Installation and Configuration Overview	1-3
Management Overview	1-4
Additional Documentation	1-4
2 Before You Install	
Verify Your Client and Server Operating Systems	2-1
Install Oracle Internet Directory and Oracle Directory Integration Platform	2-1
Upgrade Oracle Internet Directory to 10g (10.1.4.2.0)	2-2
Apply the Oracle Internet Directory StartTLS and MD5 Crypt Library Patch	2-2
Determine Which Product Features You Will Use	2-2
Download NIS Migration Scripts	2-3
Download and Apply DIPASSISTANT Patch	2-3
Download SUDO Package	2-4
Create and Index New Custom Attributes (Optional)	2-4
3 Installing and Configuring Oracle Authentication Services for Operating Systems	
Introduction	3-1
SSL Support	3-1
Self Signed Certificates	3-2
Certificate Authority Signed Certificates	3-2
Password Policy Enforcement	3-2
Active Directory Integration	3-3
Directory Plug-ins	3-3

Tools Used During Configuration	3-3
Configuring Oracle Authentication Services for Operating Systems on the Server	3-4
Configuring Oracle Authentication Services for Operating Systems on the Client.....	3-5
Replacing Self-Signed Certificates with CA-Signed Certificates	3-7
Configuring Oracle Internet Directory for Centralized Password Policies.....	3-8
Disabling Value Policies Local to the Operating System	3-8
Disabling State Policies Local to the Operating System	3-9
Switching Between SSL Authentication and Non-SSL Configurations	3-9
Rerunning the Configuration Scripts.....	3-9
Restoring the Client and Server to Their Pre-Configuration State	3-10
Restoring the Client	3-10
Restoring the Server.....	3-11

4 Migrating Entries to Oracle Internet Directory

Migrating Entries to Oracle Internet Directory	4-1
Migrating from NIS to Oracle Internet Directory.....	4-2
Migrating from Operating System Files to Oracle Internet Directory	4-2
Migrating from Another LDAP Directory to Oracle Internet Directory	4-3
Schema Migration.....	4-3
Data Migration	4-4
Setting Access Control on User Entry Attributes.....	4-6
Using Custom Attributes in Oracle Internet Directory	4-7
Migrating SUDO	4-7
Migrating SUDO Entries to Oracle Internet Directory on the Server.....	4-7
Configuring a Client to Use LDAP for SUDO Information	4-8
Reconfiguring a Client to Use /etc/sudoers.....	4-10

5 Configuring Active Directory Integration

Setting up a Plug-in to Augment Active Directory Entries for Linux Authentication	5-1
Configuring Oracle Directory Integration Platform	5-2
Configuring SSL Between Oracle Directory Integration Platform and Active Directory	5-3
Configuring SSL Between Oracle Directory Integration Platform and Oracle Internet Directory .	5-3
Setting Up the External Authentication Plug-in	5-4

6 Managing Oracle Authentication Services for Operating Systems

Creating Home Directories.....	6-1
Managing Users and Groups With libuser Tools.....	6-1
Managing Oracle Internet Directory With Oracle Directory Manager and Command-Line Utilities	6-2
Testing Whether a User Has Been Added	6-2
Changing a User's Password by Using ldapmodify	6-3
Adding a User by Using ldapadd.....	6-3
Adding a Group by Using ldapadd	6-4
Managing Password Policies	6-4

A Troubleshooting

Data Migration Errors	A-1
Sudo Conversion Script Errors.....	A-1
Management Tool Problems	A-1
Error in system-config-users	A-1
The libuser Tools Fail with Python Errors.....	A-2
Linux Management Tools Cause Inconsistencies.....	A-2
ldapsearch Error	A-2
Testing and Log File Messages	A-3
Enabling Log Messages for All Operations.....	A-3
Testing StartTLS	A-3
Password Syntax Errors	A-4
User Login Errors	A-4
Users Cannot Log In	A-4
User's Home Directory Does Not Exist	A-5
User's Shell Does Not Exist.....	A-5
Password Policy Not Consistently Enforced	A-5

B Properties File for LDAP Migration

C Sample Mapfiles

Template Mapfile	C-1
Sample Mapfile 1	C-1
Sample Mapfile 2	C-2
Sample Mapfile 3	C-2
Sun Java System Directory Server Mapfile 1.....	C-2
Sun Java System Directory Server Mapfile 2.....	C-3
eDirectory Mapfile.....	C-3

D Synchronization Profile for Active Directory Integration

Index

List of Figures

1-1	Features of Oracle Authentication Services for Operating Systems.....	1-2
1-2	Authentication Using Oracle Internet Directory	1-3

Preface

This is the Administrator's Guide for Oracle Authentication Services for Operating Systems, Release 10g (10.1.4.0.1-OAS4OS). It explains how to install, configure, and manage Oracle Authentication Services for Operating Systems on server and client systems.

Audience

This document is intended for system administrators who install, configure, and maintain Oracle Internet Directory and operating system authentication services.

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, 7 days a week. For TTY support, call 800.446.2398. Outside the United States, call +1.407.458.2479.

Related Documents

For more information about Oracle Authentication Services for Operating Systems 10g (10.1.4.0.1-OAS4OS), see:

- The README document accompanying this release
- Metalink Note 558907.1: *OAS4OS 10.1.4.0.1 Installation*, on Oracle MetaLink, <http://metalink.oracle.com>

Also see the following documents in the Oracle Application Server 10g (10.1.4.0.1) documentation set, at

<http://www.oracle.com/technology/documentation/oim1014.html>:

- *Oracle Internet Directory Administrator's Guide*
- *Oracle Application Server Installation Guide*
- *Oracle Identity Management Integration Guide*
- *Oracle Identity Management User Reference*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<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.

Product Overview

Oracle Authentication Services for Operating Systems enables you to centralize storage, authentication, and management of user identities using Oracle Internet Directory.

This chapter contains the following topics:

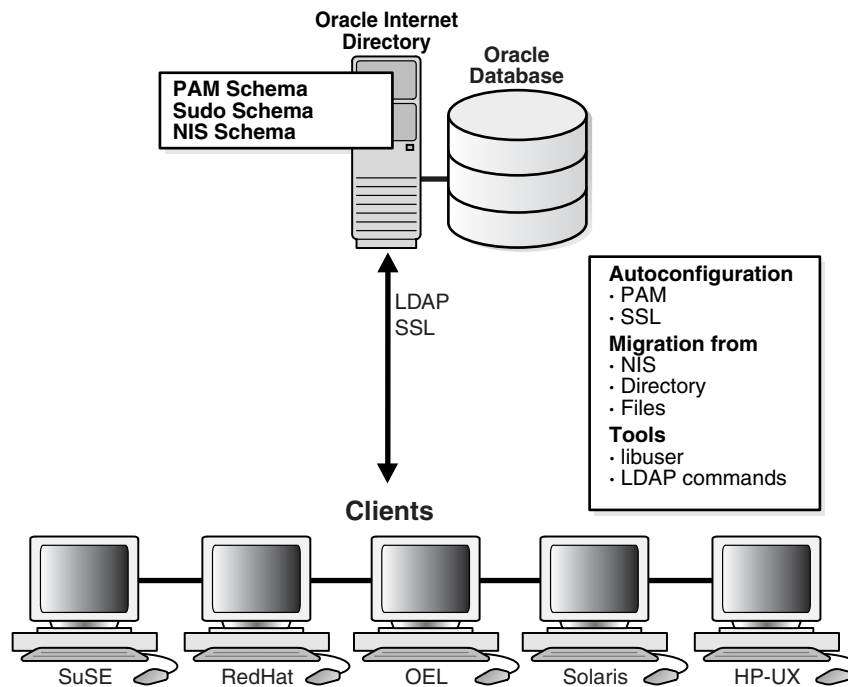
- [Introduction to Oracle Internet Directory](#)
- [Features of Oracle Authentication Services for Operating Systems](#)
- [Components of Oracle Authentication Services for Operating Systems](#)
- [Installation and Configuration Overview](#)
- [Management Overview](#)
- [Additional Documentation](#)

Introduction to Oracle Internet Directory

Oracle Internet Directory is a standards-based directory server that leverages the security, scalability, and reliability of Oracle Database 10g to store users, groups, and other types of entries. Oracle Internet Directory supports password policy enforcement. Oracle Internet Directory can be synchronized with third-party directory servers, such as Active Directory.

Features of Oracle Authentication Services for Operating Systems

Oracle Authentication Services for Operating Systems enables you to use Oracle Internet Directory for authentication on Linux and UNIX-based operating systems. Configuration scripts automate the configuration of Pluggable Authentication Modules (PAM) and Secure Sockets Layer (SSL). You can then migrate existing entries from NIS, files, or another LDAP-compliant directory, and optionally configure features such as password policy enforcement, `sudo`, and automount. Oracle Internet Directory tools are available for entry management, and `libuser` tools can be used for many operations. These features are summarized in [Figure 1-1](#).

Figure 1–1 Features of Oracle Authentication Services for Operating Systems

Components of Oracle Authentication Services for Operating Systems

Oracle Authentication Services for Operating Systems requires the Oracle Internet Directory patch tracked by Bug 6843350, which adds the following capabilities to 10g (10.1.4.2.0):

- Start-TLS—enables you configure the same port for both SSL and non-SSL connections on most operating systems
- MD5 Crypt Library—provides native MD5 crypt password hashing

The Oracle Authentication Services for Operating Systems download contains the following components:

- SSL and non-SSL server configuration scripts
- SSL and non-SSL client configuration scripts
- Support for migration from NIS as well as from flat file-based authentication
- Support for migration from a third party LDAP directory to Oracle Internet Directory. A separate patch is required.
- Support for migration of sudo policy from a `sudoers` file to Oracle Internet Directory
- Support for migration of automounts to Oracle Internet Directory

How User Authentication Works With Oracle Internet Directory

When a user provides credentials (a username and password) to `login`, `xdm`, `ssh`, `su`, or some other client login program, the following events occur.

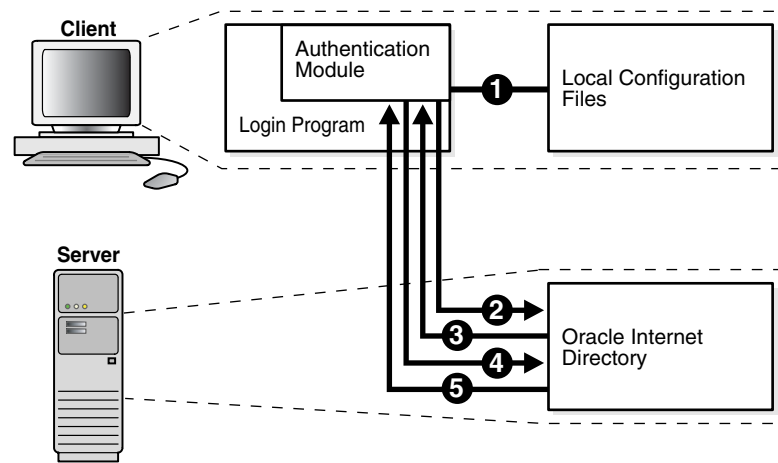
1. An authentication module in the login program examines local configuration files to determine how to authenticate the user. The files contain information such as

the method to use (LDAP), the location of the server, and, if SSL is configured, the certificate to use.

2. The authentication module attempts to perform an LDAP `bind` operation to the Oracle Internet Directory server with the user's credentials. If SSL is configured, the module first establishes the SSL communications channel using the certificate.
3. If Oracle Internet Directory determines that the credentials are correct and the account is active, the `bind` succeeds. Otherwise, the `bind` fails, and the user's login attempt fails.
4. If the `bind` succeeds, the module queries Oracle Internet Directory again for the user's group membership information.
5. Oracle Internet Directory returns the group membership information.

These events are shown in [Figure 1-2](#).

Figure 1-2 Authentication Using Oracle Internet Directory



Installation and Configuration Overview

To install and configure Oracle Authentication Services for Operating Systems, you perform the following steps:

1. Install Oracle Internet Directory 10g (10.1.4.2.0).
2. Install the patch tracked by Bug 6843350 on the Oracle Internet Directory 10.1.4.2.0 server.
3. Download the release.
4. Execute the configuration scripts to configure the server and clients for user authentication.
5. Configure password policies.
6. Migrate entries from NIS, local files, or another LDAP-compliant directory to Oracle Internet Directory.
7. Configure `sudo` authentication and migrate `sudo` entries to Oracle Internet Directory.
8. Optionally, you can configure integration with Active Directory so that you can use credentials stored in Active Directory for authentication on a Linux or UNIX-based operating system.

Management Overview

After you install Oracle Authentication Services for Operating Systems and migrate your data to Oracle Internet Directory, you must use specific tools to manage users, passwords, and other data. Specifically, you must use:

- Oracle Directory Manager
- The LDAP tools and bulk tools in `$ORACLE_HOME/bin`
- The `passwd` command

You can also use the `libuser` tools on Linux distributions that support it, with some limitations.

Additional Documentation

For more information about Oracle Authentication Services for Operating Systems 10g (10.1.4.0.1-OAS4OS), see:

- The README document accompanying this release
- Metalink Note 558907.1: *OAS4OS 10.1.4.0.1 Installation*, on Oracle MetaLink, <http://metalink.oracle.com>

Before You Install

Before installing Oracle Authentication Services for Operating Systems, ensure that you are using a supported operating system and the supported version of Oracle Internet Directory. Then, before you start the install, determine which of the optional product features you will use and locate the scripts you will use for migration.

This chapter contains the following topics:

- [Verify Your Client and Server Operating Systems](#)
- [Install Oracle Internet Directory and Oracle Directory Integration Platform](#)
- [Upgrade Oracle Internet Directory to 10g \(10.1.4.2.0\)](#)
- [Apply the Oracle Internet Directory StartTLS and MD5 Crypt Library Patch](#)
- [Determine Which Product Features You Will Use](#)
- [Download NIS Migration Scripts](#)
- [Download and Apply DIPASSISTANT Patch](#)
- [Download SUDO Package](#)
- [Create and Index New Custom Attributes \(Optional\)](#)

Verify Your Client and Server Operating Systems

Oracle Authentication Services for Operating Systems has both server and client components. The server is the computer that runs Oracle Internet Directory. The client is a computer that uses the services of Oracle Internet Directory for authentication.

For up-to-date information about supported server and client operating systems, please consult the following documents:

- The README document accompanying this release
- Metalink Note 558907.1: *OAS4OS 10.1.4.0.1 Installation*, on Oracle MetaLink, <http://metalink.oracle.com>

Install Oracle Internet Directory and Oracle Directory Integration Platform

Before you can install the patches described in the next two sections, you must install Oracle Internet Directory 10g (10.1.4.0.1). If you plan to migrate entries from an existing LDAP-compliant directory, or to synchronize Oracle Internet Directory with another directory, such as Active Directory, you must install Oracle Directory Integration Platform along with Oracle Internet Directory.

See Also:

- The 10g (10.1.4.0.1) *Oracle Application Server Installation Guide* for your platform
- The section entitled "Using the OPatch Utility" in Appendix I of the *Oracle Application Server Administrator's Guide*.

for information about installing and patching Oracle Internet Directory. Both documents are located at:

<http://www.oracle.com/technology/documentation/oim1014.html>.

Upgrade Oracle Internet Directory to 10g (10.1.4.2.0)

Oracle Authentication Services for Operating Systems requires Oracle Internet Directory 10g (10.1.4.2.0) on the server. If you have not already done so, use `$ORACLE_HOME/OPatch` to apply the patch for 10.1.4.2.0. Oracle Internet Directory 10g (10.1.4.2.0) contains the necessary schemas for authentication on a Linux or UNIX-based operating system. The tracking bug for this patch is 5983637. See <http://metalink.oracle.com>.

Apply the Oracle Internet Directory StartTLS and MD5 Crypt Library Patch

Oracle Authentication Services for Operating Systems requires a patch that adds StartTLS capability and the MD5 Crypt Library to Oracle Internet Directory. Use `$ORACLE_HOME/OPatch` to apply this patch on the server before you install Oracle Authentication Services for Operating Systems. The tracking bug for this patch is 6843350. See <http://metalink.oracle.com>.

Determine Which Product Features You Will Use

Before you begin the installation, consider which features of the product you are likely to use. For basic functionality, you need to apply the Oracle Internet Directory patch, run the server script on the system where you are running the Oracle Internet Directory server, then run the client script on each client. These scripts configure the server and clients for LDAP authentication. In addition to configuring basic LDAP authentication, you can choose from the following options:

- Secure Socket Layer (SSL)—Unless your server and clients are isolated from the internet, you should enable SSL. To do so, use the SSL versions of the server and client configuration scripts. The `libuser` tool `system-config-users` requires SSL when you use it with Oracle Authentication Services for Operating Systems on Red Hat or Oracle Enterprise Linux.
- Certificate and wallet to use with SSL—The SSL server configuration script generates a self-signed certificate, which is not designed for production mode. You can substitute a certificate signed by a certificate signing authority. You can also choose to use a customized wallet instead of the default wallet. See *Oracle Application Server Administrator's Guide* for information on Oracle wallets.
- Current authentication source to migrate from—If you are using files, NIS, or another LDAP server, you can migrate to Oracle Internet Directory.
- Whether to configure the `libuser` tools to use LDAP—The GUI tool `system-config-users` and the command-line utilities (`luseradd`, `luserdelete`, etc.) exist, by default, on Red Hat Enterprise Linux and Oracle Enterprise Linux. You can configure the `libuser` tools to work with LDAP, so

that adding a user with `useradd`, for example, adds the user entry to Oracle Internet Directory. If you do not use the `libuser` tools, you must use Oracle Directory Manager, Oracle Internet Directory bulk tools, or Oracle Internet Directory LDAP tools to configure entries directly in Oracle Internet Directory. If your client is Red Hat Enterprise Linux or Oracle Enterprise Linux, the client script will prompt you as to whether you want to configure `libuser`.

Note:

- To use `libuser` tools, you must configure your client and server for SSL.
 - If you plan to use Oracle Internet Directory to enforce password policies, you cannot use tools in the `libuser` package to add passwords or entries containing passwords.
 - You cannot use the non-`libuser` commands `useradd`, `userdel`, `groupadd`, or `groupdel` for user or group administrative tasks.
-
-
- Data to migrate—Open Source scripts such as those described in the next section support migration of users and groups and other configuration data from NIS or from files. Oracle Authentication Services for Operating Systems includes tools for migrating from a third-party LDAP directory server.
 - Whether to migrate `sudo`—You can use Oracle Internet Directory instead of a `sudoers` configuration file to authenticate `sudo` commands.
 - How to enforce password policies—You can continue to use the operating system for password enforcement. Alternatively, you can use Oracle Internet Directory for centralized password policies.
 - Whether to integrate with Active Directory—You can use credentials stored in Active Directory for user authentication on Linux or UNIX-based operating systems.

Download NIS Migration Scripts

If you have `user`, `group`, and other entries maintained in the local file system or in NIS/NIS+, you can move to LDAP as your storage mechanism for these entries. A number of free tools are available for this purpose. These tools enable you to extract the existing information and produce output files in the LDAP Data Interchange Format (LDIF). Once you have your information in LDIF files, you can use the `ldapadd` tool to load the information into Oracle Internet Directory.

We have validated the process of migrating information using the LDAP migration tools available at:

<http://www.padl.com/>

If you have the `openldap` packages installed on your host, you will find the same migration tools at: `/usr/share/openldap/migration`.

Download and Apply DIPASSISTANT Patch

If you are migrating entries from a third-party, LDAP-compliant directory to Oracle Internet Directory, use `$ORACLE_HOME/OPatch` to apply the `dipassistant` patch, which simplifies the syntax of the `properties` file you will use with the Oracle Directory Integration Platform migration tool `dipassistant`.

Note: If you encounter an error when using OPatch to apply this patch, set the environment variable `OPATCH_PLATFORM_ID` to 0 and try again.

The tracking bug for this patch is 6849766. See <http://metalink.oracle.com>.

Download SUDO Package

If you want to migrate the contents of the `sudoers` file to LDAP, you must run a migration script and build `sudo` with LDAP enabled. You can obtain the `sudo` package from:

<http://www.gratisoft.us/sudo>

Create and Index New Custom Attributes (Optional)

You cannot successfully search for an attribute in Oracle Internet Directory unless the attribute is indexed. If you plan to add custom attributes, you can index them at the time you create them by using Oracle Directory Manager. You can also use `ldapmodify` to create an indexed attribute. You would use an LDIF file such as this:

```
dn: cn=catalogs
changetype: modify
add: orclindexedattribute
orclindexedattribute: attribute_name
```

Alternatively, you can index attributes after they have been created in Oracle Internet Directory by using `catalog`, as explained in "[Using Custom Attributes in Oracle Internet Directory](#)" on page 4-7.

Note: If you attempt to perform a search with a non-indexed attribute specified as a required attribute, the server will return the error:

Function not implemented. DSA unwilling to perform.

See <http://metalink.oracle.com>.

Installing and Configuring Oracle Authentication Services for Operating Systems

This chapter contains the following topics:

- [Introduction](#)
- [Configuring Oracle Authentication Services for Operating Systems on the Server](#)
- [Configuring Oracle Authentication Services for Operating Systems on the Client](#)
- [Replacing Self-Signed Certificates with CA-Signed Certificates](#)
- [Configuring Oracle Internet Directory for Centralized Password Policies](#)
- [Switching Between SSL Authentication and Non-SSL Configurations](#)
- [Rerunning the Configuration Scripts](#)
- [Restoring the Client and Server to Their Pre-Configuration State](#)

Before you begin the procedures described in this chapter, you must perform the prerequisite procedures described in [Chapter 2](#).

Introduction

This introduction contains the following sections:

- [SSL Support](#)
- [Password Policy Enforcement](#)
- [Active Directory Integration](#)
- [Directory Plug-ins](#)
- [Tools Used During Configuration](#)

SSL Support

Oracle Internet Directory can be configured for SSL-no authentication, SSL-server authentication and SSL-mutual authentication modes. In all three modes, the data is encrypted during transmission. Oracle Internet Directory comes pre-configured with the SSL-no authentication mode. However, some clients such as the PAM_LDAP clients used for Linux user authentication do not support this mode and only support SSL-server authentication mode.

For administrative ease, the initial server configuration process enables you to configure Oracle Internet Directory for SSL-server authentication mode, using self-signed certificates.

Note: Self-signed certificates are not intended for production use. See ["Replacing Self-Signed Certificates with CA-Signed Certificates"](#) on page 3-7 for information on using certificates issued by a trusted certificate authority.

Self Signed Certificates

The SSL server configuration script generates two Oracle wallets:

1. Test Certificate Authority (CA) Wallet—used to sign the Oracle Internet Directory SSL Server Certificate. This consists of the following files in `$_ORACLE_HOME/wallet/root`:
 - `cakey.txt`—a 1024 bit RSA private key
 - `cacert.txt`—base64 encoded certificate
2. Oracle Internet Directory SSL Server Certificate. This consists of the following files in `$_ORACLE_HOME/wallet/server`:
 - `creq.txt`—Oracle Internet Directory SSL Server Certificate Request
 - `cert.txt`—Oracle Internet Directory SSL Server Certificate signed by Test CA Wallet
 - `cwallet.sso`—Oracle Internet Directory SSL Server Wallet for auto-login
 - `ewallet.p12`—PKCS12 encoded Oracle Internet Directory SSL wallet

Note: The PKCS12-encoded wallets contain the private keys for the relevant entities and are protected by a wallet password that you set when running the SSL server configuration script.

For a client to trust the Oracle Internet Directory SSL Server Certificate (2) it must trust the Test CA Wallet (1). Since most Linux clients work with the PEM format, a copy of the Test CA Wallet (1) in PEM format is available at: `$_ORACLE_HOME/wallet/pem.cert`.

Certificate Authority Signed Certificates

If you have access to a Public Key Infrastructure (PKI) deployment, you can use certificates issued by a trusted CA in that PKI to secure your Oracle Internet Directory deployment. The procedure for swapping certificates is described in ["Replacing Self-Signed Certificates with CA-Signed Certificates"](#) on page 3-7.

Password Policy Enforcement

Oracle Internet Directory ships with a rich set of password policies that can be leveraged for centralized password policy management. See the chapter on Password Policies in the *Oracle Internet Directory Administrator's Guide 10g* (10.1.4.0.1) to understand the concepts governing these features.

Oracle Internet Directory supports two types of password policies: value policies and state policies. Value policies govern password construction requirements, such as

minimum length. State policies govern things like password expiration and logout. On Linux and UNIX-based operating systems, state policies are traditionally handled in the shadow password file using the password aging feature. These policies can be applied in a fine-grained manner down to the level of a single user entry.

You can use Oracle Internet Directory to enforce both value and state policies. Value policy violations result in visible error message on the Linux client, but state policy violations simply result in login failures. This is because the `pam_ldap` client does not display the messages that Oracle Internet Directory sends as additional information with the LDAP bind failure.

To use Oracle Internet Directory for centralized password policies, you must disable value and state policies local to the operating system. The procedure for doing this is described in ["Configuring Oracle Internet Directory for Centralized Password Policies"](#) on page 3-8.

If you do not want to use Oracle Internet Directory for password policy enforcement, you must disable password policies in Oracle Internet Directory by setting `orclpwdpolicyenable` to 0. To avoid messages about password syntax, you must also disable the password syntax check by setting `pwdCheckSyntax` to 0.

See Also: The Password Policies chapter in the *Oracle Internet Directory Administrator's Guide*.

Active Directory Integration

If you have users in Active Directory, and you want to use the credentials stored in Active Directory for Linux authentication, you can configure Oracle Directory Integration Platform to integrate with Active Directory. The configuration process is described in [Chapter 5, "Configuring Active Directory Integration."](#)

Directory Plug-ins

A directory server plug-in is a customized program that extends the capabilities of the Oracle Internet Directory server. The procedures for augmenting Active Directory entries and for setting up external authentication with Active Directory both include setting up plug-ins. These procedures are described in [Chapter 5, "Configuring Active Directory Integration."](#)

See Also: *Oracle Internet Directory Administrator's Guide* for more information about directory server plug-ins.

Tools Used During Configuration

Some of the tasks described in this chapter require you to use Oracle Internet Directory or Oracle Directory Integration Platform tools. These tools include:

- The Oracle Internet Directory LDAP command-line tools—These are located in the `$ORACLE_HOME/bin` directory. These tools are `ldapsearch`, `ldapbind`, `ldapmodify`, `ldapdelete`, `ldapcompare`, `ldapmoddn`, `ldapaddmt` and `ldapmodifymt`. For interaction with the Oracle Internet Directory server, you must use the LDAP tools in `$ORACLE_HOME/bin` and not those shipped in the operating system base image.
- The Oracle Internet Directory bulk tools—These are also located in the `$ORACLE_HOME/bin` directory. These tools are `bulkload`, `bulkmodify`, `catalog`, `bulkdelete` and `ldifwrite`. The bulk tools allow you to perform bulk operations, such as adding or deleting a large number of entries.

One important bulk tool is the `catalog` tool. This tool enables you to add indexes to attributes in Oracle Internet Directory. Attributes must be indexed in order to be searchable. This example adds an index to the attribute `uid`:

```
catalog connect="connect_str" add="TRUE" attribute="uid"
```

- The `oidctl` command—You use this to stop and start the Oracle Internet Directory server.
- The `dipassistant` command—You use this when configuring SSL for communication between Oracle Directory Integration Platform and Active Directory and when migrating data from another LDAP-compliant directory to Oracle Internet Directory. If you are using `dipassistant` for data migration, you must apply the `dipassistant` patch, which simplifies the syntax of the properties file you will use with the migration tool `dipassistant`. The tracking bug for the patch is 6849766.

See Also:

- *Oracle Internet Directory Administrator's Guide* and the *Oracle Identity Management User Reference* for information about the Oracle Internet Directory LDAP tools, bulk tools, and `oidctl`.
- The chapter entitled "Oracle Directory Integration Platform Tools" in the *Oracle Identity Management User Reference* and the chapter entitled "Configuration of Directory Synchronization Profiles" in the *Oracle Identity Management Integration Guide* for more information on `dipassistant`.

Configuring Oracle Authentication Services for Operating Systems on the Server

Use the server configuration script to configure the server for UNIX or Linux authentication, as follows:

1. As a precaution, perform a backup of the Oracle Internet Directory schemas and database.
2. If you have old versions of the server configuration scripts in `$ORACLE_HOME/ldap/bin`, you might want to save them elsewhere before copying the new script to `$ORACLE_HOME/ldap/bin`.
3. If you want to configure SSL, copy `sslConfig_OIDclient.sh` and `sslConfig_OIDserver.sh` to `$ORACLE_HOME/ldap/bin`. Otherwise, copy `config_OIDclient.sh` and `config_OIDserver.sh` to `$ORACLE_HOME/ldap/bin`.

Note:

- You can switch between SSL and non-SSL configurations. See ["Switching Between SSL Authentication and Non-SSL Configurations"](#) on page 3-9.
 - You can disable either the SSL port or the non-SSL port if you are not using it. You do this by changing the value of the configuration attribute `orclSSEnable`. See the entry for `orclSSEnable` in the Attribute Reference chapter of the *Oracle Identity Management User Reference*.
-
-

4. Copy `oasconfig.ldif` to `$ORACLE_HOME/ldap/admin`.
5. Execute the server script on the server as the same user who installed Oracle Internet Directory. Type:

```
./ sslConfig_OIDserver.sh
```

or

```
./ config_OIDserver.sh
```

6. You will be prompted for `ORACLE_HOME`, realm (naming context), non-SSL port, password for `cn=orcladmin`, and wallet password. Supply the appropriate values in response to the prompts. (If you have set `ORACLE_HOME` as an environment variable, you will not be prompted for it.)

The server script edits `oasconfig.ldif` so that it contains the necessary information about the server, then loads the information into Oracle Internet Directory.

The SSL version of the script configures Oracle Internet Directory for SSL server side authentication mode with self-signed certificates. This mode can be used with `pam_ldap` to enable user authentication.

The SSL version of the script configures port 389 for StartTLS, which allows SSL and non-SSL connections to use the same port. The script also configures port 636, the SSL port, for connections from clients that do not support StartTLS.

The server script edits the client script, `sslConfig_OIDclient.sh` or `config_OIDclient.sh`, customizing it for your environment.

The script updates several Oracle Internet Directory server parameters with the information it has gathered. The SSL version of the script restarts the Oracle Internet Directory server. The non-SSL version does not.

Configuring Oracle Authentication Services for Operating Systems on the Client

You configure each client for UNIX or Linux authentication by running a client configuration script. Follow these steps:

Solaris 9 Only

1. On Solaris 9 only, download the Sun Java System Directory Server Resource Kit SDRK52 and install it as `root`. This kit is currently available at: <http://www.sun.com/download/products.xml?id=3f74a0db>
2. After installing the Sun Java System Directory Server Resource Kit, before you run the client configuration script, modify the environment variables `PATH` and `LD_LIBRARY_PATH` so that `PATH` includes `installroot/lib/nss/bin` and `LD_LIBRARY_PATH` includes `installroot/lib`, where `installroot` is the directory where you installed the Sun Java System Directory Server Resource Kit. For example, if you installed the software in `/usr`, add `/usr/lib/nss/bin` to `PATH` and add `/usr/lib` to `LD_LIBRARY_PATH`.
3. Proceed as described for all client platforms.

AIX Without SSL Only

1. Install the AIX LDAP client package. You can find it in the `ldap.client` file sets located on the AIX 5L product media. Execute the following command to install the package:

```
installp -acgXd LPPSOURCE ldap.client
```

where *LPPSOURCE* is the source device for the product images.

2. Proceed as described for all client platforms.

AIX With SSL Only

1. The following packages are required for SSL Configuration on an AIX 5L Version 5.3 client:

- gskta.rte
- ldap.clt_max_cryptobitsizerelease.rte

where *bitsize* is 32bit or 64bit and *release* is the release number.

If these packages are not already installed, install them from the AIX 5L Version 5.3 Expansion Package CD (5705-603) or from the equivalent package in Tivoli Directory Server, which is available at the IBM web site.

2. Verify the installed packages by typing:

```
lsllp -l | grep "gskta*" "*ldap*"
```

3. If necessary, create a symbolic link in `/usr/lib` to the new LDAP client library. For example:

```
ln -s /opt/IBM/ldap/release/lib/libidsldap.a /usr/lib/libibmldap.a
```

4. Proceed as described for all client platforms.
5. Verify that LDAP SSL is enabled by using `ldapsearch`, for example:

```
ldapsearch -h myserver.oracle.com -Z -K /etc/security/ldap/key.kdb
-P keystore_password -b "" -s base objectclass=*
```

6. Verify that authentication is working correctly by logging into your client machine using `telnet`, `rlogin`, `ssh`, or a similar program.

All Client Platforms

1. Copy the client configuration script from the server to the client after you have run the server configuration script. The server script edits the client script, customizing it for your environment.

For SSL Server Authentication enabled Linux clients, use the client script `sslConfig_OIDclient.sh`. For non-SSL Linux clients, use `config_OIDclient.sh`. Copy the script from `$ORACLE_HOME/ldap/bin` on the server to each client you want to configure.

2. Execute the client configuration script on the client as the `root` user. Type:

```
./ sslConfig_OIDclient.sh
```

or

```
./ config_OIDclient.sh
```

3. When prompted, confirm that you want to configure the client to authenticate against the LDAP server.
4. If the client is Red Hat Enterprise Linux or Oracle Enterprise Linux, the client script prompts you as to whether you want to configure the `libuser` package to work with LDAP. Respond `y` if you want `libuser` to be configured. If you

configure `libuser` to work with LDAP, adding a user with `luseradd`, for example, adds the user entry to Oracle Internet Directory.

The script configures Pluggable Authentication Modules (PAM) on the client operating system to use Oracle Internet Directory for user authentication. The exact tasks performed depend on the operating system type. The script performs the following basic tasks:

- Makes configuration changes to `nsswitch.conf` so that `ldap` is an option for `passwd`, `group` and `shadow`.
- Configures `/etc/ldap.conf` and `/etc/openldap/ldap.conf` with the correct URI, Base DN
- Optionally, configures the `libuser` package (via `libuser.conf`) for user management on Red Hat Enterprise Linux and Oracle Enterprise Linux.

Note: The script makes backup copies of the files it touches in subdirectories of the `/etc` directory. These subdirectories have names of the form `oracle_backup_time_stamp`. For example, a backup directory created 18:54:46 on Jan. 13 2008 would have the name `/etc/oracle_backup_20080113185446`.

In addition, `sslConfig_OIDclient.sh` performs the following steps:

- Writes out `/etc/oracle-certs/oid-test-ca.pem`, the pem format encoded certificate for the Test CA created during configuration on the Oracle Internet Directory Server. This is equivalent to `pem.cert` in "Self Signed Certificates" on page 3-2.
- Adds `oid-test-ca.pem` as a trusted CA in `/etc/ldap.conf` and `/etc/openldap/ldap.conf`
- Configures `/etc/ldap.conf` to use cleartext passwords and enable SSL

On most client operating systems, the script configures the client to use the StartTLS port on the server for SSL communication. The script does not configure StartTLS if the operating system on the client is HP-UX or Solaris. These clients use the standard SSL port, 636, on the server for SSL communication.

After you have successfully executed the client configuration script, your Linux or UNIX-based client can use Oracle Internet Directory to authenticate users.

Replacing Self-Signed Certificates with CA-Signed Certificates

If you select SSL-server authentication mode during the initial Oracle Internet Directory configuration, the server configuration script produces test self-signed certificates. If you have access to a Public Key Infrastructure (PKI) deployment, you can use certificates issued by a trusted CA in that PKI to secure your Oracle Internet Directory deployment. To do so, you must swap out the test self-signed certificates produced by the setup script with those your own trusted CA issues.

To swap out the certificates, perform the following steps:

1. Use the tools you already use with your PKI to create a signed SSL server certificate for your Oracle Internet Directory server. At the end of this process you should have two files:

- A PKCS#12-formatted file containing the Oracle Internet Directory SSL Server Certificate, Associated Private Key, Trusted Signing CA certificate and any other Trusted CA certificates
- The signing CA certificate in PEM format (X509v3 or PKCS#7).

Note: The password used to secure the PKCS#12 file should be the same as the one you selected as the password for your Directory Administrator (cn=orcladmin) during initial Oracle Internet Directory configuration.

2. Shut down Oracle Internet Directory.

3. As root, type:

```
mv /$ORACLE_HOME/wallet/server $ORACLE_HOME/wallet/server-old
mkdir $ORACLE_HOME/wallet/server
```

4. Copy the.p12 file containing the Oracle Internet Directory SSL Server Certificate you generated offline into `$ORACLE_HOME/wallet/server` and rename it `ewallet.p12`

5. Execute `orapki` to create an auto-login wallet for use by Oracle Internet Directory:

```
$ORACLE_HOME/bin/orapki wallet create \
-wallet $ORACLE_HOME/wallet/server -pwd wallet_password \
-auto_login
```

6. Start Oracle Internet Directory.

On all clients you configure, you must replace the contents of `/etc/oracle-certs/oid-test-ca.pem` with the PEM format certificate of your signing CA.

Configuring Oracle Internet Directory for Centralized Password Policies

To use Oracle Internet Directory for centralized password policies, you must disable value and state policies local to the operating system.

After you do that, users can invoke the `passwd` tool as usual to change their password. Violations of Oracle Internet Directory password value policies produce error messages in the log files beginning with `Password Policy Error`.

Disabling Value Policies Local to the Operating System

Most Linux distributions are configured by default to use the `cracklib` library to perform end-user supplied password quality validations. When using a centralized password policy enforced in Oracle Internet Directory, you might want to disable the local validations in order to avoid conflicts between the two policies.

On Oracle Enterprise Linux and Red Hat Linux, you can do this as follows:

1. Locate the following line in `/etc/pam.d/system-auth` and comment it out:

```
password requisite /lib/security/$ISA/pam_cracklib.so retry=3
```

2. Locate all subsequent lines beginning with `password` and remove `use_authtok` from those lines.

Disabling State Policies Local to the Operating System

As mentioned previously, state policies on Linux are enforced through the password aging feature enabled by the shadow password information. The operating system parses the shadow information on each account and enforces state policies locally.

In Red Hat Enterprise Linux or Oracle Enterprise Linux, you can disable password ageing for accounts created under Oracle Internet Directory by modifying `/etc/libuser.conf` to use `-1` as the default value for `LU_SHADOWINACTIVE`, `LU_SHADOWEXPIRE`, `LU_SHADOWWARNING` in the `[userdefaults]` section of the file.

For accounts that already exist in Oracle Internet Directory, or that are to be migrated to Oracle Internet Directory, you must set `shadowmax=99999` and `shadowexpire=-1` to disable password expiration.

Switching Between SSL Authentication and Non-SSL Configurations

If you have configured non-ssl authentication, you can switch to SSL authentication as follows:

1. Copy `sslConfigure_OIDserver.sh` to `$ORACLE_HOME/ldap/bin`. Copy `oasconfig.ldif` to `$ORACLE_HOME/ldap/admin`.
2. On the server, run the script `sslConfigure_OIDserver.sh`. Optionally, you can disable the non-ssl port by following the instructions in the *Oracle Internet Directory Administrator's Guide*.
3. Copy the `sslConfigure_OIDclient.sh` script generated on the server to the client machine and run this script as root.

If you have configured SSL authentication, you can switch to non-ssl authentication as follows:

1. On the server, run the script `config_OIDserver.sh`. Optionally, you can disable the ssl port by following the instructions in the *Oracle Internet Directory Administrator's Guide*.
2. Copy the `config_OIDclient.sh` generated on the server to the client machine and run this script as root.

Rerunning the Configuration Scripts

There are occasions when you might need to rerun the configuration scripts. For example, you might want to regenerate the wallet or certificate if the old one is compromised or expired.

First, rerun the configuration script on the server.

1. Copy the following scripts from the release to `$ORACLE_HOME/ldap/bin`:
 - `config_OIDclient.sh` or `sslConfig_OIDclient.sh`
 - `config_OIDserver.sh` or `sslConfig_OIDserver.sh`
2. Copy `oasconfig.ldif` from the release to `$ORACLE_HOME/ldap/admin`.
3. Execute `config_OIDserver.sh` or `sslConfig_OIDserver.sh` as the user who installed Oracle Internet Directory.

Then, rerun the script on each client.

1. Copy the latest version of the client scripts from `$ORACLE_HOME/ldap/bin` on the Oracle Internet Directory server machine to each client machine.

2. Execute `config_OIDclient.sh` or `sslConfig_OIDclient.sh` on each client machine as `root`.

Restoring the Client and Server to Their Pre-Configuration State

If necessary, you can restore your client computers to the state they were in before you ran `config_OIDclient.sh` or `sslConfig_OIDclient.sh`. To do so, locate directories under `/etc` with names of the form `oracle_backup_time_stamp`. For example, a backup directory created 18:54:46 on Jan. 13 2008 would have the name `/etc/oracle_backup_20080113185446`. If there is more than one backup directory, in most cases, you need to use the backup files in the earliest backup directory.

Restoring the Client

Perform these steps to restore the client:

1. Copy the following files, as `root`, from the backup directory to the specified destinations:
 - Copy `openldap_ldap.conf` to `/etc/openldap/ldap.conf`.
 - Copy all the files under `backup-directory/pam.d/` to `/etc/pam.d`.
 - On SuSE, copy `pam_unix2.conf` to `/etc/security/` and copy `ldap` to `/etc/sysconfig`.
 - On Solaris, copy all the files under `backup-directory/restore` to `/var/ldap/restore`.
 - Copy all other files in the backup directory to `/etc`.
2. Execute the following commands:
 - On Red Hat or Oracle Enterprise Linux:

```
authconfig --disableldapauth --update
```
 - On SuSE Linux:

```
/etc/init.d/nscd restart  
/etc/init.d/ssh restart
```
 - On Solaris:

```
ldapclient uninit
```
 - On HP-UX:

Edit the file `/etc/opt/ldapux/ldapclientd.conf`. Change the value of the `StartOnBoot` parameter to `enable=no`. Then execute the following command:

```
kill -9 `cat /etc/opt/ldapux/ldapclientd.pid`
```
 - On AIX:

```
stop-seclapclntd
```

Restoring the Server

There is nothing to restore on the server. See the *Oracle Internet Directory Administrator's Guide* if you want to stop the Oracle Internet Directory server or to disable the SSL or non-SSL port.

Migrating Entries to Oracle Internet Directory

This chapter contains the following topics:

- [Migrating Entries](#)
- [Setting Access Control on User Entry Attributes](#)
- [Using Custom Attributes in Oracle Internet Directory](#)
- [Migrating SUDO](#)

Migrating Entries

Before migrating entries from NIS, files, or another LDAP directory, perform the following tasks:

- Tune your Oracle Internet Directory database.

See Also: *Oracle Internet Directory Tuning and Configuration, a Quick Reference Guide* at <http://www.oracle.com/technology>.

- Take a cold backup of the Oracle Internet Directory database in case you need to restore it.
- Ensure that, in the event that Oracle Internet Directory becomes unavailable, the administrator will still be able to log in as `root`. Specifically:
 - Keep a local `root` account in your `/etc/passwd` and `/etc/shadow` files.
 - Do not modify the `passwd` or `shadow` precedence in `nsswitch.conf`. The configuration script sets them to:

```
passwd: files ldap
shadow: files ldap
```

Note: Exercise security precautions in your handling of files that contain sensitive information.

This section contains the following topics:

- [Migrating from NIS to Oracle Internet Directory](#)
- [Migrating from Operating System Files to Oracle Internet Directory](#)
- [Migrating from Another LDAP Directory to Oracle Internet Directory](#)

Migrating from NIS to Oracle Internet Directory

You can migrate entries from NIS to Oracle Internet Directory. The steps are as follows:

1. Run the LDAP migration scripts, described in "[Download NIS Migration Scripts](#)" on page 2-3, on your NIS master files. This will generate LDIF files containing the entries.
2. For compatibility with a variety of clients, as well as with the `system-config-users` tool, ensure that the entries include all the required attributes shown in the following example. (Substitute the user's password for *password*.)

```
uid: jueno
homedirectory: /home/jueno
loginshell: /bin/tcsh
uidnumber: 506
gidnumber: 506
cn: juri ueno
objectclass: posixAccount
objectclass: shadowAccount
objectclass: account
objectclass: top
userpassword: password
shadowwarning: -1
shadowmax: 99999
shadowlastchange: 13916
shadowexpire: -1
shadowmin: 0
shadowinactive: -1
gecos: jueno
```

The `shadowAccount` objectclass and attributes are typically missing in user entries migrated from an HP-UX server.

3. Use the `ldapadd` client tool shipped with Oracle Internet Directory to load the LDIF entries into Oracle Internet Directory. Use a command line of the form:

```
ldapadd -p port -h host -D binddn -w bindpwd -v -f ldif_file
```

Note:

- If you are using the same naming context created during installation, these scripts will generate parts of the DIT (Directory Information Tree) that already exist. This will cause `ldapadd` failures because you are attempting to add an existing entry. You can avoid these failures by specifying the `-c` option to continue upon encountering such errors.
 - The `binddn` you use must be the directory administrator so that you have the proper privileges when performing these additions.
-
-

Migrating from Operating System Files to Oracle Internet Directory

Migrating from operating system files is basically the same as migrating from NIS, except that you might have different versions of your configuration files on different machines. If you have multiple versions, run the migration scripts on each version and combine the LDIF files. You must resolve conflicts manually, using a text editor. Each

user must have a unique user name and uid, and each group must have a unique group name and gid.

Migrating from Another LDAP Directory to Oracle Internet Directory

You can migrate entries from a third-party, LDAP-compliant directory to Oracle Internet Directory. Before you do so, you must download and apply the dipassistant patch referenced by Bug 6849766, if you have not done so already.

Note: This section describes how to do a one-time migration of data from an LDAP-compliant source directory to Oracle Internet Directory. If you are planning to set up ongoing synchronization between a source directory and Oracle Internet Directory by using Oracle Directory Integration Platform, refer to the *Oracle Identity Management Integration Guide*.

Migration of entries from a third-party source directory to Oracle Internet Directory occurs in two phases: schema migration and data migration.

Schema Migration

The steps for migrating schema are as follows:

1. Analyze the schema difference between the directories by running the `schemasync` tool. The syntax is:

```
$ORACLE_HOME/bin/schemasync -srchost srchost -srcport srcport -srcdn binddn \  
                             -srcpwd bindpwd -dsthost oidhost -dstport oidport \  
                             -dstdn oiddn -dstpwd oidpwd
```

where `srchost` and `srcport` are the connection details of the source directory and `srcdn` and `srcpwd` are the credentials to connect to the source directory.

See Also: The command reference for `schemasync` in the Oracle Directory Integration Platform Tools chapter of the *Oracle Identity Management User Reference*.

The command produces four output files that list differences between the source directory and Oracle Internet Directory schema. They are:

- `$ORACLE_HOME/ldap/odi/log/attributetypes.log`—difference in the schema definition of the common attributes between the source directory and Oracle Internet Directory.
 - `$ORACLE_HOME/ldap/odi/log/objectclasses.log`—difference in the schema definition of the common object classes between the source directory and Oracle Internet Directory
 - `$ORACLE_HOME/ldap/odi/data/attributetypes.ldif`—attributes that are available only in the source directory and not in Oracle Internet Directory.
 - `$ORACLE_HOME/ldap/odi/data/objectclasses.ldif`—object classes that are available only in the source directory and not in Oracle Internet Directory.
2. If necessary, extend the schema elements in Oracle Internet Directory.
 - a. Based on the analyses in Step 1 and Step 2, determine what new schema elements you must load onto Oracle Internet Directory. Modify the files

`attributetypes.ldif` and `objectclasses.ldif` (from step 1) to have only the attributes and object classes that you must load. Name the modified files `modified_attributetypes.ldif` and `modified_objectclasses.ldif`.

For example, assume that the objectclass of the user entry in the third-party directory is `inetorgperson`, `organizationalperson`, `person`, `srcuser` and the objectclass of user entry in Oracle Internet Directory is `inetorgperson,organizationalperson,person,orcluser`. In Step 1, if the objectclass definitions of `inetorgperson`, `organizationalperson`, and `person` are different between Oracle Internet Directory and the third-party directory, the difference will be written to the `objectclasses.log` files. After looking at the file, you might decide to make the required changes in the objectclass definitions of Oracle Internet Directory. Since `srcuser` is a third-party directory specific objectclass, the objectclass definition will be in the `objectclasses.ldif` file. Modify the `objectclass.ldif` file to contain the objectclass definition and rename it `modified_objectclasses.ldif`. Modify the `attributetypes.ldif` file to contain the definitions of the attributes required for the objectclasses in `objectclasses.ldif`.

- b. Upload the required schema using the `ldapmodify` command as follows:

```
ldapmodify -h oidhost -p oidport -D 'cn=orcladmin' -w pwd \
           -f modified_attributetypes.ldif
ldapmodify -h oidhost -p oidport -D 'cn=orcladmin' -w pwd \
           -f modified_objectclasses.ldif
```

Data Migration

Migration of data is more complicated because you must include some entries and exclude others. Even in the entries that are included, you might want to include only specific attributes. In general, user and group are migrated. The attributes representing access control definitions, password policy-related attributes, and other operational attributes such as `createtimestamp`, `modifytimestamp`, `creatorsname`, `modifiersname`, `entrydn`, `numsubordinates`, `parentid`, `entryid`, and `nsuniqueid` are excluded. You might want to include `userpassword` as an attribute to be migrated. Do so only if both the directories support the same kind of encryption or hashing techniques.

You can get the exact data to be migrated by filtering the data either while exporting it from the source directory (Step 1) or as a separate step (Step 2).

1. Export the data from the source directory into LDIF file format, using the appropriate LDAP tool on your system, and analyze it. See the documentation for your directory server to determine what command to use. If you filter and export only the required LDAP entries with only the required attributes during the export operation, proceed to Step 3. Otherwise, filter it in Step 2.
2. If you did not filter out the entries and attributes not to be migrated in Step 1, remove them in this step by using `dipassistant`.

The `dipassistant` tool filters the entries based on the configuration and also supports mapping and transformation of attributes. You specify the configuration of filtering, mapping and transformation in the `mapfile`. Sample `mapfiles` are provided in [Appendix C](#).

- a. If you are migrating entries other than user and group from source directories, update the `mapfile` accordingly.

- b. Make a copy of the sample file `$ORACLE_HOME/ldap/odi/samples/migrate.properties` and name it `migrate.properties`.

See Also: The command reference for `dipassistant` in the Oracle Directory Integration Platform Tools chapter of the *Oracle Identity Management Integration Guide* for documentation of the parameters used in the properties file.

- c. In the properties file, you must specify the name of the LDIF file containing the entries to be migrated as source file and a name for the file that is to be generated by `dipassistant` as the destination file name. The containers to be included/excluded and the attributes to be included/excluded are specified in the `mapfile` parameter of the properties file. Note: This `mapfile` can be used only for migration purposes and is not supported for synchronization. A sample properties file is shown in [Appendix B](#).
- d. Generate a new LDIF file in the format required by Oracle Internet Directory by running the command:

```
$ORACLE_HOME/bin/dipassistant bootstrap -f testmigrate.properties
```

- 3. Optionally, you can use an Oracle Internet Directory plug-in to augment entries. See "[Setting up a Plug-in to Augment Active Directory Entries for Linux Authentication](#)" on page 5-1. This method has been shown to work for iPlanet (Sun Java System Directory Server) 5.2 as well as Active Directory
- 4. Get the filtered LDIF file resulting from Step 1 or Step 2 and use either `$ORACLE_HOME/bin/ldapadd` or `$ORACLE_HOME/ldap/bin/bulkload` to add the data to Oracle Internet Directory. If you have more than a few thousand entries, use `bulkload` in preference to `ldapadd`.

- a. The syntax for `ldapadd` is:

```
ldapadd -h oidhost -p oidport -d oiddn -w oidpwd -f ldif_file
```

If you use `ldapadd`, once the data is successfully added, update the Oracle Internet Directory database statistics using `$ORACLE_HOME/ldap/admin/oidstats.sql`. Log in to the Oracle Internet Directory database as the ODS database user and execute this SQL script.

See Also: The `oidstats.sql` command reference in the Oracle Internet Directory Database Tools chapter of the *Oracle Identity Management User Reference*.

- b. If you decide to use `bulkload`, then proceed to Step 4.
- 5. Bulk load LDIF data into Oracle Internet Directory. In the following steps, the file `/home/jdoe/migrationdata.ldif` is the filtered LDIF file.

- a. Stop all Oracle Internet Directory processes by executing the command:

```
$ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=OID
```

- b. Take a cold backup of the database if you have not done so already.
- c. Use `bulkload` to check for schema errors, duplicate entries and other errors and to generate intermediate files for a subsequent data load. The syntax is:

```
$ORACLE_HOME/ldap/bin/bulkload connect=oid-db check=true generate=true  
file=/home/jdoe/migrationdata.ldif
```

When you specify both `check` and `generate` options, `bulkload` checks the entries for schema compliance and duplicates and generates the intermediate files that are used during the load phase in the `$ORACLE_HOME/ldap/load` directory.

If there are any `check`-related errors, `bulkload` reports them on the screen. The tool logs entries in `$ORACLE_HOME/ldap/log/duplicatedn.log` and logs schema-related violations in `$ORACLE_HOME/ldap/log/bulkload.log`. It writes entries that have errors to `$ORACLE_HOME/ldap/load/badentry.ldif`.

If `bulkload` detects any errors in the entries, you might have to fix the entries or schema or both in Oracle Internet Directory. After you fix the problems, re-run the `bulkload` command. Repeat this until there are no errors or the errors reported are acceptable. For example, if you encounter some schema check error for a small number of entries, you can choose to `ldapadd` them from `badentry.ldif` later by fixing the entries or schema in Oracle Internet Directory.

When you use the `check` and `generate` options, `bulkload` generates the intermediate files for entries that had no `check`-related errors. The `generate` occurs even if there are erroneous entries. For example, if the LDIF file has 100 entries and 10 entries have `check` errors, `bulkload` generates the intermediate files for 90 good entries.

- d. Use `bulkload` to load the data, recreate all indexes and generate db statistics. Execute the command:

```
$ORACLE_HOME/ldap/bin/bulkload connect=oid-db load=true
file=/home/jdoe/migrationdata.ldif
```

This command accomplishes three things: loading data from `$ORACLE_HOME/ldap/load` directory into the database using `SQL*Loader`, creating indexes, and generating database statistics.

If it detects an error, `bulkload` indicates the error on the screen. If it reports an error during loading of data, you must restore the database from the backup taken in Step b and then repeat the `bulkload load=true` command. If `bulkload` reports an error during indexing, use the following command to recreate all indexes:

```
bulkload connect=oid-db index=true
```

If `bulkload` reports an error during database statistics generation, you can use the following command to generate the statistics:

```
$OH/ldap/admin/oidstats.sql
```

- e. Start all Oracle Internet Directory processes by executing the command:

```
$ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=OID
```

Setting Access Control on User Entry Attributes

To protect sensitive user attributes from unauthorized modification, set an access control item. Type:

```
ldapmodify -h oidhost -p oidport -D 'cn=orcladmin' -w pwd \
-f aci.ldif
```

where `aci.ldif` looks like this:

```
dn:
changetype: modify
add: orclaci
orclaci: access to attr=(uidnumber,gidnumber,homedirectory,uid)
  by group="cn=OracleUserSecurityAdmins,cn=Groups,cn=OracleContext"
  (search,read,write,compare) by group="cn=directoryadmingroup,cn=oracle internet
  directory" (search,read,write,compare) by * (search,compare,nowrite,nocompare)
```

Using Custom Attributes in Oracle Internet Directory

You can search for an attribute in Oracle Internet Directory only if the attribute is indexed. By default, standard attributes of the user and group entries are indexed. If you use a custom attribute, you can index it by using the `catalog` command. For example, if you migrate automount data to be used by automount programs such as `amd` or `autoofs`, index the `automountKey` attribute by using the `catalog` command, as follows:

```
catalog connect="connect_str" add="TRUE" attribute="automountKey"
```

Note: If you attempt to perform a search with a non-indexed attribute specified as a required attribute, the server will return a "Function not implemented. DSA unwilling to perform" error. See ["Create and Index New Custom Attributes \(Optional\)"](#) on page 2-4.

Some attributes, such as `uid` and user name, must be unique. Oracle Internet Directory will enforce uniqueness if you create a uniqueness constraint for that attribute. For more information see the chapter "Attribute Uniqueness in the Directory" in the *Oracle Internet Directory Administrator's Guide*.

Note: The attribute uniqueness feature works on indexed attributes only.

Migrating SUDO

You can migrate entries from `/etc/sudoers` into Oracle Internet Directory using the `sudo` package you downloaded from:

<http://www.gratisoft.us/sudo>

This `sudo` package includes `sudo` software and the scripts to convert `sudo` data to LDAP data (LDIF). Read the documentation included in the package before you begin the migration process.

Note: After migrating `sudo`, run some security tests to ensure that your `sudo` policy is being enforced correctly.

Migrating SUDO Entries to Oracle Internet Directory on the Server

To move the contents of your `sudoers` file to Oracle Internet Directory, perform these steps:

1. Add a Sudoers container to Oracle Internet Directory using the command:

```
ldapadd -h oid_hostname -p port -D cn=orcladmin \  
-w password -f sudocontainer.ldif
```

where `sudocontainer.ldif` looks like this:

```
dn:ou=Sudoers,dc=us,dc=oracle,dc=com  
objectclass:top  
objectclass:organizationUnit  
ou=sudoers
```

2. Using the `/etc/sudoers` file from your existing `sudo` client, generate an LDIF file by running the conversion script supplied with the `sudo` package you downloaded. Follow the instructions at the download site. Please see the `sudo` package documentation for known limitations.
3. View the resulting LDIF file in a text editor and correct any obvious errors.
4. Add the contents of the `ldif` file to Oracle Internet Directory by using the command:

```
ldapadd -h oid_hostname -p port -D cn=orcladmin \  
-w password -f sudoers.ldif
```

where `sudoers.ldif` is the file generated from your `/etc/sudoers` file.

If `ldapadd` encounters an error, it will stop and report the error. Correct the error and repeat the command until it runs successfully and adds all the entries.

Once you have migrated your `sudo` entries to Oracle Internet Directory, you must use LDAP tools to modify them. See the documentation in the downloaded `sudo` package for information about LDAP browsers you can use for editing `sudo` entries.

Configuring a Client to Use LDAP for SUDO Information

On most client operating systems, you can configure `sudo` with the native LDAP and SSL libraries for that operating system. On a few operating systems, you must use OpenLDAP and OpenSSL.

When you configure `sudo`, the `make install` step will install a new copy of `/etc/ldap.conf`. If you already have an `ldap.conf` file, you must make a copy before you configure `sudo` or the file will be overwritten. Once you have performed a `make install`, copy that file back to `/etc/ldap.conf`.

SuSE 10 Client

1. Download, build, and install the OpenLDAP and OpenSSL packages.
2. If you already have the file `/etc/ldap.conf`, make a copy. For example

```
cp /etc/ldap.conf /etc/ldap.conf.save
```

3. In the directory where you downloaded the `sudo` package, build `sudo` by typing the following commands:

```
./configure --with-ldap-type=openldap --with-pam --enable-ssl  
make all  
make compile  
make install
```

4. If you made a copy of your `ldap.conf` file, copy it back to its original name. For example:

```
cp /etc/ldap.conf.save /etc/ldap.conf
```

5. If there is no `libpam.so` link, make one by typing:

```
cd /usr/lib
ln -s libpam.so.0 libpam.so
```

6. Edit `/etc/pam.d/sudo`. Add the following line above the first `auth` line:

```
auth    sufficient    /lib/security/pam_ldap.so debug
```

7. Modify `/etc/ldap.conf` so that `sudoers_base` points to the base sudoers container. For example:

```
sudoers_base    ou=Sudoers,dc=us,dc=oracle,dc=com
```

If you want to configure `ssl` for `sudo` you must specify `startTLS` in `ldap.conf` since the current `sudo` implement does not support SSL only. For example:

```
ssl startTLS
```

Solaris 9, Solaris 10, or HP-UX 11.23 Client

On these operating systems, the native LDAP client does not support StartTLS. If you plan to use `sudo` with SSL, download, build, and install the OpenLDAP and OpenSSL packages and build `sudo` as described for SuSE 10 clients. Once you have completed those steps, add the following lines to `/etc/ldap.conf` to specify the target LDAP host and port and the SSL certificate authority certificate path and certificate filename:

```
host ldap_host
port ldap_port
tls_cacertdir /etc/ca_certs_dir
tls_cacertfile /etc/ca_cert_file
```

If you plan to use `sudo` in non-SSL mode only, build it using the native LDAP client libraries, as described for other clients.

Other Clients

1. If the `sudo` binary you are using was not built using the `--with-ldap` option, then rebuild the `sudo` command using the `--with-ldap` option, as described in the documentation in the downloaded `sudo` package. Before rebuilding `sudo`, save a copy of `/etc/ldap.conf` to a different name. Be sure to check the documentation and the README files for other options you might need to use. For example, you might have to specify your library and header location or a different configuration file if they are non-standard. You might also have to modify the Makefile by adding an `-lldif` flag to `SUDO_LIBS` if you are using an SDK other than OpenLDAP. Once you have rebuilt `sudo`, copy your saved `ldap.conf` file back to its original name.
2. Modify `/etc/ldap.conf` so that `sudoers_base` points to the base sudoers container you created in Server Step 1. For example:

```
sudoers_base    ou=Sudoers,dc=us,dc=oracle,dc=com
```

If you want to configure SSL for `sudo` you must specify `startTLS` in `ldap.conf` because the current `sudo` implementation does not support SSL only. For example:

```
ssl startTLS
```

Optionally, enable sudo debugging by adding the following line to `/etc/ldap.conf`:

```
Sudoers_debug 2
```

3. Prevent sudo from using the `/etc/sudoers` file by adding the `ignore_local_sudoers` suboption to the sudoers defaults. You do this by running this command:

```
ldapmodify -h oid_hostname -p port -D cn=orcladmin \  
-w password -f ignore_local_sudoers.ldif
```

where `ignore_local_sudoers.ldif` looks like this:

```
dn:cn=defaults, ou=Sudoers, dc=us, dc=oracle, dc=com  
changetype:modify  
add: sudooption  
sudooption: ignore_local_sudoers
```

Reconfiguring a Client to Use `/etc/sudoers`

If you have configured a client computer to use LDAP for sudo, you can reconfigure it to use the `sudoers` file again by commenting out the line that begins with `sudoers_base` in `/etc/ldap.conf`.

Configuring Active Directory Integration

If you have users in Active Directory, and you want to use the credentials stored in Active Directory for Linux or UNIX authentication, you can configure integration with Active Directory. Setting up integration with Active Directory requires several steps:

- You use the Oracle Directory Integration Platform to synchronize user and group entries to Oracle Internet Directory when they are added to or changed in Active Directory.
- You use an Oracle Internet Directory plug-in to add required attributes to the user and group entries in Oracle Internet Directory after they are synchronized from Active Directory to Oracle Internet Directory.
- You use another Oracle Internet Directory plug-in to enable Active Directory authentication of Linux or UNIX users.
- To secure communication, you configure SSL between Oracle Directory Integration Platform and Active Directory and between Oracle Directory Integration Platform and Oracle Internet Directory.

This chapter contains the following sections:

- [Setting up a Plug-in to Augment Active Directory Entries for Linux Authentication](#)
- [Configuring Oracle Directory Integration Platform](#)
- [Configuring SSL Between Oracle Directory Integration Platform and Active Directory](#)
- [Configuring SSL Between Oracle Directory Integration Platform and Oracle Internet Directory](#)
- [Setting Up the External Authentication Plug-in](#)

Setting up a Plug-in to Augment Active Directory Entries for Linux Authentication

User entries in Active Directory do not include key information required for Linux authentication. Therefore, when you synchronize users from Active Directory into Oracle Internet Directory by using the Active Directory connector of Oracle Directory Integration Platform, you must augment those user entries with the required information. To facilitate this, the product includes a PL/SQL plug-in that can be enabled on Oracle Internet Directory.

Enable the plug-in as follows:

1. Use a text editor to make the following changes to `$ORACLE_HOME/ldap/admin/posixattr_when_add.pls`:

- In line 71, replace the value of `v_homeDirectory` with the desired home directory.
 - In line 72, replace the value of `v_loginShell` with the desired login shell.
 - In line 73, replace the value of `v_gidNumber` with the GID number of the users
2. Load the plug-in package into the database by typing:


```
sqlplus ods/odspwd@$ORACLE_HOME/ldap/admin/posixattr_when_add.pls
```

 where `odspwd` is the password of the ODS user.
 3. Use a text editor to make the following change in `$ORACLE_HOME/ldap/admin/posixattr_when_add.ldif`: Replace the value of `orclpluginsubscriberdnlist` with your realm's DN.
 4. Add the plug-in to Oracle Internet Directory by running the following command:


```
ldapadd -h host -p port -D cn=orcladmin -w password \
-f $ORACLE_HOME/ldap/admin/posixattr_when_add.ldif
```

Configuring Oracle Directory Integration Platform

Oracle Directory Integration Platform is documented in the *Oracle Identity Management Integration Guide*. The following procedure refers to that document in several places.

To enable Oracle Directory Integration Platform for Active Directory integration with Oracle Authentication Services for Operating Systems, perform these steps:

1. Verify the synchronization requirements, as described in "Verifying Synchronization Requirements," under "Configuring Synchronization with a Third-Party Directory," in Chapter 18 of the *Oracle Identity Management Integration Guide*.
2. Create a synchronization profile by running `dipassistant expressconfig`, as described in Step 1 of "Creating Synchronization Profiles with Express Configuration," under "Configuring Synchronization with a Third-Party Directory," in Chapter 18 of the *Oracle Identity Management Integration Guide*.
3. Edit the profiles resulting from the express configuration. To understand mapping rules, see: "Configuring Mapping Rules," in Chapter 6 of the *Oracle Identity Management Integration Guide*.

Make the following changes:

- a. Change the domain rules to point to the following domain in Oracle Internet Directory: `ou=People,dc=us,dc=oracle,dc=com`.
- b. Comment out this line:


```
userPrincipalName: :user:uid: inetorgperson:userPrincipalName
```
- c. Uncomment this line


```
#sAMAccountName: :user:uid: inetorgperson
```
- d. Add this line:


```
cn: :person:gecos: :person:
```

See the sample synchronization profile in [Appendix D](#). The customizations are shown in **boldface**.

4. Continue with Steps 2-5 of "Creating Synchronization Profiles with Express Configuration," under "Configuring Synchronization with a Third-Party Directory," in Chapter 18 of the *Oracle Identity Management Integration Guide*.

Configuring SSL Between Oracle Directory Integration Platform and Active Directory

To secure communications between Oracle Directory Integration Platform and Active Directory using SSL, perform the following steps:

1. Shut down the Oracle Directory Integration Platform server by executing the following command as the user who installed Oracle Internet Directory:

```
oidctl configset=1 connect=db_connect_string instance=1 server=odisrv stop
```

where `db_connect=string` is the backend database connect string that was set during installation of Oracle Internet Directory.

2. Configure Oracle Directory Integration Platform to use SSL server authentication by executing the following command:

```
dipassistant modifyprofile -h oid_host -profile profile_name -p oid_port \
-D oid_dn odip.profile.condirurl=host:port:2
```

The value 2 in the URL specifies SSL server authentication.

3. Export the Active Directory SSL server certificate to a file and import the result into an Oracle Wallet by executing the following commands:

```
orapki wallet create -wallet /usr/lib/oracle/oid/wallet/ad -pwd wallet_pwd
orapki wallet add -cert Exported_AD_Cert_File -trusted_cert \
-pwd wallet_pwd
```

4. Edit the file `$ORACLE_HOME/ldap/odi/conf/odi.properties` to set values for the wallet location (`certWalletFile`) and the file to store the wallet password (`certWalletPwdF`), as follows:

```
certWalletFile: /usr/lib/oracle/oid/wallet/ad/cert
certWalletPwdF: /usr/lib/oracle/oid/wallet/ad/certWalletPwd
```

Ensure that there are no trailing spaces at the ends of the lines.

5. Create the `certWalletPwdF` file by executing the following command:

```
dipassistant wpasswd
```

Enter your wallet password when prompted.

6. To start the Oracle Directory Integration Platform server, execute the following command as root:

```
oidctl configset=1 connect=xe instance=1
server=odisrv flags='port=OID_port grpId=defaultgroup' start
```

where `OID_port` is the Oracle Internet Directory port number.

Configuring SSL Between Oracle Directory Integration Platform and Oracle Internet Directory

To secure communications between Oracle Directory Integration Platform and Oracle Internet Directory using SSL, perform the following steps:

1. To shut down the Oracle Directory Integration Platform server, execute the following command as root:

```
oidctl configset=1 connect=x e instance=1 server=odisrv stop
```

2. Edit the file `$ORACLE_HOME/ldap/odi/conf/odi.properties` to set values for the wallet location (`certWalletFile`) and the file to store the wallet password (`certWalletPwdf`), as follows:

```
certWalletFile: /usr/lib/oracle/oid/wallet/server
certWalletPwdf: /usr/lib/oracle/oid/wallet/server/certWalletPwdf
```

Ensure that there are no trailing spaces at the ends of the lines.

3. Create the `certWalletPwdf` file by executing the following command:

```
dipassistant wpasswd
```

Enter your wallet password when prompted.

4. Start the Oracle Directory Integration Platform server by executing the following command as root:

```
oidctl configset=1 connect=x e instance=1
server=odisrv flags='port=OID_port grpId=defaultgroup' start
```

where `OID_port` is the Oracle Internet Directory port number.

Setting Up the External Authentication Plug-in

Enable the External Authentication plug-in shipped with Oracle Internet Directory so that Linux authentication uses the credentials stored in Active Directory.

To configure and enable this plug-in, use the `extauth` operation of the Directory Integration Assistant (`dipassistant`) utility. The command syntax is:

```
dipassistant extauth [-h hostName] [-p port] -D bindDN -w bindPassword \
-t extDirType
```

See the `dipassistant` section of the chapter entitled "Oracle Directory Integration Platform Tools" in the *Oracle Identity Management User Reference* for more information on how to use the `extauth` operation.

If you want to set up an external authentication plug-in to work with multiple external authentication domains, you must perform some manual instructions after you run the external configuration tool. See "Configuring External Authentication Against Multiple Domains," under "Configuring External Authentication Plug-ins," in Chapter 18 of the *Oracle Identity Management Integration Guide*.

Managing Oracle Authentication Services for Operating Systems

This chapter contains the following topics:

- [Creating Home Directories](#)
- [Managing Users and Groups With `libuser` Tools](#)
- [Managing Oracle Internet Directory With Oracle Directory Manager and Command-Line Utilities](#)
- [Managing Password Policies](#)

Creating Home Directories

On Linux systems, you do not have to create each user's home directory when you migrate or add that user to Oracle Internet Directory. The client configuration script that you ran on each client computer enabled the creation of each user's home directory on first login. On operating systems other than Linux, however, you must manually create user home directories.

Managing Users and Groups With `libuser` Tools

If your client has the `libuser` library and you have configured it to use Oracle Internet Directory, you can use `system-config-users` or `luseradd` to add users. When you invoke one of the `libuser` commands, it will prompt you for the password for logging into Oracle Internet Directory. See your operating system documentation for more information about `system-config-users`.

Note:

- If you use `system-config-users` or other tools in the `libuser` package to add passwords or entries containing passwords, Oracle Internet Directory cannot enforce its password policies on those passwords. The reason is that the `libuser` tools generate a hashed password before sending it to Oracle Internet Directory, so Oracle Internet Directory cannot determine whether the password meets policy criteria or not.
 - The `system-config-users` tool requires that you configure your client and server for SSL.
 - Before using `system-config-users`, ensure that the user entries have all the required attributes shown in "[Migrating from NIS to Oracle Internet Directory](#)" on page 4-2. The tool may report errors if fields are missing.
 - You cannot use the non-`libuser` commands `useradd`, `userdel`, `groupadd`, or `groupdel` for user or group administrative tasks.
-

If you do not have `libuser`, or you have not configured it to use Oracle Internet Directory, you can use ODM, LDAP commands, or bulk tools. See the *Oracle Internet Directory Administrator's Guide* for more information.

Managing Oracle Internet Directory With Oracle Directory Manager and Command-Line Utilities

The *Oracle Internet Directory Administrator's Guide* contains information about managing Oracle Internet Directory. See the "Directory Administration and Monitoring Tools" chapter for information on Oracle Directory Manager. See the "Process Management" chapter for information on starting and stopping Oracle Internet Directory. See the Using Bulk Tools chapter for information on the bulk tools.

The *Oracle Identity Management User Reference* provides the syntax for Oracle Internet Directory command-line tools, including the bulk tools and LDAP tools.

Please see the *Oracle Internet Directory Administrator's Guide* for information about modifying or deleting users and groups.

Testing Whether a User Has Been Added

You can test whether a user has been added by using the following command:

```
ldapsearch -D cn=orcladmin -w password -b 'searchbase' -s -sub '(uid=username)'
```

where `searchbase` is the realm, for example, `dc=us,dc=oracle,dc=com`. You can also test the account by logging in as the user. For example, you can log in to one client from another by using `ssh`. For example:

```
ssh -l username hostname
```

Once you are logged in, type:

```
id
```

to confirm that you are logged in as the correct user.

Changing a User's Password by Using `ldapmodify`

To change a user's password, you use the command:

```
ldapmodify -p port -h host -D binddn -w old_password -v -f passwd_file
```

where `passwd_file` looks like this:

```
dn: userDN
changetype: modify
replace: userpassword
userpassword: new_password
```

Note:

- After you have used `passwd_file`, delete it or remove the cleartext password.
 - Users can change their own passwords by using the `passwd` command.
-
-

Adding a User by Using `ldapadd`

To add users and groups from the command line you use a command line such as:

```
ldapadd -p port -h host -D binddn -w bindpwd -v -f ldif_file
```

where `ldif_file` contains the information about the entry you are adding in LDIF format.

In the following `ldif_file` example, we create a user called `jueno`. The user is created in the user container `ou=People, dc=us, dc=oracle, dc=com` under the realm `dc=us, dc=oracle, dc=com`. To create a user, you must provide the following attributes: `uid`, `homedirectory`, `loginshell`, `uidnumber`, `gidnumber`, `cn`, `objectclass`, and `userpassword` (in cleartext). For compatibility with a variety of clients and with the `system-config-users` management tool, use all the object classes shown in the example.

```
dn: uid=jueno,ou=People,dc=us,dc=oracle,dc=com
uid: jueno
homedirectory: /home/jueno
loginshell: /bin/tcsh
uidnumber: 506
gidnumber: 506
cn: juri ueno
objectclass: posixAccount
objectclass: shadowAccount
objectclass: account
objectclass: top
userpassword: password
shadowwarning: -1
shadowmax: 99999
shadowlastchange: 13916
shadowexpire: -1
shadowmin: 0
shadowinactive: -1
gecos: jueno
```

After you have used the LDIF file, delete it or remove the cleartext password.

Adding a Group by Using Idapadd

To add groups from the command line, you use the same command line you use to add users. That is:

```
ldapadd -p port -h host -D binddn -w bindpwd -v -f ldif_file
```

In the following example, we create a group called `kobukuro` with group ID 505. The group is created in the group container `ou=Group, dc=us, dc=oracle, dc=com` in the realm `dc=us, dc=oracle, dc=com`. We also add a member, `juero`, at the same time, by specifying the `memberuid` and the value. The LDIF file looks like this:

```
dn: cn=kobukuro,ou=Group,dc=us,dc=oracle,dc=com
cn: kobukuro
gidnumber: 505
objectclass: posixGroup
objectclass: groupOfUniqueNames
objectclass: top
memberuid: juero
```

Adding a member to the group at the same time is optional.

Managing Password Policies

See the Managing Password Policies chapter in *Oracle Internet Directory Administrator's Guide*.

Note: If you use `system-config-users` or other tools in the `libuser` package to add passwords or entries containing passwords, Oracle Internet Directory cannot enforce its password policies on those passwords. The reason is that the `libuser` tools generate a hashed password before sending it to Oracle Internet Directory, so Oracle Internet Directory cannot determine whether the password meets policy criteria or not.

Troubleshooting

This appendix lists problems you might encounter when configuring or managing Oracle Authentication Services for Operating Systems. It contains the following topics:

- [Data Migration Errors](#)
- [Management Tool Problems](#)
- [Testing and Log File Messages](#)
- [User Login Errors](#)

Data Migration Errors

This section lists errors you might encounter when migrating entries to Oracle Authentication Services for Operating Systems.

Sudo Conversion Script Errors

Problem

The `sudo` conversion tool reports parse errors while converting your `/etc/sudoers` file to LDIF format.

Solution

The conversion script in the `sudo` package might not cover all intricacies of your `sudoers` file format. For example, if command aliases are preceded by an exclamation mark (!), remove the exclamation mark. Please see the `sudo` package documentation for known limitations.

Management Tool Problems

This section lists errors you might encounter when using management tools with Oracle Authentication Services for Operating Systems.

Error in `system-config-users`

Problem

You encounter errors when using the `system-config-users` tool.

Solution

Ensure that user entries have all the attributes described in ["Migrating from NIS to Oracle Internet Directory"](#) on page 4-2.

Solution

For errors when creating a new group on Red Hat Enterprise Linux, version 4, edit the file `/usr/share/system-config-users/userGroupCheck.py`.

Change:

```
def isGroupnameOk(str, widget):
```

to:

```
def isGroupnameOk(name, widget):
```

The libuser Tools Fail with Python Errors

Problem

You see Python errors when invoking libuser tools such as `system-config-users` and `luseradd`.

Solution

To use libuser tools, you must configure your client and server for SSL. See ["Switching Between SSL Authentication and Non-SSL Configurations"](#) on page 3-9.

Linux Management Tools Cause Inconsistencies

Problem

Using Linux tools such as `useradd`, `userdel`, `groupadd`, or `groupdel` causes inconsistencies or unexpected behavior.

Solution

These tools are not supported. After you install Oracle Authentication Services for Operating Systems and migrate your data to Oracle Internet Directory, you must use specific tools to manage users, passwords, and other data. Specifically, you must use:

- Oracle Directory Manager
- The LDAP tools and bulk tools in `$ORACLE_HOME/bin`
- The `passwd` command

You can also use the libuser tools on Linux distributions that support it, with some limitations. See ["Password Policy Not Consistently Enforced"](#) on page A-5.

Idapsearch Error

Problem

When you attempt to perform a search, the server returns this error:

```
Function not implemented. DSA unwilling to perform.
```


Solution

You have attempted to perform a search with a non-indexed attribute specified as a required attribute.

You can search for an attribute in Oracle Internet Directory only if the attribute is indexed. By default, standard attributes of the user and group entries are indexed. If you use a custom attribute, you can index it by using the `catalog` command. For example:

```
catalog connect="connect_str" add="TRUE" attribute="automountKey"
```

Testing and Log File Messages

This section describes some testing techniques and explains some messages you might find in log files when running Oracle Authentication Services for Operating Systems.

Enabling Log Messages for All Operations

Problem

Administrators need to monitor Oracle Internet Directory.

Solution

You can set a debug level that causes Oracle Internet Directory to generate log messages for all operations.

Set the function trace debug level on Oracle Internet Directory by using the following command line:

```
ldapmodify -p port -h host -D cn=orcladmin -w password -v -f debug.ldif
```

where `debug.ldif` looks like this:

```
dn:
changetype: modify
replace: orcldebugflag
orcldebugflag: 117440511
-
replace: orcldebugforceflush
orcldebugforceflush: 1
```

Testing StartTLS

Problem

StartTLS, which enables you to negotiate an SSL connection on a previously clear connection, is transparent to the user. Administrators need a way to verify that StartTLS is working.

Note: StartTLS is not used on HP-UX and Solaris Oracle Internet Directory servers. On these platforms, SSL is configured on a different port from non-SSL connections.

Solution

To verify that StartTLS is working, set a debug level that causes Oracle Internet Directory to generate a log message when an SSL negotiation begins. Because the clients are all pointing to the non-SSL port, generation of this message implies that startTLS is working.

Perform the following steps:

1. Set the function trace debug level on Oracle Internet Directory by using the following command line:

```
ldapmodify -p port -h host -D cn=orcladmin -w password -f debug.ldif -v
```

where `debug.ldif` looks like this:

```
dn:  
changetype: modify  
replace: orcldebugflag  
orcldebugflag: 25165824  
-  
replace: orcldebugforceflush  
orcldebugforceflush: 1
```

2. Perform an authentication operation that invokes the Oracle Internet Directory server. For example, use `ssh` to connect to a client that is configured to authenticate against Oracle Internet Directory.
3. Examine the log files in `$ORACLE_HOME/ldap/log`. Look for messages containing the string `gslsflnNegotiateSSL`.

Password Syntax Errors

Problem

Log files contain messages about password syntax, and Oracle Internet Directory is not being used for password policy enforcement.

Solution

If you are not using Oracle Internet Directory for password policy enforcement, you must disable password policies in Oracle Internet Directory by setting `orclpwdpolicyenable` to 0. To avoid messages about password syntax, you must also disable the password syntax check by setting `pwdCheckSyntax` to 0.

User Login Errors

This section lists errors users might encounter when attempting to log in when Oracle Authentication Services for Operating Systems is used for authentication.

Users Cannot Log In

Problem

Users cannot log in after you run the client configuration script.

Solution

On some operating systems, if `nscd` or `sshd` is running while you execute the `config_OIDclient.sh` or `sslConfig_OIDclient.sh` script, user authentication might not work after the configuration. Restart `sshd` or `nscd` to correct the problem.

User's Home Directory Does Not Exist

Problem

Adding or migrating a user to Oracle Internet Directory does not create that user's home directory.

Solution

On Linux systems, you do not have to create a user's home directory on the client computer when you add that user to Oracle Internet Directory. The client configuration script that you ran on each client computer enabled the creation of each user's home directory on first login. On operating systems other than Linux, however, you must manually create user home directories.

User's Shell Does Not Exist

Problem

When attempting to log in, the user sees a message such as:

```
No shell
Connection closed by foreign host.
```

Solution

This problem occurs when a user entry in Oracle Internet Directory specifies a shell pathname that does not exist on the computer where the user is logging in. Supported shells and shell pathnames vary from one operating system to another. For example, one operating system might have `sh`, `csh`, `bash`, and `tcsh` under `/bin`, and another might have `sh` and `csh` under `/usr/bin`.

If the user must be able to log in on computers with different shell pathnames, you might have to create a symbolic link to the shell on one of the computers.

Password Policy Not Consistently Enforced

Problem

Oracle Internet Directory fails to enforce password policies, or password policy enforcement is not as expected.

Solution

If you use Oracle Internet Directory to enforce password policies, you cannot use tools in the `libuser` package to add passwords or entries containing passwords. The reason is that the `libuser` tools generate a hashed password before sending it to Oracle Internet Directory, so Oracle Internet Directory cannot determine whether the password meets policy criteria or not. Use the LDAP tools or Oracle Directory Manager instead.

Solution

If you are using Oracle Internet Directory for password policy enforcement, you must set `shadowmax` to 99999 and `shadowexpire` to -1 to disable password expiration by the operating system.

Properties File for LDAP Migration

This is a sample of a properties file, discussed in "[Migrating from Another LDAP Directory to Oracle Internet Directory](#)" on page 4-3.

```
#####
## This configuration file provides necessary information for      ##
## performing the bootstrapping of OiD and a Connected directory. ##
#####

# Source Type : Specifies whether, source end of the bootstrapping is
# LDAP or LDIF.
#
#
odip.bootstrap.srctype = LDIF

# Source URL : In case of LDAP source type it specifies the source directory
# location. In case of LDIF it specifies the location of the LDIF file.
#
# NOTE - e.x for LDAP the expected format is host[:port]
#         for LDIF the expected format is absolute path of the file
#
odip.bootstrap.srcurl = oracle/ldap/odip/scr/IPlanet.ldif

# Source DN : This information supplements the Source URL. In case of LDIF
# binding this parameter is meaningless. However in case of LDAP this parameter
# specifies the Bind DN.
#
#odip.bootstrap.srcdn

# Source Password : Bind password. In case of LDAP binding this is used as
# security credential
#
#odip.bootstrap.srcpasswd

# Destination Type : Specifies whether, destination end of the bootstrapping
# is LDAP or LDIF.
#
# NOTE - In future bootstrapping with a TAGGED and PLSQL based interfaces
# would be supported.
#
odip.bootstrap.desttype = LDIF

# Destination URL : In case of LDAP it specifies the directory location
# In case of LDIF it specifies the location of the LDIF file.
#
# NOTE - e.x for LDAP the expected format is host[:port]
#         for LDIF the expected format is absolute path of the file
```

```

#
odip.bootstrap.desturl = /oracle/ldap/odip/scr/OiD.ldif

# Destination DN : This information supplements the destination URL.
# In case of LDIF binding this parameter is meaningless. However in case of
# LDAP this parameter specifies the Bind DN.
#
#odip.bootstrap.destdn

# Destination Password : Bind password. In case of LDAP binding this is
# used as security credential
#
# NOTE - It is not recommended to specify the password in this file.
#
#odip.bootstrap.destpasswd

# and domain mappings.
#
odip.bootstrap.mapfile = /oracle/ldap/odip/scr/bootstrap.map

#
# NOTE - If this file already exists then it will be backed up and a new
# version will be created
#
odip.bootstrap.logfile = /oracle/ldap/odip/scr/bootstrap.log

# Log Messages Severity : Specifies the type of the log messages that needs
# to be logged
#
#           INFO      ---- 1
#           WARNING   ---- 2
#           DEBUG     ---- 4
#           ERROR     ---- 8
#
# NOTE - A combination of these types could also be given. for ex if you are
# interested
# only in WARNING and ERROR message then specify value 8+1 i.e 9 Similarly for all
# types of message use 1 + 2 + 4 + 8 = 15
#
odip.bootstrap.logseverity = /oracle/ldap/odip/scr/bootstrap.log

# Trace file : Specifies the location of the trace file. The default
# trace file will be bootstrap.trc created under $OH/ldap/odi/log directory
#
# NOTE - If this file already exists then it will be backed up and a new
# version will be created
#
odip.bootstrap.trcfile = /oracle/ldap/odip/scr/bootstrap.trc

```

Sample Mapfiles

This appendix contains a template mapfile and some sample mapfiles.

This appendix includes the following sections:

- [Template Mapfile](#)
- [Sample Mapfile 1](#)
- [Sample Mapfile 2](#)
- [Sample Mapfile 3](#)
- [Sun Java System Directory Server Mapfile 1](#)
- [Sun Java System Directory Server Mapfile 2](#)
- [eDirectory Mapfile](#)

Template Mapfile

```
DomainRules
# Specify the list of domain rules
DomainExclusionList
# Specify the list of domains to be excluded in migration
###
AttributeRules
# List the attributes that are to be migrated
AttributeExclusionList
# Specify the list of attributes that are to be excluded
~
```

Sample Mapfile 1

```
# This file contains the domain rules with the list of containers to be migrated
# and the list of attributes to be migrated.
DomainRules
ou=groups,dc=us,dc=oracle,dc=com
ou=people,dc=us,dc=oracle,dc=com
ou=system administrators,dc=us,dc=oracle,dc=com
###
AttributeRules
Cn
Sn
Givenname
Objectclass
```

Sample Mapfile 2

```
# This file contains the domain rules with the list of containers to be migrated
# and the list of attributes to be filtered
DomainRules
ou=groups,dc=us,dc=oracle,dc=com
ou=people,dc=us,dc=oracle,dc=com
ou=system administrators,dc=us,dc=oracle,dc=com
###
AttributeRules
*.*
AttributeExclusionList
modifytimestamp
createtimestamp
modifiersname
creatorsname
```

Sample Mapfile 3

```
# This file contains domain rules with the list of containers to be excluded and
# the list of attributes to be excluded
DomainRules
*.*
DomainExclusionList
ou=system administrators,dc=us,dc=oracle,dc=com
###
AttributeRules
*.*
AttributeExclusionList
modifytimestamp
createtimestamp
modifiersname
creatorsname
```

Sun Java System Directory Server Mapfile 1

```
# This file contains domain rules with the list of containers to be excluded and
# the list of attributes to be excluded
DomainRules
ou=groups,dc=us,dc=oracle,dc=com:ou=groups,dc=us,dc=oracle,dc=com:cn=%,ou=group,dc
=us,dc=oracle,dc=com
ou=people,dc=us,dc=oracle,dc=com: ou=people,dc=us,dc=oracle,dc=com:uid=%
ou=people,dc=us,dc=oracle,dc=com
ou=system
administrators,dc=us,dc=oracle,dc=com:ou=people,dc=us,dc=oracle,dc=com:uid=%,
ou=people,dc=us,dc=oracle,dc=com
DomainExclusionList
###
AttributeRules
*.*
AttributeExclusionList
modifytimestamp
createtimestamp
modifiersname
creatorsname
nsuniqueid
aci
```


Sun Java System Directory Server Mapfile 2

```
# This file contains domain rules with the list of containers to be excluded and
# the list of attributes to be excluded
DomainRules
*.*
###
AttributeRules
*.*
AttributeExclusionList
modifytimestamp
createtimestamp
modifiersname
creatorsname
nsuniqueid
aci
```

eDirectory Mapfile

```
# This file contains domain rules with the list of containers to be excluded and
# the list of attributes to be excluded
DomainRules
*.*
###
AttributeRules
*.*
AttributeExclusionList
modifytimestamp
createtimestamp
modifiersname
creatorsname
```

Synchronization Profile for Active Directory Integration

This properties file was generated by running `dipassistant express` and then customizing the file, as described in ["Configuring Oracle Directory Integration Platform"](#) on page 5-2. The customizations are shown in **boldface**.

```
# USE THIS MAP FILE, IF DOMAIN IN ACTIVE DIRECTORY IS DIFFERENT FROM DOMAIN IN OID
# FOR ONE-TO-ONE DOMAIN MAPPING USE ACTIVECHG.MAP.MASTER IN ODI/CONF DIRECTORY
DomainRules
CN=USERS,DC=sgttest01v1oimad,DC=com:ou=People,dc=us,dc=oracle,dc=com:
###
AttributeRules
# attribute rule common to all objects
objectguid: :binary: :orclobjectguid: : :bin2b64(objectguid)
ObjectSID: :binary: :orclObjectSID: : :bin2b64(ObjectSID)
distinguishedName: : :orclSourceObjectDN: :orclADObject
# attribute rule for mapping windows organizationalunit
ou: : :organizationalunit:ou: : organizationalunit:
# attribute rule for mapping directory containers
cn: : :container: cn: :orclContainer:
# attribute rule for mapping directordomains
dc: : :domain: dc: :domain:
# USER ENTRY MAPPING RULES
# attribute rule for mapping windows LOGIN id
sAMAccountName,userPrincipalName: : :user:orclSAMAccountName:
:orclADUser:toupper(truncl(userPrincipalName,'@'))+"$"+sAMAccountName
# attribute rule for mapping Active Directory LOGIN id
userPrincipalName: : :user:orclUserPrincipalName: :orclADUser:userPrincipalName
# Map the userprincipalname to the nickname attr by default
#userPrincipalName: : :user:uid: :inetorgperson:userPrincipalName
# Map the SamAccountName to the nickname attr if required
# If this rule is enabled, userprincipalname rule needs to be disabled
sAMAccountName: : :user:uid: :inetorgperson
# Assign the userprincipalname to Kerberos principalname
userPrincipalName: : :user:krbPrincipalName:
:orcluserv2:trunc(userPrincipalName,'@')+'@'+toupper(truncl(userPrincipalName,'@')
)
# This rule is mapped as SAMAccountName is a mandatory attr on AD
# and sn is mandatory on OID. sn is not mandatory on Active Directory
sAMAccountName: : :user:sn: : person:
# attributes to map to cn - normally this is the given name
cn: : :person:cn: :person:
departmentNumber: : :inetorgperson:departmentnumber: :organizationalperson:
# attribute rule for mapping entry and to create orclUserV2
# There should be a mapping rule with orcluserv2 objectclass
# without which the PORTAL may not function properly
```

```
# The next rule shows any attribute of any objectclass can be mapped
# to different attribute of different objectclass so long as the
# schema and syntax are compatible.
givenName: : :user:displayName: :orclUserV2:
employeeID: : :user:employeeNumber: :inetOrgPerson:
physicalDeliveryOfficeName: : :user:physicalDeliveryOfficeName:
:organizationalPerson:
title: : :user:title: :organizationalPerson:
mobile: : :organizationalperson:mobile: :inetorgperson:
telephonenumber: : :organizationalperson:telephonenumber: :inetorgperson:
facsimileTelephoneNumber: : :organizationalperson:facsimileTelephoneNumber:
:inetorgperson:
l: : :user:l: :person:
# mail needs to be assigned valid value for default settings in DAS
userPrincipalName: : :user:mail: :inetorgperson:
# GROUP ENTRY MAPPING RULES
cn: : :group:cn: :groupofuniquenames:
# displayname needs to be assigned a valid value for default settings on DAS
SAMAccountName: : :group:displayName: :orclgroup:
# Description needs to be assigned a valid value for default settings on DAS
Description: : :group:Description: :groupOfUniqueNames:
member: : :group:uniquemember: :groupofUniqueNames:dnconvert(member)
managedby: : :group:owner: :orclprivilegegroup:dnconvert(managedby)
sAMAccountName: : :group:orclSAMAccountName: :orclADGroup:
# Add parameter for Oracle Authentication Services for Operating Systems
cn: : :person:gecos: :person:
```

Index

A

- Active Directory integration
 - configuring Directory Integration Platform, 5-2
 - configuring SSL between DIP and Active Directory, 5-3
 - configuring SSL between DIP and Oracle Internet Directory, 5-3
 - external authentication plug-in, 5-4
 - general, 5-1
 - plug-in to augment entries, 5-1
- adding a group, 6-4
- adding a user, 6-3
- authentication
 - configuring on client, 3-5
 - configuring on server, 3-4

C

- changing a user's password, 6-3
- choosing product features, 2-2
- client configuration, 3-5
- configuration
 - restoring client and server, 3-10
- configuration scripts
 - rerunning, 3-9
- configuration tools, 3-3
- creating home directories, 6-1
- custom attributes
 - indexing, 2-4, 4-7

D

- data migration from another LDAP directory, 4-4

E

- enabling log messages, A-3

H

- home directories
 - creating, 6-1
 - not created, A-5

I

- indexing custom attributes, 2-4, 4-7

L

- ldapsearch
 - errors while using, A-2
- libuser
 - errors while using, A-2
- libuser tools, 6-1
- log messages
 - enabling, A-3
 - password syntax, A-4
 - StartTLS, A-3
- login errors, A-4

M

- managing password policies, 6-4
- mapfile
 - examples, C-1
- migrating entries
 - from another LDAP directory, 4-3
 - from files, 4-2
 - from NIS, 4-2
 - general, 4-1
- migrating sudo, 4-7

N

- no shell error, A-5

P

- password
 - changing, 6-3
- password policy
 - configuration, 3-8
 - disabling local policies, 3-8
 - enforcement, 3-2
 - inconsistent enforcement, A-5
 - managing, 6-4
- password syntax errors, A-4
- plug-in
 - external authentication by Active Directory, 5-4
 - to augment Active Directory Entries, 5-1

- prerequisites
 - dipassistant patch, 2-3
 - NIS migration scripts, 2-3
 - operating system, 2-1
 - Oracle Directory Integration Platform, 2-1
 - Oracle Internet Directory, 2-1
 - Oracle Internet Directory patches, 2-2
 - sudo package, 2-4
- product features
 - choosing, 2-2

R

- rerunning configuration scripts, 3-9

S

- schema migration from another LDAP directory, 4-3
- server configuration, 3-4
- shell does not exist, A-5
- SSL
 - certificates, 3-2, 3-7
 - configuring between DIP and Active Directory, 5-3
 - configuring between Oracle Internet Directory and Active Directory, 5-3
 - support, 3-1
 - switching between SSL and non-SSL authentication, 3-9
- StartTLS
 - testing, A-3
- sudo
 - configuring a client, 4-8
 - conversion script errors, A-1
 - migration, 4-7
 - reconfiguring a client to use sudoers file, 4-10
- sudoers file
 - parsing errors, A-1
- system-config-users
 - errors when using, A-1

T

- testing whether a user has been added, 6-2
- tools
 - adding a group, 6-4
 - adding a user, 6-3
 - changing a user's password, 6-3
 - command line, 6-2
 - configuration, 3-3
 - errors during use, A-1
 - ldapadd, 6-3, 6-4
 - ldapmodify, 6-3
 - libuser, 6-1
 - libuser errors, A-2
 - Oracle Internet Directory management, 6-2
 - unsupported, A-2

U

- unsupported Linux management tools, A-2