NetIQ® Identity Manager
Driver for Top Secret® Implementation Guide

December, 2021

Legal Notice

Third-Party Materials

Contents

About this Book and the Library

This guide describes implementation of the NetIQ® Identity Manager 4.8 driver for CA Top Secret on mainframes (z/OS operating system).

The driver synchronizes data from a connected mainframe system using CA Top Secret Security with NetIQ Identity Manager 4.8, the comprehensive identity management suite that allows organizations to manage the full user life cycle, from initial hire, through ongoing changes, to ultimate retirement of the user relationship.

Other Information in the Library

The library provides the following information resources:

Identity Manager Setup Guide

Provides overview of Identity Manager and its components. This book also provides detailed planning and installation information for Identity Manager.

Designer Administration Guide

Provides information about designing, testing, documenting, and deploying Identity Manager solutions in a highly productive environment.

User Application: Administration Guide

Describes how to administer the Identity Manager User Application.

User Application: User Guide

Describes the user interface of the Identity Manager User Application and how you can use the features it offers, including identity self-service, the Work Dashboard, role and resource management, and compliance management.

User Application: Design Guide

Describes how to use the Designer to create User Application components, including how to work with the Provisioning view, the directory abstraction layer editor, the provisioning request definition editor, the provisioning team editor, and the role catalog.

Identity Reporting Module Guide

Describes the Identity Reporting Module for Identity Manager and how you can use the features it offers, including the Reporting Module user interface and custom report definitions, as well as providing installation instructions.

Analyzer Administration Guide

Describes how to administer Analyzer for Identity Manager.

Identity Manager Common Driver Administration Guide

Provides information about administration tasks that are common to all Identity Manager drivers.

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. If you have suggestions for improvements, click **Add Comment** at the bottom of any page in the HTML versions 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

Qmunity, 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, Qmunity 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 http://community.netiq.com.

1 Overview

The NetIQ® Identity Manager 4.8 driver for CA Top Secret synchronizes data between the Identity Vault and a connected system running Top Secret Security for z/OS. The driver runs on the targeted z/OS system. The Identity Vault runs on any Identity Manager supported platform and communicates with the driver on the target z/OS system over a secure network link.

The driver uses embedded Remote Loader technology to communicate with the Identity Vault, bidirectionally synchronizing changes between the Identity Vault and the connected system. The embedded Remote Loader component, also called the driver shim, runs as a started task on the connected z/OS system. There is no requirement to install Java* on the connected system.

The Subscriber shim commits changes to the security system using customizable REXX execs that issue native TSO commands.

The Publisher shim uses standard security system exit routines to capture events of interest and submits them to the Metadirectory engine.

The driver uses a scriptable framework, designed so that you can easily add support for existing and future applications.

The Identity Manager 4.8 driver for Top Secret combines the flexibility of the Fan-Out driver and the bidirectional support and Identity Manager policy options available from traditional Identity Manager drivers. Key features of the driver include:

- Bidirectional synchronization of data without requiring Java or a separate Remote Loader
- Customizable schema to integrate all aspects of account administration
- Customizable REXX execs to handle all data to be synchronized
- Configuration on the z/OS system using traditional sequential files
- Driver shim implemented as a traditional z/OS started task
- Operator command control for starting and stopping the driver shim, configuring Remote Loader options, and displaying status information

The following sections present a basic overview of the driver:

- "Driver Architecture" on page 11
- "Configuration Overview" on page 16

Driver Architecture

The driver synchronizes information between the Identity Vault and the security system on the connected z/OS system.

Identity Manager detects relevant changes to identities in the Identity Vault and notifies the Subscriber component of the driver. After customizable policy processing, events are sent to the Subscriber channel of the driver shim started task on the connected system. The Subscriber shim securely passes the information to customizable REXX execs that perform the required actions.

Standard security system exit routines capture password and other changes and place them on a memory queue. The change log started task moves events from the memory queue to the change log data set, where they are stored for processing. At configurable intervals, the Publisher shim polls the change log for events and submits them to the Metadirectory engine. The Metadirectory engine processes these events using customizable policies, and posts relevant changes to the Identity Vault.

The following illustration shows an overview of the architecture.

Figure 1-1   *Top Secret Driver Architecture*
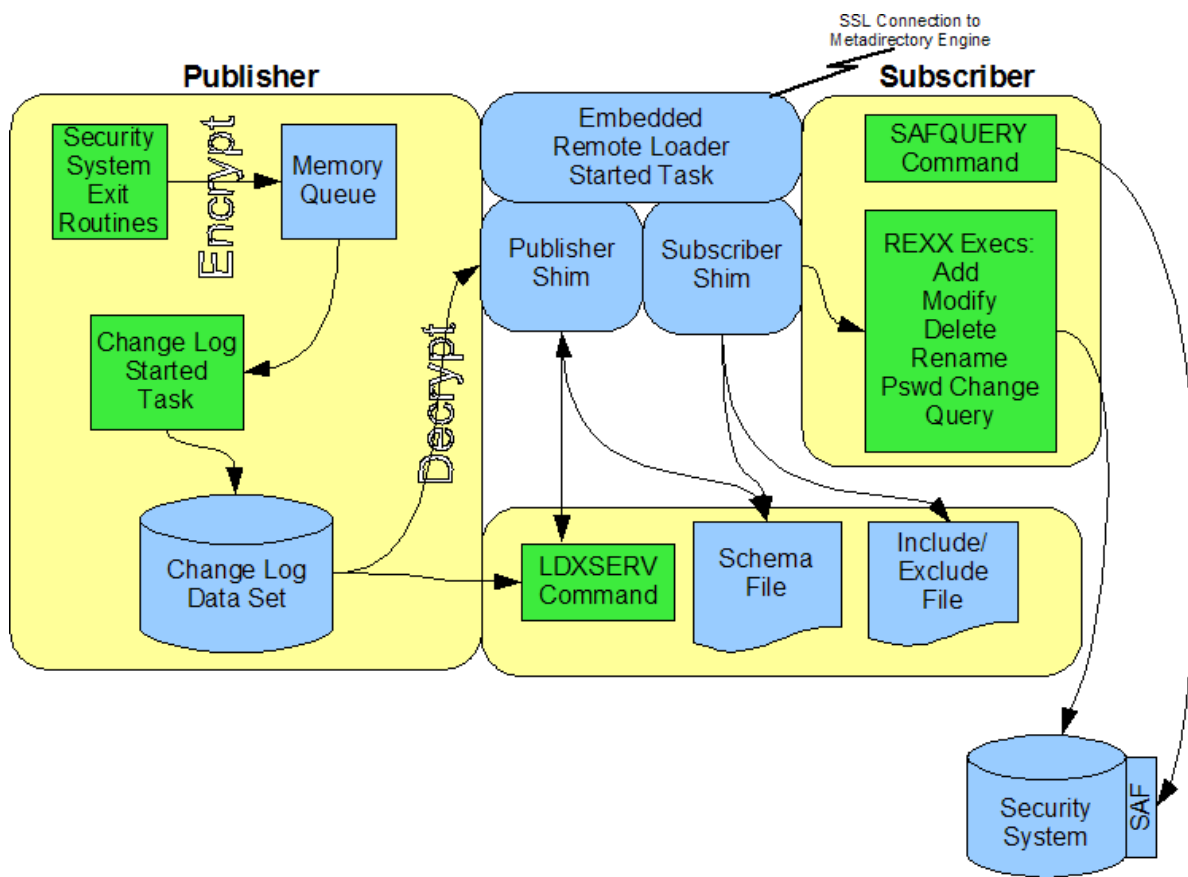
The following topics describe the driver architecture in more detail:

- "Publisher Channel" on page 13
- "Subscriber Channel" on page 14
- "Driver Shim Started Task" on page 14
- "LDXSERV Command" on page 15
- "Scriptable Framework" on page 15
- "Schema File" on page 15
- "Include/Exclude File" on page 15

Publisher Channel

The Publisher shim periodically examines the change log for events. When the Publisher shim finds events in the change log, it decrypts, processes, and sends them to the Metadirectory engine in XDS format over a Secure Sockets Layer (SSL) network link. The Metadirectory engine applies policies, takes the appropriate actions, and posts the events to the Identity Vault.

z/OS object names are uppercase. The Publisher Placement policy converts object names to lowercase.

Security System Exit Routines

Top Secret provides an installation exit named TSSINSTX. The driver uses this exit to detect activity of interest and to place events in the memory queue. When the driver exit routines place an event in the memory queue, they notify the change log started task. The change log started task then moves the event information to the change log data set. Each system that shares the security system database must run the exit routines provided by the driver in module IDMTSSIX.

The driver exit routines perform the following tasks:

- Monitor password changes from the local security system and record user and password information in the memory queue.
- Monitor security system administrative commands entered by users, either directly from the TSO command line, or as generated by the administrative panels. The exit routines record these commands and related information, such as the issuer and time stamp, in the memory queue.

Memory Queue

The memory queue is an encrypted, in-storage buffer that holds events. Events are added to the memory queue by the security system exit routines, and are removed from the queue by the change log started task. The memory queue is located in Subpool 231 (fetch-protected ECSA).

Change Log Started Task

The change log started task is notified of events added to the memory queue by the driver exit routines and moves them to the change log data set.

Each system that shares a security system database must run the change log started task. The change log started task must be started as part of your normal z/OS system initialization procedure and stopped during normal system shutdown.

Change Log Data Set

The change log started task removes encrypted events from the memory queue and stores them in the change log data set for processing by the Publisher shim. The Publisher shim removes events from the change log at configurable intervals and submits them to the Metadirectory engine. If communication with the Metadirectory engine is temporarily lost, events remain in the change log until communication becomes available again.

The change log data set is a standard z/OS direct access (DSORG=DA) data set. There is one change log data set for the set of systems that share the security system database. The change log data set must reside on a shared device unless the security system database is not shared.

Subscriber Channel

The Subscriber channel of the driver shim started task receives XDS command documents from the Metadirectory engine, stores them using z/OS name/token callable services, then calls the appropriate REXX execs to handle the command.

The Subscriber Creation policy converts object names to uppercase before the command documents are sent to the Subscriber. z/OS object names are uppercase.

The Subscriber shim calls the `LDXSERV` command on startup to identify itself to the security system exit routines for loopback detection. This prevents the exit routines from generating events for commands issued by the Subscriber shim.

SAFQUERY Command

`SAFQUERY` is an APF-authorized TSO command that is used by the driver to query security system information. `SAFQUERY` uses the RACROUTE macro for z/OS to retrieve information from the security system database through the system authorization facility (SAF).

REXX Execs

The provided REXX execs support adds, modifies, deletes, and renames for User and Group objects, and handle password synchronization. The REXX execs use standard TSO commands to apply the changes. You can extend the REXX execs to support other object types and events. The REXX execs have secure access to the original XDS command data using the `IDMGETV` command. `IDMGETV` accesses z/OS name/token callable services and places the data in REXX variables.

Driver Shim Started Task

The driver shim runs as a started task. Only one system that shares the security system database runs the driver shim started task. The driver shim started task must be started as part of your normal z/OS system initialization procedure and stopped during normal system shutdown.

LDXSERV Command

`LDXSERV` is an APF-authorized TSO command that provides services for the driver shim. The Subscriber channel calls `LDXSERV` with the NOTSSLOG parameter at startup. This uses z/OS name/token callable services to create a token in the address space. If the security system exit routines find the token, they do not generate events. This prevents publication of events for actions taken by the Subscriber (loopback).

You can use the `LDXSERV STATUS` command to display information about the Publisher channel event subsystem. To use the `LDXSERV` command, you must include the driver load library in the logon procedure STEPLIB concatenation.

Scriptable Framework

The interface between the security system and the driver shim uses customizable REXX execs. You can extend the execs that are provided with the driver to support other applications and databases.

Several utility execs and helper commands are provided with the driver to enable communication with the driver shim and the change log. An extensible connected system schema file allows you to add your own objects and attributes to those already supported by the driver.

For more information about the REXX execs and the scriptable framework, see "The Scriptable Framework" on page 59.

Schema File

The configuration of class and attribute definitions for the connected system is specified using the schema file. You can modify and extend this file to include new objects and attributes. For details about configuring the schema file, see "The Connected System Schema File" on page 61.

The driver uses the keywords and functions of the Top Secret `TSS` administrative command to define the schema. The schema includes two classes: USER and GROUP. These correspond to Top Secret users and groups.

Some items in the schema refer to keywords used to create and modify Top Secret users, but cannot be queried or synchronized. These attributes can be used only by Identity Manager policies to make event-time decisions that affect the behavior of the `TSS` administrative command. The auxiliary schema used to extend eDirectory™ does not include these attributes.

The schema contains some attributes that consolidate multiple Top Secret attributes.

Include/Exclude File

The include/exclude file allows local system policies to enforce which objects are included or excluded from provisioning by the Subscriber channel. This allows for administrative rules to be set and enforced locally rather than having processing decisions made by the Metadirectory engine. For details about using the include/exclude file, see "The Connected System Include/Exclude File" on page 63.

To control which objects are processed by the Publisher channel, use policies. For details about customizing policies, see the policy documentation on the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).

Configuration Overview

This section discusses driver configuration details specific to the Identity Manager Driver for Top Secret. For basic configuration information, see the *Identity Manager 4.8 Administration Guide* on the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/). For detailed information about configuring the driver, see Chapter 5, "Configuring the Top Secret Driver," on page 47.

Topics include

- ◆ "Data Flow" on page 16
- ◆ "OMVS Information Management" on page 16
- ◆ "TSO Information Management" on page 17
- ◆ "Filter and Schema Mapping" on page 17
- ◆ "Policies" on page 17

Data Flow

Filters and policies control the data flow of users and groups to and from the connected system and the Identity Vault. The **Data Flow** option, specified during driver import, determines how these filters and policies behave.

- ◆ **Bidirectional:** Sets classes and attributes to be synchronized on both the Subscriber and Publisher channels.
- ◆ **Application to Identity Vault:** Sets classes and attributes to be synchronized on the Publisher channel only.
- ◆ **Identity Vault to Application:** Sets classes and attributes to be synchronized on the Subscriber channel only.

OMVS Information Management

The **Set Preconfigured OMVS Data** option, specified during driver import, determines whether the driver sets preconfigured OMVS (UNIX System Services) attributes for new users in the security system.

The attributes you can configure are:

- ◆ **OMVSPGM:** The default program (login shell)
- ◆ **UID Assignment:** Whether UID and GID numbers are assigned by the security system or by the Identity Vault
- ◆ **HOME:** The default home directory

TSO Information Management

The **Set Preconfigured TSO Data** option, specified during driver import, determines whether the driver sets preconfigured Time Sharing Option (TSO) information for new users in the security system.

The attributes you can configure are:

- **TSOLACCT:** The default account number
- **TSOLPROC:** The default logon procedure
- **TSOUNIT:** The default unit name

Filter and Schema Mapping

The Metadirectory engine uses filters to control which objects and attributes are shared. The default filter configuration for the driver allows objects and attributes to be shared as described in the following table:

Table 1-1 *Default Filter and Schema Mapping*

eDirectory Class	eDirectory Attribute	Top Secret Class	Top Secret Attribute
User	CN	USER	ACID
User	Group Membership	USER	GROUP
User	Login Disabled	USER	SUSPEND
User	Login Expiration Time	USER	UNTIL
User	Password Expiration Interval	USER	PASSINT
User	Surname	USER	NAME
Group	CN	GROUP	ACID

Policies

The Metadirectory engine uses policies to control the flow of information into and out of the Identity Vault. The following table describes the policy functions for the driver in the default configuration:

Table 1-2 *Default Driver Policy Functions*

Policy	Description
Mapping	Maps the Identity Vault User and Group objects and selected attributes to a user or group in the security system.
Publisher Input	Parses security system commands to produce XDS events.
Publisher Event	None is provided.

Policy	Description
Publisher Matching	Restricts privileged accounts and defines matching criteria for placement in the Identity Vault.
Publisher Create	Defines creation rules for users and groups before provisioning into the Identity Vault.
Publisher Placement	Defines where new users and groups are placed in the Identity Vault. Converts object names to lowercase.
Publisher Command	Defines password publishing policies.
Subscriber Matching	Defines rules for matching users and groups in the connected system and restricts events from a configurable container.
Subscriber Create	Defines required creation criteria. Converts object names to uppercase.
Subscriber Command	Defines password subscribing policies.
Subscriber Output	Sends e-mail notifications for password failures and converts information formats from the Identity Vault to the connected system.

2 Planning for the Top Secret Driver

This section helps you plan for deployment of the NetIQ® Identity Manager 4.8 driver for CA Top Secret. Topics include

- "Deployment Planning" on page 19
- "Migration Planning" on page 20
- "Customization Planning" on page 20
- "Started Task User IDs" on page 21
- "Establishing a Security-Equivalent User" on page 21

For more information about planning, see the *Identity Manager 4.8 Installation Guide* on the Identity Manager Documentation Web site (https://www.netiq.com/documentation/idm45/).

Deployment Planning

- Review Chapter 3, "Installing the Top Secret Driver," on page 23 and Chapter 5, "Configuring the Top Secret Driver," on page 47.
- Is this a new installation, or are you replacing a Fan-Out driver Platform Services installation? For details about upgrading from the Fan-Out driver, see Chapter 4, "Upgrading from the Fan-Out Driver," on page 43.
- Consider where and how you will install each component.
  - You must install the driver libraries (samples library, load library, and REXX exec library) on a volume that is shared by each system that shares the security system database.
  - You must run the driver shim started task on only one system that shares the security system database.
  - You must create the change log data set on a volume that is shared by all systems that share the security system database.
  - You must run the change log started task on each system that shares the security system database.
  - You must install the exit routines on each system that shares the security system database.
- Consider how you will respond to the installation prompts and other installation decisions.
- You must provide a connected system schema file during installation. A file with the required classes and attributes is provided in the driver samples library member `SCHEMDEF`.

  For details about the connected system schema file, see "The Connected System Schema File" on page 61.
- You must provide a driver shim configuration file during installation. A file that you can customize is provided in the driver samples library member `DRVCONF`.

  For details about the driver shim configuration file, see "The Driver Shim Configuration File" on page 54.

- You must provide an include/exclude file during installation. A file with basic suggested content is provided in the driver samples library member `INCEXC`.

  You can use the include/exclude file on the connected system to limit your initial deployment to a small number of users and groups.

  For details about the include/exclude file, see "The Connected System Include/Exclude File" on page 63.

- How will you prototype, test, and roll out your deployment?

- What user ID will you use to run the change log started task? What user ID will you use to run the driver shim started task?

  For details about the requirements for these user IDs, see "Started Task User IDs" on page 21.

- What are the host names or IP addresses of your Metadirectory server and the system that will run the driver shim started task?

- Will you use the default TCP port numbers?

Table 2-1 *Default TCP Port Numbers*

Purpose	TCP Port Number
Driver shim connection to the Metadirectory engine	8090
Driver shim HTTP services for log viewing	8091
Secure LDAP port	636
Non-secure LDAP port	389

Migration Planning

- Where are the objects that you plan to manage with the driver?

- Can you use a Matching policy to select the objects to manage based on criteria such as department, group membership, or some other attribute?

Customization Planning

- Do you plan to customize the REXX execs provided with the driver?

  For details about the provided execs, see Table 6-1, "Identity Vault Command Processing Execs," on page 60; Table 6-2, "Other Execs," on page 60; and the execs themselves.

- Do you plan to add attributes or classes to the connected system schema file?

  For details about the connected system schema file, see "The Connected System Schema File" on page 61.

- What options do you plan to use in your driver shim configuration file?

  For details about the driver shim configuration file, see "The Driver Shim Configuration File" on page 54.

- How will you use the include/exclude file?

For details about the include/exclude file, see "The Connected System Include/Exclude File" on page 63.

◆ Do you plan to customize policies?

For details about customizing policies, see the policy documentation on the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).

◆ Are the resources needed to perform the customization available within your organization?

Started Task User IDs

You must prepare user IDs for each started task to use. For details, see "Preparing User IDs for the Started Tasks" on page 34.

These user IDs have special requirements.

◆ "Change Log Started Task User Requirements" on page 21
◆ "Driver Shim Started Task User Requirements" on page 21

Change Log Started Task User Requirements

The change log started task must run as a user that can update the change log data set.

Driver Shim Started Task User Requirements

The driver shim started task user must have rights to update the change log data set and to perform the Subscriber channel actions carried out by the REXX execs, such as creating and modifying users and groups, defining alias information in the catalog, and creating home directories. For details about using the `TSS ADMIN` command to assign administrative authorities, see your *CA Top Secret Security for z/OS Command Functions Guide*.

Establishing a Security-Equivalent User

The Driver object must run with Security Equivalence to a user with sufficient rights. You can set the driver equivalent to Admin or a similar user. For stronger security, you can define a user with only the minimal rights necessary for the operations you want the driver to perform.

The driver user must be a trustee of the containers where synchronized users and groups reside, with the rights shown in Table 2-2. Inheritance must be set for [Entry Rights] and [All Attribute Rights].

Table 2-2   *Base Container Rights Required by the Driver Security-Equivalent User*

Operation	[Entry Rights]	[All Attribute Rights]
Subscriber notification of account changes (recommended minimum)	Browse	Compare and Read

Operation	[Entry Rights]	[All Attribute Rights]
Creating objects in the Identity Vault without group synchronization	Browse and Create	Compare and Read
Creating objects in the Identity Vault with group synchronization	Browse and Create	Compare, Read, and Write
Modifying objects in the Identity Vault	Browse	Compare, Read, and Write
Renaming objects in the Identity Vault	Browse and Rename	Compare and Read
Deleting objects from the Identity Vault	Browse and Erase	Compare, Read, and Write
Retrieving passwords from the Identity Vault	Browse and Supervisor	Compare and Read
Updating passwords in the Identity Vault	Browse and Supervisor	Compare, Read, and Write

If you do not set Supervisor for [Entry Rights], the driver cannot set passwords. If you do not want to set passwords, set the Subscribe setting for the User class nspmDistributionPassword attribute to Ignore in the filter to avoid superfluous error messages. For details about accessing and editing the filter, see the policy documentation on the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).

For complete information about rights, see the *NetIQ eDirectory™ Administration Guide*.

3 Installing the Top Secret Driver

This section provides the information you need to install the NetIQ® Identity Manager 4.8 driver for CA Top Secret.

Topics include

Before You Begin

- Review Chapter 2, "Planning for the Top Secret Driver," on page 19.
- Ensure that you have the most recent distribution, support pack, and patches for the driver.
- Review the most recent support information for the driver on the NetIQ Support Web site (http://support.netiq.com).

Required Knowledge and Skills

To successfully install, configure, and use the driver, you must have system administration skills and rights for Identity Manager, z/OS, and Top Secret. You must be proficient with using iManager to configure Identity Manager drivers. You must be familiar with the facilities of the driver, and you must have developed a deployment plan.

For an overview of driver facilities, see Chapter 1, "Overview," on page 11.

For information about planning for the driver, see Chapter 2, "Planning for the Top Secret Driver," on page 19.

For information about administering your target systems, see your IBM* and Top Secret documentation.

Prerequisites

- "Connected System Requirements" on page 24
- "Identity Vault Requirements" on page 24

Connected System Requirements

❒ z/OS

❒ CA Top Secret Security for z/OS

For information about supported platforms and operating environments, see the Identity Manager 4.8 Drivers Documentation Web site (https://www.netiq.com/documentation/identity-manager-47-drivers). From this index page, you can select a readme file associated with the platform(s) for which you need support.

Identity Vault Requirements

❒ NetIQ Identity Manager 4.7 or later

Getting the Installation Files

1 Obtain the most recent distribution of the Identity Manager 4.8 Driver for CA Top Secret from the NetIQ Downloads Web site (https://dl.netiq.com/index.jsp).

At the time of this *Implementation Guide*'s release, the driver was included in the following ISO package:

```
NIdM_Integration_Module_4.8.3_Mainframes_Midrange.iso
```

2 The following files will be needed for the mainframe installation:

```
SAMPLIB.XMT
IDMLOAD.XMT
TSSEXEC.XMT
```

These files are located under `bidirectional/TSS` on the ISO distribution.

Creating the Driver in Designer

The Top Secret Driver supports Designer 4 Package features, which allows you to create a driver by selecting which packages to install. After you create and configure the driver, you need to deploy it to the Identity Vault and start it.

Topics in this section include

Importing the Current Driver Packages

Driver packages can be updated at any time and are stored in the Package Catalog. Packages are initially imported into the Package Catalog when you create a project, import a project, or convert a project. It is important to verify you have the latest packages imported into the Package Catalog before you install the driver.

To verify you have the latest packages imported into 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.

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

5 Click **Import Package**.

Figure 3-1   **Import Package**
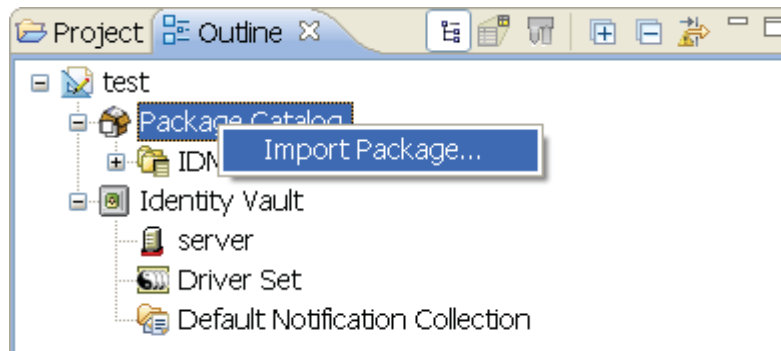
6 Select the Top Secret Packages

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.

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 to the next section, Installing the Driver Packages.

Installing the Driver Packages

After you have imported the current driver packages into the Package Catalog, you can install the driver packages to create a new driver.

1. In Designer, open your project.

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

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

4. Select the optional features to install for the Top Secret driver. The options are:

   IMPORTANT: Publications referenced in the following option descriptions can be accessed at the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).

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

   Entitlements: This package contains configuration information for synchronizing Top Secret accounts and policies that enable account creation and auditing for the Top Secret driver. To enable account creation and auditing, verify that this option is selected. For more information, see the *Identity Manager 4.8 Entitlements Guide*.

   Password Synchronization: This package contains the policies that enable the Top Secret driver to synchronize passwords. To synchronize passwords, verify that this option is selected. For more information, see the *Identity Manager 4.8 Password Management Guide*.

   Data Collection: This package contains 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 *Identity Reporting Module Guide*.

   Account Tracking: This package contains the policies that enable you to track accounts 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. After selecting the optional packages, click **Next**.

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

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) The Common Settings page is displayed only if the Common Settings package is installed as a dependency. On the Install Common Settings page, fill in the following fields:

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

   If you want a unique location for this driver, set the value for all drivers on this page. After the driver is created, change the value on the driver's Global Configuration Values page.

   Group Container: Since the Top Secret driver does not synchronize Group objects, this setting can be ignored.

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 Accounts" in the *NetIQ Identity Manager 4.8 Common Driver Administration Guide*.

10 On the Driver Information page, fill in the following field:

Driver Name: Specify a name for the driver that is unique within the driver set.

11 On the Install Top Secret Base page, fill in the following fields to connect to the Remote Loader and click Next:

Connect to Remote Loader: By default, the driver is configured to connect using the Remote Loader. You must select Yes for this option.

Host Name: Specify the port number where the Remote Loader is installed and is running for this driver. The default port number is 8090.

Port: Specify the Remote Loader's password as defined on the Remote Loader. The Metadirectory server (or Remote Loader shim) requires this password to authenticate to the Remote Loader.

Remote Password: Specify the Remote Loader's password as defined on the Remote Loader. The Metadirectory server (or Remote Loader shim) requires this password to authenticate to the Remote Loader.

Driver Password: Specify the driver object password that is defined in the Remote Loader service. The Remote Loader requires this password to authenticate to the Metadirectory server.

12 On the Install Top Secret Base page, fill in the following fields for the driver parameters:

Connected System or Driver Name: The name of the connected system, application or Identity Manager driver. This value is used by the e-mail notification templates.

Synchronize Top Secret Passwords to Identity Vault: Specifies whether password changes in Top Secret should be synchronized with the Identity Vault password.

Publish TSO Commands: Specifies whether the original TSO commands on the publisher channel should be published as status documents (for logging purposes).

13 (Conditional) This page is displayed only if you selected to install the Managed System Information packages. On the Install Top Secret Managed System Information page, fill in the following fields, then click Next:

Classification: Select the classification of the Top Secret system. This information is displayed in the reports. Options include:

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

If you select Other, you must specify a custom classification for the Top Secret system.

Environment: Select the type of environment the Top Secret system provides. Options include:

- Development
- Test
- Staging

- **Production**
- **Other**

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

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

14 (Conditional) On the System Ownership page, fill in the following fields to define the ownership of the Top Secret system, then click **Next**:

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

15 (Conditional) On the General Information page, fill in the following fields to define your Top Secret system, then click **Next**:

Name: Specify a descriptive name for this Top Secret system. The name is displayed in reports.

Description: Specify a brief description for this Top Secret system. The description is displayed in reports.

Location: Specify the physical location for this Top Secret system. The location is displayed in reports.

Vendor: Leave CA as the vendor of Top Secret. This information is displayed in reports.

Version: Specify the version of this Top Secret system. The version is displayed in reports.

16 (Conditional) On the Top Secret Entitlements page, review the default values for Top Secret entitlement options. Change any, if necessary, and click **Next**.

17 (Conditional) On the Account Tracking page, review the default values for Top Secret Account Tracking options. Change any, if necessary, and click **Next**:

18 (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. Identity Manager uses the CSV file to map Top Secret entitlements into corresponding resources in the Identity Manager catalog.

Entitlement Name: Specify a descriptive name for the entitlement to map it to the CSV file that contains the Top Secret entitlement details.

Entitlement Name is the name of the entitlement. This parameter corresponds to the Entitlement Assignment Attribute in Top Secret. For example, you could define an entitlement called *ParkingPass*.

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

Entitlement Assignment Attribute holds the entitlement values in Top Secret. 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 as the driver. This file contains the values for the application entitlements.

Multi-valued: Set the value of this parameter to **True** if you want to assign resources and entitlements multiple times with different values to the same user. Otherwise, set it to **False**.

19 (Conditional) On the Driver Parameters page, fill in the following fields, then click **Next**:

Create Users with (USING): Specifies a User to be used as a model for creating new Users in Top Secret.

Default Department: Enter a default Department for new Accounts in Top Secret.

User Default Group: Enter a default group for new Users in Top Secret.

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

The driver is created. You can modify the configuration settings by continuing with the next section, Configuring the Driver. If you don't need to configure the driver, skip ahead to Deploying the Driver.

Configuring the Driver

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 and the Global Configuration Values. These settings must be configured properly for the driver to start and function correctly.

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 Modify the driver settings as necessary.

> **IMPORTANT:** In addition to the driver settings, 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 Top Secret*, 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 "Configuration Overview" on page 16.

4 Continue with the next section, Deploying the Driver.

Deploying the Driver

After a driver 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. Whatever rights that the driver needs to have on the server, the DriversUser object must have the same security rights:

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

7b Click **OK** twice.

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.

8b Click **OK**.

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

8d Click **OK**.

9 Click **OK**.

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**.

Creating the Driver in iManager

Drivers are created with packages, and iManager does not support packages. In order to create or modify drivers, you must use Designer. See "Creating the Driver in Designer" on page 24.

Installing the Driver Shim on the Connected System

The driver shim and its files are installed into data sets that you specify, and into files created by the installation process in the HFS.

The driver uses an embedded Remote Loader. It is not necessary to install Java on the connected system.

For all procedures in this section that are performed using the target system, use a user ID with administrative rights.

- "Setting Up the Libraries on Your z/OS System" on page 31
- "Authorizing the Driver TSO Commands" on page 32
- "Securing the Driver Shim with SSL" on page 33

- "Configuring the Remote Loader and Driver Object Passwords" on page 33
- "Allocating and Initializing the Change Log Data Set" on page 33
- "Setting Up the Started Tasks" on page 34
- "Testing before Installing the Security System Exit" on page 36
- "Installing the Driver Security System Exit IDMTSSIX" on page 37
- "Testing the Completed Connected System Installation" on page 40

Setting Up the Libraries on Your z/OS System

The driver shim is packaged as z/OS partitioned data sets (PDS) unloaded with the `TRANSMIT` command.

- **Driver Samples Library:** `samplib.xmt` contains sample cataloged procedures, other JCL, and sample configuration-related files.
- **Driver Load Library:** `idmload.xmt` contains executable programs for the driver shim.
- **Driver REXX Exec Library:** `tssexec.xmt` contains the REXX execs for the scriptable framework and to perform configuration tasks.

To upload these files to the target system and extract them:

1 Use FTP to upload the files to the target system from the workstation where you placed them in Step 2 on page 24.

```
c:\> ftp Your-z/OS-Host
User: Your-User-ID
Password:
ftp> quote site lrecl=80 recfm=fb
ftp> binary
ftp> put samplib.xmt
ftp> put tssexec.xmt
ftp> quote site pri=30 sec=5 cyl
ftp> put idmload.xmt
ftp> quit
```

2 Log on to z/OS using the same user ID that you used for the FTP session.

3 Use the TSO `RECEIVE` command to extract the data sets. When `RECEIVE` prompts you for parameters, specify the appropriate data set names and volumes according to your standards.

Place these data sets on a disk volume that is shared by the systems that share the security system database.

```
READY
receive indataset(samplib.xmt)
INMR901I Dataset IDM.SAMPLIB from ADMIN on SYSB
INMR906A Enter restore parameters or 'DELETE' or 'END' +
dsname('sys3.ts.samplib') volume(work0a)
. . . many IEBCOPY messages . . .
INMR001I Restore successful to dataset 'SYS3.TS.SAMPLIB'
READY
receive indataset(idmload.xmt)
INMR901I Dataset IDM.LOAD from ADMIN on SYSB
INMR906A Enter restore parameters or 'DELETE' or 'END' +
dsname('sys3.ts.load') volume(work0a)
. . . many IEBCOPY messages . . .
INMR001I Restore successful to dataset 'SYS3.TS.LOAD'
READY
receive indataset(tssexec.xmt)
INMR901I Dataset IDM.EXECLIB from ADMIN on SYSB
INMR906A Enter restore parameters or 'DELETE' or 'END' +
dsname('sys3.ts.execlib') volume(work0a)
. . . many IEBCOPY messages . . .
INMR001I Restore successful to dataset 'SYS3.TS.EXECLIB'
READY
```

4 Add the driver load library to the APF list.

Use the `PARMLIB IEAAPFxx` or `PROGxx` member as appropriate. If you use the dynamic APF facility, you can use the `SET PROG` command to activate your changes. Otherwise, you must IPL for the change to take effect.

5 Restrict access to the driver load library.

WARNING: Do not put the driver load library in the linklist unless you use program protection to secure its contents against unauthorized use. Failure to protect the driver load library introduces security exposures.

---

6 Customize the JOB card and run the job in the samples library member `HFSINST`.

This creates the HFS file system structure for the driver.

Authorizing the Driver TSO Commands

`LDXSERV`, `SAFQUERY` and `TSSLPBK` require APF authorization. They reside in the driver load library, which you added to the APF list in . You must also add them to the list of authorized TSO commands.

1 Add `LDXSERV`, `SAFQUERY` and `TSSLPBK` to the `AUTHCMD NAMES(...)` statement in member `IKJTSOxx` of `SYS1.PARMLIB` or its equivalent.

Example 3-1 *Example:*

```
AUTHCMD NAMES(...
  other commands...
  LDXSERV SAFQUERY TSSLPBK)
```

2 Use the `PARMLIB` TSO command to activate your changes.

Example 3-2 *Example:*

```
PARMLIB CHECK(00)
PARMLIB UPDATE(00)
```

For more information about the `PARMLIB` command, see the *TSO/E System Programming Command Reference* for your system.

Securing the Driver Shim with SSL

1 Run the REXX exec in the REXX exec library member `SETCERT`.

2 When prompted, enter the Metadirectory server host name or IP address and secure LDAP port number (default is 636).

3 When prompted, enter `Y` to accept the certificate authority presented.

```
You are about to connect to the eDirectory LDAP server to retrieve
the eDirectory Tree Trusted Root public certificate.

Enter the LDAP Server Host Address [localhost]: sr.digitalairlines.com
Enter the LDAP Server Port [636]:

Certificate Authority:
   Subject:        ou=Organizational CA,o=TREENAME
   Not Before:     20160821144845Z
   Not After:      20260821144845Z
Do you accept the Certificate Authority? (Y/N) y
```

Configuring the Remote Loader and Driver Object Passwords

Run the REXX exec in the driver REXX exec library member `SETPWDS`, and respond to the prompts.

Use the same passwords that you used in when setting up the driver on the Metadirectory server.

Allocating and Initializing the Change Log Data Set

The change log data set is a standard z/OS direct access data set. The change log data set must reside on a shared device unless it is used by only a single system.

Create one change log data set. It is shared by each z/OS system that shares the security system database. The log file utility `LDXUTIL` is used to initialize the change log data set. The change log data set must be initialized before you start the driver shim started task for the first time.

To allocate and initialize the change log data set:

1 Customize the samples library member `LOGINIT`.

Update the JCL to conform to your local installation requirements, and specify the following:

- ◆ The name of your driver load library.
- ◆ A name for your change log data set.
- ◆ The shared disk volume where the change log is to be allocated. Specify a different unit name if appropriate.

2 Run the LOGINIT job.

An IEC031I D37 message is normal and should be ignored.

3 Ensure that your change log data set is protected appropriately for the sensitive nature of its contents.

WARNING: If you initialize a change log data set that contains data, the data is lost.

Setting Up the Started Tasks

Preparing User IDs for the Started Tasks

You can use any name for the user IDs.

1 Create the administrative user ID for the change log started task by entering the following single command line:

```
TSS CREATE(LDXLOGR) TYPE(USER) NAME('CHANGE LOG ACID')
  DEPARTMENT(deptname)
  PASSWORD(NOPW,0) FACILITY(STC)
```

Using `NOPW` creates the user ID without a password. If you assign a password, you will be prompted for it upon starting the change log started task.

The above example uses the ACID `TYPE(USER)` for managing ACIDs in a specific Department. Use an appropriate type based on your intended management scope. For example, use `TYPE(SCA)` to manage ACIDs for any Department or Division.

2 Create the user for the driver shim started task by entering the following single command line:

```
TSS CREATE(TSDRV) TYPE(DCA) NAME('DRIVER SHIM ACID')
  DEPARTMENT(deptname)
  PASSWORD(NOPW,0) FACILITY(STC)
```

In the above example, a Top Secret type DCA is assigned to the user ID, so it can be an administrator type capable of managing ACIDs within a department.

Using `NOPW` creates the user ID without a password. If you assign a password, you will be prompted for it upon starting the change log started task.

3 Assign OMVS attributes to the driver shim ACID, which is required to run the driver shim started task, by entering the following command lines:

```
TSS ADDTO(TSDRV) ID(0) HOME(/) OMVSPGM(/bin/sh) DFLTGRP(OMVSGRP)

TSS MODIFY(OMVSTABS)
```

In this example, `UID(0)` and `DFLTGRP(OMVSGRP)` are used. Any UNIX user ID and group may be assigned here, provided they have read/write access to the HFS directories created in "Securing the Driver Shim with SSL" on page 33 and "Configuring the Remote Loader and Driver Object Passwords" on page 33.

4 Assign necessary administrator privileges to the driver shim ACID by entering the following single command:

```
TSS ADMIN(TSDRV) ACID(ALL) MISC1(ALL) MISC2(ALL) MISC9(ALL)
  DATA(RESOURCE, XAUTH, INSTDATA, CICS, PROFILE, ADMIN, NAMES, ACID,
  PASSWORD, ALL)
```

In the above example, administrator privileges are assigned to LIST/CREATE/DELETE/MODIFY ACIDs and all the data within their scope.

5 Add the user ACIDs to the STC table to assign them to the started tasks by entering the following command lines:

```
TSS ADDTO(STC) PROCNAME(LDXLOGR) ACID(LDXLOGR)
```

```
TSS ADDTO(STC) PROCNAME(TSDRV) ACID(TSDRV)
```

6 Use the include/exclude file to exclude these users from provisioning.

Example 3-3   *Example Include/Exclude File Fragment:*

```
EXCLUDE
   ...
   LDXLOGR
   TSDRV
   ...
ENDEXCLUDE
```

For details about the include/exclude file, see "The Connected System Include/Exclude File" on page 63.

Setting Up the Change Log Started Task

You must install and run the change log started task on each system that shares the security system database.

To install the change log started task:

1 Copy member `LDXLOGR` from the samples library to your started task procedure library (`SYS1.PROCLIB` or its equivalent). You can give the change log started task a different name if necessary.

2 Update the JCL to specify the following:

- ◆ The name of your driver load library
- ◆ The name of your change log data set

3 Add the change log started task to your system startup and shutdown procedures.

For information about starting and stopping the change log started task, see "Starting and Stopping the Change Log Started Task" on page 71.

The change log started task should be started during your system startup procedure before user processing begins. Any events of interest that occur are stored in the memory queue until the change log started task has initialized.

The change log started task should be stopped during your system shutdown procedure after all user processing has ended. Any events of interest that occur after the change log started task shuts down remain in the memory queue and are lost when the system is shut down.

4   Review your Workload Manager definitions to ensure that the change log started task is assigned to a Service Class appropriate for its role.

Setting Up the Driver Shim Started Task

Install and run the driver shim started task on only one system that shares the security system database.

To install the driver shim started task:

1   Copy member `TSDRV` from the samples library to your started task procedure library (`SYS1.PROCLIB` or its equivalent). You can give the driver shim started task a different name if necessary.

2   Update the JCL to specify the following:

- The name of your driver load library

- The name of your driver shim configuration file

  You can use your driver samples library member `DRVCONF` as a model. For details, see "The Driver Shim Configuration File" on page 54.

- The name of your connected system schema file

  You can use your driver samples library member `SCHEMDEF` as a model. For details, see "The Connected System Schema File" on page 61.

- The name of your include/exclude file

  You can use your driver samples library member `INCEXC` as a model. For details, see "The Connected System Include/Exclude File" on page 63.

- The name of your change log data set

- The name of your driver REXX exec library

3   Add the driver shim started task to your system startup and shutdown procedures.

For information about starting and stopping the driver shim started task, see "Starting and Stopping the Driver Shim Started Task" on page 72.

The driver shim started task should be started during your system startup procedure before user processing begins. The driver shim started task should be stopped during your system shutdown procedure after all user processing has ended.

4   Review your Workload Manager definitions to ensure that the driver shim started task is assigned to a Service Class appropriate for its role.

Testing before Installing the Security System Exit

You can use the `LDXSERV` command to test your installation before you install the exit.

1   If it is not already running, start the change log started task.

For details about starting the change log started task, see "Starting and Stopping the Change Log Started Task" on page 71.

2 Issue the following command from a TSO session that has the driver load library included in its STEPLIB concatenation:

```
LDXSERV STATUS
```

Examine the output of the command. You should see information about the memory queue, information about the change log started task, and a valid, empty change log data set.

Installing the Driver Security System Exit IDMTSSIX

Follow your normal procedure for applying system-level changes to your z/OS system. We recommend that you do the following:

- Install and test the exit on a test system or partition first.
- Make a copy of applicable libraries before applying any changes.
- Plan a back off procedure.

This exit uses the Top Secret Recovery File Exit to capture TSS commands. There are three different procedures for installing the driver exit module IDMTSSIX into the Top Secret installation exit TSSINSTX. Use the following table to select the procedure to use based on your Top Secret version and your current use of TSSINSTX.

Table 3-1 *Exit Installation Procedure Choices*

Top Secret Version	Your Use of TSSINSTX	Installation Procedure to Use
Version 12	Not used	"Exit Installation Procedure 1" on page 37
Version 12	Using TSSINSTX, but not using either the security file change or password functions	"Exit Installation Procedure 2" on page 38
Any version supported by the driver other than version 12	Not using TSSINSTX, or using TSSINSTX but not using either the security file change or password functions	"Exit Installation Procedure 2" on page 38
Any version supported by the driver	Already using the security file change or password functions of TSSINSTX	"Exit Installation Procedure 3" on page 39

Exit Installation Procedure 1

1 Allocate and reformat a Recovery File for Top Secret, using `TSSMAINT`. It will be used in a later step.

2 Use `IEBCOPY` to copy member `TSSINSTX` from the driver load library to your TSS load library.

This member was built based on the sample provided in the `TSSOPMAT` library for CA Top Secret version 12.

3 If your TSS load library is in the z/OS linklist, refresh `LLA` with the following operator command:

```
F LLA,REFRESH
```

4 Activate the exit using the following operator command:

```
F TSS,EXIT(ON)
```

5 Add the following statements to your Top Secret control options parameter file if they are not already used or specified on the TSS JCL:

```
EXIT(ON)
RECOVER(ON)
RECFILE(dataset-name)
```

Exit Installation Procedure 2

1 If you are not already using a Top Secret Recovery File, allocate and format a Recovery File for Top Secret, using `TSSMAINT`.

2 Add the following statements to your modified `TSSINSTX` source at both the `CHANGE` and `PASSWORD` labels:

```
        GETMAIN R,LV=72            Get standard savearea
        LR   R11,R13              Save original R13
        LR   R13,R1               New savearea addr into R13
        LR   R1,R9                Copy parmlist base to R1
        L    R15,=V(IDMTSSIX)     Get addr of IDM module
        BALR R14,R15              Call it
        LR   R1,R13               Copy temp savearea ptr to R1
        LR   R13,R11              Restore R13
        FREEMAIN R,LV=72,A=(1)    Get rid of savearea
        B    EXIT0
```

These statements are in the driver samples library member `TSSINSTX`.

3 In the `TSSINSTX` function matrix (label `MATRIX` near the beginning of the source module), set the following two entries to `#####YES`:

```
(32) New Password Verification
(48) Security File Change
```

You can use the MATRIX table in driver samples library member `TSSINSTX` as an example.

4 Assemble and link `TSSINSTX` to replace your existing `TSSINSTX` module. Add the following statements to the link step:

```
//SYSLIB  DD  DISP=SHR,DSN=<driver load library>
//SYSLIN  DD  DISP=OLD,DSN=<TSSINSTX object from ASM step>
//        DD  *
  INCLUDE SYSLIB(IDMTSSIX)
  ENTRY TSSINSTX
  NAME TSSINSTX(R)
```

5 If your TSS load library is in the z/OS linklist, refresh LLA with the following operator command:

```
F LLA,REFRESH
```

6 Activate the exit using the following operator command:

```
F TSS,EXIT(ON)
```

7 Add the following statements to your Top Secret control options parameter file if they are not already used or specified on the TSS JCL:

```
EXIT(ON)
RECOVER(ON)
RECFILE(dataset-name)
```

Exit Installation Procedure 3

1 If you are not already using a Top Secret Recovery File, allocate and reformat a Recovery File for Top Secret, using `TSSMAINT`. It will be used in a later step.

2 Determine the calling sequence for your functions and the driver module IDMTSSIX.

  ◆ The driver exit functions never fail a request, and they expect the current request to succeed.

  ◆ If your functions might reject a request, call them before IDMTSSIX.

  ◆ Do not call IDMTSSIX for a request that your exit functions reject.

  ◆ If your exit functions never reject a request, it does not matter whether IDMTSSIX is called before or after your functions.

3 Add the following statements to your modified `TSSINSTX` source in both the `CHANGE` and `PASSWORD` functions:

```
         GETMAIN R,LV=72              Get standard savearea
         LR    R11,R13               Save original R13
         LR    R13,R1                New savearea addr into R13
         LR    R1,R9                 Copy parmlist base to R1
         L     R15,=V(IDMTSSIX)      Get addr of IDM module
         BALR  R14,R15               Call it
         LR    R1,R13                Copy temp savearea ptr to R1
         LR    R13,R11               Restore R13
         FREEMAIN R,LV=72,A=(1)      Get rid of savearea
         B     EXIT0
```

These statements are in the driver samples library member `TSSINSTX`.

4 In the `TSSINSTX` function matrix (label `MATRIX` near the beginning of the source module), set the following two entries to #####YES:

```
(32) New Password Verification
(48) Security File Change
```

You can use the MATRIX table in driver samples library member `TSSINSTX` as an example.

5 Assemble and link `TSSINSTX` to replace your existing `TSSINSTX` module. Add the following statements to the link step:

```
//SYSLIB  DD  DISP=SHR,DSN=<driver load library>
//SYSLIN  DD  DISP=OLD,DSN=<TSSINSTX object from ASM step>
//        DD  *
  INCLUDE SYSLIB(IDMTSSIX)
  ENTRY TSSINSTX
  NAME TSSINSTX(R)
```

6 If your TSS load library is in the z/OS linklist, refresh LLA with the following operator command:

```
F LLA,REFRESH
```

7 Activate the exit using the following operator command:

```
F TSS,EXIT(ON)
```

8 Add the following statements to your Top Secret control options parameter file, if they are not already used or specified on the TSS JCL:

```
EXIT(ON)
RECOVER(ON)
RECFILE(dataset-name)
```

Testing the Completed Connected System Installation

1 If it is not already running, start the change log started task.

For details about starting the change log started task, see "Starting and Stopping the Change Log Started Task" on page 71.

2 Perform some actions to exercise the security system exit routines and create some sample events.

    2a Change a password using the logon screen.

    2b Create new user ID.

3 Issue the following command from a TSO session that has the driver load library included in its STEPLIB concatenation:

```
LDXSERV STATUS
```

Examine the output of the command. You should see the exit routines loaded, information about the memory queue, information about the change log started task, and a valid, non-empty change log data set.

Post-Installation Tasks

1 If desired, set **Startup Option** on the Driver Configuration page to **Auto start**. This causes the driver to start when the Metadirectory engine starts.

2 Activate the driver.

Identity Manager and Identity Manager drivers must be activated within 90 days of installation or they shut down. At any time during the 90 days, or afterward, you can activate Identity Manager products.

For details about activating NetIQ Identity Manager Products, see the *Identity Manager 4.8 Installation Guide* on the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).

Uninstalling the Driver

- ◆ "Uninstalling the Security System Exit" on page 41
- ◆ "Uninstalling the Driver Shim" on page 41
- ◆ "Uninstalling the Driver Object from eDirectory" on page 42

Uninstalling the Security System Exit

If you were not using the TSSINSTX installation exit before installing the driver, follow "Exit Uninstallation Procedure 1" on page 41.

If you were using the TSSINSTX installation exit before installing the driver, follow "Exit Uninstallation Procedure 2" on page 41.

Exit Uninstallation Procedure 1

1 Deactivate the exit using the following operator command:

   `F TSS,EXIT(OFF)`

2 Replace the `EXIT(ON)` statement to your Top Secret control options parameter file with `EXIT(OFF)`.

Exit Uninstallation Procedure 2

1 Replace member `TXXINSTX` in your TSS load library with a backup copy you made before installing the driver.

2 If your TSS load library is in the z/OS linklist, refresh LLA with the following operator command:

   `F LLA,REFRESH`

3 Deactivate the exit using the following operator command:

   `F TSS,EXIT(OFF)`

4 Reactivate the exit using the following operator command:

   `F TSS,EXIT(ON)`

Uninstalling the Driver Shim

1 Remove the change log started task and driver shim started task from your system startup and shutdown procedures.

2 Stop the change log started task and driver shim started task.

   For details, see "Starting and Stopping the Change Log Started Task" on page 71 and "Starting and Stopping the Driver Shim Started Task" on page 72.

3 Remove members `LDXLOGR` and `TSDRV` from your started task procedure library.

4 Delete the users you created for the started tasks in "Preparing User IDs for the Started Tasks" on page 34.

5 Remove the driver load library from your APF list.

Reverse the action you took in Step 4 on page 32.

6 Remove the `LDXSERV` and `SAFQUERY` commands from IKJTSO*xx*.

Reverse the actions you took in "Authorizing the Driver TSO Commands" on page 32.

7 Remove the driver files from the HFS. They were created in Step 6 on page 32.

```
rm -Rf /opt/novell
rmdir -p /opt/novell/
```

8 Delete the driver samples library, load library, and REXX exec library that you created in Step 3 on page 31.

9 Delete the change log data set that you created in "Allocating and Initializing the Change Log Data Set" on page 33.

Uninstalling the Driver Object from eDirectory

1 In iManager, select **Identity Manager Overview** from the Identity Manager task list on the left side of the window.

2 Navigate to your driver set by searching the tree or by entering its name.

3 Click **Delete Driver** on the Identity Manager Overview page.

4 Select the Driver object to be deleted, then click **OK**.

4 Upgrading from the Fan-Out Driver

This section provides the information you need if you are upgrading from the Identity Manager Fan-Out driver to the Identity Manager 4.8 driver for CA Top Secret.

Topics include

- "Preparing for Migration" on page 43
- "Migrating Fan-Out Driver Platform Services to the Top Secret Driver" on page 44
- "Configuring the Driver" on page 44
- "Post-Migration Tasks" on page 44

The Fan-Out driver provides one-way synchronization to a heterogeneous mix of systems including Linux and UNIX systems, and IBM i5/OS* (OS/400* operating system) and z/OS systems. The Fan-Out driver also provides authentication redirection from those systems.

Moving to the Identity Manager 4.8 driver for CA Top Secret provides two main advantages.

- **Bidirectional Synchronization:** The driver allows synchronization from the connected system.
- **Standard Identity Manager Policies That Simplify Customization:** The Fan-Out driver makes minimal use of Identity Manager policies.

Consider the following before migrating from the Fan-Out driver.

- **Heterogeneity:** The Fan-Out driver supports operating system environments besides Top Secret. You can continue to use the Fan-Out driver for those systems while using the Identity Manager 4.8 driver for CA Top Secret on your Top Secret systems.
- **Authentication Redirection:** The Fan-Out driver provides authentication redirection using the password exit. The Identity Manager 4.8 driver for CA Top Secret provides bidirectional password synchronization.

Preparing for Migration

We recommend that you perform the upgrade in a test environment similar to your production environment before upgrading production systems.

Before beginning the upgrade process, review Chapter 3, "Installing the Top Secret Driver," on page 23.

To prepare for installing the upgrade:

1 Verify that you have the required knowledge and skills.

For details, see "Required Knowledge and Skills" on page 23.

2 Ensure that the prerequisites are met.

For details, see "Prerequisites" on page 23.

3 Prepare the distribution files for installation.

For details, see "Getting the Installation Files" on page 24.

4 If necessary, migrate the UID and GID numbers from the appropriate Fan-Out driver Platform Set to the OMVS attributes of the Identity Vault users or groups. You can assign OMVS attributes, such as HOME (home directory) and OMVSPGM (login shell), to objects in the Identity Vault.

Migrating Fan-Out Driver Platform Services to the Top Secret Driver

Perform the following steps on your target platform system:

1 Stop the following started tasks:
    ◆ PLATRCVR
    ◆ ASCLIENT

2 Remove ASCLIENT and PLATRCVR from your system startup and shutdown procedures.

3 Remove the Fan-Out driver Top Secret exit.

4 Install the driver shim on the connected system.

   For details, see "Installing the Driver Shim on the Connected System" on page 30.

Configuring the Driver

1 Install and set up the Identity Manager Driver for Top Secret on the Metadirectory server.

   For details, see "Creating the Driver in Designer" on page 24.

2 Make any required policy modifications.

   Create or modify an appropriate policy to use the alternative naming attribute if one was used by the Fan-Out driver. For more information about policy customization, see the policy documentation on the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).

3 Start the driver.

   Click the upper right corner of the driver icon, then click **Start driver**.

4 Migrate the users to make new associations. For details, see "Migrating Identities from the Identity Vault to the Connected System" on page 56 and "Migrating Identities from the Connected System to the Identity Vault" on page 57.

Post-Migration Tasks

Perform the steps listed in "Post-Installation Tasks" on page 40.

After the new driver is operating properly, you can remove the Fan-Out driver components.

1 Delete the Platform object from the Fan-Out driver configuration.

2 On the connected system, uninstall Platform Services.

3 If this is the last platform being served by the Fan-Out driver, you can uninstall the Fan-Out core driver.

   3a Remove the `ASAM` directory from the file system.

   3b Remove the ASAM System container object and all of its subordinates from the tree.

   3c Uninstall the Fan-Out driver plug-ins.

5 Configuring the Top Secret Driver

After you have installed the Identity Manager 4.8 driver for CA Top Secret, use the information in this section for configuration. Topics include

- "Driver Parameters and Global Configuration Values" on page 47
- "The Driver Shim Configuration File" on page 54
- "Setting the Remote Loader and Driver Object Passwords" on page 55
- "Migrating Identities" on page 56
- "International Considerations" on page 58

Driver Parameters and Global Configuration Values

You can control the operation of the driver by modifying the properties described in the following sections.

IMPORTANT: Changing these values requires a restart of the driver.

- "Driver Configuration Page" on page 48
- "Global Configuration Values Page" on page 50

Driver properties and Global Configuration Values are first created during the deployment of Designer Top Secret Driver packages. Driver properties and Global Configuration Values are first created during the deployment of Designer Top Secret Driver packages. For details, see "Creating the Driver in Designer" on page 24.

To edit the properties shown on the Driver Configuration page and the Global Configuration Values page:

1 In iManager, select **Identity Manager Overview** from the Identity Manager task list on the left side of the window.

2 Navigate to your driver set by searching the tree or by entering its name.

3 Click the driver to open its overview.

4 Click the driver icon.

5 Select **Driver Configuration** or **Global Config Values** as appropriate.

6 Edit the property values as desired, then click **OK**.

Driver Configuration Page

Table 5-1 *Driver Configuration Page*

Property Name	Values or Format
Driver Module	Connect to Remote Loader must be selected
Driver Object Password	Text value
Authentication ID	Not used
Authentication Context	Not used
Remote Loader Connection Parameters	Host name or IP address and port number of the driver shim on the connected system, and the RDN of the object with the server certificate
Driver Cache Limit	The recommended value is 0 (zero)
Application Password	Not used
Remote Loader Password	Text value
Startup Option	Auto start Manual
Automatic Loopback Detection	Yes No
Create Home Directories	Yes No
User Catalog Alias	Catalog data set name
Group Catalog Alias	Catalog data set name
Polling Interval	Number of seconds
Heartbeat Interval	Number of seconds
Publisher Disabled	Yes No

Driver Object Password

The Driver object password is used by the driver shim (embedded Remote Loader) to authenticate itself to the Metadirectory engine. This must be the same password that is specified as the Driver object password on the connected system driver shim.

Remote Loader Connection Parameters

The Remote Loader Connection Parameters option specifies information that the driver uses for Secure Sockets Layer (SSL) communication with the connected system.

Table 5-2 *Remote Loader Connection Parameters*

Parameter	Description
`host=hostName`	Connected system host name or IP address.
`port=portNumber`	Connected system TCP port number. The default is 8090.
`kmo=objectRDN`	The RDN of the object with the server certificate signed by the tree's certificate authority. Enclose the RDN in double quotes (") if the name contains spaces.

The following is an example Remote Loader connection parameter string:

```
hostname=192.168.17.41 port=8090 kmo="SSL CertificateDNS"
```

Remote Loader Password

The Remote Loader password is used to control access to the driver shim (embedded Remote Loader). This must be the same password that is specified as the Remote Loader password on the connected system driver shim.

Automatic Loopback Detection

Specifies whether the driver shim discards events that would cause loopback conditions. This function supplements the loopback detection provided by the Metadirectory engine.

Create Home Directories

Specifies whether the driver automatically creates home directories in the hierarchical file system when users are created.

User Catalog Alias

Specifies the data set name of the catalog used for new users created by the driver.

If you specify a value for **User Catalog Alias**, the REXX exec to add a new user issues the following command:

```
DEFINE ALIAS(NAME('user') RELATE('UserCatalogAlias'))
```

Group Catalog Alias

Specifies the data set name of the catalog used for new groups created by the driver.

If you specify a value for **Group Catalog Alias**, the REXX exec to add a new group issues the following command:

```
DEFINE ALIAS(NAME('group') RELATE('GroupCatalogAlias'))
```

Polling Interval

Specifies the number of seconds that the Publisher shim waits after running the polling exec and sending events from the change log to the Metadirectory engine. The default interval is 60 seconds.

Publisher Disabled

Specifies whether the Publisher shim is active.

Select **Yes** if you are using Identity Vault to Application (one-way) data flow. This saves processing time.

Heartbeat Interval

Specifies how often, in seconds, the driver shim contacts the Metadirectory engine to verify connectivity. Specify 0 to disable the heartbeat.

Global Configuration Values Page

Table 5-3   *Global Configuration Values*

Property Name	Values or Format
Connected System or Driver Name	Text value
Create Users With	Text value
User Default Department	Text value
User Default Group	Text value
User Default TSO Account Number	Text value
User Default TSO Proc	Text value
User Default TSO Unit	Text value
UID Assignment	Assign by Top Secret Assign by Identity Vault
UID Range	Numeric range
GID Range	Numeric range
Default Home Directory	Text value
Default Program	Text value
The Top Secret Connected System Accepts Passwords from the Identity Vault	Yes No
The Identity Vault Accepts Passwords from the Top Secret Connected System	Yes No
The Identity Vault Accepts Administrative Password Resets from the Top Secret Connected System	Yes No

Property Name	Values or Format
Publish Passwords to NDS Password	Yes No
Publish Passwords to Distribution Password	Yes No
Require Password Policy Validation before Publishing Passwords	Yes No
Reset User's External System Password to the Identity Manager Password on Failure	Yes No
Notify the User of Password Synchronization Failure via E-Mail	Yes No
User Base Container	Identity Vault Container object
Group Base Container	Identity Vault Container object

To view and edit Password Management GCVs, select **Show** for **Show Password Management Policy**.

To view and edit User and Group Placement GCVs, select **Show** for **Show User and Group Placements**.

Connected System or Driver Name

Specifies the name of the driver. This value is used by the e-mail notification templates.

Create Users With

Specifies the ACID of a user to be used as a model for creating new users.

User Default Department

Specifies the default department for new users.

User Default Group

Specifies the default group for new users.

User Default TSO Account Number

Specifies the default account number for new users.

User Default TSO Proc

Specifies the default cataloged procedure name for new users. For example, `IKJACCNT`.

User Default TSO Unit

Specifies the default unit name for new users. For example, `SYSALLDA`.

UID Assignment

Specifies how UID and GID numbers are assigned to new users and groups. Select **Assign by Top Secret** or **Assign by Identity Vault**.

UID Range

Specifies a range of numbers used when Top Secret assigns UID numbers for new users. The REXX exec to add a new user uses this value with the `RANGE` keyword on the `TSS` command. Use a pair of values separated by a comma, similar to the following:

`10000,200000`

GID Range

Specifies a range of numbers used when Top Secret assigns GID numbers for new groups. The REXX exec to add a new group uses this value with the `RANGE` keyword on the `TSS` command. Use a pair of values separated by a comma, similar to the following:

`10000,200000`

Default Home Directory

Specifies the default OMVS home directory path for new users. Include the ending slash (`/`) in the directory path. The user's ACID is appended to the value that you specify. Use a value similar to the following:

`/home/`

In this example, the home directory that is assigned by the driver for a user whose ACID is `IBMUSER` is `/home/IBMUSER`.

Default Program

Specifies the default OMVS program (login shell). Use a value similar to the following:

`/bin/sh`

The Top Secret Connected System Accepts Passwords from the Identity Vault

Specifies whether the driver allows passwords to flow from the Identity Vault to the connected system.

The Identity Vault Accepts Passwords from the Top Secret Connected System

Specifies whether the driver allows passwords to flow from the connected system to the Identity Vault.

The Identity Vault Accepts Administrative Password Resets from the Top Secret Connected System

Specifies whether the driver allows passwords to be reset from the connected system in the Identity Vault. An administrative user can use the `TSS REPLACE` command to set another user's password.

Publish Passwords to NDS Password

Specifies whether the driver uses passwords from the connected system to set NDS® passwords in the Identity Vault. NDS passwords in the Identity Vault are not bidirectional and cannot be synchronized to another system.

Publish Passwords to Distribution Password

Specifies whether the driver uses passwords from the connected system to set NMAS™ Distribution Passwords, which are used for Identity Manager password synchronization.

Require Password Policy Validation before Publishing Passwords

Specifies whether the driver applies NMAS password policies to published passwords. If so, a password is not written to the Identity Vault if it does not conform.

Reset User's External System Password to the Identity Manager Password on Failure

Specifies whether, on a publish Distribution Password failure, the driver attempts to reset the password on the connected system using the Distribution Password from the Identity Vault.

Notify the User of Password Synchronization Failure via E-Mail

Specifies whether the driver sends an e-mail to a user if the password cannot be synchronized.

User Base Container

Specifies the base container object in the Identity Vault for user synchronization. This container is used in the Subscriber channel Event Transformation policy to limit the Identity Vault objects being synchronized. This container is used in the Publisher channel Placement policy as the destination for adding objects to the Identity Vault. Use a value similar to the following:

`users.myorg`

Group Base Container

Specifies the base container object in the Identity Vault for group synchronization. This container is used in the Subscriber channel Event Transformation policy to limit the Identity Vault objects being synchronized. This container is used in the Publisher channel Placement policy as the destination when adding objects to the Identity Vault. Use a value similar to the following:

```
groups.myorg
```

The Driver Shim Configuration File

The driver shim configuration file controls operation of the driver shim. You can specify the configuration options listed in Table 5-4, one per line. You can also specify these options on the command line. For details about driver shim command line values, see "Driver Shim Command Line Options" on page 109.

The driver shim configuration file must be a sequential file or a member of a partitioned data set. The DRVCONF DD statement in the driver shim started task JCL identifies the driver shim configuration file. An example driver shim configuration file is provided in the driver samples library member DRVCONF.

Table 5-4 *Driver Shim Configuration File Statements*

Option (Short and Long Forms)	Description

`-conn <connString>`
 	
`-connection <connString>`
	A string with connection options. Enclose the string in double quotes ("). If you specify more than one option, separate the options with spaces.
	
	`port=<driverShimPort>`
	
	`ca=<Certificate Authority Key File>`

`-hp <httpPort>`
 	
`-httpport <httpPort>`
	Specifies the HTTP services port number. The default HTTP services port number is 8091.
	
	You can connect to this port to view log files. For details, see "The Trace File" on page 77 and "The Status Log" on page 78.

`-path <driverPath>`	Specifies the path for driver files. The default path is `/opt/novell/tsdrv`.

`-sp <RLpassword>,<DOpassword>,`
 	
`-setpassword <RLpassword>,<DOpassword>,`
	Sets the Remote Loader and Driver object passwords.

`-t <traceLevel>`
 	
`-trace <traceLevel>`
	Sets the level of debug tracing. 0 is no tracing, and 10 is all tracing. For details, see "The Trace File" on page 77.
	
	The output file location is specified by the `tracefile` option.

Option (Short and Long Forms)	Description

`-tf <fileName>`
 	
`-tracefile <fileName>`
	Sets the trace file location.
	
	The default is `/opt/novell/tsdrv/logs/trace.log`.

`-tfm size`
 	
`-tracefilemax size`
	Specifies the limit to the size of the trace file for this instance. Specify the value in kilobytes, megabytes, or gigabytes, using the abbreviation for the byte type. The minimum value is 100K. For example:
	
	◆ `-tracefilemax 1000K`
	
	◆ `-tracefilemax 100M`
	
	◆ `-tracefilemax 10G`
	
	NOTE:
	
	◆ When you add this option to the configuration file, the application uses the specified name for the tracefile and includes up to 9 "roll-over" files. Each file size is 1/10th of the total size specified. The roll-over files are named using the base of the main trace filename plus `_n`, where `n` is 1 through 9.
	
	◆ If the trace file data is larger than the specified maximum when the Driver Shim is started, the trace file data remains larger than the specified maximum until roll-over is completed through all 10 files.

Example Driver Shim Configuration File

```
-tracefile /opt/novell/tsdrv/logs/trace.log
-tracefilemax 100M
-trace 3
-connection "ca=/opt/novell/tsdrv/keys/ca.pem"
-path /opt/novell/tsdrv/
```

Setting the Remote Loader and Driver Object Passwords

The Remote Loader password is used by the Metadirectory engine to authenticate itself to the driver shim (embedded Remote Loader). The Driver object password is used by the driver shim to authenticate itself to the Metadirectory engine.

These passwords are set during installation. You can set them at any time later using the procedures in the following sections. The corresponding passwords you set on the connected system and in the Identity vault must be identical.

◆ "Connected System" on page 56
◆ "Identity Vault" on page 56

Connected System

The Remote Loader and Driver object passwords are stored on the connected system under `/opt/novell/tsdrv/keys` in encrypted files `dpwdlf40` (Driver object password) and `lpwdlf40` (Remote Loader password).

To set the passwords on the connected system:

1 Run the REXX exec in the REXX exec library member `SETPWDS` and respond to the prompts.

2 Restart the driver shim started task.

Identity Vault

The Remote Loader and Driver object passwords are set for the driver through iManager and are stored in the Identity Vault. Each password on the connected system must exactly match its counterpart in the Identity vault.

To change the passwords in the Identity Vault after driver installation:

1 In iManager, navigate to the Driver Overview for the driver.

2 Click the driver icon.

3 Specify the Driver object password.

4 Specify the Remote Loader password.

   The Remote Loader password follows the Authentication heading.

5 Click **Apply**.

6 Restart the driver.

Migrating Identities

When you first run the driver, you might have identities in the Identity Vault that you want to provision to the connected system, or vice versa. Identity Manager provides a built-in migration feature to help you accomplish this.

Migrating Identities from the Identity Vault to the Connected System

1 In iManager, open the Identity Manager Driver Overview for the driver.

2 Click **Migrate from Identity Vault**. An empty list of objects to migrate is displayed.

3 Click **Add**. A browse and search dialog box that allows you to select objects is displayed.

4 Select the objects you want to migrate, then click **OK**.

To view the results of the migration, click **View the Driver Status Log**. For details about the log, see "The Status Log" on page 78.

If a user has a Distribution Password, the Distribution Password is migrated to the connected system as the user's password. Otherwise, no password is migrated. For information about Universal Passwords and Distribution Passwords, see the *Password Management Administration Guide* (https://www.netiq.com/documentation/password_management33/).

Migrating Identities from the Connected System to the Identity Vault

1 In iManager, open the Identity Manager Driver Overview for the driver.

2 Click **Migrate into Identity Vault** to display the Migrate Data into the Identity Vault window.

3 Specify your search criteria:

   3a To view the list of eDirectory™ classes and attributes, click **Edit List**.

   3b Select class User or class Group.

   > **IMPORTANT:** Identity Manager imports objects by class in the order specified in the list. Migrate users before you migrate groups so that the users can be added to the newly created groups.

   3c Select the attributes to be used as search criteria for objects of the selected class, then click **OK**.

   The eDirectory attributes map to Top Secret attributes as specified by the driver schema: CN maps to ACID, etc. For the default mappings, see Table 1-1, "Default Filter and Schema Mapping," on page 17.

   3d Specify values for the selected attributes, then click **OK**.

   The values can include basic regular expressions. For details about basic regular expressions, use the OMVS `man grep` command.

4 Click **OK**.

To view the results of the migration, click **View the Driver Status Log**. For details about the log, see "The Status Log" on page 78.

Because local passwords cannot be retrieved from Top Secret, they cannot be submitted to the Metadirectory engine until they are changed. The password change exit routine captures password changes.

Synchronizing the Driver

To generate events for associated objects that have changed since the driver's last processing, open the Identity Manager Driver Overview page for the driver in iManager, then click **Synchronize**.

International Considerations

The Identity Manager driver for Top Secret assumes the Top Secret system is using the EBCDIC Latin 1 (Open Systems) codepage for translating data to and from the Identity Manager Metadirectory engine. This codepage, IBM-1047, is configured in the `SAMPLIB` member `TSDRV`, the JCL for starting the Top Secret driver shim. If your system uses a codepage other than IBM-1047, you will need to change the JCL for starting the Top Secret driver to ensure correct character translation.

The following line, specified in the `SAMPLIB(TSDRV)` member, illustrates how the environment is configured for the default codepage IBM-1047:

```
//  SET ENV='ENVAR("LC_CTYPE=IBM-1047")'
```

You may change this text to reflect the codepage of your Top Secret system. The format is IBM-*CCSID*, where *CCSID* is a coded character set identifier, represented in decimal.

For a list of valid codepage identifiers, see the IBM CCSID reference table (http://www-01.ibm.com/software/globalization/ccsid/ccsid_registered.html). The following table lists a subset of sample values:

Codepage	Description
IBM-858	IBM-PC
IBM-1047	Latin 1 (Open Systems)
IBM-1140	US/Canada
IBM-1141	Austria/Germany
IBM-1142	Denmark/Norway
IBM-1143	Finland/Sweden
IBM-1144	Italy
IBM-1145	Spain/Spanish Latin America
IBM-1146	UK
IBM-1147	France
IBM-1148	International
IBM-1149	Iceland

6 Customizing the Top Secret Driver

This section provides information about available resources for customizing the Identity Manager 4.8 driver for CA Top Secret.

Topics include

For details about the filters and policies provided with the driver, see "Filter and Schema Mapping" on page 17 and "Policies" on page 17.

The Scriptable Framework

The driver provides a comprehensive scriptable framework that you can use to add to the built-in support for the security system, and to add support for other applications and security system fields that have been customized for a particular installation.

The driver scriptable framework includes components that simplify the job of extending the driver to support new applications and fields.

- Embedded Remote Loader
    - Full SSL support, and an installer to easily configure the certificates
    - Web access to debugging information from the embedded Remote Loader
- Encrypted change log that stores changes from the application to the Identity Vault if there is a communication problem
- Loopback detection system to prevent subscribed events from being published back to the Identity Vault
- z/OS name/token callable services helper programs that provide for securely passing large variables to and from the REXX execs
- Easily extendable connected system schema file to support any application
- Include/exclude file for simplified testing and deployment by the platform administrator
- Event support, both for applications that have exits or callouts, and for applications that must be polled for changes

The names of objects and attributes in the REXX execs are the names specified in the connected system schema file.

The following tables describe the major REXX execs.

Table 6-1 *Identity Vault Command Processing Execs*

REXX Exec	Identity Vault Event
IDMADDG	Add Group
IDMADDU	Add User
IDMCONNU	Add User to Group
IDMDELG	Delete Group
IDMDELU	Delete User
IDMDSABL	Disable User
IDMENABL	Enable User
IDMMODG	Modify Group
IDMMODPW	Password Change
IDMMODU	Modify User
IDMQUERY	Query
IDMRENG	Rename Group
IDMRENU	Rename User
IDMRMVU	Remove User from Group

Table 6-2 *Other Execs*

REXX Exec	Purpose
IDMSUB	Calls the appropriate command processing exec based on the type of event and object. This is executed for every Subscriber event.
IDMPOLL	Not used for CA Top Secret. You can use this exec as needed to support your own applications if they do not generate events when changes are made.
IDMHRTBT	Heartbeat exec.
IDMGLBLS	Holds configurable options that all REXX execs can use during event processing.
IDMSTATS	Sends a status document to report the health of the application.
IDMTSOEX	Executes a TSO command and returns the command return code and command output.
SETPWDS	Sets the Remote Loader and Driver object passwords, which are used to authenticate and authorize the connection between the driver shim started task and the Metadirectory system.

REXX Exec	Purpose
SETCERT	Retrieves the certificate authority for the Metadirectory engine that uses SSL to communicate with the driver shim started task.

The Connected System Schema File

The schema file on the connected system is used to specify the classes and attributes that are available on the system.

The schema file is read by the driver shim when the Metadirectory engine requests it. This typically happens at driver startup. The schema file is also used by the Policy Editor to map the schema of the Identity Vault to the schema of the external application.

If you change the schema file, you must restart the driver shim and the driver.

The REXX execs that are provided with the driver depend on the classes and attributes in the schema file that is provided with the driver.

The connected system schema file must be a sequential file or a member of a partitioned data set. The SCHEMDEF DD statement in the driver shim started task JCL identifies the schema file. An example schema file with the required classes and attributes is provided in the driver samples library member SCHEMDEF.

Schema File Syntax

Each line in the schema file represents an element and must begin with the element name: SCHEMA, CLASS, or ATTRIBUTE.

The first element of the schema file is the schema definition. The schema definition is followed by class definitions. Each class definition can contain attribute definitions.

Except for the values of class and attribute names, the contents of the schema file are case insensitive.

Comments

Lines that begin with an octothorpe (#) are comments.

```
# This is a comment.
```

Schema Definition

The first line in the schema file that is not a comment must be the schema definition.

```
SCHEMA [HIERARCHICAL]
```

HIERARCHICAL specifies that the target application is not a flat set of users and groups, but is organized by hierarchical components, such as a directory-based container object.

Class Definition

CLASS *className* [CONTAINER]

You must specify a class name. Enclose the class name in double quotes (") if it contains spaces.

Add the CONTAINER keyword if objects of this class can contain other objects.

The class definition is ended by another class definition or by the end of the file.

Attribute Definition

Any number of attribute definitions can follow a class definition. Attribute definitions define attributes for the class whose definition they follow.

ATTRIBUTE *attributeName* [*TypeAndProperties*]

An attribute name is required. Enclose the attribute name in double quotes (") if it contains spaces.

If no attribute type is specified, the attribute has the string type. The allowable types are:

- STRING
- INTEGER
- STATE
- DN

The allowable attribute properties are:

- REQUIRED
- NAMING
- MULTIVALUED
- CASESENSITIVE
- READONLY

Example Schema File

For a complete example connected system schema file, see the driver samples library member SCHEMDEF. An excerpt from that file follows.

```
SCHEMA

  CLASS USER

    ATTRIBUTE ACID NAMING REQUIRED
    ATTRIBUTE ACTION MULTIVALUED
    ATTRIBUTE AFTER
    ATTRIBUTE AUDIT STATE
    ATTRIBUTE AUTOUID STATE
    . . .
    ATTRIBUTE XCOMMAND MULTIVALUED
    ATTRIBUTE XSUSPEND STATE
    ATTRIBUTE XTRANSACTIONS

  CLASS GROUP

    ATTRIBUTE ACID NAMING REQUIRED
    ATTRIBUTE AUTOGID STATE
    ATTRIBUTE GID
```

The Connected System Include/Exclude File

You can use an optional include/exclude file on the connected system to control which identities are or are not synchronized from the Identity Vault to the connected system.

To control which objects are synchronized from the connected system to the Identity Vault, use policies. For details about customizing policies, see the policy documentation on the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).

The connected system include/exclude file must be a sequential file or a member of a partitioned data set. The INCEXC DD statement in the driver shim started task JCL identifies the include/exclude file. An example include/exclude file that excludes many common z/OS users and groups, such as JES, OMVS, and INIT, is provided in the driver samples library member INCEXC.

The file is read when the driver shim starts. If you make changes to it, you must restart the driver shim.

The include/exclude file can contain include rules and exclude rules. To ensure optimal performance, each include/exclude file should contain no more than 50 entries total.

You can use the include/exclude file to phase in your deployment of the driver, excluding most users and groups at first, and then adding more as you gain confidence and experience.

Include/Exclude Processing

Identity Vault events for identities that match an exclude rule are discarded by the Subscriber shim.

Included identities are treated normally by the Subscriber shim.

Identities that do not match an include rule or an exclude rule in the file are included.

Identities are matched in the following priority:

1. Exclude rules
2. Include rules

Within each level of this matching priority, identities are matched against rules in the order that the rules appear in the file. The first rule that matches determines whether the identity is included or excluded.

Include/Exclude File Syntax

Except for class names, attribute names, and the values to match, the contents of the include/exclude file are case insensitive.

The Subscriber Creation policy converts object names to uppercase. Use uppercase names in the include/exclude file to match identities.

The include/exclude file can contain any number of include sections, exclude sections, and single-line rules.

Include sections and exclude sections can contain class matching rules, and class matching rules can contain attribute matching rules. Include sections and exclude sections can also contain association matching rules.

Class and attribute names used in the include/exclude file must correspond to the names specified in the schema file. For details about the schema file, see "The Connected System Schema File" on page 61.

Comments

Lines that begin with an octothorpe (#) are comments.

```
# This is a comment.
```

Include and Exclude Sections

Include and exclude sections provide rules to specify which objects are to be included or excluded from synchronization.

An include section begins with an include line and ends with an endinclude line.

```
INCLUDE
   .
   .
   .
ENDINCLUDE
```

An exclude section begins with an exclude line and ends with an endexclude line.

```
EXCLUDE
   .
   .
   .
ENDEXCLUDE
```

You can use class matching rules and association matching rules within an include section and an exclude section.

Class Matching Rules

Use a class matching rule within an include section or an exclude section to specify the name of a class of objects to include or exclude.

A class matching rule is defined by a class line that specifies the name of the class and ends with an endclass line.

```
CLASS className
   .
   .
   .
ENDCLASS
```

You can use attribute matching rules within a class matching rule.

Attribute Matching Rules

You can use attribute matching rules within a class matching rule to limit the objects that are included or excluded. If no attribute matching rules are specified for a class, all objects of the specified class are included or excluded.

An attribute matching rule comprises an attribute name, an equals sign (=), and an expression. The expression can be an exact value, or it can use limited regular expressions. For details about limited regular expressions, see "Limited Regular Expressions" on page 66.

```
attributeName=expression
```

Multiple attribute matching rules can be specified for a given class.

Attribute matching rules within a class matching rule are logically ANDed together. To logically OR attribute matching rules for a class, specify multiple class matching rules. For example, the following include/exclude file excludes both user01 and user02:

```
# Exclude the User object if its ACID is USER01 or USER02.
EXCLUDE
CLASS USER
    ACID=USER01
ENDCLASS
CLASS USER
    ACID=USER02
ENDCLASS
ENDEXCLUDE
```

Association Matching Rules

You can specify association matching rules in an include or exclude section. Association matching rule expressions can specify an exact association or a limited regular expression. For details about limited regular expressions, see "Limited Regular Expressions" on page 66.

By default, an association is the ACID. Association formation can be customized in the Subscriber REXX execs.

For example, to exclude the `root` user, specify

```
EXCLUDE
   ROOT
ENDEXCLUDE
```

Single-Line Rules

INCLUDE|EXCLUDE [*className*] *objectSelection*

Where *objectSelection* can be

{*associationMatch* | *attributeName=expression*}

You must specify whether the rule is to include or exclude the objects it matches.

You can specify a class name to limit matches to only objects of that class.

You must specify either an association or an attribute matching expression. The syntax of the association and attribute matching expression is the same as that of association matching rules and attribute matching rules previously described. For details, see "Association Matching Rules" on page 66 and "Attribute Matching Rules" on page 65.

For example, to ignore events from the Admin user in the Identity Vault:

```
# Do not subscribe to events for the Admin user.
EXCLUDE ADMIN
```

Limited Regular Expressions

A limited regular expression is a pattern used to match a string of characters.

Character matching is case sensitive.

Any literal character matches that character.

A period (`.`) matches any single character.

A bracket expression is a set of characters enclosed by left (`[`) and right (`]`) brackets that matches any listed character. Within a bracket expression, a range expression is a pair of characters separated by a hyphen, and is equivalent to listing all of the characters that sort between the given characters. For example, `[0-9]` matches any single digit.

An asterisk (`*`) indicates that the preceding item is matched zero or more times.

A plus sign (`+`) indicates that the preceding item is matched one or more times.

A question mark (`?`) indicates that the preceding item is matched zero or one times.

You can use parentheses to group multiple expressions into a single item. For example, `(abc)+` matches `abc`, `abcabc`, `abcabcabc`, etc. Nesting of parentheses is not supported.

Example Include/Exclude Files

Example 6-1 *Example 1*

```
# Exclude users whose names start with TEMP
EXCLUDE
    CLASS USER
        ACID=TEMP.*
    ENDCLASS
ENDEXCLUDE
```

Example 6-2 *Example 2*

```
# Exclude USERA and USERB
# Because attribute rules are ANDed, these must be in separate
# CLASS sections.
EXCLUDE
    CLASS USER
        ACID=USERA
    ENDCLASS
    CLASS USER
        ACID=USERB
    ENDCLASS
ENDEXCLUDE
```

Managing Additional Attributes

You can add additional attributes to the driver for both the Publisher and Subscriber channels. These attributes can be accessed by the REXX execs for all event types.

To publish or subscribe to additional attributes, you must add them to the filter and add support for them into the REXX execs.

- ◆ "Modifying the Filter" on page 67
- ◆ "Modifying the Rexx Execs for New Attributes" on page 68
- ◆ "Modifying the Publisher Channel for Additional Top Secret Fields" on page 68

Modifying the Filter

1 On the iManager Driver Overview page for the driver, click the **Filter** icon on either the Publisher or Subscriber channel. It is the same object.

2 In the Filter Edit dialog box, click the class containing the attribute to be added.

3 Click **Add Attribute**, then select the attribute from the list.

4 Select the flow of this attribute for the Publisher and Subscriber channels.

- ◆ **Synchronize:** Changes to this object are reported and automatically synchronized.

* **Ignore:** Changes to this object are not reported and not automatically synchronized.
* **Notify:** Changes to this object are reported, but not automatically synchronized.
* **Reset:** Resets the object value to the value specified by the opposite channel. (You can set this value on either the Publisher or Subscriber channel, but not both.)

5 Click **Apply**.

If you want to map this attribute to an existing attribute in the connected system schema file, modify the Schema Mapping policy for the driver.

For complete details about managing filters and Schema Mapping policies, see the policy documentation on the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).

Modifying the Rexx Execs for New Attributes

In the Subscriber channel, a specific REXX exec is called to take the appropriate action for each type of event. For example, if the additional attribute is required for adds and modifies of users, modify `IDMADDU` and `IDMMODU` to process the additional attribute.

Publishing additional attributes requires that you act on changes made in the source application.

Modifying the Publisher Channel for Additional Top Secret Fields

New field names can be added to the Top Secret Field Descriptor Table (FDT). This table also allows installations to remap existing fields to additional data, supplied by the customer. For more information about the FDT, see your Top Secret documentation.

The Publisher channel makes use of Java classes and XSLT to help you publish new field names for a customized Top Secret installation.

TSO commands, such as `TSS`, are sent from the change log to the Identity Manager Input Transformation Policies as the original command that was entered by the user, including all keywords and operand values that were supplied. A set of helper functions, written in Java, helps you to parse these keywords appropriately with XSLT, through the use of XSLT Java extension calls.

The default Input Transformation is named `TSO (TSS) Input Transformation`, and is the first policy to act on data for the Publisher channel. This policy is implemented as an XSLT script. When invoked by the processor, the script retrieves a parser that is created once per lifetime of the Identity Manager JVM* instance and is unique to this particular driver instance and command instance. The parser registers known keywords, and you can register custom keywords. After being registered, the parser is invoked to parse the command image and produces a node set containing a single XML document that represents the command image. This document can then be transformed into XDS documents for event types processed by the Metadirectory engine.

The following table describes the Java classes and methods.

Table 6-3 *Java Classes and Methods*

Java Class	Java Method	Description
TSOCommandParser	static TSOCommandParser registerParser(String name, String driver)	Static method that creates a new TSOCommandParser object with a uniquely specified name and driver string
TSOCommandParser	static boolean hasParser(String name, String driver)	Static method that returns true if a parser is available or false if one is not
TSOCommandParser	static TSOCommandParser getParser(String name, String driver)	Static method that returns the parser that has been registered previously
TSOCommandParser	void registerCommand(String cmd)	Registers a TSO command with this parser instance
TSOCommandParser	NodeSet parse(String cmd)	Parses the TSO command and returns a NodeSet that can be used for XSLT processing
TSOCommand	TSOCommand(String name)	Constructor method to create a new TSO command with a case-insensitive command string prefix
TSOCommand	void registerKeyword(TSOKeyword keyword)	Registers a keyword to be associated with this TSOCommand instance
TSOCommand	void registerPositional(TSOPositional pos)	Registers a positional parameter to be associated with this TSOCommand instance
TSOKeyword	TSOKeyword(String name)	Constructor method to create a new TSO keyword instance with a case-insensitive keyword string value
TSOKeyword	void registerKeyword(TSOKeyword keyword)	Registers a sub-keyword for this TSOKeyword instance
TSOKeyword	void registerPositional(TSOPositional pos)	Registers a positional parameter to be associated with this TSOCommand instance
TSOKeyword	void setDefaultValue(String v)	Assigns a default value for a keyword that contains a value during parsing
TSOKeyword	void setImpliedName(String n)	Assigns a name to be used to describe the keyword in lieu of its string name
TSOKeyword	void mapValue(String key, String value)	Assigns a mapped value to the key string when found during parsing
TSOKeyword	void sensitive()	Sets this keyword to be case sensitive by using the XML attribute `is-sensitive=true`

Customizing the System Exit, IDMTSSIX

The default exit module, IDMTSSIX, queues events to cross-memory for the logger started task, which, in turn, updates the change log. If the logger is unavailable, or the system cannot process events quickly enough, these events will become backed up into ECSA storage. To prevent memory exhaustion and alert operators of the problem, IDMTSSIX includes these hard-coded features:

  - When the number of events in ECSA reach 1000, the warning message LDX9998E is issued.
  - When the number of events in ECSA reach 10,000, the warning message LDX9999E is issued and events are no longer queued to ECSA.

The above values, 1000 and 10,000, can be changed by applying a ZAP modification to the Exit module, IDMTSSIX. Included in the `SAMPLIB` dataset is the job, ZAPTSSIX. If you wish to modify these values, customize ZAPTSSIX with the location of IDMTSSIX and the new cutoff values and submit.

For calculation reference, an event may be a password-change event or a command-image event. With password-change events, a single event occupies 52 bytes. Single command-image events are variable in length, averaging about 250 bytes.

7 Using the Top Secret Driver

This section provides information about operational tasks commonly used with the Identity Manager 4.8 driver for CA Top Secret.

Topics include

- "Starting and Stopping the Driver" on page 71
- "Starting and Stopping the Change Log Started Task" on page 71
- "Starting and Stopping the Driver Shim Started Task" on page 72
- "Displaying Driver Shim Status" on page 72
- "Changing the Driver Shim Trace Level" on page 72
- "Monitoring Driver Messages" on page 72

Starting and Stopping the Driver

To start the driver:

1 In iManager, navigate to the Driver Overview for the driver.
2 Click the upper right corner of the driver icon.
3 Click **Start driver**.

To stop the driver:

1 In iManager, navigate to the Driver Overview for the driver.
2 Click the upper right corner of the driver icon.
3 Click **Stop driver**.

Starting and Stopping the Change Log Started Task

The change log started task must be run on each system that shares the security system database.

To start the change log started task, issue the following operator command:

```
START LDXLOGR
```

To stop the change log started task, issue the following operator command:

```
STOP LDXLOGR
```

Starting and Stopping the Driver Shim Started Task

The driver shim started task must be run on only one system that shares the security system database.

To start the driver shim started task, issue the following operator command:

```
START TSDRV
```

To stop the driver shim started task, issue the following operator command:

```
STOP TSDRV
```

Displaying Driver Shim Status

To see status and version information for the driver shim, issue the following operator command:

```
MODIFY TSDRV,APPL=STATUS
```

You can use the `LDXSERV` TSO command to display information about the Publisher channel event subsystem. Enter the following TSO command:

```
LDXSERV STATUS
```

To use the `LDXSERV` command, you must include the driver load library in your STEPLIB concatenation.

Changing the Driver Shim Trace Level

To change the trace level setting for the driver shim, issue the following operator command with the desired trace level:

```
MODIFY TSDRV,APPL='CTL(desired_trace_level)'
```

Example 7-1 *For example*

```
MODIFY TSDRV,APPL='CTL(9)'
```

For details about the trace file and trace levels, see "The Trace File" on page 77.

Monitoring Driver Messages

The driver shim started task writes messages to the system console, SYSLOG, and the driver operational log. The driver operational log data set is defined by the DRVLOG DD statement in the TSDRV started task JCL. Monitor driver activity there in the same way you monitor other key system functions. For details about the messages written by the driver, see Appendix B, "System and Error Messages," on page 85.

8 Securing the Top Secret Driver

This section describes best practices for securing the Identity Manager 4.8 driver for CA Top Secret. Topics include

- "Using SSL" on page 73
- "Physical Security" on page 73
- "Network Security" on page 73
- "Auditing" on page 74
- "Driver Security Certificates" on page 74
- "Driver REXX Execs" on page 74
- "The Change Log" on page 75
- "Driver Passwords" on page 75
- "Driver Code" on page 75
- "Administrative Users" on page 75
- "Connected Systems" on page 75

For additional information about Identity Manager security, see the *NetIQ® Identity Manager 4.8 Administration Guide* on the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).

Using SSL

Enable SSL for communication between the Metadirectory engine and the driver shim on the connected system.

If you don't enable SSL, you are sending information, including passwords, in the clear.

Physical Security

Keep your servers in a physically secure location with access by authorized personnel only.

Network Security

Require users outside of the corporate firewall to use a VPN to access corporate data.

Auditing

Track changes to sensitive information. Examine audit logs periodically.

For details about using NetIQ Audit to monitor driver operation, see the NetIQ Audit Documentation Web site (http://www.novell.com/documentation/novellaudit20/index.html).

For details about auditing Top Secret, see your *CA Top Secret Auditor Guide*.

Driver Security Certificates

SSL uses security certificates to control, encrypt, and authenticate communications.

Ensure that the security certificate directory `/opt/novell/tsdrv/keys` is appropriately protected. The installation program sets secure file permissions for this directory.

The Driver Shim and the Identity Manager engine communicate through SSL using a certificate created in the Identity Vault and retrieved by the driver shim during the installation process. For more information on this certificate and how to renew or install third-party certificates, refer to the *Identity Manager Administration Guide*.

The Embedded Remote Loader web interface uses a dynamically generated, self-signed certificate for SSL communication. The details of this certificate are as follows:

Table 8-1   *Security Certificate Details*

Property Name	Values / Parameters
Subject	SSL Server
Issuer	SSL Server
Validity	1 year
Serial Number	0
Key	1024-bit RSA

Renewal of this certificate automatically occurs when the Driver Shim is restarted on the connected platform.

Driver REXX Execs

The driver uses REXX execs to perform updates on the connected system, and to collect changes made there.

Ensure that the driver REXX exec library is appropriately protected.

The Change Log

The change log data set contains information about events on the connected system, including passwords. It is encrypted, but it should be protected against access by unauthorized users.

Ensure that the change log data set is appropriately protected.

Driver Passwords

Use strong passwords for the Driver object and Remote Loader passwords, and restrict knowledge of them to authorized personnel. These passwords are stored in encrypted form in the security certificate directory `/opt/novell/tsdrv/keys`. The installation program sets secure file permissions for this directory.

Driver Code

Ensure that the driver load library is appropriately protected.

Do not put the driver load library in the linklist unless you use program protection to secure its contents against unauthorized use.

Administrative Users

Ensure that accounts with elevated rights on the Metadirectory system, Identity Vault systems, and the connected systems are appropriately secure. Protect administrative user IDs with strong passwords.

Connected Systems

Ensure that connected systems can be trusted with account information, including passwords, for the portion of the tree that is configured as their base containers.

A   Troubleshooting

This section provides information about troubleshooting the Identity Manager 4.8 driver for CA Top Secret. Topics include

Driver Status and Diagnostic Files

There are several log files that you can view to examine driver operation.

The System Log

SYSLOG is used by the driver shim to record urgent, informational, and debug messages. Examining these should be foremost in your troubleshooting efforts. For detailed message documentation, see Appendix B, "System and Error Messages," on page 85.

The Trace File

The default trace file exists on the connected system at `/opt/novell/tsdrv/logs/trace.log`. A large amount of debug information can be written to this file. Use the trace level setting in the driver shim configuration file to control what is written to the file. For details about the driver shim configuration file, see "The Driver Shim Configuration File" on page 54.

Table A-1   *Driver Shim Trace Levels*

Trace Level	Description
0	No debugging.
1–3	Identity Manager messages. Higher trace levels provide more detail.
4	Previous level plus Remote Loader, driver, driver shim, and driver connection messages.

Trace Level	Description
5–7	Previous level plus change log and loopback messages. Higher trace levels provide more detail.
8	Previous level plus driver status log, driver parameters, driver security, driver Web server, driver schema, driver encryption, and driver include/exclude file messages.
9	Previous level plus low-level networking and operating system messages.
10	Previous level plus maximum low-level program details (all options).

The following is an example `the driver shim configuration file` line to set the trace level:

`-trace 9`

To view the trace file:

1 Use a Web browser to access the driver shim at `https://driver-address:8091`. Substitute the DNS name or IP address of your driver for *driver-address*.

2 Authenticate by using any user name and the password that you specified as the Remote Loader password.

3 Click **Trace**.

The REXX Exec Output File

Output from the REXX execs is written to DDNAME `SYSTSPRT` of the driver shim started task. This file captures the standard error output from all execs executed by the driver shim.

DSTRACE

You can view Identity Manager information using the DSTRACE facility on the Metadirectory server. Use iManager to set the tracing level. For example, trace level 2 shows Identity Vault events in XML documents, and trace level 5 shows the results of policy execution. Because a high volume of trace output is produced, we recommend that you capture the trace output to a file. For details about using DSTRACE, see the *NetIQ® Identity Manager 4.8 Administration Guide* on the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).

The Status Log

The status log is a condensed summary of the events that have been recorded on the Subscriber and Publisher channels. This file exists on the connected system at `/opt/novell/tsdrv/logs/dirxml.log`. You can also view the status log in iManager on the Driver Overview page. You can change the log level to specify what types of events to log. For details about using the status log, see the *NetIQ Identity Manager 4.8 Administration Guide* on the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).

To view the status log:

1 Use a Web browser to access the driver shim at `https://driver-address:8091`. Substitute the DNS name or IP address of your driver for *driver-address*.

2 Authenticate by using any user name and the password that you specified as the Remote Loader password.

3 Click **Status**.

The Operational Log

The operational log contains both important and informational messages that indicate the operational status of the driver shim. These messages indicate items that are not urgent enough to warrant operator response, but useful for tracking the progress of the driver. The location of the operational log is specified by the DRVLOG DD statement in the driver shim started task JCL.

Change Log Started Task Message Log

The change log started task writes important and informational messages to DDNAME `SYSPRINT`.

Troubleshooting Common Problems

Driver Shim Installation Failure

Ensure that you use binary mode to FTP the driver samples library, load library, and REXX exec library XMT files to the target system.

Driver Rules Installation Failure

Ensure that you use a version of iManager compatible with your version of Identity Manager.

Schema Update Failure

- Examine the log file at `/var/nds/schema.log`.

- Ensure that you specify the correct parameters (host name, Admin FDN in dotted format, and password).
- Ensure that you have network connectivity to the Metadirectory server.

Driver Certificate Setup Failure

To set up certificates, the driver shim communicates with the Metadirectory server using the LDAP secure port (636).

- Ensure that eDirectory™ is running LDAP with SSL enabled. For details about configuring eDirectory, see the *NetIQ eDirectory Administration Guide*.
- Ensure that the connected system has network connectivity to the Metadirectory server.

You can use the driver REXX exec library member `SETCERT` to configure the certificate at any time.

If you cannot configure SSL using LDAP, you can install the certificate manually.

1 In iManager, browse the Security container to locate your tree's certificate authority (typically named *treeName* CA).

2 Click the certificate authority object.

3 Click **Modify Object**.

4 Select the **Certificates** tab.

5 Click **Public Key Certificate**.

6 Click **Export**.

7 Select **No** to export the certificate without the private key, then click **Next**.

8 Select **Base64 format**, then click **Next**.

9 Click **Save the exported certificate to a file**, then specify a location to save the file.

10 Use FTP or another method to store the file on the connected system as `/opt/novell/tsdrv/keys/ca.pem`.

Driver Start Failure

- Examine the status log and DSTRACE output.
- The driver must be specified as a Remote Loader driver. You can set this option in the iManager Driver Edit Properties window.
- You must activate both Identity Manager and the driver within 90 days. The Driver Set Overview page in iManager shows when Identity Manager requires activation. The Driver Overview page shows when the driver requires activation.

  For details about activating NetIQ Identity Manager Products, see the *Identity Manager 4.8 Installation Guide* on the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).
- Ensure that the driver load library is APF-authorized.

  You can use the `DISPLAY PROG,APF` operator command to display your APF-authorized libraries.

- Ensure that the `LDXSERV` and `SAFQUERY` commands are listed as authorized TSO commands in your active `IKJTSOxx` member.

  You can use the `DISPLAY IKJTSO,AUTHCMD` operator command to display authorized TSO commands.

For more information about troubleshooting Identity Manager engine errors, see the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).

Driver Shim Startup or Communication Failure

- Examine the trace file.
- Ensure that the connected system's operating system and security system versions are supported. For a list of supported operating systems, see "Connected System Requirements" on page 24.
- Apply all maintenance for your operating system and security system.
- Ensure that the Remote Loader and Driver object passwords that you specified while setting up the driver on the Metadirectory server match the passwords stored with the driver shim.

  To update these passwords on the connected system, use the `SETPWDS` REXX exec. The passwords are stored under `/opt/novell/tsdrv/keys` in encrypted files `dpwdlf40` (Driver object password) and `lpwdlf40` (Remote Loader password).

  To update these passwords on the Metadirectory server, use iManager to update the driver configuration. For details, see "Driver Configuration Page" on page 48.
- Ensure that the correct host name and port number of the connected system are specified in the Driver Configuration Remote Loader connection parameters. You can change the port number (default 8090) in the driver shim configuration file.
- Ensure that the user ID that the driver shim started task uses has been set up properly. For details, see "Preparing User IDs for the Started Tasks" on page 34.
- Ensure that only one system in a complex that shares the security system database is running the driver shim started task.

Users or Groups Are Not Provisioned to the Connected System

- Examine the status log, DSTRACE output, trace file, and REXX exec output file.
- To be provisioned, users and groups must be in the appropriate base container. You can view and change the base containers in iManager on the Global Configuration Values page of the Driver Edit Properties window. For more details, see "Global Configuration Values Page" on page 50.
- To provision identities from the Identity Vault to the connected system, the driver Data Flow property must be set to Bidirectional or Identity Vault to Application. To change this value, re-import the driver rules file over your existing driver.
- The user that the driver is security equivalent to must have rights to read information from the base container. For details about the rights required, see Table 2-2, "Base Container Rights Required by the Driver Security-Equivalent User," on page 21.

Users or Groups Are Not Provisioned to the Identity Vault

- Examine the status log, DSTRACE output, and trace file.

- Examine the **User Base Container** and **Group Base Container** GCV values. For more details, see "Global Configuration Values Page" on page 50.

- To provision identities from the connected system to the Identity Vault, the driver Data Flow property must be set to Bidirectional or Application to Identity Vault. To change this value, re-import the driver rules file over your existing driver.

- The user that the driver is security equivalent to must have rights to update the base container. For details about the rights required, see Table 2-2, "Base Container Rights Required by the Driver Security-Equivalent User," on page 21.

- Ensure that the security system exit has been installed, that LLA has been refreshed, and that the exit has been activated. For details, see "Installing the Driver Security System Exit IDMTSSIX" on page 37.

Identity Vault User Passwords Are Not Provisioned to the Connected System

- Examine the status log, DSTRACE output, and REXX exec output file.

- Several password management properties are available in iManager on the Global Configuration Values page of the Driver Edit Properties window. Ensure that the connected system accepts passwords from the Identity Vault. To determine the right settings for your environment, view the help for the options, or see the *NetIQ Identity Manager 4.8 Administration Guide* on the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).

- Ensure that the user's container has an assigned Universal Password policy and that the **Synchronize Distribution Password When Setting Universal Password** option is set for this policy.

Connected System User Passwords Are Not Provisioned to the Identity Vault

- Examine the status log, DSTRACE output, and the trace file.

- Several password management properties are available in iManager on the Global Configuration Values page of the Driver Edit Properties window. Ensure that at least one of the following options is set:

  - **The Identity Vault Accepts Passwords from the Top Secret Connected System**

  - **The Identity Vault Accepts Administrative Password Resets from the Top Secret Connected System**

  To determine the right settings for your environment, view the help for the options, or see the *NetIQ Identity Manager 4.8 Administration Guide* on the Identity Manager 4.8 Documentation Web site (https://www.netiq.com/documentation/identity-manager-47/).

- If the **Require Password Policy Validation before Publishing Passwords** GCV is set, the user's password must satisfy the password rules in the password policy assigned to the user container.

- Ensure that the change log started task is running on all systems that share the security system database.

- Ensure that the security system exit has been installed, that LLA has been refreshed, and that the exit has been activated. For details, see "Installing the Driver Security System Exit IDMTSSIX" on page 37.

Users or Groups Are Not Modified, Deleted, Renamed, or Moved

- Examine the status log, DSTRACE output, trace file, and REXX exec output file.

- Examine the driver Data Flow setting to verify the authoritative source for identities.

- Identity Vault and connected system identities must be associated before events are synchronized. To view an identity's associations, use Modify User/Group in iManager and click the **Identity Manager** tab. You can migrate identities to establish associations. For details, see "Migrating Identities" on page 56.

- Identity Vault move events can remove the identity from the base container monitored by the driver to a container that is not monitored by the driver. This makes the move appear to be a delete.

- Moving a user or group is not supported by Top Secret.

Change Log Errors

- Examine the change log started task messages.

- Ensure that the change log started task is running on all systems that share the security system database.

- Ensure that the user ID that the change log started task uses has been set up properly. For details, see "Preparing User IDs for the Started Tasks" on page 34.

- Ensure that you initialized the change log data set during installation. For details about initializing the change log data set, see "Allocating and Initializing the Change Log Data Set" on page 33.

- You can use the `LDXSERV` TSO command to display information about the change log data set. Enter the following TSO command:

  ```
  LDXSERV STATUS
  ```

  To use the `LDXSERV` command, you must include the driver load library in your STEPLIB concatenation.

B   System and Error Messages

Components of the Identity Manager 4.8 driver for CA Top Secret write messages to report operational status and problems. For detailed troubleshooting information, see Appendix A, "Troubleshooting," on page 77.

Each message begins with a code of 3-6 characters associated with the driver component that generated the message. Use this code to find message information quickly as follows:

- "CFG Messages" on page 85
- "DOM Messages" on page 86
- "DRVCOM Messages" on page 86
- "HES Messages" on page 87
- "LDX0 Messages" on page 87
- "LDXL Messages" on page 89
- "LDXS Messages" on page 92
- "LDXU Messages" on page 93
- "LDXV Messages" on page 95
- "LWS Messages" on page 97
- "NET Messages" on page 105
- "RDXML Messages" on page 105

CFG Messages

Messages beginning with CFG are issued by configuration file processing.

CFG001E Could not open configuration file *filename*.

Explanation:	Could not open the configuration file.
Possible cause:	The file does not exist.
Possible cause:	You don't have permission to read the file.
Action:	Ensure that the configuration file exists at the correct location and that you have file system rights to read it.

CFG002E Error parsing configuration file line: *<configline>*.

Explanation:	The line is not formatted as a valid configuration statement and cannot be parsed.
Action:	Correct the line in the configuration file.

CFG003W Configuration file line was ignored. No matching statement name found: *<configline>*.

Explanation:	This line is formatted as a valid configuration file statement, but the statement is not recognized. The line is ignored.
Possible cause:	The statement is incorrectly typed or the statement name is used only in a newer version of the software.
Action:	Correct the statement.

CFG004E Error parsing configuration file line. No statement name was found: *<configLine>*.

Explanation:	Could not find a statement name on the configuration line.
Action:	Correct the line in the configuration file to supply the required statement.

CFG005E A required statement *statement_id* is missing from the configuration file.

Explanation:	The *statement_id* statement was not specified in the configuration file, but is required for the application to start.
Action:	Add the required statement to the configuration file.

DOM Messages

Messages beginning with DOM are issued by driver components as they communicate among themselves.

DOM0001W XML parser error encountered: *errorString*.

Explanation:	An error was detected while parsing an XML document.
Possible cause:	The XML document was incomplete, or it was not a properly constructed XML document.
Action:	See the error string for additional details about the error. Some errors, such as no element found, can occur during normal operation and indicate that an empty XML document was received.

DRVCOM Messages

Messages beginning with DRVCOM are issued by the include/exclude system.

DRVCOM000I *nameversion* Copyright 2005 Omnibond Systems, LLC. ID=*code_id_string*.

Explanation:	This message identifies the system component version.

Action: No action is required.

DRVCOM001W Invalid include/exclude CLASS statement.

Explanation: The include/exclude configuration file contains an invalid CLASS statement.

Action: Correct the include/exclude configuration file with proper syntax.

DRVCOM002D An include/exclude Rule was added for class: *class*.

Explanation: The include/exclude configuration supplied a rule for the specified class.

Action: None.

DRVCOM003D An include/exclude Association Rule was added for association *association*.

Explanation: The include/exclude configuration supplied an association rule for the specified association.

Action: None.

HES Messages

Messages beginning with HES are issued by driver components as they use HTTP to communicate.

HES001E Unable to initialize the HTTP client.

Explanation: Communications in the client could not be initialized.

Possible cause: Memory is exhausted.

Action: Increase the amount of memory available to the process.

HES002I Connecting to host *host_name* on port *port_number*.

Explanation: The client is connecting to the specified server.

Action: None.

HES003W SSL communications have an incorrect certificate. rc = *rc*.

Explanation: The security certificate for SSL services could not be verified.

Possible cause: The certificate files might be missing or invalid.

Action: Obtain a new certificate.

LDX0 Messages

Messages beginning with LDX0 are issued by the driver security system exit module IDMTSSIX and the `LDXSERV` command.

LDX0001E There are old events on the LDX queue. Ensure that LDXLOGR is started.

Explanation: The memory queue access routine in the security system exit found events in the memory queue that have been unprocessed for at least fifteen minutes. During normal operation, the change log started task processes events from the queue immediately.

Possible cause: The change log started task is not running.

Action: Ensure that the change log started task is running.

LDX0002I Unexpected RC *xxxxxxxx* during token processing routine.

Explanation: An unexpected return code was received from z/OS name/token callable services by a driver component.

Possible cause: Internal system error.

Action: Collect diagnostic information and contact NetIQ® Technical Support.

LDX0103E Unable to parse command line.

Explanation: The `LDXSERV` command contained invalid operands and was unable to prompt for correct information.

Action: Correct the syntax of the `LDXSERV` command and reissue it. If the command was issued by the driver shim, collect diagnostic information and contact NetIQ Technical Support.

LDX0105E Internal error: *description*.

Explanation: An unexpected error occurred in the `LDXSERV` command. The message contains a description of the problem.

Possible cause: Internal error.

Action: Collect diagnostic information and contact NetIQ Technical Support.

LDX0106E Unable to open the log file.

Explanation: `LDXSERV` was unable to open the change log data set.

Possible cause: The user ID running the `LDXSERV` command does not have access to the change log data set.

Action: Check the session log and message files for additional messages concerning the failure. If you are unable to determine and correct the cause of the error, collect diagnostic information and contact NetIQ Technical Support.

LDX0107E No preallocated log file and no valid environment.

Explanation: The `LDXSERV` command was unable to find the change log data set because there was no LOGFILE DD statement and there was no valid LDX environment. The LDX environment is created when the security system exit is invoked for the first time after an IPL or when the change log started task first starts.

Action: Ensure that you are logged on to a system where the driver is installed and that the security system exit has been properly installed and is active. If you are unable to determine and correct the cause of the error, collect diagnostic information and contact NetIQ Technical Support.

LDX0108E No preallocated log file and logger is not active.

Explanation: The `LDXSERV` command was unable to find the change log data set because there was no LOGFILE DD statement and the change log started task was not active.

Action: If you are unable to determine and correct the cause of the error, collect diagnostic information and contact NetIQ Technical Support.

LDX0109E Dynamic allocation failed for log file *dsname*, s99rc=*rc*, s99error=*err*.

Explanation: The `LDXSERV` command was unable to dynamically allocate the change log data set. The dynamic allocation return code and reason codes are given in the message by *rc* and *err* respectively.

Dynamic allocation return codes and reason codes are documented in the IBM publication *z/OS Programming: Authorized Assembler Services Guide*.

Action: If you are unable to determine and correct the cause of the error, collect diagnostic information and contact NetIQ Technical Support.

LDXL Messages

Messages beginning with LDXL are issued by the change log started task.

LDXL000 LOGGING STARTED AT *hh:mm:ss* ON *mm/dd/yyyy*.

Explanation: The change log started task has initialized.

Action: Informational only. No action is required.

LDXL001 MESSAGE LOG DISABLED, SYSPRINT DD MISSING.

Explanation: During initialization, the change log started task was unable to open the SYSPRINT DD statement.

The change log started task continues processing, but no messages are written to SYSPRINT.

Possible cause: The SYSPRINT DD statement is missing from the JCL for the change log started task.

Action: Ensure that a SYSPRINT DD statement is present in the JCL and that it defines a file that the change log started task can write to.

LDXL002 EXECUTE STATEMENT PARAMETERS: *parm-values*.

Explanation: During initialization, the change log started task found the listed parameters present on the EXEC statement PARM parameter.

Action: Informational only. No action is required.

LDXL003 START COMMAND PARAMETERS: *parameters*.

Explanation: During initialization, the change log started task found the listed parameters present on the command line.

Action: Informational only. No action is required.

LDXL004 STOP COMMAND RECEIVED.

Explanation: An operator entered a `STOP` command for the change log started task. The change log started task ends.

Action: Informational only. No action is required.

LDXL005 MODIFY COMMAND PARAMETERS: *parameters*.

Explanation: An operator entered a `MODIFY` command for the change log started task with the listed parameters.

Action: Informational only. No action is required.

LDXL006 UNRECOGNIZED CIBVERB TYPE: X'*hh*', COMMAND IGNORED.

Explanation: During processing, the change log started task received a command input buffer (CIB) with a verb other than `STOP` or `MODIFY`. Processing continues.

Possible cause: Internal system error.

Action: Collect diagnostic information and contact NetIQ Technical Support.

LDXL007 OPERATOR CANCEL DETECTED, ATTEMPTING NORMAL SHUTDOWN.

Explanation: An operator has issued a `CANCEL` command without the `DUMP` parameter for the change log started task. The change log started task attempts a clean shutdown.

Action: Wait for the change log started task to end. If the change log started task does not end within a reasonable amount of time, issue another `CANCEL` command specifying the `DUMP` parameter. If you are unable to determine and correct the cause of the error, collect diagnostic information and contact NetIQ Technical Support.

LDXL008 EVENT TRACING ENABLED.

Explanation: An operator has issued a `MODIFY` command for TRACE ON to the change log started task.

Event tracing is turned on.

Action: Informational only. No action is required.

LDXL009 EVENT TRACING DISABLED.

Explanation: An operator has issued a `MODIFY` command for TRACE OFF to the change log started task.

Event tracing is turned off.

Action: Informational only. No action is required.

LDXL010 MODIFY COMMAND IGNORED, INVALID OR MISSING PARAMETERS.

Explanation: An operator has issued a `MODIFY` command to the change log started task, but the command parameters are not recognized.

The `MODIFY` command is ignored.

Action: Reissue the `MODIFY` command with the intended parameters.

LDXL011 EVENT RC(*rc*) DATA: *event_data*.

Explanation: Event tracing is turned on and an event has been processed.

The return code from ProcessEvent is *rc*. The content of the event record is *event_data*.

Processing continues.

Action: Informational only. No action is required.

LDXL012 TERMINATING BECAUSE LOGGING ALREADY ACTIVE.

Explanation: On startup, the change log started task has detected that another change log started task is already running.

This instance of the change log started task terminates.

To detect this condition, the change log started task enqueues exclusively on qname `ldxlogr`, rname `#LDXENVIRONTOKEN` when it initializes. If the ENQ macro fails, this message is issued. The change log started task dequeues this resource on shutdown.

Possible cause: A `START` command for the change log started task has been issued more than once.

Action: Do not start more than one instance of the change log started task at a time.

LDXL013 LOGGING TO DATASET: *dsname*.

Explanation: The name of the change log data set in use is *dsname*.

Action: Informational only. No action is required.

LDXL999 LOGGING ENDED AT *hh:mm:ss* ON *mm/dd/yyyy*.

Explanation: The change log started task is ending.

Possible cause: An operator entered a `STOP` command for the change log started task.

Action: Informational only. No action is required.

LDXS Messages

Messages beginning with LDXS are issued by the driver shim change log API.

LDXS000I *nameversion* Copyright 2006 Omnibond Systems, LLC. ID=*code_id_string*.

Explanation: This message identifies the system component version.

Action: No action is required.

LDXS001A Error executing script *scriptName*. The return code is *returnCode*, the reason code is *reasonCode*, the abend code is *abendCode*.

Explanation: The driver shim could not execute *scriptName*.

Possible cause: The script or command does not exist or is not valid.

Action: Ensure that the driver shim is correctly configured to execute the command or script and that the command or script exists and is valid.

LDXS002A The change log service startup failed, rc = *rc*.

Explanation: The change log API failed to initialize.

Possible cause: The change log data set has not been initialized.

Possible cause: The driver load library is not properly configured.

Possible cause: The driver does not have the required rights to access the change log data set.

Action: Ensure that all of the steps of the installation procedure have been performed correctly and have not subsequently been reversed.

LDXS003A Unable to create token, return code from IEANTCR is *rc*.

Explanation: z/OS name/token callable services failed to create a token. The return code from IEANTCR is *rc*.

Possible cause: Internal error.

Action: Collect diagnostic information and contact NetIQ Technical Support.

LDXU Messages

Messages beginning with LDXU are issued by the log file utility `LDXUTIL`.

LDXU000I Log File Utility started on *mm/dd/yyyy* at *hh:mm:ss*.

Explanation: The log file utility has initialized.

Action: Informational only. No action is required.

LDXU001W Message log disabled, SYSPRINT DD missing.

Explanation: During initialization, the log file utility was unable to open the SYSPRINT DD statement. The log file utility continues processing, but no messages are written to SYSPRINT.

Possible cause: The SYSPRINT DD statement is missing from the JCL for the log file utility.

Action: Ensure that a SYSPRINT DD statement is present in the JCL and that it defines a file that the log file utility can write to.

LDXU002I Execute statement parameters: *parm-values*.

Explanation: During initialization, the log file utility found the listed parameters present on the EXEC statement PARM parameter.

Action: Informational only. No action is required.

LDXU003E Open failed for log file.

Explanation: The log file utility could not open the change log data set.

Possible cause: The LOGFILE DD statement is missing from the JCL for the log file utility.

Action: Ensure that a LOGFILE DD statement is present in the JCL and that it defines a data set that the log file utility can write to.

LDXU004I Log file blocksize: *blksize*.

Explanation: The log file utility is initializing the change log data set with a blocksize of *blksize*.

Action: Informational only. No action is required.

LDXU005I Log file blocks written: *block-count*.

Explanation: While initializing the change log data set, the log file utility has written *block-count* blocks of empty records.

Action: Informational only. No action is required.

LDXU006E Open failed for LOADIN file.

Explanation: The log file utility load function could not open the LOADIN ddname.

Possible cause: The LOADIN DD statement is missing from the JCL for the log file utility.

Action: Ensure that a LOADIN DD statement is present in the JCL and that it defines a file that the log file utility can read.

LDXU007E Unrecognized or missing execute statement parameter.

Explanation: The log file utility found an unknown parameter in the EXEC statement PARM parameter.

Processing ends.

Possible cause: The EXEC statement PARM value is missing or does not contain one of the following functions:

- INITIALIZE
- DUMP
- LOAD

Action: Correct the PARM value and resubmit the job.

LDXU008I Log file events loaded: *event-count*.

Explanation: The log file utility load function has successfully loaded *event-count* events into the change log data set from the input file.

Action: Informational only. No action is required.

LDXU009E Add event failed, error code *code*.

Explanation: The log file utility load function was unable to add an event record to the change log data set. The LDXLADD LDXIOERR code was *code*.

Possible cause: Internal system error.

Action: Collect diagnostic information and contact NetIQ Technical Support.

LDXU010E Read header failed, error code *code*.

Explanation: The log file utility dump function was unable to read the header record of the change log data set. The LDXLGETE LDXIOERR code was *code*.

Possible cause: Internal system error.

Action: Collect diagnostic information and contact NetIQ Technical Support.

LDXU011E Read event failed, error code *code*.

Explanation: The log file utility dump function was unable to read an event record from the change log data set. The LDXLGETE LDXIOERR code was *code*.

Possible cause: Internal system error.

Action: Collect diagnostic information and contact NetIQ Technical Support.

LDXU990I Open BDAM log succeeded.

Explanation: The log file utility has initialized the change log data set with empty records and has successfully opened it to complete the initialization by updating the header information.

Action: Informational only. No action is required.

LDXU991E Open BDAM log failed.

Explanation: The log file utility has initialized the change log data set with empty records, but could not reopen it to complete the initialization by updating the header information.

Possible cause: Internal system error.

Action: Collect diagnostic information and contact NetIQ Technical Support.

LDXU999I Log File Utility ended on *mm/dd/yyyy* at *hh:mm:ss*.

Explanation: The log file utility has completed processing.

Action: Informational only. No action is required.

LDXV Messages

Messages beginning with LDXV are issued by the `IDMGETV` and `IDMSETV` commands.

LDXV001E IDM token not present.

Source: `IDMGETV` command, `IDMSETV` command

Explanation: The data areas that the driver shim sets up for the `IDMGETV` or `IDMSETV` command before calling a script are not present.

Possible Cause: The command was not called by the driver shim.

Action: Ensure that the commands are invoked by the driver shim. They are not intended to be used outside of this environment.

LDXV002E Unable to parse command.

Source: `IDMGETV` command, `IDMSETV` command

Explanation: The TSO parsing routine detected an error in the command and was unable to prompt for a correction.

Possible Cause: The command had a syntax error. It was not called from an interactive session and could not prompt for a correction.

Action: Examine the associated messages from the TSO parsing routine that describe the error. Correct the operands of the command.

LDXV003E Error from TSO service routine IKJCT441, RC *<rc>*.

Source: `IDMGETV` command

Explanation: TSO routine IKJCT441 detected a problem and ended with return code *rc* (decimal).

Possible Cause: Internal error.

Action: Collect diagnostic information and contact NetIQ Technical Support.

LDXV004W *<variablename>* contains invalid characters to be a REXX variable.

Source: `IDMGETV` command, `IDMSETV` command

Explanation: The command was directed to create the variable named in the message, but the variable name contained characters that are not acceptable in a REXX variable name. The acceptable characters are as follows:

Alphanumeric characters	`A-Z, a-z, 0-9`
"At" sign	`@`
Octothorpe	`#`
"Dollar" sign	`$`
Exclamation mark	`!`
Question mark	`?`
Period	`.`
Underscore	`_`

Possible Cause: The variable named in the message was defined in eDirectory™ using one or more characters not in the list of acceptable characters. For example, some attribute names might contain spaces.

Action: Use the driver mapping rules to rename the variable to a name that meets the REXX naming requirements.

LDXV005E IDMGETV was not called from a CLIST or REXX exec.

Source: `IDMGETV` command

Explanation: The `IDMGETV` command must be called from a REXX exec, because it creates and manipulates REXX variables.

Possible Cause: The command was not called from within the REXX environment.

Action: Call the command from within the REXX environment.

LDXV006E GROUP or USER list required.

Source: `IDMSETV` command

Explanation: The `IDMSETV` command requires either the GROUP(*grouplist*) or USER(*userlist*) operand.

Possible Cause: Use one of the required operands.

Action: Correct the operands of the command.

LDXV007E GROUP and USER operands are mutually exclusive.

Source: `IDMSETV` command

Explanation: The command found both the USER and GROUP operands on the command line. These are mutually exclusive.

Possible Cause: Both GROUP and USER were specified on the `IDMSETV` command.

Action: Correct the command.

LDXV008E Error returned from *<service>*: RC *<rc>*.

Source: `IDMGETV` command, `IDMSETV` command

Explanation: The IBM service routine *service* returned the return code *rc* (decimal).

Possible Cause: Internal error.

Action: Collect diagnostic information and contact NetIQ Technical Support.

LWS Messages

Messages beginning with LWS are issued by the integrated HTTP server.

LWS0001I Server has been initialized.

Explanation: The server has successfully completed its initialization phase.

Action: None. Informational only.

LWS0002I All services are now active.

Explanation: All of the services offered by the server are now active and ready for work.

Action: None. Informational only.

LWS0003I Server shut down successfully.

Explanation: The server processing completed normally. The server ends with a return code of 0.

Action: No action is required.

LWS0004W Server shut down with warnings.

Explanation: The server processing completed normally with at least one warning. The server ends with a return code of 4.

Action: See the log for additional messages that describe the warning conditions.

LWS0005E Server shut down with errors.

Explanation: The server processing ended with one or more errors. The server ends with a return code of 8.

Action: See the log for additional messages that describe the error conditions.

LWS0006I Starting *service*.

Explanation: The server is starting the specified service.

Action: None. Informational only.

LWS0007E Failed to start *service*.

Explanation: The server attempted to start the specified service, but the service could not start. The server terminates processing.

Action: See the log for additional messages that describe the error condition.

LWS0008I Stopping all services.

Explanation: The server was requested to stop. All services are notified and will subsequently end processing.

Action: None. Informational only.

LWS0009I Local host is *host_name* (*IP_address*).

Explanation: This message shows the host name and IP address of the machine that the server is running on.

Action: None. Informational only.

LWS0010I Local host is *IP_address*.

Explanation: This message shows the IP address of the machine that the server is running on.

Action: None. Informational only.

LWS0011I Server is now processing client requests.

Explanation: The server has successfully started all configured services, and it is ready for clients to begin requests.

Action: None. Informational only.

LWS0012I *service* is now active on port *number*.

Explanation: The server *service* is running on the specified TCP port *number*. Clients can begin making requests to the specified service.

Action: None. Informational only.

LWS0013I *service* is now inactive on port *number*.

Explanation: The server *service* is not active on the specified TCP port *number*. Processing continues, but no client requests can be made to the service until it becomes active again.

Action: None. Informational only.

LWS0014E An error was encountered while parsing execution parameters.

Explanation: An error occurred while parsing the execution parameters. The server terminates with a minimum return code of 8.

Action: Collect diagnostic information and contact NetIQ Technical Support.

LWS0015E *service* failed to start with error *number*.

Explanation: The specified service failed to start. The server terminates with a minimum return code of 8.

Action: Collect diagnostic information and contact NetIQ Technical Support.

LWS0020I Server *version* level: *level*.

Explanation: This message contains information detailing the current service level for the server program being executed. The value of *version* indicates the current release of the server. The value of *level* is a unique sequence of characters that can be used by NetIQ Technical Support to determine the maintenance level of the server being executed.

Action: Normally, no action is required. However, if you report a problem with the server to NetIQ Technical Support, you might be asked to provide the information in the message.

LWS0023I Listen port *number* is already in use.

Explanation: The displayed listen port is already in use by another task running on the local host. The server retries establishing the listen port.

Action: Determine what task is using the required port number and restart the server when the task is finished, or specify a different port in the configuration file. If the port number is changed for the server, the client must also specify the new port number.

LWS0024W Too many retries to obtain port *number*.

Explanation: The server tried multiple attempts to establish a listen socket on the specified port number, but the port was in use. The server terminates with a return code of 4.

Action: Determine what task is using the required port number, and restart the server when the task is finished, or specify a different port in the configuration file. If the port number is changed for the server, the client must also specify the new port number.

LWS0025I Local TCP/IP stack is down.

Explanation: The server detected that the local host TCP/IP service is not active or is unavailable. The server retries every two minutes to reestablish communication with the TCP/IP service.

Action: Ensure that the TCP/IP service is running.

LWS0026E Unrecoverable TCP/IP error *number* returned from *internal_function_name*.

Explanation: An unrecoverable TCP/IP error was detected in the specified internal server function name. The server ends with a minimum return code of 8. The error number reported corresponds to a TCP/IP errno value.

Action: Correct the error based on TCP/IP documentation for the specified errno.

LWS0027W Listen socket was dropped for port *number*.

Explanation: The server connection to the displayed listen port was dropped. The server attempts to reconnect to the listen port so that it can receive new client connections.

Action: Determine why connections are being lost on the local host. Ensure that the host TCP/IP services are running.

LWS0028E Unable to reestablish listen socket on port *number*.

Explanation: The listen socket on the specified port number was dropped. The server tried multiple attempts to reestablish the listen socket, but all attempts failed. The server ends with a return code of 8.

Action: Determine if the host's TCP/IP service is running. If the host's TCP/IP service is running, determine if another task on the local host is using the specified port.

LWS0029I *<id>* Client request started from *ip_address* on port *number*.

Explanation: A new client request identified by *id* has been started from the specified IP address on the displayed port number.

Action: None. Informational only.

LWS0030I *<id>* Client request started from *host* (*ip_address*) on port *number*.

Explanation: A new client request identified by *id* has been started from the specified host and IP address on the displayed port number.

Action: None. Informational only.

LWS0031W Unable to stop task *id*: *reason*.

Explanation: The server attempted to terminate a service task identified by *id*. The server could not stop the task for the specified reason. The server ends with a return code of 4.

Action: See the *reason* text for more information about why the task could not terminate.

LWS0032I *<id>* Client request has ended.

Explanation: The client requested identified by *id* has ended.

Action: None. Informational only.

LWS0033I *<id>* Client request: *resource*.

Explanation: The client connection identified by *id* issued a request for *resource*.

Action: None. Informational only.

LWS0034W *<id>* Write operation for client data has failed.

Explanation: A write operation failed for the connection identified by *id*. This is normally because the client dropped the connection. The client connection is dropped by the server.

Action: Ensure that the client does not prematurely drop the connection. Retry the client request if necessary.

LWS0035W *<id>* Read operation for client data has timed out.

Explanation: A read operation on the connection identified by *id* has timed out because of inactivity. The client connection is dropped by the server.

Action: Ensure that the client does not prematurely drop the connection. Retry the client request if necessary.

LWS0036W *<id>* Client request error: *error_code - error_text*.

Explanation: The server encountered an error while processing the client request. The server terminates the request.

Action: Determine why the request was in error by viewing the error code and error text that was generated.

LWS0037W *<id>* Client request error: *code*.

Explanation: The server encountered an error while processing the client request. The server terminates the request.

Action: Determine why the request was in error by viewing the error code and error text that was generated.

LWS0038I Received command: *command_text*.

Explanation: The server has received the displayed command from the operator. The server processes the command.

Action: None. Informational only.

LWS0043E Task *id* ended abnormally with RC=*retcode*.

Explanation: The server detected a task that ended with a non-zero return code. The server ends with a minimum return code of 8.

Action: View the log for other messages that might have been generated regarding the error.

LWS0045I Idle session time-out is *number* seconds.

Explanation: The message shows the idle time limit for connections. The server automatically terminates sessions that are idle for longer than the specified number of seconds.

Action: None. Informational only.

LWS0046I Maximum concurrent sessions limited to *number*.

Explanation: The message shows the maximum number of concurrent sessions allowed. The server allows only the specified number of concurrent sessions to be active at any given time. All connections that exceed this limit are forced to wait until the total number of connections drops below the specified value.

Action: None. Informational only.

LWS0047W Unable to delete log file *filename*.

Explanation: The log file could not be deleted as specified.

Possible cause: The user service or daemon does not have file system rights to delete old log files.

Action: Verify that the user service or daemon has the appropriate rights.

Action: Examine the current logs for related messages.

LWS0048I Log file *filename* successfully deleted.

Explanation: The log file has been deleted as specified.

Action: None. Informational only.

LWS0049E Error *error* authenticating to the directory as *fdn*.

Explanation: The connection manager could not connect to the directory as user *fdn*. The error was *error*.

Possible cause: The configuration parameters do not contain the correct user or password.

Action: Correct the cause of the error as determined from *error*.

Action: Verify that the User object has the appropriate rights.

Action: Verify that the password given for the User object in the configuration parameters is correct.

LWS0050E Server application initialization failure was detected.

Explanation: During server initialization, an error was detected while initializing the server Application object.

Possible Cause   This message is commonly logged when the driver is started and then immediately shut down. This can happen during installation, when the shim is started to generate keys or configure SSL. You can safely ignore this message in those cases.

Action: See the error logs for additional messages that indicate the cause of the error.

LWS0051E Server initialization failure was detected.

Explanation: The server failed to initialize properly because of an initialization error specific to the operating system.

Action: See the log for additional messages that indicate the cause of the error.

LWS0052W This server is terminating because of another instance already running (*details*).

Explanation: The server is shutting down because there is another active instance of this server running on the host.

Possible cause: A previous instance of the server was not stopped before starting a new instance.

Action: Stop or cancel the previous server instance before starting a new one.

LWS0053I The parameter *keyword* is no longer supported.

Explanation: The specified parameter is not supported in this release and might be removed in future releases.

Possible cause: An execution parameter was specified that is no longer supported.

Action: Do not specify the unsupported parameter.

LWS0054I The execution parameter *keyword* is in effect.

Explanation: The specified execution parameter is in effect for the server.

Action: Informational only. Processing continues.

LWS0055W Invalid execution parameter detected: *keyword*.

Explanation: An invalid execution parameter was detected.

Action: Do not specify the invalid or unknown execution parameter.

LWS0056I Not accepting new connections because of the MAXCONN limit. There are *number* active connections now for *service*.

Explanation: The specified service has a maximum connection limit that has been reached. The service no longer accepts new connections until at least one of the active connections ends.

Action: If you receive this message frequently, increase the MAXCONN limit for this service or set the MAXCONN to unlimited connections.

LWS0057I New connections are now being accepted for *service*.

Explanation: The service was previously not accepting new connections because of the imposed MAXCONN limit. The service can now accept a new connection because at least one active connection has ended.

Action: None. Informational only.

LWS0058I Listen socket on port *number* has been re-established.

Explanation: The previously dropped listen socket has been reestablished. Services using the specified port can now continue. The listen socket previously dropped because of an error or TCP/IP connectivity problems has been reestablished. Client connection processing continues.

Action: None. Informational only.

LWS0059W Server is terminating because the required service *serviceName* is ending.

Explanation: The specified required service has ended. The server terminates because it cannot continue running without the required service.

Action: See related log messages to determine why the required service ended. Correct the problem and restart the server.

NET Messages

Messages beginning with NET are issued by driver components during verification of SSL certificates.

NET001W Certificate verification failed. Result is *result*.

Explanation: A valid security certificate could not be obtained from the connection client. Diagnostic information is given by *result*.

Possible cause: A security certificate has not been obtained for the component.

Possible cause: The security certificate has expired.

Possible cause: The component certificate directory has been corrupted.

Action: Respond as indicated by *result*. Obtain a new certificate if appropriate.

RDXML Messages

Messages beginning with RDXML are issued by the embedded Remote Loader.

RDXML000I *nameversion* Copyright 2005 Omnibond Systems, LLC. ID=*code_id_string*.

Explanation: This message identifies the system component version.

Action: No action is required.

RDXML001I Client connection established.

Explanation: A client has connected to the driver. This can be the Metadirectory engine connecting to process events to and from the driver, or a Web-based request to view information or publish changes through the SOAP mechanism.

Action: No action required.

RDXML002I Request issued to start Driver Shim.

Explanation: The driver received a command to start the driver shim and begin processing events.

Action: No action required.

RDXML003E An unrecognized command was issued. The driver shim is shutting down.

Explanation: The driver received an unrecognized command from the Metadirectory engine. The driver shim is shutting down to avoid further errors.

Possible cause: Network error.

Possible cause: Invalid data sent to the driver.

Possible cause: The Metadirectory engine version might have been updated with new commands that are unrecognized by this version of the driver.

Possible cause: This message is logged when the driver shim process is shut down from the connected system rather than from a Driver object request. The local system can queue an invalid command to the driver shim to simulate a shutdown request and terminate the running process.

Action: Ensure that the network connection is secured and working properly.

Action: Apply updates for the engine or driver if necessary.

Action: If the driver shim process was shut down from the local system, no action is required.

RDXML004I Client Disconnected.

Explanation: A client has disconnected from the driver. This might be the Metadirectory engine disconnecting after a driver shutdown request or a Web-based request that has ended.

Action: No action required.

RDXML005W Unable to establish client connection.

Explanation: A client attempted to connect to the driver, but was disconnected prematurely.

Possible cause: The client is not running in SSL mode.

Possible cause: Mismatched SSL versions or mismatched certificate authorities.

Possible cause: Problems initializing SSL libraries because of improperly configured system entropy settings.

Action: Ensure that both the Metadirectory engine and the driver are running in the same mode: either clear text mode or SSL mode.

Action: If you are using SSL, ensure that the driver and Metadirectory engine have properly configured certificates, and that the driver system is configured properly for entropy.

RDXML006E Error in Remote Loader Handshake.

Explanation: The Metadirectory engine attempted to connect to the driver, but the authorization process failed. Authorization requires that both supply mutually acceptable passwords. Passwords are configured at installation.

Possible cause: The Remote Loader or Driver object passwords do not match.

Action: Set the Remote Loader and Driver object passwords to the same value for both the driver and the driver shim. Use iManager to modify the driver properties. Re-configure the driver shim on the connected system.

RDXML007I Driver Shim has successfully started and is ready to process events.

Explanation: The Metadirectory engine has requested the driver to start the shim for event processing, and the driver shim has successfully started.

Action: No action required.

RDXML008W Unable to establish client connection from *remoteName*.

Explanation: A client attempted to connect to the driver, but was disconnected prematurely.

Possible cause: The client is not running in SSL mode.

Possible cause: Mismatched SSL versions or mismatched certificate authorities.

Possible cause: Problems initializing SSL libraries because of improperly configured system entropy settings.

Action: Ensure that both the Metadirectory engine and the driver are running in the same mode: either clear text mode or SSL mode.

Action: If you are using SSL, ensure that the driver and Metadirectory engine have properly configured certificates, and that the driver system is configured properly for entropy.

RDXML009I Client connection established from *remoteName*.

Explanation: A client has connected to the driver. This can be the Metadirectory engine connecting to process events to and from the driver, or a Web-based request to view information or publish changes through the SOAP mechanism.

Action: No action required.

C Technical Details

Topics in this section include

- *"Driver Shim Command Line Options" on page 109*
- *"SAF Interface" on page 110*

Driver Shim Command Line Options

The following options can be specified on the driver shim command line. You can also specify driver shim configuration file statements as command line options. For details about the driver shim configuration file, see *"The Driver Shim Configuration File" on page 54*.

Options Used to Set Up Driver Shim SSL Certificates

The following command line options are used to set up the driver shim SSL certificates:

Table C-1   *Driver Shim Command Line Options for Setting Up SSL Certificates*

Option (Short and Long Forms)	Description
-s -secure	Secures the driver by creating SSL certificates, then exits.
-p -password	Specifies the Remote Loader password.

Other Options

Table C-2   *Other Driver Shim Command Line Options*

Option (Short and Long Forms)	Description
-c *<congFile>* -config *<configFile>*	Instructs the driver shim to read options from the specified configuration file. Options are read from ddname DRVCONF by default.
-? -help	Displays the command line options, then exits.
-v -version	Displays the driver shim version and build date, then exits.

SAF Interface

The driver query processor uses the system authorization facility (SAF) to retrieve information from the security system. Queries are used by the Metadirectory engine for matching and merging. Some fields in Top Secret, including custom fields, cannot be queried, because they are not recognized by SAF. These fields cannot be migrated from the connected system to the Identity Vault. Merge operations, which occur when objects in both the Identity Vault and the connected system are matched for the first time, might not include these fields for the event being processed.

Structured attributes are not supported by the schema. These attributes occur in certain Top Secret commands where more than one operand is used to define a field. Because these operands must be specified atomically on a single command, the corresponding auxiliary attribute in eDirectory™ must provide for this. These operands are filtered out and not synchronized by the default driver configuration. You can customize policies to process these fields if necessary.