# NetIQ® Identity Manager

## Driver for LDAP Implementation Guide

**July 2014**

NetIQ.

# Contents

# About this Book and the Library

The *Identity Manager Driver for LDAP Implementation Guide* explains how to install, configure, and manage the Identity Manager Driver for LDAP.

## Intended Audience

This guide is for NetIQ Identity Manager administrators who are using the Identity Manager Driver for LDAP.

## Other Information in the Library

The library provides the following information resources:

**Identity Manager Framework Installation Guide**

Provides detailed planning and installation information for Identity Manager components.

**Identity Manager Integrated Installation Guide**

Provides integrated installation information for installing Identity Manager components.

**Identity Manager Overview Guide**

Provides conceptual information about Identity Manager. This book also provides an overview of the various components and many administration tasks.

**Identity Manager Driver Guides**

Provides implementation information about Identity Manager drivers.

# About NetIQ Corporation

We are a global, enterprise software company, with a focus on the three persistent challenges in your environment: Change, complexity and risk—and how we can help you control them.

## Our Viewpoint

**Adapting to change and managing complexity and risk are nothing new**

In fact, of all the challenges you face, these are perhaps the most prominent variables that deny you the control you need to securely measure, monitor, and manage your physical, virtual, and cloud computing environments.

**Enabling critical business services, better and faster**

We believe that providing as much control as possible to IT organizations is the only way to enable timelier and cost effective delivery of services. Persistent pressures like change and complexity will only continue to increase as organizations continue to change and the technologies needed to manage them become inherently more complex.

## Our Philosophy

**Selling intelligent solutions, not just software**

In order to provide reliable control, we first make sure we understand the real-world scenarios in which IT organizations like yours operate—day in and day out. That's the only way we can develop practical, intelligent IT solutions that successfully yield proven, measurable results. And that's so much more rewarding than simply selling software.

**Driving your success is our passion**

We place your success at the heart of how we do business. From product inception to deployment, we understand that you need IT solutions that work well and integrate seamlessly with your existing investments; you need ongoing support and training post-deployment; and you need someone that is truly easy to work with—for a change. Ultimately, when you succeed, we all succeed.

## Our Solutions

- Identity & Access Governance
- Access Management
- Security Management
- Systems & Application Management
- Workload Management
- Service Management

## Contacting Sales Support

For questions about products, pricing, and capabilities, contact your local partner. If you cannot contact your partner, contact our Sales Support team.

| | |
|---|---|
| **Worldwide:** | www.netiq.com/about_netiq/officelocations.asp |
| **United States and Canada:** | 1-888-323-6768 |
| **Email:** | info@netiq.com |
| **Web Site:** | www.netiq.com |

## Contacting Technical Support

For specific product issues, contact our Technical Support team.

| | |
|---|---|
| **Worldwide:** | www.netiq.com/support/contactinfo.asp |
| **North and South America:** | 1-713-418-5555 |
| **Europe, Middle East, and Africa:** | +353 (0) 91-782 677 |
| **Email:** | support@netiq.com |
| **Web Site:** | www.netiq.com/support |

## Contacting Documentation Support

Our goal is to provide documentation that meets your needs. The documentation for this product is available on the NetIQ Web site in HTML and PDF formats on a page that does not require you to log in. If you have suggestions for documentation improvements, click **Add Comment** at the bottom of any page in the HTML version of the documentation posted at www.netiq.com/documentation. You can also email Documentation-Feedback@netiq.com. We value your input and look forward to hearing from you.

## Contacting the Online User Community

NetIQ Communities, the NetIQ online community, is a collaborative network connecting you to your peers and NetIQ experts. By providing more immediate information, useful links to helpful resources, and access to NetIQ experts, NetIQ Communities helps ensure you are mastering the knowledge you need to realize the full potential of IT investments upon which you rely. For more information, visit community.netiq.com.

# 1 Understanding the LDAP Driver

The Identity Manager Driver for LDAP (LDAP driver) synchronizes data between the Identity Vault and LDAP-compliant directories. The driver supports the Subscriber and Publisher channels, uses filters to control objects and attributes, and uses policies to control data.

## 1.1 Data Transfer Between Systems

### 1.1.1 Synchronizing Data

The Identity Manager Driver for LDAP synchronizes data between an Identity Vault and LDAP-compliant directories. The driver can run anywhere that a Metadirectory server or Identity Manager Remote Loader is running. See Section 1.2.1, "Local and Remote Platforms," on page 12.

The driver uses the Lightweight Directory Access Protocol to bidirectionally synchronize changes between an Identity Vault and the connected LDAP-compliant directory.

Because of this flexible model for communicating, the driver can synchronize with LDAP-compliant directories running on platforms that are not supported by an Identity Vault, such as HP-UX, OS/400, and OS/390.

### 1.1.2 Publication Methods

The driver can use either of two publication methods to recognize data changes and communicate them to an Identity Vault through Identity Manager:

- "Changelog Method" on page 9
- "LDAP-Search Method" on page 10

#### Changelog Method

This method is preferred when a changelog is available. Changelogs are found on the following:

- Critical Path InJoin Directory
- IBM SecureWay Directory
- IBM Tivoli Directory

- iPlanet Directory Server
- Isode M-Vault
- Netscape Directory Server
- Oracle Internet Directory
- Sun Java System Directory

### LDAP-Search Method

Some servers don't use the changelog mechanism. The LDAP-search method enables the LDAP driver to publish data about the LDAP server to an Identity Vault by searching for changes in predefined contexts in the LDAP directory.

The LDAP-search method synchronizes changes that occur from one poll to the next.

## 1.1.3 How the Driver Works

Channels, filters, and policies control data flow.

### Publisher and Subscriber Channels

The LDAP driver supports Publisher and Subscriber channels:

- The Publisher channel reads information from the target directory changelog or an LDAP search and submits that information to an Identity Vault via the Metadirectory engine.

  By default, the Publisher channel checks the log every 60 seconds, processing up to 1000 entries at a time, starting with the first unprocessed entry.

- The Subscriber channel watches for additions and modifications to Identity Vault objects and issues LDAP commands that make changes to the target directory.

### Filters

Identity Manager uses filters to control which objects and attributes are shared. The default filter configurations for the LDAP driver allow objects and attributes to be shared, as illustrated in the following figure:

**Figure 1-1** *LDAP Driver Filters*
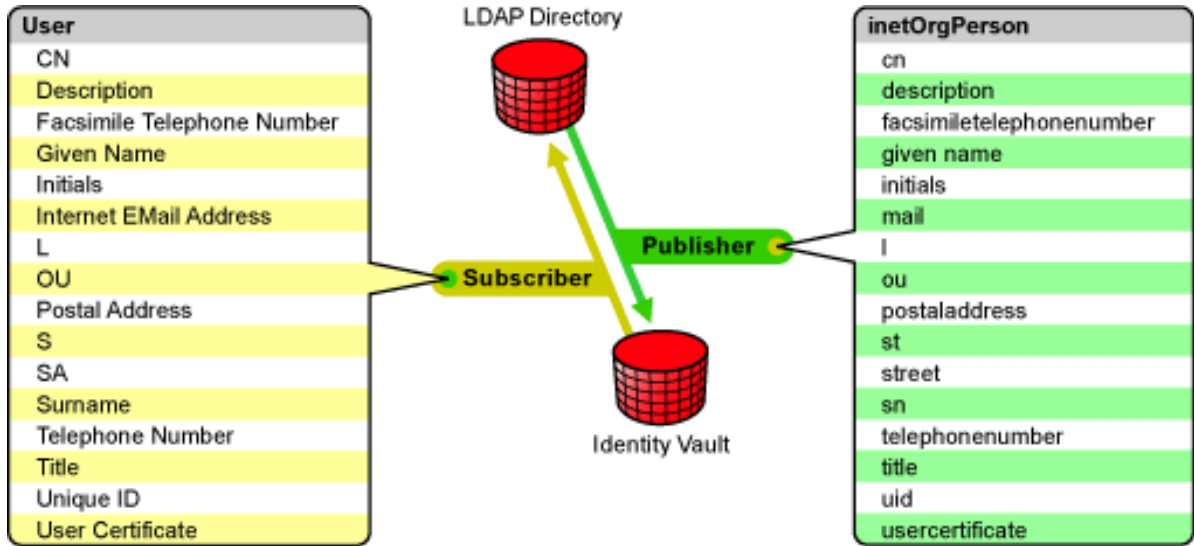


## Policies

Policies are used to control data synchronization between the driver and an Identity Vault.

The following table provides information on default policies. These policies and the individual rules they contain can be customized as explained in Chapter 6, "Synchronizing Data," on page 35.

**Table 1-1** *Default Policies*

| Policy | Description |
| --- | --- |
| Schema Mapping | Maps the Identity Vault User object and selected properties to an LDAP inetOrgPerson. |
| | Maps the Identity Vault Organizational Unit to an LDAP organizationalUnit. |
| | By default, more than a dozen standard properties are mapped. |
| Publisher Create | Specifies that in order for a User to be created in an Identity Vault, the cn, sn, and mail attributes must be defined. In order for an Organizational Unit to be created, the OU attribute must be defined. |
| Publisher Placement | Specifies that new User objects created in the LDAP directory are placed under a specified Identity Vault container in the same structure that mirrors the object's LDAP container structure. In other words, an Identity Vault container (defined during creation of the driver) becomes the root container in which the LDAP objects are mirrored exactly as they exist in the target directory. |
| Matching | Specifies that a user object in an Identity Vault is the same object as an inetOrgPerson in the target directory when the e-mail attributes match. |
| Subscriber Create | Specifies that in order for a user to be created in the target directory, the CN, Surname, and Internet Email Address attributes must be defined. In order for an Organizational Unit to be created, the OU attribute must be defined. |

| Policy | Description |
| --- | --- |
| Subscriber Placement | Specifies that new User objects created in the Identity Vault are placed under a specified LDAP container in the same structure that mirrors the object's Identity Vault container structure. In other words, an LDAP container (defined during creation of the driver) becomes the root container in which the Identity Vault objects are mirrored exactly as they exist in the Identity Vault. |

## 1.2 Key Driver Features

### 1.2.1 Local and Remote Platforms

You can install the LDAP driver locally or remotely.

An installation on the same computer where an Identity Vault and the Metadirectory engine are installed is referred to as a local configuration. The following figure illustrates a local configuration:

**Figure 1-2**  *A Local Configuration*
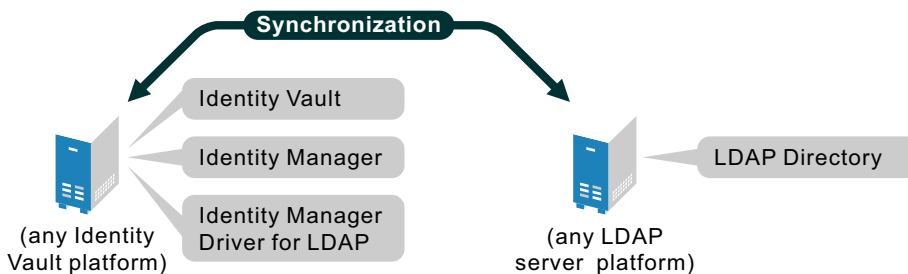


If platform or policy constraints make a local configuration difficult, you can install the LDAP driver on the server hosting the target LDAP server. This installation is referred to as a remote configuration and requires the use of the Remote Loader service.

Although a remote configuration is possible, it provides little additional flexibility because of the following:

- The driver can run on any Identity Vault platform.
- The driver communicates with the LDAP server on any platform across the wire via the LDAP protocol.

See "System Requirements" in the *Identity Manager 4.0.2 Integrated Installation Guide* for information about the supported platforms for the Metadirectory server and Remote Loader.

### 1.2.2 Entitlements

The LDAP driver supports entitlements. However, an action such as provisioning an account in the target directory is delayed until the proper approvals have been made. In Role-Based Services, rights assignments are made based on attributes of a user object. Entitlements standardize a method of

recording this information on objects in the Identity Vault. From the driver perspective, an entitlement grants or revokes the user's right to objects in the target directory. You can use entitlements to grant the right to an account in the target directory and to control group membership. The User Application either grants or revokes entitlements for a user based upon administrator defined rules.

The LDAP driver uses Permission Collection and Reconciliation service to map entitlements to resources and automatically assign those entitlements to users when permissions change in the target directory. The driver updates the Resource Catalog so that it reflects the exact state of user permissions in the target directory. The driver content includes an enhanced entitlement package that contains the following entitlements by default:

- **Account:** This entitlement grants or denies an account in the target directory for the specified user. When this entitlement is granted, the driver provides an enabled logon account. When this entitlement is revoked, the driver either disables or deletes the logon account, depending on the driver configuration.

- **Group:** This entitlement grants or denies membership to a group in the target directory. When the entitlement is revoked, Identity Manager removes the user from the group.

The LDAP driver also supports custom entitlements other than the default set provided, creating and automatically managing the relationship of identities to resource assignments. The driver uses a CSV file to map the target directory permissions into corresponding resources in the Resource Catalog. If an administrator then assigns a resource to a user in the User Application or in iManager, that change is reflected in the target directory and similarly, if the target directory administrator makes a change to the user permission, that change is reflected in the Identity Vault and the corresponding resource is updated with the permission assignment.

The following packages contain the content necessary for collecting and reconciling permission assignments in the target directory:

- NOVLACOMSET 2.0.0 (Common Settings Advanced Edition)
- NOVLLDAPENT 2.2.0 (LDAP Entitlements Support)
- NOVLLDAPDCFG 2.1.0 (LDAP Default Configuration)

If you want the driver to support custom entitlement and use Permission Collection and Reconciliation service, ensure that these packages are installed on the driver. You can turn this functionality on or off using the new set of GCVs included with the driver.

You should enable entitlements for the driver only if you plan to use the User Application or Role-Based Entitlements with the driver. For more information about entitlements, see the *Identity Manager 4.0.2 Entitlements Guide*.

## Prerequisites

Before continuing, ensure that you go through the prerequisites needed for enabling this functionality. For general prerequisites, see "Prerequisites" in "Understanding Permission Collection and Reconciliation Service " in the *NetIQ Identity Manager 4.0.2 Common Driver Administration Guide*. In addition to the general prerequisites, ensure that the LDAP driver version is 4.0.0.1.

Also, you need to set up administrative user accounts and configure a password policy for them. For more information, see "Setting Up Administrative User Accounts" and "Setting Up Administrative Passwords" in the *NetIQ Identity Manager 4.0.2 Common Driver Administration Guide*.

To use the new functionality included in the LDAP driver, you can either create a new driver with the latest packages or upgrade packages on an existing driver. For more information about creating a driver, see Section 4.1, "Creating the Driver Object in Designer," on page 21 or Section 4.3, "Adding Packages to an Existing Driver," on page 29.

### CSV File Format

The LDAP driver can consume the entitlement information from the CSV file, which is present on the server where Identity Manager is installed.The CSV file must contain values of the target system permission information in the format specified below. The target directory administrator should maintain a separate CSV file for every custom entitlement. For example, a CSV file can contain details about issuing parking passes to the employees for the *ParkingPass* entitlement. A CSV file that holds *ParkingPass* entitlement details represents this information in the following format:

```
North, North Lot, North Parking Lot
```

where *North* is the entitlement value, *North Lot* is the display name in the User Application for the entitlement value *North*, and *North Parking Lot* is the description of the entitlement value, which is displayed in the User Application.

## 1.2.3 Password Synchronization

The LDAP driver supports password synchronization on the Subscriber channel, meaning that you can send passwords from the Identity Vault to any connected target directory.

Password synchronization on the Publisher channel (LDAP directory to Identity Vault) is supported only with Sun Java System Directory version 5.2. See Section 2.2, "Installing the Identity Manager Plug-Ins for Password Synchronization with Sun Java System Directory," on page 15.

# 2 Installing the Driver Files

By default, the LDAP driver files are installed on the Metadirectory server at the same time as the Metadirectory engine. The installation program extends the Identity Vault's schema and installs the driver shim and the driver configuration file. It does not create the driver in the Identity Vault (see Chapter 4, "Creating a New Driver Object," on page 21) or upgrade an existing driver's configuration (see Chapter 5, "Upgrading an Existing Driver," on page 33).

The following sections explain what to do if the LDAP driver files are not on the server you want and how to install them:

- Section 2.1, "Installing the Driver Files," on page 15
- Section 2.2, "Installing the Identity Manager Plug-Ins for Password Synchronization with Sun Java System Directory," on page 15

## 2.1 Installing the Driver Files

If you performed a custom installation and did not install the LDAP driver on the Metadirectory server, you have two options:

- Install the files on the Metadirectory server, using the instructions in "Installing Identity Manager" in the *Identity Manager 4.0.2 Integrated Installation Guide*.
- Install the Remote Loader (required to run the driver on a non-Metadirectory server) and the driver files on a non-Metadirectory server where you want to run the driver. See "Installing Identity Manager" in the *Identity Manager 4.0.2 Integrated Installation Guide*.

## 2.2 Installing the Identity Manager Plug-Ins for Password Synchronization with Sun Java System Directory

The LDAP driver supports password synchronization on the Subscriber channel, meaning that you can send passwords from the Identity Vault to any connected LDAP directory.

Password synchronization on the Publisher channel (LDAP directory to Identity Vault) is supported only with Sun Java System Directory version 5.2 and Sun Java System Directory Server Enterprise Edition version 6.3.x. It requires you to install an Identity Manager plug-in to the Sun Java System Directory.

The following sections provide information to help you use the plug-in:

- Section 2.2.1, "How the Plug-In Works," on page 16
- Section 2.2.2, "Where to Find the Plug-In," on page 16
- Section 2.2.3, "Installing the Plug-In," on page 16

### 2.2.1 How the Plug-In Works

The plug-in is a post-operation plug-in. Sun Java System Directory notifies the plug-in whenever a password is set or changed. The plug-in then encrypts the password by using the Advanced Encryption Standard (AES) and stores the encrypted password on the novellDistPassword attribute. The LDAP driver can then synchronize the encrypted password to NetIQ Identity Manager. The LDAP driver decrypts the password and uses it to set the Identity Manager distribution password.

**IMPORTANT:** Only passwords that are set or modified after the plug-in is installed can be synchronized.

### 2.2.2 Where to Find the Plug-In

The plug-in is located on the Identity Manager 4.0.2 DVD for the Windows, Linux, and Solaris platforms.

***Table 2-1***   *Plug-In Location*

| DVD | Location | Filename |
| --- | --- | --- |
| Identity Manager 4.0.2 DVD - Windows | `IDM4.0.2_Win\products\IDM\wind ows\setup\utilities\sun_passwo rd_plugins\`*`platform`* | `novl-idm-pswd.dll` |
| Identity Manager 4.0.2 DVD - Linux | `IDM4.0.2_Lin/products/IDM/ linux/setup/utilities/ sun_password_plugins/`*`platform`* | `novl-idm-pswd.so` |
| Identity Manager 4.0.2 DVD - Solaris | `IDM4.0.2_Solaris/products/IDM/ solaris/setup/utilities/ sun_password_plugins/`*`platform`* | `novl-idm-pswd.so` |

### 2.2.3 Installing the Plug-In

1 Locate the correct plug-in file. See Section 2.2.2, "Where to Find the Plug-In," on page 16 for information.

2 Copy the binary plug-in file to the `lib` directory in your Sun Java System Directory installation location.

For example, on Windows the default installation location for Sun Java System Directory is `C:\Program Files\Sun\MPS` and inside that directory is a `lib` directory. Place `novl-idm-pswd.dll` in the `lib` directory.

On other platforms, the default installation location is often `/var/Sun/mps`. You need to locate the Sun Java System Directory installation location on your system, and put the plug-in file inside the `lib` directory.

On Solaris SPARC computers, the Sun Java System Directory installation includes two versions of most libraries: a 32-bit version and a 64-bit version. By default, the 32-bit version is found at `/var/Sun/mps/lib`.  The 64-bit version is found at `/var/Sun/mps/lib/64`.

Both a 32-bit and a 64-bit version of the plug-in are provided. Copy both versions to their respective locations on your Solaris installation. At runtime, the Sun Java System Directory determines which version is the appropriate version to load.

**3** Locate and edit the `novl-idm-pswd.ldif` or `novl-idm-pswd-win32.ldif` file. The file is located in the `sun_password_plugins` directory on your DVD image.

The `.ldif` file contains plug-in configuration information that you apply to the directory. It also contains two schema definitions:

- One definition is for the novellDistPassword attribute that stores the encrypted password.
- The other definition is for the novellDistPasswordUser auxiliary class that is applied to your users to allow the use of the novellDistPassword attribute.

As a convenience, the `.ldif` file also contains an instruction to turn on the Retro Changelog Plugin, which most customers want turned on to enable Publisher channel operations with the Identity Manager LDAP driver. If you know that the changelog is already enabled, or if you don't want to enable the changelog, you can remove the Retro Changelog Plugin section from the `.ldif` file.

Most users need to edit only two items in the `.ldif` file:

- The nsslapd-pluginPath attribute
- The nsslapd-pluginarg0 attribute

Ensure that the value of nsslapd-pluginPath is the path where you installed the plug-in. For example, if you installed the plug-in in the `/var/Sun/mps/lib` directory, the value should be `/var/Sun/mps/lib/novl-idm-pswd.so`. Set the value of nsslapd-pluginarg0 to a password that will be used to generate an AES key used to encrypt user passwords. When you create the LDAP driver, you will configure the driver with this same encryption password.

Solaris users should set the value of nsslapd-pluginPath to the path of the 32-bit version of the plug-in, even if the operating system is 64-bit. (See Step 2.) At runtime, the directory determines whether to load the 32-bit or the 64-bit version of the plug-in.

**4** Apply the `novl-idm-pswd.ldif` or `novl-idm-pswd-win32.ldif` file to the Sun directory.

To complete this step, you need to know the configuration administrator's DN and password. Typically, the DN will be `"uid=admin,ou=Administrators,ou=TopologyManagement,o=NetscapeRoot"`. However, the password will vary. You also need to know the LDAP port used by your Sun directory.

The `ldapmodify` command line utility that was installed with your Sun Java System Directories can be used to apply the `.ldif` file. Use a command similar to the following:

```
ldapmodify -h localhost -p 389 -D
"uid=admin,ou=Administrators,ou=TopologyManagement,o=NetscapeRoot" -w password
-f novl-idm-pswd.ldif
```

**5** Restart Sun Java System Directory so that your changes take affect and the plug-in starts.

For troubleshooting, note any errors that might appear on the console.

# 3 Preparing the LDAP Server

You need to do the following to prepare the LDAP server to which you are connecting:

- Create a user account through which the LDAP driver can authenticate to the LDAP directory.
- If you plan to use changelog as the publication method (see Section 1.1.2, "Publication Methods," on page 9), you need to verify that the changelog mechanism of the LDAP server is enabled. This is required only for synchronization of data from the LDAP server to the Identity Vault (Publisher channel). It is not required if you plan only to push data from the Identity Vault to the LDAP directory.

The following sections provide instructions

- Section 3.1, "Creating an LDAP User Object with Authentication Rights," on page 19
- Section 3.2, "Enabling the Changelog," on page 20

## 3.1 Creating an LDAP User Object with Authentication Rights

When you use the changelog publication method, the driver attempts to prevent loopback situations where an event that occurs on the Subscriber channel is sent back to the Metadirectory engine on the Publisher channel. However, the LDAP-search publication method relies on the Metadirectory engine to prevent loopback.

With the changelog method, one way that the driver prevents loopback from happening is to look in the changelog to see which user made the change. If the user that made the change is the same user that the driver uses to authenticate with, the Publisher assumes that the change was made by the driver's Subscriber channel.

NOTE: If you use Critical Path InJoin Server, the changelog implementation on that server is somewhat limited because it doesn't provide the DN of the object that initiated the change. Therefore, the creator/modifier DN can't be used to determine whether the change came from an Identity Vault or not.

In that case, all changes found in the changelog are sent by the Publisher to the Metadirectory engine, and the Optimize/Modify discards unnecessary or repetitive changes.

To stop the Publisher channel from discarding legitimate changes, make sure the User object that the driver uses to authenticate with is not used for any other purpose.

For example, suppose you are using the Netscape Directory Server and have configured the driver to use the administrator account CN=Directory Manager. If you want to manually make a change in the Netscape Directory Server and have that change synchronize, you can't log in and make the change with CN=Directory Manager. You must use another account.

To avoid this problem:

1 Create a user account that the driver uses exclusively.

2 Assign that user account rights to see the changelog and to make any changes that you want the driver to be able to make.

For example, at the VMP company, you create a user account for the driver called uid=ldriver,ou=Directory Administrators,o=lansing.vmp.com. You then assign the appropriate rights to the user account by applying the following LDIF to the server by using the LDAPModify tool or the NetIQ Import Conversion Export utility.

```
# give the new user rights to read and search the changelog
dn: cn=changelog
changetype: modify
add: aci
aci: (targetattr = "*")(version 3.0; acl "LDAP DirXML Driver"; allow
(compare,read,search) userdn = "ldap:///uid=ldriver,ou=Directory
Administrators,o=lansing.vmp.com"; )
-
# give the new user rights to change anything in the o=lansing.vmp.com
container
dn: o=lansing.vmp.com
changetype: modify
add: aci
aci: (targetattr = "*")(version 3.0; acl "LDAP DirXML Driver"; allow (all)
userdn = "ldap:///uid=ldriver,ou=Directory
Administrators,o=lansing.vmp.com"; )
```

## 3.2 Enabling the Changelog

The changelog is the part of the LDAP server that enables the driver to recognize changes that require publication from the LDAP directory to an Identity Vault. The LDAP directories supported by this driver support the changelog mechanism.

Critical Path InJoin and Oracle Internet Directory have the changelog enabled by default. Unless the changelog has been turned off, you don't need to perform any additional steps to enable it.

IBM SecureWay, Netscape Directory Server, and iPlanet Directory Server require you to enable the changelog after installation. For information on enabling the changelog, refer to the documentation supporting your LDAP directory.

**TIP:** The iPlanet changelog requires you to enable the Retro Changelog Plug-in. See for instructions.

# 4 Creating a New Driver Object

After the LDAP driver files are installed on the server where you want to run the driver (see Chapter 2, "Installing the Driver Files," on page 15), you can create the driver in the Identity Vault.

- Section 4.1, "Creating the Driver Object in Designer," on page 21
- Section 4.2, "Activating the Driver," on page 29
- Section 4.3, "Adding Packages to an Existing Driver," on page 29
- Section 4.4, "Viewing Permission Collection and Reconciliation Service Configuration Objects," on page 30

## 4.1 Creating the Driver Object in Designer

You create the LDAP driver object by installing the driver packages and then modifying the configuration to suit your environment. After you create and configure the driver, you need to deploy it to the Identity Vault and start it.

- Section 4.1.1, "Importing the Current Driver Packages," on page 21
- Section 4.1.2, "Installing the Driver Packages," on page 22
- Section 4.1.3, "Configuring the Driver Object," on page 26
- Section 4.1.4, "Deploying the Driver Object," on page 28
- Section 4.1.5, "Starting the Driver," on page 28

### 4.1.1 Importing the Current Driver Packages

The driver packages contain the items required to create a driver, such as policies, entitlements, filters, and Schema Mapping policies. These packages are only available in Designer and can be updated after they are initially installed. You must have the most current version of the packages in the Package Catalog before you can create a new driver object.

To verify that you have the most recent version of the driver packages in the Package Catalog:

1 Open Designer.

2 In the toolbar, click *Help > Check for Package Updates*.

3 Click *OK* if there are no package updates

or

Click *OK* to import the package updates. If prompted to restart Designer, click *Yes* and save your project, then wait until Designer restarts.

4 In the Outline view, right-click the Package Catalog.

5 Click *Import Package*.

**6** Select the LDAP driver packages

or

Click *Select All* to import all of the packages displayed, then click *OK*.

By default, only the base packages are displayed. Deselect *Show Base Packages Only* to display all packages.

---

**NOTE:** If you want the driver to support the Permission Collection and Reconciliation service, ensure you import the following packages to the driver:

- NOVLACOMSET 2.0.0 (Common Settings Advanced Edition)
- NOVLLDAPENT 2.2.0 (LDAP Entitlements Support)
- NOVLLDAPDCFG 2.1.0 (LDAP Default Configuration)

For information about the Permission Collection and Reconciliation service, see "Understanding Permission Collection and Reconciliation Service " in the *NetIQ Identity Manager 4.0.2 Common Driver Administration Guide*.

---

**7** Click *OK* to import the selected packages, then click *OK* in the successfully imported packages message.

**8** After the current packages are imported, continue with Section 4.1.2, "Installing the Driver Packages," on page 22.

## 4.1.2 Installing the Driver Packages

To install the driver packages:

**1** In Designer, open your project.

**2** In the Modeler, right-click the driver set where you want to create the driver, then click *New > Driver*.

**3** Select *LDAP Base* from the list of base packages, then click *Next*.

**4** Select the optional features to install for the LDAP driver, then click *Next*.

All options are selected by default. The options are:

**Default Configuration:** This package contains the default configuration information for the LDAP driver. Always leave this option selected.

**Entitlements:** This package contains configuration information for synchronizing accounts and policies that enable account creation and auditing for the LDAP driver. If you want account creation and auditing enabled, verify that this option is selected.

The entitlement package also contains policies that allow Identity Manager to consume permission information from the target LDAP server, and dynamically create an entitlement and dynamic resource for each permission type, and load the permission data as entitlement values into Identity Manager Role-Based Provisioning Module. This package contains GCVs to control the resource mapping. Select this package if you want to enable the permission reconciliation feature for this driver. For more information, see "Understanding Permission Collection and Reconciliation Service " in the *NetIQ Identity Manager 4.0.2 Common Driver Administration Guide*.

For general information about entitlements, see the *Identity Manager 4.0.2 Entitlements Guide*.

**Password Synchronization:** This packages contains the policies that enable the LDAP driver to synchronize passwords. If you want to synchronize passwords, verify that this option is selected. For more information, see the *Identity Manager 4.0.2 Password Management Guide*.

**Data Collection:** These packages contain the policies that enable the driver to collect data for reports. If you are using the Identity Reporting Module, verify that this option is selected. For more information, see the *Using Identity Manager 4.0.2 ReportsIdentity Reporting Module Guide*.

**Account Tracking:** These packages contain the policies that enables account tracking information for reports. If you are using the Identity Reporting Module, verify that this option is selected. For more information, see the *Identity Reporting Module Guide*.

**5** Click *Next*.

**6** (Conditional) If there are package dependencies for the packages you selected to install, you must install them to install the selected package. Click *OK* to install the package dependencies listed.

**7** (Conditional) If more than one type of package dependency must be installed, you are presented with these packages separately. Continue to click *OK* to install any additional package dependencies.

**8** (Conditional) If not already configured, fill in the following fields on the Common Settings page, then click *Next*:

**User Container:** Select the Identity Vault container where the LDAP accounts will be added if they don't already exist in the vault. This value becomes the default for all drivers in the driver set.

**Group Container:** Select the Identity Vault container where the LDAP accounts will be added if they don't already exist in the vault. This value becomes the default for all drivers in the driver set.

---

**NOTE:** This page is displayed only if the Common Settings package is installed as a dependency.

---

**9** (Conditional) If not already configured, fill in the following fields on the Common Settings Advanced Edition page, then click *Next*:

   ◆ **User Application Provisioning Services URL:** Specify the User Application Identity Manager Provisioning URL.

   ◆ **User Application Provisioning Services Administrator:** Specify the DN of the User Application Administrator user. This user should have the rights for creating and assigning resources. For more information, see "Setting Up Administrative User Accounts" in the *NetIQ Identity Manager 4.0.2 Common Driver Administration Guide*.

   ---

   **NOTE:** This page is only displayed if you installed the Common Settings Advanced Edition package.

   ---

**10** On the Driver Information page, specify a name for the driver, then click *Next*.

**11** On the Application Authentication page, fill in the following fields, then click *Next*:

**Authentication ID:** Specify the authentication ID for the driver in LDAP format.

**Connection Information:** Specify the connection information for the driver to connect to the LDAP directory.

**Password:** Specify the password for the authentication ID.

**12** On the Remote Loader page, fill in the following fields to configure the driver to connect using the Remote Loader, then click *Next*:

**Connect To Remote Loader:** By default, the driver is configured to connect using the Remote Loader. If you want to run the driver locally, select *No*, then click *Next*. Otherwise, fill in the remaining fields to configure the driver to connect by using the Remote Loader.

**Host Name:** Specify the IP address or DNS name of the server where the Remote Loader is installed and running.

**Port:** Specify the port number for this driver. Each driver connects to the Remote Loader on a separate port. The default value is 8090.

**Remote Loader Password:** Specify a password to control access to the Remote Loader. It must be the same password that is specified as the Remote Loader password on the Remote Loader.

**Driver Password:** Specify a password for the driver to authenticate to the Metadirectory server. It must be the same password that is specified as the Driver Object Password in the Remote Loader service.

**13** On the Account Status Support page, fill in the following fields, then click *Next*:

**LDAP Server Type:** Select the LDAP server type this driver connects to. Based on the type, the appropriate attribute is modified in the LDAP directory to disable the account. The options are:

- ◆ eDirectory
- ◆ openLDAP
- ◆ iPlanet | SunOne | OID

**nsManageDisabledRole DN:** This options is displayed only if you select *iPlanet | SunOne |OID*. This is the DN of the role used to disable users in the connected LDAP system. If a user is enabled in the Identity Vault, this role must first be removed from the inetOrgPerson object before the attribute that disables the account can be cleared. By default, this role should be name cn=nManagedDisabledRole, plus the name of your directory servers' root DN. For example cn=nManagedDisabledRole,dc=example,dc=com.

**14** (Conditional) On the Entitlements Name to CSV File Mappings page, click the *Add Name to File Mapping* ➕ icon to populate the page with the entitlement configuration options.

---

**NOTE:** This page is only displayed if you installed the Entitlements package.

---

The information that you specify in this page is used for creating the permission catalog. Fill in the following fields, then click *Next*:

- ◆ **Entitlement Name:** Specify a descriptive name for the entitlement to map it to the CSV file that contains the entitlement details from the connected LDAP directory.

    *Entitlement Name* is the name of the entitlement. This parameter corresponds to the Entitlement Assignment Attribute on the connected LDAP directory. For example, you could define an entitlement called *ParkingPass*.

    This parameter is used to create a resource in the User Application.

- ◆ **Entitlement Assignment Attribute:** Specify a descriptive name for the assignment attribute for an entitlement.

    *Entitlement Assignment Attribute* holds the entitlement values on the target directory. For example, you could have an attribute called *Parking*.

    You must add this parameter to *Field Names* in the Driver Parameters page or modify it in driver settings after creating the driver.

- ◆ **CSV File:** Specify the location of the CSV file. This file must be located on the same server where Identity Manager engine is installed. This file contains the values for the application permissions.

- ◆ **Multi-valued?:** Set the value of this parameter to *True* if you want to assign a resource with multiple entitlement values to the same user. Otherwise, set it to *False*.

**15** On the Synchronization Settings page, fill in the following fields to configure the driver's synchronization settings, then click *Next*:

**Subscriber Channel Placement Type:** Select the desired form of placement for the Subscriber channel. This option determines the Subscriber channel Placement policies.

- ◆ **mirrored:** Places objects hierarchically within the base container.
- ◆ **flat:** Places objects only in the base container.

**LDAP Directory Base Container:** Specify the container where user objects reside in the LDAP directory. If you are using a flat Placement rule, this is the container where the users are placed. If you are using a mirrored Placement rule, this is the root container. For example, ou=people,dc=example,dc=com.

**Publisher Channel Placement Type:** Select the desired form of placement for the Publisher channel. This option determines the Publisher channel Placement policies.

- ◆ **mirrored:** Places object hierarchically within the base container.
- ◆ **flat:** Places objects only in the base container.

**16** (Conditional) On the Install LDAP Managed System Information page, fill in the following fields to define your LDAP system, then click *Next*:

This page is displayed only if you selected to install the Data Collection and Account Tracking packages.

- ◆ **Name:** Specify a descriptive name for this LDAP system. The name is displayed in reports.
- ◆ **Description:** Specify a brief description for this LDAP system. The description is displayed in reports.
- ◆ **Location:** Specify the physical location of this LDAP system. The location is displayed in reports.
- ◆ **Vendor:** Specify the vendor of LDAP system. This information is displayed in reports.
- ◆ **Version:** Specify the version of this LDAP system. The version is displayed in the reports.

**NOTE:** This page is only displayed if you installed the Managed System package.

**17** (Conditional) On the Install LDAP Managed System Information page, fill in the following fields to define the classification of the LDAP system, then click *Next*:

This page is displayed only if you selected to install the Data Collection and Account Tracking packages.

**Classification:** Select the classification of the LDAP system. This information is displayed in the reports. You options are:

- ◆ Mission-Critical
- ◆ Vital
- ◆ Not-Critical
- ◆ Other

  If you select *Other*, you must specify a custom classification for the LDAP system.

**Environment:** Select the type of environment the LDAP system provides. The options are:

- ◆ Development
- ◆ Test
- ◆ Staging
- ◆ Production
- ◆ Other

  If you select *Other*, you must specify a custom classification for the LDAP system.

NOTE: This page is only displayed if you installed the Managed System package.

**18** (Conditional) On the Install LDAP Managed System Information page, fill in the following fields to define the ownership of the LDAP system, then click *Next*:

This page is displayed only if you selected to install the Data Collection and Account Tracking packages.

**Business Owner:** Select a user object in the Identity Vault that is the business owner of the LDAP system. This can only be a user object, not a role, group, or container.

**Application Owner:** Select a user object in the Identity Vault that is the application owner of the LDAP system. This can only be a user object, not a role, group, or container.

NOTE: This page is only displayed if you installed the Managed System package.

**19** (Conditional) On the Install LDAP Account Tracking page, fill in the following field, then click *Next*:

This page is displayed only if you selected to install the Account Tracking packages.

**Realm:** Specify the name of the realm, security domain, or namespace where the account name is unique.

NOTE: This page is only displayed if you installed the Account Tracking package.

**20** Review the summary of tasks that will be completed to create the driver, then click *Finish*.

The driver is now created. You can modify the configuration settings, by continuing with the next section, Section 4.1.3, "Configuring the Driver Object," on page 26. If you don't need to configure the driver, continue to Section 4.1.4, "Deploying the Driver Object," on page 28

## 4.1.3 Configuring the Driver Object

There are many settings that can help you customize and optimize the driver. The settings are divided into categories such as Driver Configuration, Engine Control Values, and Global Configuration Values (GCVs). Although it is important for you to understand all of the settings, your first priority should be to review the Driver Parameters located on the Driver Configuration page. The Driver Parameters let you configure the LDAP directory type, publication method, and other parameters associated with the Publisher channel.

To access the Driver Properties page:

**1** Open your project.

**2** In the Modeler, right-click the driver icon ▓ or the driver line, then select *Properties*.

**3** (Conditional) Click *GCVs > Entitlements* and review the following settings:

NOTE: These settings are only displayed if you installed the Entitlements package.

- **Use Entitlements to Control LDAP Accounts:** Ensure the value of this parameter is set to *true* to enable the driver to manage user account permissions using the Account entitlement. By default, the value is set to *true*.
- **Use Group Entitlement:** Ensure the value of this parameter is set to *true* to enable the driver to manage group memberships using the Group entitlement. By default, the value is set to *true*.
- **Enable Permissions Collection and Reconciliation:** Set the value of this parameter to *true* for permission reconciliation and entitlement assignment. By default, the value is set to *false*.

The following settings are only displayed if you set *Enable Permissions Collection and Reconciliation* to *true*:

- **Enable Permissions Reconciliation for Account Entitlement:** Ensure the value of this parameter is set to *Yes* to enable the driver to map the target directory user accounts to users in the Identity Vault and assign user account entitlements through the Publisher channel. By default the value is set to *Yes*.

- **Allow User add via publisher channel:** Set the value of this parameter to *Yes* to allow the driver to add new user accounts to the Identity Vault through the Publisher channel. By default, the value is set to *No*.

- **Enable Permissions Reconciliation for Group entitlement:** Ensure the value of this parameter is set to *Yes* to enable the driver to assign group entitlements through the Publisher channel. By default, the value is set to *Yes*.

  **IMPORTANT:** If you set the value of this parameter to *No*, the user and group resource is not created in the User Application.

- **Enable Permission Reconciliation for all Custom entitlements:** If the value of this parameter is set to *No*, this parameter allows you to select custom entitlements for permission reconciliation. By default, the value is set to *Yes*, which allows permission reconciliation of all custom entitlements.

- **Select Permissions to be reconciled:** This parameter is presented if the value of *Enable Permission Reconciliation for all Custom entitlements* is set to *No*.

  Click the *Add* ✚ icon to add custom entitlements you want to selectively onboard, specifying an Assignment Attribute Name for each entitlement.

4 Click *Apply*.

5 Modify any other settings as necessary.

You should review the set of default policies and rules provided by the basic driver configuration. Although these policies and rules are suitable for synchronizing with LDAP directory, your synchronization requirements for the driver might differ from the default policies. If this is the case, you need to change them to carry out the policies you want. The default policies and rules are discussed in Section 1.1.3, "How the Driver Works," on page 10.

- **Configure the driver filter:** Modify the driver filter to include the object classes and attributes you want synchronized between the Identity Vault and the LDAP directory. For instructions, see Chapter 6, "Synchronizing Data," on page 35.

- **Configure policies:** Modify the policies as needed. For information about the default configuration policies, see "Policies" on page 11.

- **Secure the driver connection:** Using a secure connection (SSL) between the Identity Vault and the LDAP directory is strongly recommended. To set up this secure connection, see Chapter 7, "Configuring SSL Connections," on page 41.

- **Configure password synchronization:** The basic driver configuration is set up to support password synchronization through Universal Password. If you don't want this setup, see "Configuring Password Flow" in the *Identity Manager 4.0.2 Password Management Guide*.

6 Click *OK* when finished.

After completing the configuration tasks, continue with the next section, Deploying the Driver Object.

## 4.1.4 Deploying the Driver Object

After the driver object is created in Designer, it must be deployed into the Identity Vault.

**1** In Designer, open your project.

**2** In the Modeler, right-click the driver icon ⬛ or the driver line, then select *Live > Deploy*.

**3** If you are authenticated to the Identity Vault, skip to Step 5; otherwise, specify the following information:

**Host:** Specify the IP address or DNS name of the server hosting the Identity Vault.

**Username:** Specify the DN of the user object used to authenticate to the Identity Vault.

**Password:** Specify the user's password.

**4** Click *OK*.

**5** Read through the deployment summary, then click *Deploy*.

**6** Read the successful message, then click *OK*.

**7** Click *Define Security Equivalence* to assign rights to the driver.

The driver requires rights to objects within the Identity Vault. The Admin user object is most often used to supply these rights. However, you might want to create a *DriversUser* (for example) and assign security equivalence to that user.

   **7a** Click *Add*, then browse to and select the object with the correct rights.

   **7b** Click *OK* twice.

      For more information about defining a Security Equivalent User in objects for drivers in the Identity Vault, see "Establishing a Security Equivalent User" in the Identity Manager 4.0.2 Security Guide.

**8** Click *Exclude Administrative Roles* to exclude users that should not be synchronized.

You should exclude any administrative User objects (for example, Admin and DriversUser) from synchronization.

   **8a** Click *Add*, then browse to and select the user object you want to exclude, then click *OK*.

   **8b** Repeat Step 8a for each object you want to exclude.

   **8c** Click *OK*.

**9** Click *OK*.

## 4.1.5 Starting the Driver

When a driver is created, it is stopped by default. To make the driver work, you must start the driver and cause events to occur. Identity Manager is an event-driven system, so after the driver is started, it won't do anything until an event occurs.

To start the driver:

**1** In Designer, open your project.

**2** In the Modeler, right-click the driver icon ⬛ or the driver line, then select *Live > Start Driver*.

For information about management tasks for the driver, see Chapter 8, "Managing the Driver," on page 45.

## 4.2 Activating the Driver

If you created the driver in a driver set where you have already activated the Metadirectory engine and service drivers, the driver inherits the activation. If you created the driver in a driver set that has not been activated, you must activate the driver within 90 days. Otherwise, the driver stops working.

If driver activation has expired, the following error message is displayed in the ndstrace window:

```
DirXML Log Event -------------------
Driver: \META-RHEL6\system\DriverSet\eDirDriver-BulkOperations
Channel: Subscriber
Status: Error
Message: Code(-9075) Shutting down because DirXML engine evaluation period
has expired. Activation is required for further use.
```

To use the driver, you must reactivate it.

For information on activation, refer to "Activating Novell Identity Manager Products" in the *Identity Manager 4.0.2 Integrated Installation Guide*.

## 4.3 Adding Packages to an Existing Driver

You can add new functionality to an existing driver by adding new packages to it.

1 Right-click the driver, then click *Properties*.

2 Click *Packages*, then upgrade the already installed LDAP Base package.

   2a Select the package from the list of packages, then click the *Select Operation* cell.

   2b Click *Upgrade* from the drop-down list, then click *Apply*.

   2c Click *OK* to close the Package Management page.

   You can upgrade the Password Synchronization package in a similar way.

3 Click the *Add Packages* icon 🟢.

4 Select the packages to install.

---

**NOTE:** The LDAP Entitlements package contains the content for Permission Collection and Reconciliation service. Select it to enable this service.

---

5 (Optional) If you want to see all available packages for the driver, clear the *Show only applicable package versions* option, if you want to see all available packages for the driver, then click *OK*.

   This option is only displayed on drivers. By default, only the packages that can be installed on the selected driver are displayed.

6 Click *Apply* to install all of the packages listed with the Install operation.

7 (Conditional) Fill in the fields with appropriate information to install the package you selected for the driver, then click *Next*.

8 Read the summary of the installation, then click *Finish*.

9 Click *OK* to close the Package Management page after you have reviewed the installed packages.

10 Modify the driver configuration settings. See Section 4.1.3, "Configuring the Driver Object," on page 26.

11 Deploy the driver. See Section 4.1.4, "Deploying the Driver Object," on page 28.

12 Start the driver. See Section 4.1.5, "Starting the Driver," on page 28.

**13** (Conditional) Review the newly created or modified configuration objects. See Section 4.4, "Viewing Permission Collection and Reconciliation Service Configuration Objects," on page 30.

**14** Repeat Step 1 through Step 9 for each driver where you want to add the new packages.

## 4.4 Viewing Permission Collection and Reconciliation Service Configuration Objects

**NOTE:** This section contains information about verifying the objects that are either newly created or modified as part of enabling the Permission Collection and Reconciliation service. If this service is not enabled for the driver, skip this section.

After the driver is deployed and configured with the new Permission Collection and Reconciliation service, verify that the driver correctly creates and updates the entitlements information in the Identity Vault.

Complete the following steps:

**1** In iManager, click ⓦ to display the Identity Manager Administration page.

**2** In the *Administration* list, click *Identity Manager Overview*.

  **2a** (Conditional) If the driver set is not listed on the *Driver Sets* tab, use the *Search In* field to search for and display the driver set.

  **2b** Click the driver set to open the Driver Set Overview page.

**3** Click the LDAP driver icon.

**4** Click the *Jobs* tab. The *PermissionOnboarding* job is displayed in the Jobs page. For more information, see "PermissionOnboarding Job" in the *NetIQ Identity Manager 4.0.2 Common Driver Administration Guide*.

**5** Click *Advanced > Mapping Tables*. The DNs of the Entitlement objects are displayed in the Mapping Table page based on the InitEntitlementResourceObjects policy and data from the configuration objects. For more information, see "Mapping Tables" in the *NetIQ Identity Manager 4.0.2 Common Driver Administration Guide*.

**6** In iManager, click *Driver Set > Edit Driver Set properties*.

**7** Click *Global Config Values* to display the driver set GCV page.

This page contains two sets of GCVs that are consumed by the drivers in the driver set. Ensure that you configure them for the driver set containing the drivers for reconciliation of identity, resources, and permission assignments.

  ◆ **NOVLCOMSET:** This GCV object contains the following:

    ◆ **User Container:** Specifies the Identity Vault container where the users are added, if they don't already exist in the Identity Vault. This value is the default value for all drivers in the driver set.

    ◆ **Group Container:** Specifies the Identity Vault container where the groups are added, if they don't already exist in the Identity Vault. This value is the default value for all drivers in the driver set.

  ◆ **Advanced Settings:** This GCV object contains the following:

    ◆ **User Application Provisioning Services URL:** Specifies the User Application Identity Manager Provisioning URL.

- **User Application Provisioning Services Administrator:** Specifies the DN of the provisioning services administrator. This user should have the rights for creating and assigning resources.

# 5 Upgrading an Existing Driver

The following sections provide information to help you upgrade an existing driver:

## 5.1 Supported Upgrade Paths

You can upgrade from any 3.*x* version of the LDAP driver. Upgrading a pre-3.*x* version of the driver directly to version 4.0.2 is not supported.

## 5.2 What's New in Version 4.0.2

Version 4.0.2 of the driver does not include any new features.

From 4.0 version of the driver, driver content is delivered in packages instead of through a driver configuration file.

## 5.3 Upgrade Procedure

The process for upgrading the LDAP driver is the same as for other Identity Manager drivers. For detailed instructions, see "Upgrading Drivers to Packages" in the *Identity Manager 4.0.2 Upgrade and Migration Guide*.

# 6 Synchronizing Data

The following sections provide information to help you control which classes and attributes are synchronized between your Identity Vault and the connected LDAP directory. Not only can you choose which classes and attributes are synchronized, but you can also determine which direction they flow (Identity Vault to LDAP, LDAP to Identity Vault, or both).

- Section 6.1, "Determining Which Objects Are Synchronized," on page 35
- Section 6.2, "Defining Schema Mapping," on page 35
- Section 6.3, "Netscape Directory Server Configuration," on page 37
- Section 6.4, "Migrating and Resynchronizing Data," on page 38

## 6.1 Determining Which Objects Are Synchronized

Identity Manager uses the driver filter, located on both the Publisher and Subscriber channels, to control which objects are synchronized and to define the authoritative data source for these objects.

The following steps provide instructions for editing the filter in iManager. For information about editing the filter in Designer, see "Controlling the Flow of Objects with the Filter" in the *Policies in Designer 4.0.2* guide.

1 In iManager, open the LDAP driver Overview page:

    1a Click 🌐 to display the Identity Manager Administration page.

    1b In the *Administration* list, click *Identity Manager Overview*.

    1c If the driver set is not listed on the *Driver Sets* tab, use the *Search In* field to search for and display the driver set.

    1d Click the driver set to open the Driver Set Overview page.

    1e Click the LDAP driver icon to display its Overview page.

2 Click the Publisher or Subscriber filter icon and make the appropriate changes.

For every object and attribute selected in the filter, the Schema Mapping policy must have a corresponding entry unless the class or attribute names are the same in both directories (see Section 6.2, "Defining Schema Mapping," on page 35). Before mapping an attribute, verify that a corresponding attribute actually exists in the target directory.

## 6.2 Defining Schema Mapping

Different LDAP servers have different schemas. When the driver is first started, it queries the server for the specific schema.

You must be familiar with the characteristics of directory attributes and the LDAP server attributes. The driver handles all LDAP attribute types (cis, ces, tel, dn, int, bin). It also handles the eDirectory Facsimile Telephone Number.

When you map attributes, follow these guidelines:

- Verify that every class and attribute specified in the Subscriber and Publisher policies is mapped in the Mapping policy unless the class or attribute names are the same in both directories.

- Before mapping a directory attribute to an LDAP server attribute, verify that an LDAP server attribute actually exists. For example, the Full Name attribute is defined for a User object on an Identity Vault, but fullname doesn't exist in an inetOrgPerson object on Netscape.

- Always map attributes to attributes of the same type. For example, map strings attributes to strings attributes, octet attributes to binary attributes, or telenumber attributes to telenumber attributes.

- Map multivalue attributes to multivalue attributes.

The driver doesn't provide data conversion between different attribute types or conversions from multivalue to single-value attributes. The driver also doesn't understand structured attributes except for Facsimile Telephone Number and Postal Address.

Identity Manager is flexible about the syntax that it accepts from the Publisher:

- **Accepting Non-Structured/Non-Octet Syntax:** Identity Manager accepts any non-structured/ non-octet syntax for any other non-structured/non-octet syntax as long as the actual data can be coerced to the appropriate type. That is, if the Identity Vault is looking for a numeric value, the actual data should be a number.

- **Coercing the Data to Octet:** When Identity Manager is expecting octet data and gets another non-octet/non-structured type, Identity Manager coerces the data to octet by serializing the string value to UTF-8.

- **Coercing the Data to a String:** When Identity Manager is passed octet data and another non-structured type is expected, Identity Manager coerces the data to a string by decoding the Base64 data. Identity Manager next tries to interpret the result as a UTF-8 encoded string (or the platform's default character encoding if it is not a valid UTF-8 string) and then applies the same rules as Accepting Non-Structured/Non-Octet Syntax.

- **FaxNumber:** For faxNumber, if a non-structured type is passed in, Accepting Non-Structured/ Non-Octet Syntax and Coercing the Data to a String are applied to the data to get the phone number portion of the fax number. The other fields are defaulted.

- **State:** State. For state, False, No, F, N (in either uppercase or lowercase), 0 and " " (empty string) are interpreted as False, and any other value is interpreted as True.

The following steps provide instructions for modifying the Schema Mapping Policy in iManager. For information about using Designer, see "Defining Schema Map Policies" in the *Policies in Designer 4.0.2* guide.

1 In iManager, open the LDAP driver Overview page:

   1a Click 🌐 to display the Identity Manager Administration page.

   1b In the *Administration* list, click *Identity Manager Overview*.

   1c If the driver set is not listed on the *Driver Sets* tab, use the *Search In* field to search for and display the driver set.

   1d Click the driver set to open the Driver Set Overview page.

   1e Click the LDAP driver icon to display its Overview page.

2 Click the schema mapping icon on the Publisher or Subscriber channel.

3 Click the policy to display the editing page.

4 Edit the policy as appropriate for your setup.

## 6.3  Netscape Directory Server Configuration

The following sections apply only when connecting to Netscape Directory Server:

### 6.3.1  Defining Object Placement in Netscape Directory Server

We recommend following the Netscape naming rules for objects in Netscape Directory Server. A brief explanation of naming rules is included here for your convenience.

The directory contains entries that represent people. These person entries must have names. In other words, you must decide what the relative distinguished name (RDN) is for each person entry. The DN must be a unique, easily recognizable, permanent value. We recommend that you use the uid attribute to specify a unique value associated with the person. An example DN for a person entry is:

uid=jsmith,o=novell

The directory also contains entries that represent many things other than people (for example, groups, devices, servers, network information, or other data). We recommend that you use the cn attribute in the RDN. Therefore, if you are naming a group entry, name it as follows:

cn=administrators,ou=groups,o=novell

The directory also contains branch points or containers. You need to decide what attributes to use to identify the branch points. Because attribute names have a meaning, use the attribute name with the type of entry it is representing. The Netscape recommended attributes are defined as follows:

| Attribute Name | Definition |
| --- | --- |
| c | Country name |
| o | Organization name |
| ou | Organizational Unit |
| st | State |
| l | Locality |
| dc | Domain Component |

A Subscriber Placement policy specifies the naming attribute for a classname. The following example is for the User classname. The `<placement>` statement specifies that uid is used as the naming attribute.

```
<placement-rule>
   <match-class class-name="User"/>
   <match-path prefix="\Novell-Tree\Novell\Users"/>
   <placement>uid=<copy-name/>,ou=People,o=Netscape</
 placement>
</placement-rule>
```

The following Subscriber Placement specifies that ou is used as the naming attribute for class-name Organizational Unit.

```
<placement-rule>
   <match-class class-name="Organizational Unit"/>
   <match-path prefix="\Novell-Tree\Novell\Users"/>
   <placement>ou=<copy-name/>,ou=People,o=Netscape</placement>
</placement-rule>
```

To configure a placement policy:

**1** In iManager, *click Identity Manager > Identity Manager Overview*.

**2** Locate the driver in its driver set.

**3** Open the Identity Manager Driver Overview page by clicking the driver.

**4** Click the Publisher or Subscriber Placement policy icon, then make the appropriate changes.

**5** Click *Close*.

## 6.3.2 Working with eDirectory Groups and Netscape

Because group attributes are different in an Identity Vault and Netscape Directory Server, some special processing is required by the driver. On the Publisher channel, special processing takes place when the driver sees the attribute uniquemember in the classname groupofuniquenames.

The driver also sets the Equivalent To Me attribute in the eDirectory Group. The Equivalent To Me attribute must be included in the Publisher filter. The Equivalent To Me attribute does not need to be in the Schema Mapping policy because the eDirectory attribute name is used. There is no equivalent attribute name in Netscape Directory Server. No special processing is required on the Subscriber channel.

# 6.4 Migrating and Resynchronizing Data

Identity Manager synchronizes data as the data changes. If you want to synchronize all data immediately, you can choose from the following options:

- ◆ **Migrate Data from the Identity Vault:** Allows you to select containers or objects you want to migrate from an Identity Vault to an LDAP server. When you migrate an object, the Metadirectory engine applies all of the Matching, Placement, and Create policies, as well as the Subscriber filter, to the object.

   NOTE: When you migrate data from an Identity Vault into the LDAP directory, you might need to change your LDAP server settings to allow migration of large numbers of objects. See Section 9.2, "Migrating Users into an Identity Vault," on page 47.

- ◆ **Migrate Data into the Identity Vault:** Allows you to define the criteria that Identity Manager uses to migrate objects from an LDAP server into an Identity Vault. When you migrate an object, the Metadirectory engine applies all of the Matching, Placement, and Create policies, as well as the Publisher filter, to the object. Objects are migrated into the Identity Vault by using the order you specify in the Class list.

- ◆ **Synchronize:** Identity Manager looks in the Subscriber class filter and processes all objects for those classes. Associated objects are merged. Unassociated objects are processed as Add events.

**1** In iManager, open the LDAP driver Overview page:

   **1a** Click 🔵 to display the Identity Manager Administration page.

   **1b** In the *Administration* list, click *Identity Manager Overview*.

> **1c** If the driver set is not listed on the *Driver Sets* tab, use the *Search In* field to search for and display the driver set.
>
> **1d** Click the driver set to open the Driver Set Overview page.
>
> **1e** Click the LDAP driver icon to display its Overview page.

**2** Click *Migrate*, then click the appropriate migration button.

# 7 Configuring SSL Connections

The driver uses the LDAP protocol to communicate with the LDAP server. Most LDAP servers allow non-encrypted (clear-text) connections. Additionally, when configured correctly, some LDAP servers allow SSL-encrypted connections. SSL connections encrypt all traffic on the TCP/IP socket by using a public/private key pair. The actual LDAP protocol doesn't change, but the communication channel performs the encryption.

The procedure for enabling SSL connections differs slightly from one LDAP server to another. This document covers the process for enabling SSL connections when you are using Netscape Directory Server 4.12. If you are using another LDAP server, the procedure is similar.

## 7.1 Step 1: Generating a Server Certificate

You first need to install a server certificate. The LDAP server itself can generate a certificate, but the certificate must then be signed by a certificate authority (CA) that is trusted by the server. One way to get the certificate signed is to use the CA that comes with an Identity Vault.

To generate a certificate request:

**1** In the navigation tree in Netscape Console, select the server that the driver will communicate with.

**2** Click *Open Server*.

**3** Click *Tasks > Certificate Setup Wizard.*

**4** Provide information to request a certificate.

Depending on the certificates or tokens that might already be installed on the host system, you might see some or all of the following fields:

**Select a Token (Cryptographic Device):** Select *Internal (Software)*.

**Is the Server Certificate Already Requested and Ready to Install?** Select *No*.

If a trust database doesn't already exist for this host, one is generated for you.

A trust database is a key pair and certificate database installed on the local host. When you use an internal token, the trust database is the database into which you install the key and certificate.

**5** Type and confirm the password.

The password must contain at least eight characters, and at least one of them must be numeric. This password helps secure access to the new key database you're creating.

**6** Continue providing information as prompted, then click *Next*.

**7** After a trust database is created, click *Next*.

**8** Type the requested information, then click *Next*.

**9** Type the password for the token you selected earlier, then click *Next*.

The Certificate Setup Wizard generates a certificate request for your server. When you see the page, you can send the certificate request to the certification authority.

**10** Continue with Step 2: Sending the Certificate Request.

# 7.2  Step 2: Sending the Certificate Request

**1** Copy the server certificate request into Notepad or another text editor.

**2** Save the file as `csr.txt`.

Your certificate request e-mail should look like the following:

```
-----BEGIN NEW CERTIFICATE REQUEST-----

        .

        .

        .
-----END NEW CERTIFICATE REQUEST----
```

**3** In iManager, *select NetIQ Certificate Server > Issue Certificate*.

**4** In the *Filename* field, browse to `csr.txt`, then click *Next*.

**5** Select *Organizational Certificate Authority*.

**6** Specify SSL as the key type, then click *Next*.

**7** Specify the certificate parameters, click *Next*, then click *Finish*.

**8** Save the certificate in Base64 format as `cert.b64` to a local disk or an external storage device.

**9** Continue with Step 3: Installing the Certificate.

# 7.3  Step 3: Installing the Certificate

**1** In the navigation tree in Netscape Console, select the server that the driver will be connecting to.

**2** Click *Open*.

**3** Click *Tasks > Certificate Setup Wizard*.

**4** Start the wizard and indicate that you are ready to install the certificate.

**5** When prompted, provide the following information:

**Select a Token (Cryptographic Device):**  Select *Internal (Software)*.

**Is the Server Certificate Already Requested and Ready to Install?** Select *Yes*.

**6** Click *Next*.

**7** In the *Install Certificate For* field, select *This Server*.

**8** In the *Password* field, type the password you used to set up the trust database, then click *Next*.

**9** In the *Certificate Is Located in This File* field, type the absolute path to the certificate (for example, `A:\CERT.B64`).

**10** After the certificate is generated, click *Add*.

**11** After the certificate is successfully installed, click *Done*.

**12** Continue with Step 4: Activating SSL in Netscape Directory Server 4.12.

## 7.4 Step 4: Activating SSL in Netscape Directory Server 4.12

After you install the certificate, complete the following to activate SSL:

**1** In the navigation tree in Netscape Console, select the server you want to use SSL encryption with.

**2** Click *Open > Configuration > Encryption*.

**3** Specify the following information:

**Enable SSL:** Select this option.

**Cipher Family:** Select *RSA*.

**Token to Use:** Select *Internal (Software)*.

**Certificate to Use:** Select *Server-Cert*.

**Client Authentication:** Because the driver doesn't support client authentication, deselect *Allow Client Authentication*.

**4** Click *Save*.

**5** Click *Tasks*, then restart the server for the changes to take effect.

**6** Continue with Step 5: Exporting the Trusted Root from the Directory Tree.

## 7.5 Step 5: Exporting the Trusted Root from the Directory Tree

**1** In iManager, select *eDirectory Administration > Modify Object*.

**2** Browse to the Certificate Authority (CA) object, then click *OK*.

**3** Select *Certificates* from the drop-down list.

**4** Click *Export*.

**5** Click *No* at the prompt that displays *Do you want to export the private key with the certificate?"*

**6** Click *Next*.

**7** In the *Filename* field, specify a filename (for example, `PublicKeyCert`), then select *Base64* as the format.

**8** Click *Export*.

**9** Continue with Step 6: Importing the Trusted Root Certificate.

## 7.6 Step 6: Importing the Trusted Root Certificate

You need to import the trusted root certificate into the LDAP server's trust database and the client's certificate store.

- Section 7.6.1, "Importing into the LDAP Server's Trust Database," on page 44
- Section 7.6.2, "Importing into the Client's Certificate Store," on page 44

### 7.6.1 Importing into the LDAP Server's Trust Database

You need to import the trusted root certificate into the LDAP server's trust database. Because the server certificate was signed by the Identity Vault's CA, the trust database needs to be configured to trust the Identity Vault CA.

**1** In the Netscape Console, click *Tasks > Certificate Setup Wizard > Next*.

**2** In *Select a Token*, accept the default for Internal (*Software*).

**3** In *Is the Server Certificate Already Requested and Ready to Install*, select *Yes*.

**4** Click *Next* twice.

**5** In Install Certificate For dialog box, select *Trusted Certificate Authority*.

**6** Click *Next*.

**7** Select *The Certificate Is Located in This File*, then type the full path to the `.b64` file containing the trusted root certificate.

**8** Click *Next*.

**9** Verify the information on the screen, then click *Add*.

**10** Click *Done*.

**11** Continue with Importing into the Client's Certificate Store.

### 7.6.2 Importing into the Client's Certificate Store

You need to import the trusted root certificate into a certificate store (also called a keystore) that the driver can use.

**1** Use the KeyTool class found in `rt.jar`.

For example, if your public key certificate is saved as `PublicKeyCert.b64` on a your local disk and you want to import it into a new certificate store file named `.keystore` in the current directory, enter the following at the command line:

```
java sun.security.tools.KeyTool -import -alias TrustedRoot -file
A:\PublicKeyCert.b64

-keystore .keystore -storepass keystorepass
```

**2** When you are asked to trust this certificate, select *Yes*, then click *Enter*.

**3** Copy the `.keystore` file to any directory on the same file system that has the Identity Vault files.

**4** In iManager, select *Identity Manager > Identity Manager Overview*.

**5** Search for drivers.

**6** Click the LDAP Driver object, then click it again in the *Identity Manager Driver Overview* page.

**7** In the *Keystore Path* parameter, enter the complete path to the `.keystore` file.

**8** Continue with Step 7: Adjusting Driver Settings.

## 7.7 Step 7: Adjusting Driver Settings

You need to enable the driver's SSL parameter and adjust the other SSL parameters as needed. For information, see Section A.1.5, "Driver Parameters," on page 53.

# 8 Managing the Driver

As you work with the LDAP driver, there are a variety of management tasks you might need to perform, including the following:

- Starting, stopping, and restarting the driver
- Viewing driver version information
- Using Named Passwords to securely store passwords associated with the driver
- Monitoring the driver's health status
- Backing up the driver
- Inspecting the driver's cache files
- Viewing the driver's statistics
- Using the DirXML Command Line utility to perform management tasks through scripts
- Securing the driver and its information
- Synchronizing objects
- Migrating and resynchronizing data
- Activating the driver

Because these tasks, as well as several others, are common to all Identity Manager drivers, they are included in one reference, the *NetIQ Identity Manager 4.0.2 Common Driver Administration Guide*.

# 9 Troubleshooting

## 9.1 Troubleshooting Driver Processes

Viewing driver processes is necessary to analyze unexpected behavior. To view the driver processing events, use DSTrace. You should only use it during testing and troubleshooting the driver. Running DSTrace while the drivers are in production increases the utilization on the Identity Manager server and can cause events to process very slowly. For more information, see "Viewing Identity Manager Processes" in the *NetIQ Identity Manager 4.0.2 Common Driver Administration Guide*.

## 9.2 Migrating Users into an Identity Vault

Some LDAP servers have settings that limit the number of entries that an LDAP query can return. For example, iPlanet Directory Server 5.1 has a default limit of 5000 objects.

When you migrate user data from LDAP into an Identity Vault, the driver makes an LDAP query to the server and returns the objects that match the criteria (such as objectclass=User).

A limit on the number of entries that can be returned on an LDAP query can cause a migration to stop before it is complete, even though the Identity Manager driver continues to run normally.

To fix this, change the limit. For example, do the following in iPlanet:

1 Go to the *Configuration* tab, then select *Database* settings.

2 Raise the look-through limit on the LDBM plug-in tab from the default of 5000 to an appropriate number.

   This is the number of records the query is allowed to look at while fulfilling the query.

3 Go to the *Configuration* tab, select *Directory Server Settings*, select the *Performance* tab, then raise the size limit according to the number of user accounts you need to migrate.

This is the actual number of records that the query is allowed to return.

After these settings have been adjusted, the migration should finish correctly.

## 9.3 OutOfMemoryError

If you use the LDAP-Search method and the driver shuts down with a java.lang.OutOfMemoryError:

1 Try setting or increasing the DHOST_JVM_INITIAL_HEAP and *DHOST_JVM_MAX_HEAP* environment variables.

2 Restart the driver.

3 Monitor the driver to make sure that the variables provide enough memory.

For more information, see "Configuring Java Environment Parameters" in the *NetIQ Identity Manager 4.0.2 Common Driver Administration Guide*.

## 9.4 LDAP v3 Compatibility

The LDAP driver works with most LDAP v3 compatible LDAP servers. The driver is written to the LDAP specification (IETF RFCs 4510-4519). To increase compatibility with some LDAP servers that don't fully meet the RFC 2251 requirements, we have added workarounds to the LDAP driver.

One compatibility issue that cannot be ignored or worked around is the RFC 4511 section 4.1.1 requirement that servers allow Message ID values up to 2,147,483,647 (integer values using four bytes).

Oracle Internet Directory version 2.1.1.0.0 (which is part of Oracle 8i) allows only Message ID values up to 32,767 (integer values using two bytes). Therefore, it can't function properly with the LDAP driver.

If you need compatibility with Oracle Internet Directory, NetIQ recommends upgrading to version 9.2.0.1.0 (included with Oracle 9i) or later.

## 9.5 Synchronizing Data after Remote Loader Failover on Linux High Availability Cluster

To synchronize data after the Remote Loader is restarted on the Linux High Availability Cluster with LDAPsearch as the publication method, change the *Search Results to Synchronize on First Startup* option to *Synchronize Everything* to capture the events that occur after the LDAP driver starts.

If you set *Synchronize Everything* option for migrating a large number of objects, it might cause a performance issue when you start the driver for the first time after you create the driver. Set the driver option to *Synchronize only subsequent changes* when you start the driver for the first time, change it to *Synchronize Everything*, then restart the driver.

**NOTE:** If your LDAP server supports changelog, use it as a publication method instead of LDAPsearch for better performance and consistent search results.

## 9.6 Deletion of Users Is Not Synchronized Between Identity Vault and SunOne 5.2 Connected System

In SunOne 5.2, if you are using the changelog method, deletion of users fail to synchronize with the Identity Vault because the changelog generated for the delete entry has a leading space in it.

To work around this issue, either use the search method or remove the leading space using a policy.

## 9.7 Issues in Permission Reconciliation When the Driver is Configured in LDAPSearch Mode

- When a group is moved from a base container to another container, the user assignments for that group are revoked from the Resource Catalog.

  In a normal scenario, the driver checks the base container while polling for changes, and this container is configured as a driver parameter.

- When a group is added outside the base container, this change is not reflected in the group dynamic resource.

Because the driver does not capture the changes that occur outside the base container, the permission reconciliation does not work as expected.

To work around these issues, migrate these groups into the Identity Vault.

1 In iManager, open the LDAP driver Overview page:

    **1a** Click 🔵 to display the Identity Manager Administration page.

    **1b** In the *Administration* list, click *Identity Manager Overview*.

    **1c** If the driver set is not listed on the *Driver Sets* tab, use the *Search In* field to search for and display the driver set.

    **1d** Click the driver set to open the Driver Set Overview page.

    **1e** Click the LDAP driver icon to display its Overview page.

2 Click *Migrate*, select the *Migrate into Identity Vault* option and then select the group class to migrate all groups into the Identity Vault.

## 9.8 Frequently Asked Questions

**Question:** Does the LDAP-search method retrieve everything every time, or does it just retrieve updates since the last poll?

**Answer:** The LDAP-search method synchronizes updates from one poll to the next.

**Question:** If I have a choice between using the LDAP-search method or the changelog method, should I use the LDAP Search method?

**Answer:** Use the changelog method because it has performance advantages and is the recommended method.

# A    Driver Properties

This section provides information about the Driver Configuration and Global Configuration Values properties for the LDAP driver. These are the only unique properties for drivers. All other driver properties (Named Password, Engine Control Values, Log Level, and so forth) are common to all drivers. Refer to "Driver Properties" in the *NetIQ Identity Manager 4.0.2 Common Driver Administration Guide* for information about the common properties.

The information is organized according to tabs that display in iManager. If a field is different in Designer, it is marked with a Designer 🟠 icon.

## A.1    Driver Configuration

In iManager:

1 Click 🔵 to display the Identity Manager Administration page.

2 Open the driver set that contains the driver whose properties you want to edit:

  2a In the *Administration* list, click *Identity Manager Overview*.

  2b If the driver set is not listed on the *Driver Sets* tab, use the *Search In* field to search for and display the driver set.

  2c Click the driver set to open the Driver Set Overview page.

3 Locate the driver icon, then click the upper right corner of the driver icon to display the *Actions* menu.

4 Click *Edit Properties* to display the driver's properties page.

  By default, the Driver Configuration page is displayed.

In Designer:

1 Open a project in the Modeler.

2 Right-click the driver icon or line, then select click *Properties > Driver Configuration*.

The Driver Configuration options are divided into the following sections:

## A.1.1 Driver Module

The driver module changes the driver from running locally to running remotely or the reverse.

**Java:** Used to specify the name of the Java class that is instantiated for the shim component of the driver. This class can be located in the `classes` directory as a class file, or in the `lib` directory as a `.jar` file. If this option is selected, the driver is running locally.

The name of the Java class is: `com.novell.nds.dirxml.driver.ldap.LDAPDriverShim`

**Native:** This option is not used with the driver.

**Connect to Remote Loader:** Used when the driver is connecting remotely to the connected system. Designer includes two suboptions:

  ◆ **Remote Loader Configuration for Documentation:** Includes information on the Remote Loader client configuration when Designer generates documentation for the driver.
  ◆ **Driver Object Password:** Specifies a password for the Driver object. If you are using the Remote Loader, you must enter a password on this page. Otherwise, the remote driver does not run. The Remote Loader uses this password to authenticate itself to the remote driver shim.

## A.1.2 Driver Object Password

**Driver Object Password:** Use this option to set a password for the driver object. If you are using the Remote Loader, you must enter a password on this page. Otherwise, the remote driver does not run. The Remote Loader uses this password to authenticate itself to the remote driver shim.

## A.1.3 Authentication

The Authentication section stores the information required to authenticate to the connected system.

**Authentication information for server:** Displays or specifies the IP address or server name that the driver is associated with

**Authentication ID:** Specifies the DN of the LDAP account that the driver will use for authentication. For example: `Administrator`

**Authentication Context:** Specify the IP address or name of the LDAP server.

**Remote Loader Connection Parameter:** Used only if the driver is connecting to the application through the Remote Loader. The parameter to enter is `hostname=xxx.xxx.xxx.xxx port=xxxx kmo=certificatename`, when the host name is the IP address of the application server running the Remote Loader server and the port is the port the Remote Loader is listening on. The default port for the Remote Loader is 8090.

The `kmo` entry is optional. It is used only when an SSL connection exists between the Remote Loader and the Metadirectory engine.

Example: `hostname=10.0.0.1 port=8090 kmo=IDMCertificate`

**Application Password:** Specify the password for the user object listed in the *Authentication ID* field.

**Remote Loader Password:** Used only if the driver is connecting to the application through the Remote Loader. The password is used to control access to the Remote Loader instance. It must be the same password specified during the configuration of the Remote Loader on the connected system.

**Cache limit (KB):** Specify the maximum event cache file size (in KB). If it is set to zero, the file size is unlimited. Click *Unlimited* to set the file size to Unlimited in Designer.

## A.1.4    Startup Option

The Startup Option section allows you to set the driver state when the Identity Manager server is started.

**Auto start:** The driver starts every time the Identity Manager server is started.

**Manual:** The driver does not start when the Identity Manager server is started. The driver must be started through Designer or iManager.

**Disabled:** The driver has a cache file that stores all of the events. When the driver is set to *Disabled*, this file is deleted and no new events are stored in the file until the driver state is changed to *Manual* or *Auto Start*.

**Do not automatically synchronize the driver:** This option applies only if the driver is deployed and was previously disabled. If this is not selected, the driver re-synchronizes the next time it is started.

## A.1.5    Driver Parameters

The Driver Parameters section lets you configure the driver-specific parameters. When you change driver parameters, you tune driver behavior to align with your network environment. The parameters are divided into different categories:

### Driver Options

**LDAP Directory Type:** When Isode M-Vault is the target directory, set the LDAP Directory Type to *M-Vault*. Otherwise, use the *LDAPv3* setting.

**Enforce Matching Parenthesis in Schema Elements:** Select whether the driver enforces matching parenthesis in the LDAP schema objectclass and attribute type definitions. If you choose *No*, the driver ignores the parenthesis syntax infractions in the schema definitions.

**Additional Allowable Schema Name Characters:** Specify extra characters to allow in LDAP objectclass and attribute type names, even when those characters are specifically disallowed by RFC 2252. Some LDAP servers don't always follow the specifications.

**Use SSL:** Select *Yes* to use SSL to secure communication between the driver and the LDAP server. If you use SSL, fill in the following parameters:

- **Keystore Path for SSL Certs:** Specify the full path to the keystore file containing the SSL certificates
- **Use Mutual Authentication:** Select *Yes* if you want the driver to use SSL mutual authentication (both client and server), or select *No* for server authentication only. If you select *Yes*, you must have the appropriate certificates configured in your keystore.

**Key Alias:** The alias created when importing the public key certificate into the keystore. Typically, you only need to specify the alias when using mutual authentication.

**Keystore Password:** Specify the password used to access the keystore file that contains the SSL certificates.

**Maximum number of operations for a single bind:** Specify the number of LDAP operations after which the driver reconnects to the LDAP server. Change the default value to a large value if the driver does frequent binds.

## Subscriber Options

**LDAP Server Supports Binary Attribute Option:** Most LDAP servers support the use of the binary attribute option as defined in RFC 2251 section 4.1.5.1. If you don't know whether the LDAP server supports the binary attribute option, select *Yes*.

**Ignore empty components for Postal Address:** The LDAP driver uses a space " " as the value between the Postal Address separator ($) for missing values of Postal Address components. Selecting *yes* turns on this parameter discards the trailing components that have no value. The default setting is *no*.

## Publisher Options

**Polling Interval in Seconds:** Specify the interval at which the driver checks the LDAP server for changes. When new changes are found, they are applied to the Identity Vault.

**Enable Paged Search:** Select *Yes* to enable the paged search of the target directory objects. The default setting is *No*.

**Temporary File Directory:** Specify a directory on the local file system (the one where the driver is running) where temporary state files can be written. If you don't specify a path, the driver uses the default driver path:

- ◆ **Metadirectory server:** Defaults to the eDirectory DIB file directory.
- ◆ **Remote Loader server:** Defaults to the root of the Remote Loader directory.

These files help maintain driver consistency even when the driver is shut down. They also help prevent memory shortages during extensive data searches.

**Heartbeat interval in minutes:** Specify how many minutes of inactivity should elapse before this channel sends a heartbeat document. In practice, more than the number of minutes specified can elapse. That is, this parameter defines a lower bound.

**Publication Method:** Select whether you want to use LDAP Search or changelog as the publication method. The changelog method is the recommended method for LDAP directories that support it. For more information, see .

If you select *Changelog*, fill in the following fields:

**Changelog Entries to Process on Startup:** Select how the LDAP driver processes entries at startup.

- ◆ **All:** The Publisher attempts to process all of the changes found in the changelog. The Publisher continues until all changes have been processed. It processes new changes according to the poll rate.
- ◆ **None:** When the driver starts running, the Publisher doesn't process any previously existing entries. It processes new changes according to the poll rate.

- **Previously Unprocessed:** This setting is the default. If this is the first time the driver has been run, it behaves like the *All* option, processing all new changes.

   If the driver has been run before, this setting causes the Publisher to process only changes that are new since the last time the driver was running. Thereafter, it processes new changes according to the poll rate.

**Maximum Batch Size for Changelog Processing:** When the Publisher channel processes new entries from the LDAP changelog, the Publisher asks for the entries in batches of this size (the default is 1000). If there are fewer than this number of changelog entries, all of them are processed immediately. If there are more than this number, they are processed in consecutive batches of this size.

**Preferred LDAP ObjectClass Names:** Identity Manager requires that objects be identified by using a single object class. However, many LDAP servers and applications can list multiple object classes for a single object. By default, when the LDAP driver finds an object on the LDAP server or application that has been added, deleted, or modified, it sends the event to the Metadirectory engine and identifies it by using the object class that has the most levels of inheritance in the schema definition.

For example, a user object in LDAP is identified with the object classes of inetorgperson, organizationalperson, person, and top. Inetorgperson has the most levels of inheritance in the schema (inheriting from organizationalperson, which inherits from person, which inherits from top). By default, the driver uses inetorgperson as the object class it reports to the Metadirectory engine.

If you want to change the default behavior of the driver, you can add the optional driver Publisher parameter named preferredObjectClasses. The value of this parameter can be either one LDAP object class or a list of LDAP object classes separated by spaces.

When this parameter is present, the LDAP driver examines each object being presented on the Publisher channel to see if it contains one of the object classes in the list. It looks for them in the order they appear in the preferredObjectClasses parameter. If it finds that one of the listed object classes matches one of the values of the objectclass attribute on the LDAP object, it uses that object class as the one it reports to the Metadirectory engine. If none of the object classes match, it resorts to its default behavior for reporting the primary object class.

**Prevent Loopback:** The Prevent Loopback parameter is used only with the changelog publication method. The LDAP-search method doesn't prevent loopback, other than the loopback prevention built into the Metadirectory engine.

The default behavior for the Publisher channel is to avoid sending changes that the Subscriber channel makes. The Publisher channel detects Subscriber channel changes by looking in the LDAP changelog at the creatorsName or modifiersName attribute to see whether the authenticated entry that made the change is the same entry that the driver uses to authenticate to the LDAP server. If the entry is the same, the Publisher channel assumes that this change was made by the driver's Subscriber channel and doesn't synchronize the change.

If you select *LDAP Search*, fill in the following fields:

**LDAP Directory Base Container:** Specify the container where user objects reside in the LDAP directory. If you are using a flat Placement rule, this is the container where the users are placed. If you are using a mirrored Placement rule, this is the root container.

**Search Scope:** Indicates the depth of the polling searches. This parameter defaults to search the entire subtree that the LDAP `base-dn` points to.

**Class Processing Order:** Use this parameter to order certain events when referential attributes are an issue. The value of the parameter is a list of class names from the LDAP server, separated by spaces. For example, to make sure that new users are created before they are added to groups, make sure that `interorgperson` comes before `groupofuniquenames`.

The driver defines a special class name, `others`, to mean all classes other than those explicitly listed.

The default value for this parameter is `others groupofuniquenames`.

**Search Results to Synchronize on First Startup:** This parameter defines whether the initial search results are synchronized, or only subsequent changes are synchronized.

**LDAP search filters to filter on individual attributes:** Specify the LDAP search filters to filter the individual attributes for different classes which are in Driver filter. If you don't specify this option, the search is done based only on the object classes in the Driver filter like `objectclass=inetorgperson`. If there are `n` classes in the Driver filter, you can specify a maximum of `n` LDAP search filters separated by space. Each search filter is for its corresponding class in the driver filter. The following is an example of a search filter:

`(&(objectclass=inetorgperson)(cn=test))`

**Search Results to Synchronize on First Startup:** The first time the driver starts, it performs the defined LDAP search. This setting defines whether the initial search results are synchronized, or only the subsequent changes are synchronized.

If you select *No Publisher*, there are no additional fields.

**Use Sun Password Plugin:** Select *Yes* if you have installed and configured the Identity Manager Password plug-in on Sun Java System Directory and you want to use it to synchronize to the Identity manager distribution password.

## A.1.6  ECMAScript

Displays an ordered list of ECMAScript resource files. The files contain extension functions for the driver that Identity Manager loads when the driver starts. You can add additional files, remove existing files, or change the order the files are executed.

## A.1.7  Global Configuration

Displays an ordered list of Global Configuration objects. The objects contain extension GCV definitions for the driver that Identity Manager loads when the driver is started. You can add or remove the Global Configuration objects, and you can change the order in which the objects are executed.

# A.2  Global Configuration Values

Global configuration values (GCVs) are values that can be used by the driver to control functionality. GCVs are defined on the driver or on the driver set. Driver set GCVs can be used by all drivers in the driver set. Driver GCVs can be used only by the driver on which they are defined.

The LDAP driver includes many GCVs. You can also add your own if you discover you need additional ones as you implement policies in the driver.

To access the driver's GCVs in iManager:

**1** Click  to display the Identity Manager Administration page.

**2** Open the driver set that contains the driver whose properties you want to edit.

    **2a** In the *Administration* list, click *Identity Manager Overview*.

    **2b** If the driver set is not listed on the *Driver Sets* tab, use the *Search In* field to search for and display the driver set.

    **2c** Click the driver set to open the Driver Set Overview page.

**3** Locate the driver icon, click the upper right corner of the driver icon to display the *Actions* menu, then click *Edit Properties*.

or

To add a GCV to the driver set, click *Driver Set*, then click *Edit Driver Set properties*.

To access the driver's GCVs in Designer:

**1** Open a project in the Modeler.

**2** Right-click the driver icon 🖼 or line, then select *Properties > Global Configuration Values.*

or

To add a GCV to the driver set, right-clickthe driver set icon 🎛, then click *Properties > GCVs*.

The global configuration values are organized as follows:

- Section A.2.1, "Driver Parameters," on page 57
- Section A.2.2, "Entitlements," on page 58
- Section A.2.3, "Password Synchronization," on page 60
- Section A.2.4, "Account Status Support," on page 61
- Section A.2.5, "Account Tracking," on page 61

## A.2.1 Driver Parameters

**Subscriber Channel Placement Type:** Select the desired form of placement for the Subscriber channel. This option determines the Subscriber channel Placement policies.

- **mirrored:** Places objects hierarchically within the base container.
- **flat:** Places objects only in the base container.

**LDAP Directory Base Container:** Specify the container where user objects reside in the LDAP directory. If you are using a flat Placement rule, this is the container where the users are placed. If you are using a mirrored Placement rule, this is the root container. For example, ou=people,dc=example,dc=com.

**Publisher Channel Placement Type:** Select the desired form of placement for the Publisher channel. This option determines the Publisher channel Placement policies.

- **mirrored:** Places object hierarchically within the base container.
- **flat:** Places objects only in the base container.

## A.2.2 Entitlements

Entitlements act like an On/Off switch to control account access. When the driver is enabled for entitlements, accounts are created and removed or disabled only when the account entitlement is granted or revoked from users. For more information, see the *Identity Manager 4.0.2 Entitlements Guide*.

There are multiple sections in the Entitlements tab. Depending on which packages you installed, different options are enabled or displayed.

- "Entitlements" on page 58
- "Permission Collection and Reconciliation" on page 58
- "Data Collection" on page 59
- "Role Mapping" on page 59
- "Resource Mapping" on page 59
- "Parameter Format" on page 60
- "Entitlement Extensions" on page 60

### Entitlements

**Use Entitlements to Control LDAP Accounts?:** Select *True* to enable the driver to manage LDAP accounts based on the driver's defined entitlements.

Select *False* to disable management of LDAP accounts based on the entitlements.

**Account action on Entitlement Revoke:** Select the action to take when an LDAP User Account entitlement is revoked. The options are:

- Do Nothing
- Disable User
- Delete User

**Use Group Entitlement:** Select *True* to enable the driver to manage LDAP groups based on the driver's defined entitlements.

Select *False* to disable management of LDAP groups based on the entitlements.

**Advanced settings:** Select *show* to display the entitlement options that allow or deny additional functionality like data collection and others. These settings should rarely be changed.

### Permission Collection and Reconciliation

If you installed the Entitlements package used for permission collection and reconciliation, iManager and Designer display the following options. For more information about permission reconciliation feature, see "Understanding Permission Collection and Reconciliation Service ," in the *NetIQ Identity Manager 4.0.2 Common Driver Administration Guide*.

**Enable Permissions Collection and Reconciliation:** Set the value of this parameter to *true* for allowing permission collection and entitlement assignment. By default, the value is set to *false*, which allows the driver to override any other conditions to reconcile custom entitlements.

**Enable Permissions Reconciliation for Account Entitlement:** Ensure the value of this parameter is set to *Yes* to enable the driver to map the LDAP directory accounts to users in the Identity Vault and assign user account entitlements through the Publisher channel. By default, the value is set to *Yes*.

**Allow User add via publisher channel:** Set the value of this parameter to *Yes* to allow the driver to add new user accounts to the Identity Vault through the Publisher channel. By default, the value is set to *No*.

**Enable Permissions Reconciliation for Group entitlement:** Ensure the value of this parameter is set to *Yes* to enable the driver to assign group entitlements through the Publisher channel. By default, the value is set to *Yes*.

---

**IMPORTANT:** If you set the value of this parameter to *No*, the group resource is not created in the User Application.

---

**Enable Permissions Reconciliation for all Custom entitlements:** If the value of this parameter is set to *No*, it allows you to select the custom entitlements for reconciling them. By default, it is set to *Yes*, which allows reconciling of all custom entitlements.

**Add Custom Entitlements for Reconciliation:** This parameter is presented if the value of *Enable Permission Reconciliation* for all Custom Entitlements is set to *No*.

Click the *Add* ✚ icon add custom entitlements you want to selectively onboard and specify *Assignment Attribute Name* for them.

## Data Collection

Data collection enables the Identity Report Module to gather information to generate reports. For more information, see the *Identity Reporting Module Guide*.

**Enable data collection:** Select *Yes* to enable data collection for the driver through the Data Collection Service by the Managed System Gateway driver. If you are not going to run reports on data collected by this driver, select *No*.

**Allow data collection from user accounts:** Select *Yes* to allow data collection by the Data Collection Service through the Managed System Gateway driver for the user accounts.

**Allow data collection from groups:** Select *Yes* to allow data collection by the Data Collection Service through the Managed System Gateway driver for groups.

## Role Mapping

The Role Mapping Administrator allows you to map business roles with IT roles. For more information, see the *Novell Identity Manager Role Mapping Administrator 4.0.2 User Guide*.

**Enable role mapping:** Select *Yes* to make this driver visible to the Role Mapping Administrator.

**Allow mapping of user accounts:** Select *Yes* if you want to allow mapping of user accounts in the Role Mapping Administrator. An account is required before a role, profile, or license can be granted through the Role Mapping Administrator.

**Allow mapping of groups:** Select *Yes* if you want to allow mapping of groups in the Role Mapping Administrator.

## Resource Mapping

The Roles Based Provisioning Module allows you to map resources to users. For more information, see the *User Application: User Guide*.

**Enables resource mapping:** Select *Yes* to make this driver visible to the Roles Based Provisioning Module.

**Allow mapping of user accounts:** Select Yes if you want to allow mapping of user accounts in the Roles Based Provisioning Module. An account is required before a role, profile, or license can be granted.

**Allow mapping of groups:** Select *Yes* if you want to allow mapping of groups in the Roles Based Provisioning Module.

## Parameter Format

**Format for Account entitlement:** Select the parameter format the entitlement agent must use when granting this entitlement. The options are *Identity Manager 4* or *Legacy*.

**Format for Group entitlement:** Select the parameter format the entitlement agent must use when granting this entitlement. The options are *Identity Manager 4* or *Legacy*.

## Entitlement Extensions

**User account extensions:** The content of this field is added below the entitlement elements in the EntitlementConfiguration resource object.

**Group extensions:** The content of this field is added below the entitlement elements in the EntitlementConfiguration resource object.

## A.2.3   Password Synchronization

These GCVs enable password synchronization between the Identity Vault and the LDAP system.

In Designer, you must click the 🏺 icon next to a GCV to edit it. This displays the Password Synchronization Options dialog box for a better view of the relationship between the different GCVs.

In iManager, you should edit the Password Management Options on the *Server Variables* tab rather than under the GCVs. The Server Variables page has a better view of the relationship between the different GCVs.

For more information about how to use the Password Management GCVs, see "Configuring Password Flow" in the *Identity Manager 4.0.2 Password Management Guide*.

**Connected System or Driver Name:** Specify the name of the LDAP system or the driver name. This valued is used by the e-mail notification template to identity the source of the notification message.

**Application accepts passwords from Identity Manager:** If *True*, allows passwords to flow from the Identity Manager data store to the connected system.

**Identity Manager accepts passwords from application:** If *True*, allows passwords to flow from the connected system to Identity Manager.

**Publish passwords to NDS password:** Use the password from the connected system to set the non-reversible NDS password in eDirectory.

**Publish passwords to Distribution Password:** Use the password from the connected system to set the NMAS Distribution Password used for Identity Manager password synchronization.

**Require password policy validation before publishing passwords:** If *True*, applies NMAS password policies during publish password operations. The password is not written to the data store if it does not comply.

**Reset user's external system password to the Identity Manager password on failure:** If *True*, on a publish Distribution Password failure, attempts to reset the password in the connected system by using the Distribution Password from the Identity Manager data store.

**Notify the user of password synchronization failure via e-mail:** If *True*, notifies the user by e-mail of any password synchronization failures.

## A.2.4 Account Status Support

**LDAP Server Type:** Specifies the LDAP server type this driver connects to. Based on the type, the appropriate attribute is modified in the LDAP directory to disable the account. The options are:

- ◆ eDirectory
- ◆ openLDAP
- ◆ iPlanet | SunOne | OID

**nsManageDisabledRole DN:** This option is only displayed if you select *iPlanet | SunOne |OID*. This is the DN of the role used to disable users in the connected LDAP system. If a user is enabled in the Identity Vault, this role must first be removed from the inetOrgPerson object before the attribute that disables the account can be cleared. By default, this role is named cn=nManagedDisabledRole, plus the name of your directory servers' root DN. For example cn=nManagedDisabledRole,dc=example,dc=com.

## A.2.5 Account Tracking

Account tracking is part of the Identity Reporting Module. For more information, see the *Identity Reporting Module Guide*.

**Enable account tracking:** If *True,* this option enables account tracking policies. Set it to *False* if you do not want to execute account tracking policies.

**Realm:** Specifies the name of the realm, security domain, or namespace in which the account name is unique.

**Advanced settings:** Leave the setting as *show* to display the account tracking settings. Changing these settings might result in malfunction of the Account Tracking feature. Only change these settings if you know exactly what you are doing.

**Identifiers:** Adds the account identifier attributes. Attribute names must be in the application namespace.

**Object Class:** Adds the object class to track. Class names must be in the application namespace.

**Status attribute:** Specifies the name of the attribute in the application namespace to represent the account status.

NOTE: The Managed System Query does not return proper status of accounts because the default value for the account status attribute set by the Account Tracking package is loginDisabled. For a proper account status, manually select the appropriate LDAP directory server type or edit the XML file and add the appropriate account status attribute that represents the account status of the directory server that you are using in the Account Tracking GCV.

**Status active value:** Specifies the value of the status attribute that represents an active state.

**Status inactive value:** Specifies the value of the status attribute that represents an inactive state.

**Subscription default status:** Specifies the default status that the policies assume when an object is subscribed to the application and the status attribute is not set in the Identity Vault.

**Publication default status:** Specifies the default status that the policies assume when an object is published to the Identity Vault and the status attribute is not set in the application.

# B <br> Trace Levels

The driver supports the following trace levels:

| Level | Description |
| --- | --- |
| 1 | Minimal Tracing |
| 2 | Previous level and LDAP Server information, LDAP operations in the Subscriber channel, Changelog Information |
| 3 | Previous level and LDAPSearch messages in the Publisher channel and all the remaining messages |
| 4 | Previous level and LDAPSearch messages in the Publisher channel and all the remaining messages |
| 5 | Previous level and password synchronization messages in the Publisher channel |

For information about setting driver trace levels, see "Viewing Identity Manager Processes" in the *NetIQ Identity Manager 4.0.2 Common Driver Administration Guide*.