

Driver for Sentinel 6.1 and the Identity Vault Collector Implementation Guide

Novell[®] Identity Manager

4.0.1

April 15, 2011

www.novell.com



Legal Notices

Novell, Inc., makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc., reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc., makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc., reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. See the [Novell International Trade Services Web page \(http://www.novell.com/info/exports/\)](http://www.novell.com/info/exports/) for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Copyright © 2011 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

Novell, Inc., has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed on the [Novell Legal Patents Web page \(http://www.novell.com/company/legal/patents/\)](http://www.novell.com/company/legal/patents/) and one or more additional patents or pending patent applications in the U.S. and in other countries.

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.
www.novell.com

Online Documentation: To access the latest online documentation for this and other Novell products, see the [Novell Documentation Web page \(http://www.novell.com/documentation\)](http://www.novell.com/documentation/).

Novell Trademarks

For Novell trademarks, see [the Novell Trademark and Service Mark list \(http://www.novell.com/company/legal/trademarks/tmlist.html\)](http://www.novell.com/company/legal/trademarks/tmlist.html).

Third-Party Materials

All third-party trademarks are the property of their respective owners.

Contents

About This Guide	5
1 Overview	7
1.1 Components of Account Tracking	7
1.1.1 DirXML-Accounts Attribute	7
1.1.2 Sentinel Driver	8
1.1.3 Identity Vault Collector	8
1.1.4 Custom Events	8
1.2 How It Works	9
1.3 What's New	12
2 Checklist for Enabling Account Tracking	13
3 Preconfiguring the Sentinel Driver	15
3.1 Placing Prerequisite Files	15
3.1.1 Sentinel Server	15
3.1.2 Sentinel RD Server	16
4 Creating a New Driver	17
4.1 Using Designer to Create and Configure the Driver	17
4.1.1 Importing the Sentinel Driver Packages	17
4.1.2 Creating the Driver	17
4.1.3 Using Designer to Deploy the Driver	19
4.1.4 Using Designer to Start the Driver	20
4.2 Activating the Driver	20
5 Configuring Account Tracking	21
6 Creating Connections to the JMS Message Bus for Sentinel 6.1	23
6.1 Creating the Connection Factories	23
6.2 Creating Queues	24
7 Installing and Configuring the Identity Vault Collector	25
7.1 Prerequisites	25
7.2 Installing the Identity Vault Collector	25
7.3 Configuring the Identity Vault Collector	26
7.3.1 Configuring the Collector for Sentinel	26
7.3.2 Configuring the Collector for Sentinel RD	27
7.4 Configuring an SSL Connection	29
7.4.1 Generating the Keystore File	29
7.4.2 Moving the Keystore File	29
7.4.3 Configuring the Remote Collector Manager Installation	30
7.5 Starting the Collector	30
7.6 Starting the Sentinel Driver	30

8	Configuring Multiple Instances of the Sentinel Driver	33
9	Custom Audit Events	35
10	Managing the Driver	43
11	Troubleshooting	45
11.1	Troubleshooting the Sentinel Driver	45
11.2	Troubleshooting the Identity Vault Collector	45
11.3	Account Tracking Information Is Not Written to the Sentinel Server	45
11.4	Error -9005 Sentinel Driver Does Not Start	46
11.5	Error Occurs when Uninstalling the Driver	46
A	Driver Properties	47
A.1	Driver Configuration	47
A.1.1	Driver Module	47
A.1.2	Driver Object Password (iManager Only)	48
A.1.3	Authentication	48
A.1.4	Startup Options	49
A.1.5	Driver Parameters	50
A.1.6	ECMAScript (Designer Only)	51
A.2	Global Configuration Values	51

About This Guide

This guide explains how to use the Identity Manager driver for Sentinel (Sentinel driver) and the Novell Identity Vault Collector to integrate Identity Manager and Sentinel.

- ♦ Chapter 1, “Overview,” on page 7
- ♦ Chapter 2, “Checklist for Enabling Account Tracking,” on page 13
- ♦ Chapter 3, “Preconfiguring the Sentinel Driver,” on page 15
- ♦ Chapter 4, “Creating a New Driver,” on page 17
- ♦ Chapter 5, “Configuring Account Tracking,” on page 21
- ♦ Chapter 6, “Creating Connections to the JMS Message Bus for Sentinel 6.1,” on page 23
- ♦ Chapter 7, “Installing and Configuring the Identity Vault Collector,” on page 25
- ♦ Chapter 8, “Configuring Multiple Instances of the Sentinel Driver,” on page 33
- ♦ Chapter 9, “Custom Audit Events,” on page 35
- ♦ Chapter 10, “Managing the Driver,” on page 43
- ♦ Chapter 11, “Troubleshooting,” on page 45
- ♦ Appendix A, “Driver Properties,” on page 47

Audience

This guide is intended for administrators of Identity Manager, Sentinel, and Sentinel RD.

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation, or go to www.novell.com/documentation/feedback.html and enter your comments there.

Documentation Updates

For the most recent version of the *Identity Manager 4.0.1 Driver for Sentinel 6.1 and the Identity Vault Implementation Guide*, visit the [Novell Compliance Management Platform documentation Web site](http://www.novell.com/documentation/ncmp11/) (<http://www.novell.com/documentation/ncmp11/>).

Additional Documentation

For documentation on Identity Manager, see the [Identity Manager Documentation Web site](http://www.novell.com/documentation/idm40/) (<http://www.novell.com/documentation/idm40/>).

For documentation on Sentinel, see the [Novell Sentinel Documentation Web site](http://www.novell.com/documentation/sentinel61/) (<http://www.novell.com/documentation/sentinel61/>).

For documentation on Sentinel Rapid Deployment, see the [Novell Sentinel Rapid Deployment Documentation Web site](http://www.novell.com/documentation/sentinel61rd/) (<http://www.novell.com/documentation/sentinel61rd/>).

Overview

1

The Sentinel driver and the Identity Vault Collector seamlessly integrate Identity Manager and Sentinel to track user account information. A user account can have one or more identities per system connected to the Identity Vault. The Sentinel driver and the Identity Vault Collector are used together to track each account identity and the status of the account. For more information, see [Section 1.2, “How It Works,” on page 9](#).

The Sentinel driver and the Identity Vault Collector work with Sentinel 6.1 and Sentinel 6.1 Rapid Deployment (RD).

The account tracking solution adds functionality that allows you to rapidly solve a variety of complex business problems. For example, the account tracking solution helps you monitor for rogue administration and define what action is taken if this occurs. For more information, see the section on sending alerts when rogue administration occurs in the *Novell Compliance Management Platform 1.1 Integration Guide* (http://www.novell.com/documentation/ncmp11/ncmp_integration/?page=/documentation/ncmp11/ncmp_integration/data/bgg1d1z.html).

The Sentinel driver and the Identity Vault Collector are used with other components to provide these solutions.

- ♦ [Section 1.1, “Components of Account Tracking,” on page 7](#)
- ♦ [Section 1.2, “How It Works,” on page 9](#)
- ♦ [Section 1.3, “What’s New,” on page 12](#)

1.1 Components of Account Tracking

There are four major components and artifacts used to track account identities and the status of those accounts:

- ♦ [Section 1.1.1, “DirXML-Accounts Attribute,” on page 7](#)
- ♦ [Section 1.1.2, “Sentinel Driver,” on page 8](#)
- ♦ [Section 1.1.3, “Identity Vault Collector,” on page 8](#)
- ♦ [Section 1.1.4, “Custom Events,” on page 8](#)

1.1.1 DirXML-Accounts Attribute

The DirXML-Accounts attribute tracks and stores the different account identifiers, if account tracking is enabled. It is created on each account when the account is synchronized to the Identity Vault. For example, John Smith has an account in Active Directory and in an LDAP directory. [Table 1-1](#) shows that the DirXML-Accounts attribute stores the different identifiers for John’s account. Active Directory has four different account identifiers for the same account and the LDAP directory has one.

Table 1-1 Contents of the DirXML-Accounts Attribute

Driver/Application	Account Identifier Type	Account Identifier Sample Data
Active Directory	sAMAccountName	jsmith
Active Directory	userPrincipalName	jsmith@company.com
Active Directory	DN	cn=John Smith,cn=users,dc=company,dc=com
Active Directory	association	5d377f84f3ab534babbf12edd6540d77
LDAP	DN	cn=jsmith,cn=users,dc=company,dc=com

This allows for correlation between all of the account identities in the systems managed by Identity Manager. Business policies can be validated with this information. For more information, see [Chapter 5, “Configuring Account Tracking,” on page 21](#).

1.1.2 Sentinel Driver

The Sentinel driver is an Identity Manager driver that sends the account identifier and the account status from the Identity Vault to the Identity Vault Collector. The account identifier data is used to track the accounts, the status of the identities, and the account access information.

The Sentinel driver tracks the following status:

- ♦ Add
- ♦ Modify
- ♦ Delete
- ♦ Rename
- ♦ Move

1.1.3 Identity Vault Collector

A Sentinel Collector performs functions such as remote protocol connections and data mapping. The Identity Vault Collector is designed to provide data collection services for the Identity Vault. It parses, normalizes, and enhances data received from the Sentinel driver. The Identity Vault Collector writes the data sent from the Identity Vault to the data store. This data is used in conjunction with other Sentinel Collectors to track accounts and validate business policies.

1.1.4 Custom Events

Custom events are audit events generated by policies in each driver and sent to the Platform Agent via the Metadirectory engine. The Platform Agent forwards these events to Sentinel. Sentinel stores the events for analysis to see if business policies are being broken. You can run reports to see what business policies are being kept and which policies are not.

In the past, Sentinel tracked Add, Delete, and Modify events. Sentinel could report on how many events occurred, but not if that event was supposed to occur. The custom events track granting and revoking of entitlements. The entitlements generate Add, Delete, or Modify events. Sentinel tracks which entitlement generated the Add event, and the reports show when and why an Add event occurred, instead of just when an Add event occurred.

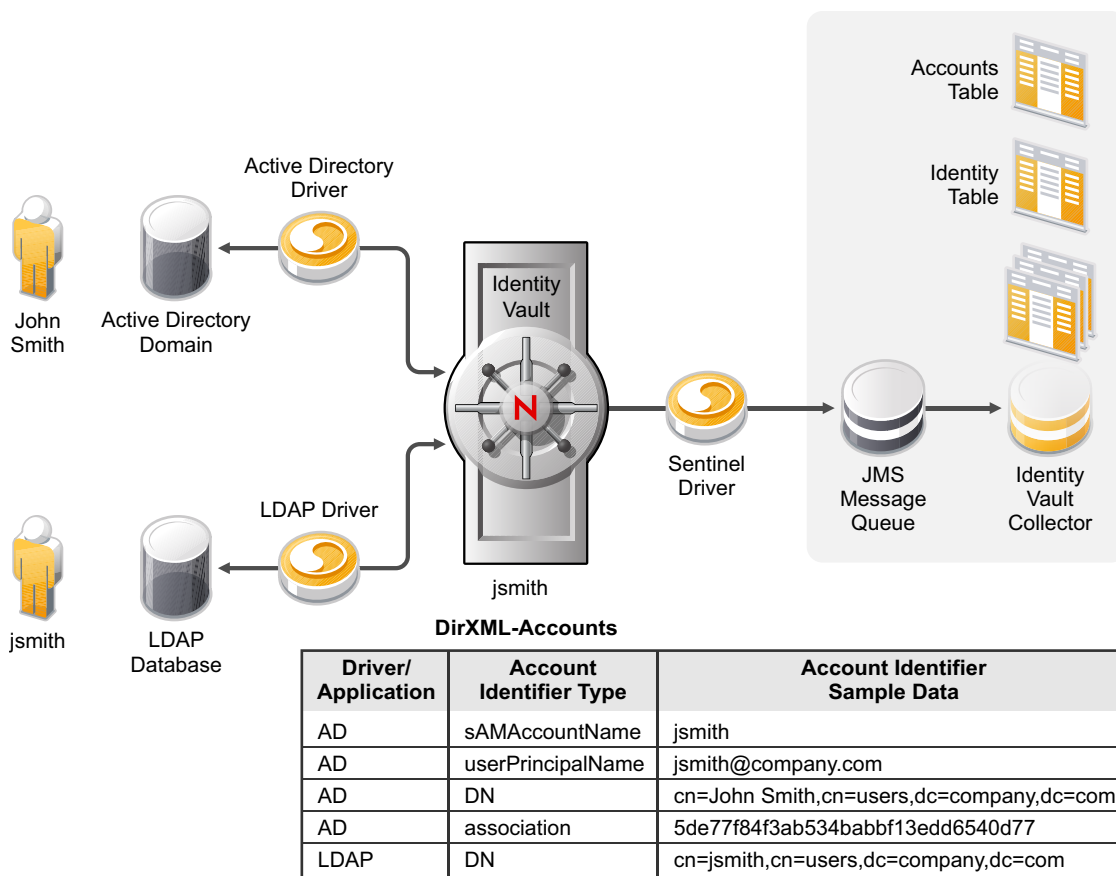
For more information, see [Chapter 9, “Custom Audit Events,”](#) on page 35.

1.2 How It Works

Without the Sentinel driver or the Identity Vault collector, Sentinel receives information from other collectors and then stores the data in tables in the data store. If the same user has different account identifiers, Sentinel treats each identifier as a unique account. Sentinel stores all of this information, but it is not able to make the connections it needs to realize that each identifier is referring to the same account.

The Sentinel driver enables you to track all account identifiers for each user and to track the status of those accounts, so you have a complete picture of user activities. [Figure 1-1](#) illustrates how the Sentinel driver works with the Identity Vault Collector to capture this information.

Figure 1-1 Synchronizing Account Data



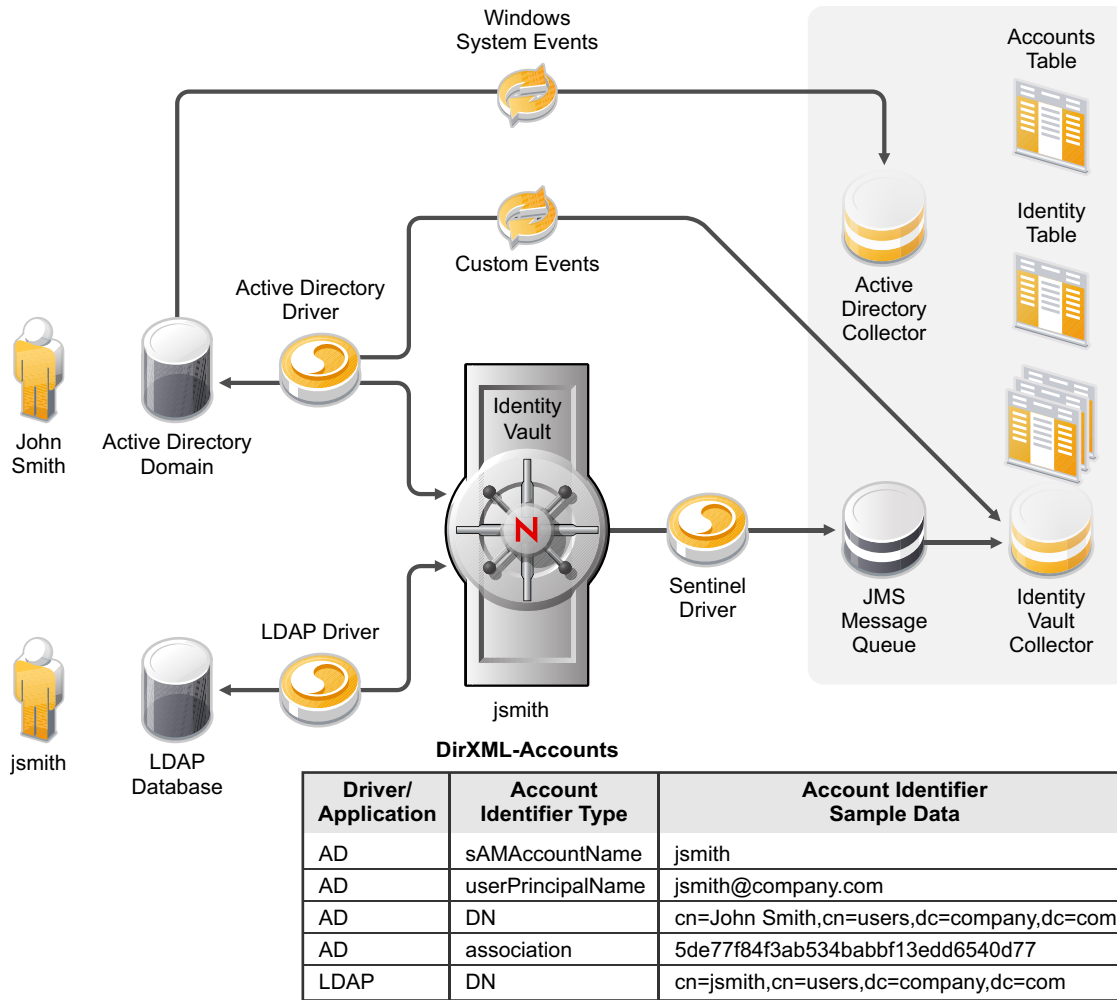
1. An account for John Smith is created in Active Directory and synchronized to the Identity Vault via the Active Directory driver.
2. An account for John Smith is created in the Identity Vault that contains the DirXML-Accounts attribute. The DirXML-Accounts attribute stores the different account identifiers from Active Directory.
3. When the new account is created in the Identity Vault, the Sentinel driver detects that the DirXML-Accounts attribute is added and sends this information to the JMS message queue that is part of Sentinel.

The different Sentinel versions use different message queues. If you are using Sentinel, it is a SonicMQ message queue. If you are using Sentinel RD, it is an ActiveMQ message queue.

4. The LDAP driver detects the new account created in the Identity Vault, then synchronizes this information to the LDAP database.
5. A new account is created for John Smith in the LDAP database as
`cn=jsmith,cn=users,dc=company,dc=com`.
6. The new account information is synchronized back to the Identity Vault and added to the DirXML-Accounts attribute as a new entry.
7. The Sentinel driver detects the change to the DirXML-Accounts attribute, then sends this information to the JMS message queue.
8. The Identity Vault Collector reads the account data from the JMS message queue.
9. The Identity Vault Collector parses, normalizes, and enhances the account data and then stores the account data in the Identity table in the data store.
10. The Sentinel correlation engine uses the information in the Identity table to generate reports of account activity per identity across all the systems provisioned by Identity Manager.

The second half of this solution allows the other Sentinel Collectors to use the account information to track whether business policies are being enforced or not. [Figure 1-2](#) shows how the custom events and the events from other Collectors are used to provide a complete record of John Smith's accounts.

Figure 1-2 Synchronizing Events



1. The account John Smith is created in the Identity Vault by the Active Directory driver.
2. The Sentinel driver detects this new account and sends the account information to the Identity Vault Collector, which stores it in the Identity Table.
3. John Smith logs in to Active Directory, and that information is sent to Sentinel through the Active Directory Collector to Sentinel.
4. The Active Directory Collector receives the login event directly from Windows without going through the Identity Vault. Information is recorded in the Accounts table indicating that `cn=John Smith,cn=users,dc=company,dc=com` logged in at a specific time.
5. If John Smith's CN in Active Directory is renamed to John D. Smith, this information is synchronized to the Identity Vault via the Active Directory driver.
6. The DirXML-Accounts attribute is updated with the new information, and the Sentinel driver detects this change.
7. The Sentinel driver synchronizes the new account information to the JMS message queue.
8. The Identity Vault collector reads the new account information and writes it to the Identity table.

9. When John Smith logs in again to Active Directory, the Active Directory collector records the login information.
10. Sentinel performs a lookup on the Identity table and detects that John Smith and John D. Smith are the same user account. Sentinel can keep a complete record of user actions.
11. Custom audit events for the Identity Vault are defined and added to each Identity Manager driver through policies. The policies add a layer of intelligence to Identity Manager and Sentinel by defining the business logic. For a list of these events, see [Chapter 9, “Custom Audit Events,” on page 35](#).

These policies are part of each driver that ships with Identity Manager.

12. You can generate useful reports about user accounts from Sentinel.

The Sentinel driver and the Identity Vault Collector provides the infrastructure to allow Sentinel to track each user’s account. This awareness allows business policies to be enforced.

1.3 What’s New

Sentinel Driver: The Sentinel driver now supports Identity Manager 4.0.1 and Sentinel RD. You can use the Sentinel driver with Sentinel or Sentinel RD.

Identity Vault Collector: The Identity Vault collector supports Sentinel RD. You can use the Identity Vault Collector with Sentinel or Sentinel RD.

Checklist for Enabling Account Tracking

2

Use the following checklist to verify that all steps are completed in order to have a complete solution with the Identity Vault Collector and the Sentinel driver.

Identity Manager 4.0 and either Sentinel 6.1 or Sentinel Rapid Deployment (RD) must be installed and configured before you proceed with any steps. See the *Identity Manager 4.0 Integrated Installation Guide* (http://www.novell.com/documentation/idm40/idm_integrated_install/?page=/documentation/idm40/idm_integrated_install/data/front.html), the *Sentinel Installation Guide* (<http://www.novell.com/documentation/sentinel61/index.html>), and the *Sentinel Rapid Deployment Installation Guide* (<http://www.novell.com/documentation/sentinel61rd/>).

- Copy the prerequisite .jar files to the Metadirectory engine or the Remote Loader. For instructions, see [Section 3.1, “Placing Prerequisite Files,”](#) on page 15.
- Create and configure the Sentinel driver. For instructions, see [Chapter 4, “Creating a New Driver,”](#) on page 17.
- Configure account tracking for the drivers that support account tracking. For instructions, see [Chapter 5, “Configuring Account Tracking,”](#) on page 21.
- Verify that the Sentinel server is running. For instructions, see [Step 1 on page 23.](#)
- (Conditional) If you are using Sentinel 6.1, create the connection factories. For instructions, see [Section 6.1, “Creating the Connection Factories,”](#) on page 23.
- (Conditional) If you are using Sentinel 6.1, create the message queues for the Sentinel driver. For instructions, see [Section 6.2, “Creating Queues,”](#) on page 24.
- Create and configure the Identity Vault collector. For instructions, see [Chapter 7, “Installing and Configuring the Identity Vault Collector,”](#) on page 25.
- Start the Identity Vault Collector. For instructions, see [Section 7.5, “Starting the Collector,”](#) on page 30.
- Start the Sentinel driver. For instructions, see [Section 4.1.4, “Using Designer to Start the Driver,”](#) on page 20.

Preconfiguring the Sentinel Driver

3

The following sections explain some simple pre-configuration steps you need to follow before creating the Sentinel driver.

- ♦ [Section 3.1, “Placing Prerequisite Files,” on page 15](#)

3.1 Placing Prerequisite Files

The following files must be copied into the correct directory for the driver to start. The files are different depending on whether you are using Sentinel or Sentinel RD.

- ♦ [Section 3.1.1, “Sentinel Server,” on page 15](#)
- ♦ [Section 3.1.2, “Sentinel RD Server,” on page 16](#)

3.1.1 Sentinel Server

1 On the Sentinel server, locate the following files:

- ♦ `mfcontext.jar`
- ♦ `sonic_Client.jar`

Platform	Location
Windows	<code>c:\Program Files\Novell\Sentinel6\3rdparty\SonicMQ\MQ7\lib</code>
Linux/UNIX	<code>/opt/Novell/Sentinel6/3rdpartySonicMQ/MQ7.0/lib</code>

2 Copy these files to the Identity Manager server.

Platform	Location
Windows	Local Installation: <code>c:\Novell\NDS\lib</code>
	Remote Installation: <code>c:\Novell\RemoteLoader\lib</code>
Linux/UNIX	Location Installation: <code>/opt/novell/eDirectory/lib/dirxml/classes</code>
	Remote Installation: <code>/opt/novell/eDirectory/lib/dirxml/classes</code>

3 Restart eDirectory™ to pick up these new classes.

- ♦ To restart eDirectory on Windows, access *Novell eDirectory Services* in the Control Panel. Select *ds.dlm*, click *Shutdown*, then click *Start*.
- ♦ To restart eDirectory on Linux/UNIX, enter:
`ndsmanage stopall`
then enter:
`ndsmanage startall`

3.1.2 Sentinel RD Server

- 1 On the Sentinel RD server, locate the `activemq-all-5.3.2.jar` file.

Platform	Location
Windows	<code>c:\Program Files\Novell\Sentinel6_rd\lib</code>
Linux/UNIX	<code>/opt/novell/sentinel6_rd_x86-64/lib</code>

- 2 Copy the file to the Identity Manager server.

Platform	Location
Windows	Local Installation: <code>c:\Novell\NDS\lib</code>
	Remote Installation: <code>c:\Novell\RemoteLoader\lib</code>
Linux/UNIX	Location Installation: <code>/opt/novell/eDirectory/lib/dirxml/classes</code>
	Remote Installation: <code>/opt/novell/eDirectory/lib/dirxml/classes</code>

- 3 Restart eDirectory to pick up these new classes.
 - ♦ To restart eDirectory on Windows, access *Novell eDirectory Services* in the Control Panel. Select *ds.dlm*, click *Shutdown*, then click *Start*.
 - ♦ To restart eDirectory on Linux/UNIX, enter:

```
ndsmanage stopall  
then enter:  
ndsmanage startall
```


Creating a New Driver

After the Sentinel driver files are installed on the server where you want to run the driver (see [Chapter 3, “Preconfiguring the Sentinel Driver,” on page 15](#)), you can use Designer to create the driver in the Identity Vault. The following sections provide instructions:

- ◆ [Section 4.1, “Using Designer to Create and Configure the Driver,” on page 17](#)
- ◆ [Section 4.2, “Activating the Driver,” on page 20](#)

4.1 Using Designer to Create and Configure the Driver

The following sections provide steps for using Designer to create and configure a new Sentinel driver.

- ◆ [Section 4.1.1, “Importing the Sentinel Driver Packages,” on page 17](#)
- ◆ [Section 4.1.2, “Creating the Driver,” on page 17](#)
- ◆ [Section 4.1.3, “Using Designer to Deploy the Driver,” on page 19](#)
- ◆ [Section 4.1.4, “Using Designer to Start the Driver,” on page 20](#)

4.1.1 Importing the Sentinel Driver Packages

Before you create the driver, verify that you have the Sentinel Driver packages updated and imported. If necessary, you may need to use the package update facility in Designer to import the Sentinel Driver packages.

You need to have the following packages:

- ◆ Sentinel Active MQ Configuration (NOVSENTAMQ)
- ◆ Sentinel Sonic MQ Configuration (NOVSENTSMQ)
- ◆ Sentinel Base (NOVSENTB)

IMPORTANT: You need to run the package update step and the import step separately for each package, starting with the Base package. If you do not do this, the Base package will be permanently unavailable.

To import the Sentinel Driver packages:

- 1 In Designer, select *Help > Check for Package Updates* to install the Sentinel Driver packages.
- 2 In the Outline View, right-click on *Package Catalog* and choose *Import* to import the packages.

4.1.2 Creating the Driver

After you have imported the Sentinel Driver packages, you are ready to create the driver:

- 1 Select *Sentinel* in the Modeler view.

- 2 Drag the icon for *Sentinel* onto the Modeler view. *Sentinel* is categorized under *Enterprise* in the palette on the right.

Designer displays the Driver Configuration Wizard.

- 3 Select *Sentinel Base* and click *Next*.
- 4 Select *Sentinel Active MQ Configuration* (for Sentinel RD) or *Sentinel Sonic MQ Configuration* (for Sentinel) and click *Next*.
- 5 Specify the name of the driver and click *Next*.

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

- 6 Indicate whether you want to connect to a remote loader and click *Next*.

Connect to Remote Loader: Select *no* if this driver will run on the Metadirectory server without using the Remote Loader. Select *yes* if you want the driver to use the Remote Loader, either locally on the Metadirectory server or remotely on another server.

- 7 (Conditional) If you chose to run the driver remotely, click *Next*, then fill in the fields listed below. Otherwise, skip to the next step.

Remote Host Name and Port: Specify the hostname or IP address of the server where the driver's Remote Loader is running.

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

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

- 8 Specify values for the Broker URL, Broker username, and Broker password, as follows:

Broker URL: Specify the IP address of the Sentinel broker. The following are examples showing the expected values for the different versions of Sentinel. The ports listed are the default ports for the brokers.

- ♦ **Sentinel:** `tcp://brokeripaddress:10012`
For Sentinel, you must use `tcp://`.
- ♦ **Sentinel RD:** `ssl://brokeripaddress:61616`
For Sentinel RD, you must use `ssl://`.


Broker Username: Specify the username used to authenticate to this broker. If you are connecting to a Sentinel system, use a random username. If you are connecting to a Sentinel RD system, you must use the username and password contained in the `SENTINEL_HOME/config/activemqusers.properties` file on the Sentinel RD server. The username is `collectormanager`.

Broker Password: Specify the password of the user used to authenticate to the broker. If you are connecting to a Sentinel RD system, the `collectormanager` password is located in the `SENTINEL_HOME/config/activemqusers.properties` file.

- 9 Click *Next*.
- 10 On the Installation Summary screen, click *Finish*.

The driver configuration settings are explained in [Appendix A, "Driver Properties,"](#) on page 47.

If you need to do additional configuration for the driver, you must access the properties page of the driver. If you do not have the Driver Properties page displayed:


- 1 In Designer, open your project.
- 2 In the Modeler, right-click the driver icon  or the driver line, then select *Properties*.
This opens the properties page for the driver.

4.1.3 Using Designer to Deploy the Driver

After a driver is created in Designer, it must be deployed into the Identity Vault, because Designer is an offline tool. Plus, additional configuration procedures must be completed for the driver to work.

- ♦ “Deploying the Driver” on page 19
- ♦ “Additional Configuration” on page 20

Deploying the Driver

- 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 follow information to authenticate:
 - ♦ **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*.

Additional Configuration

There is additional configuration that must be completed before you start the Sentinel driver.

- ♦ (Conditional) The connection factories must be created for Sentinel 6.1. Sentinel RD automatically creates the connection factories.
- ♦ (Conditional) The SonicMQ message queues must be created, if you are using Sentinel 6.1. Sentinel RD automatically creates the messages queues for ActiveMQ.
- ♦ The Identity Vault Collector must be installed and configured.

See [Chapter 5, “Configuring Account Tracking,” on page 21](#) for instructions on how to create the connection factories and message queues. For the Identity Vault Collector installation instructions, see [Chapter 7, “Installing and Configuring the Identity Vault Collector,” on page 25](#).

4.1.4 Using Designer to Start the Driver

After the driver is created, you need to start the driver. However, you first need to configure and start the collector. To start the collector, see [Section 7.5, “Starting the Collector,” on page 30](#).

For details on starting the driver, see [Section 7.6, “Starting the Sentinel Driver,” on page 30](#).

IMPORTANT: The Identity Vault collector must be started before the driver is started. When the collector starts, the JNDI destinations are created. The driver looks for the JNDI destinations when it starts and if they do not exist, the driver cannot start.

4.2 Activating the Driver

The Sentinel driver contains its own activation that you receive from the customer center. The Sentinel driver requires this new activation within 90 days of creating the driver. Otherwise, the driver stops working.

If you create the driver in a driver set where you’ve already activated the Sentinel driver, the driver inherits the activation.

For more information on activation, see the section on activation in the (http://www.novell.com/documentation/idm40/idm_integrated_install/?page=/documentation/idm40/idm_integrated_install/data/bpo948h.html) *Identity Manager 4.0 Integrated Installation Guide*.

Configuring Account Tracking

5

To enable account tracking, complete the following two tasks:

- ♦ Extend the schema by installing Identity Manager 4.0.1 or later. If you have not installed Identity Manager 4.0.1 or later, see the *Identity Manager 4.0 Integrated Installation Guide* (http://www.novell.com/documentation/idm40/idm_integrated_install/?page=/documentation/idm40/idm_integrated_install/data/front.html) for instructions.
- ♦ Enable the account tracking GCV on each driver used with the Sentinel driver. Not all drivers can be enabled for account tracking. If a driver does not have the Account Tracking GCV, then account tracking cannot be enabled. The drivers that are enabled for Account Tracking are:
 - ♦ Active Directory
 - ♦ eDirectory
 - ♦ LDAP
 - ♦ Notes
 - ♦ SAP User Management
 - ♦ SAP Portal

These steps to enable account tracking are the same for each driver.

- 1 Access the Account Tracking GCV:
 - ♦ **In Designer:** Right-click the driver icon, then select *Properties > GCVs*.
 - ♦ **In iManager:** Edit the driver properties, then click the *Global Config Values* tab.
- 2 Set the *Account Tracking > Show Account Tracking Configuration* option to *show*.
- 3 Use the information in [Table 5-1](#) to correctly enable account tracking.
- 4 Click *OK* to save the changes.

If the driver is running, it must be restarted for the changes to take effect.

Table 5-1 *Show Account Tracking Configuration Options*

Option	Description
Enable account tracking	Select <i>true</i> to enable the policies in the driver to use the DirXML-Accounts attribute.
Realm	Specify the name of your realm, security domain, or namespace where the account name is unique.
Object Class	Specify the object classes to track with account tracking. The class name must be in the application namespace.

Option	Description
Identifiers	<p>Each driver has different account identifier attribute. By default, the attributes are prepopulated for each driver.</p> <ul style="list-style-type: none"> ◆ Active Directory: association, sAMAccountName, userPrincipalName, LDAPDN ◆ eDirectory: association, CN ◆ GroupWise: CN ◆ LDAP: association, LDAPDN ◆ Notes: association, FullName ◆ SAP User Management: association, USERNAME:BABIBNAME ◆ SAP Portal: association, logonname
Status attribute	<p>Specify the name of the attribute in the application namespace that represents the account status. By default the attributes are:</p> <ul style="list-style-type: none"> ◆ Active Directory: dirxml-uACAccountDisable ◆ eDirectory: Login Disabled ◆ GroupWise: 50058 ◆ LDAP: loginDisabled ◆ Notes: AccountTrackingAccountStatus ◆ SAP User Management: LOCKUSER ◆ SAP Portal: isLocked
Status active value	<p>The value of the status attribute that represents an active state. By default, the value is <i>false</i>.</p>
Status inactive value	<p>The value of the status attribute that represents an inactive state. By default, the value is <i>true</i>.</p>
Subscription default status	<p>The default status that the policies assume when an object is subscribed to the application and the status attribute is not set in the Identity Vault. By default, the status is <i>Active</i>.</p>
Publication default status	<p>The default status that the policies assume when an object is published to the Identity Vault and the status attribute is not set in the application. By default, the status is <i>Uninitialized</i>.</p>

Creating Connections to the JMS Message Bus for Sentinel 6.1

6

In traditional Sentinel configuration, there is a connector and a collector. The connector establishes a connection to the JMS message bus. However, there is no connector for the Identity Vault Collector. The Sentinel driver connects directly to the JMS message bus through connection factories and queues. The connection factories and the queues must be created for the Sentinel driver.

If you are using Sentinel RD, skip these steps and proceed to [Chapter 7, “Installing and Configuring the Identity Vault Collector,”](#) on page 25. The queues are automatically created in Sentinel RD.

- ♦ [Section 6.1, “Creating the Connection Factories,”](#) on page 23
- ♦ [Section 6.2, “Creating Queues,”](#) on page 24

6.1 Creating the Connection Factories

- 1 Start the Sentinel server by entering the following at a command prompt:
 - ♦ **Linux:** `/etc/init.d/sentinel start`
 - ♦ **Windows:** `Net start sentinel`
- 2 Launch the Sonic Management Console by entering:
 - ♦ **Linux:** `$ESEC_HOME/3rdparty/SonicMQ/MQ7.0/bin/startmc.sh`
 - ♦ **Windows:** `%ESEC_HOME%\3rdparty\SonicMQ\MQ7.0\bin\startmc.bat`
- 3 Log in to Sonic Management Console by using the following information:
 - ♦ **Connection Name:** By default, the value is Connection1. Any value is valid.
 - ♦ **Domain Name:** `esecDomain`
 - ♦ **Connection URL:** `tcp://localhost:10012`
The default Message Bus port is 10012. If you specified a different port during the installation of Sentinel, use that port.
 - ♦ **User Name:** Specify the administrator for Sentinel. For example, `esecadm`.
 - ♦ **Password:** Specify the password of the administrator.
- 4 From the Sonic Management Console toolbar, click *Tools > JMS Administered Objects*.
- 5 Click *JNDI Naming Service*, then use the following information to create the JNDI naming service:
 - ♦ **Sonic Storage:** Select the *Sonic Storage* check box.
 - ♦ **Domain:** Specify `esecDomain` for the domain name.
 - ♦ **Context Factory:** This field is prepopulated and the value cannot be changed.
 - ♦ **Provider URL:** Specify `tcp://localhost:10012` for the provider URL. If you are not using the default port, specify the port you are using.
- 6 Click *Connect*.
- 7 Select the `localhost:10012` entry in the tree on the left, then select the *Connection Factories* tab.

- 8 Click *New*.
- 9 Specify `TopicConnectionFactory` in the *Lookup Name* field.
The connection factory name must be the specified name.
- 10 Specify `ConnectionFactory` in the *Factory Type* field.
- 11 Specify `tcp://ipaddress:10012` in the *Connection URL* field.
The *ipaddress* is the IP address of your Sentinel server.
- 12 Click *Update* to save the information.
- 13 Repeat [Step 8](#) through [Step 12](#), but use `QueueConnectionFactory` as the *Lookup Name*.
- 14 Close the JMS Administered Objects dialog box.
- 15 Continue with [Section 6.2, “Creating Queues,”](#) on page 24.

6.2 Creating Queues

There are specific queues that must be created for the Sentinel Driver to work.

- 1 In the Sonic Management Console, select the *Configuration* tab, then expand the *Brokers* folder.
- 2 Expand *esecBroker*, then select *Queues*.
- 3 Right-click *Queues* in the left pane, then select *New Queue*.
- 4 Specify `pubReceiveEvent` in the *Name* field.
- 5 Click *OK* to create the new queue.
- 6 Repeat [Step 3](#) through [Step 5](#) twice more. Use the names of `pubReceiveEventResponse` and `subReceiveResponse` for each of the new queues.
- 7 Close the Management Console after the queues are created.
- 8 Continue with [Chapter 7, “Installing and Configuring the Identity Vault Collector,”](#) on page 25.

If you have more than one instance of the Sentinel driver, you must create additional queues for each additional driver. For more information, see [Chapter 8, “Configuring Multiple Instances of the Sentinel Driver,”](#) on page 33.

Installing and Configuring the Identity Vault Collector

7

Use the information in the following sections to install and configure the Identity Vault Collector.

- ♦ Section 7.1, “Prerequisites,” on page 25
- ♦ Section 7.2, “Installing the Identity Vault Collector,” on page 25
- ♦ Section 7.3, “Configuring the Identity Vault Collector,” on page 26
- ♦ Section 7.4, “Configuring an SSL Connection,” on page 29
- ♦ Section 7.5, “Starting the Collector,” on page 30
- ♦ Section 7.6, “Starting the Sentinel Driver,” on page 30

7.1 Prerequisites

- ❑ Install Identity Manager 4.0.1 on a server in your environment, as described in the *Novell Identity Manager Integrated Installation Guide* (http://www.novell.com/documentation/idm401/idm_integrated_install/?page=/documentation/idm40/idm_integrated_install/data/front.html).
- ❑ Create and configure the Sentinel driver. For more information, see the [Chapter 4, “Creating a New Driver,” on page 17](#) and [Chapter 5, “Configuring Account Tracking,” on page 21](#).
- ❑ Install and configure the different Sentinel components. For more information, see the *Novell Sentinel Installation Guide* (http://www.novell.com/documentation/sentinel61/s61_install/data/) or the *Novell Sentinel Rapid Deployment Installation Guide* (http://www.novell.com/documentation/sentinel61rd/s61rd_install/data/).
- ❑ (Conditional) Create the JMS queues and connection factories, if you are using Sentinel. For more information, see [Chapter 6, “Creating Connections to the JMS Message Bus for Sentinel 6.1,” on page 23](#).

7.2 Installing the Identity Vault Collector

To install the Identity Vault Collector, you add it to the Event Source Manager. This step is only done once. The Identity Vault Collector is then displayed as a collector to select during configuration.

To install the Identity Vault Collector:

- 1 Download the Identity Vault Collector (`Novell_Identity-Vault_6.1r2.clz.zip`) from the [Customer Center Web site](https://secure-www.novell.com/center/regadmin/jsps/home_app.jsp) (https://secure-www.novell.com/center/regadmin/jsps/home_app.jsp) to the server where the Sentinel Control Center is running.
- 2 Log in to the Sentinel Control Center.
- 3 Select the *Event Source Management > Live View*, then select *Tools > Import plugin*.
- 4 Browse to and select the `Novell_Identity-Vault_6.1r2.clz.zip` file, then click *Next*.
- 5 Follow the remaining prompts, then click *Finish*.

7.3 Configuring the Identity Vault Collector

The steps are different if you have Sentinel or Sentinel RD.

- ♦ [Section 7.3.1, “Configuring the Collector for Sentinel,” on page 26](#)
- ♦ [Section 7.3.2, “Configuring the Collector for Sentinel RD,” on page 27](#)

7.3.1 Configuring the Collector for Sentinel

- 1 In the Event Source Management live view, right-click the Collection Manager, then click *Add Collector*.
- 2 Select *Novell* in the *Vendor* column.
- 3 Select *Identity Manager* in the *Name* column, then click *Next*.
- 4 In the *Installed Scripts* column, select *Novell_Identity_Manager_6.1r2*, then click *Next*.
- 5 Configure the Identity Vault Collector for your needs by using the following information:

Configuration Parameter	Default Value	Description
ActiveMQ JMS User	none	Leave this field blank. This field is only used with Sentinel RD.
Broker Type	Sentinel RD/ ActiveMQ	Select Sentinel/SonicMQ as the broker type.
Broker URL		The URL that the Identity Vault Collector uses to retrieve identity events stored in the SonicMQ message queue. The format is <code>brokeripaddress:10012</code> , where 10012 is the default port.
Connector Retry Behavior	no connector	Determines how the Collector retries obtaining data from the Connector if no data is received.
Execution Mode	release	Sets the execution mode for the collector. There are three options: <ul style="list-style-type: none">♦ release: Use this mode for normal operation.♦ custom: Use this mode if the Identity Manager Collector is customized.♦ debug: Use this mode for troubleshooting. It generates debug trace files.
MSSP Customer Name	unknown	Name or numeric code for a specific customer in an MSSP environment. All data that is received is flagged with his value so that data segregation can be maintained.
Script Error Severity	5 Severe (5)	Sets the severity for a script error event.
Send Script Error Message	yes	Sends a script error event when there is an error in the collector script.

Configuration Parameter	Default Value	Description
Sentinel Driver Instance ID		Enables multiple Sentinel drivers. Each Sentinel driver is paired with a specific Identity Vault Collector. This instance ID is synchronized between the Sentinel driver and the Identity Vault Collector. By default, there is no value. Use letters and numbers only.

6 Click *Next*.

7 Complete the configuration of the Identity Manager Collector with the following information:

Name: Specify a name for this collector.

Run: Select whether the collector is started whenever the Collector Manager is started.

Alert if no data received in specified time period: (Optional) Select this option to send the No Data Alert event to Sentinel if data is not received by the collector in the specified time period.

Limit Data Rate: (Optional) Select this option to set a maximum limit on the rate of data the collector sends to Sentinel. If the data rate limit is reached, Sentinel throttles back on the source in order to limit the flow of data.

Set Filter: (Optional) Specify a filter on the raw data passing through the collector.

Trust Event Source Time: (Optional) Select this option if you trust the Event Source server's time.

8 Click *Finish*.

7.3.2 Configuring the Collector for Sentinel RD

1 In the Sentinel Control Center toolbar, select *Event Source Management > Live View*.

2 Right-click the Collector Manager, then click *Add Collector*.

3 Select *Novell* in the *Vendor* column.

4 Select *Identity Manager* in the Name column, select *3.6.1* in the version column, then click *Next*.

5 In the *Installed Scripts* column, select *Novell_Identity-Vault_6.1r2*, then click *Next*.

6 Configure the Identity Vault Collector using the following information:

Configuration Parameter	Default Value	Description
ActiveMQ JMS User	system	Specify system as the username that is contained in the <code>configactivemquser.properties</code> file. System is the username that ActiveMQ JMS uses to connect to the Sentinel JMS broker to retrieve identity events.

Configuration Parameter	Default Value	Description
Broker Type	Sentinel RD/ ActiveMQ	Select the type of broker you are using. The broker type is determined by the version of Sentinel you are using. The options are: <ul style="list-style-type: none"> ◆ Sentinel RD/ActiveMQ ◆ Sentinel/SonicMQ
Broker URL		Specify the URL used to connect to the Sentinel's JMS broker. The format for Sentinel RD is: <code>ssl://localhost:61616?wireFormat.maxInactivityDuration=0</code>
Connector Retry Behavior	no connector	Specify how the Collector retries retrieving data from the Connector if no data is received.
Execution Mode	release	Sets the execution mode for the collector. There are three options: <ul style="list-style-type: none"> ◆ release: Use this mode for normal operation. ◆ custom: Use this mode if the Identity Manager Collector is customized. ◆ debug: Use this mode for troubleshooting. It generates debug trace files.
MSSP Customer Name	unknown	Name or numeric code for a specific customer in an MSSP environment. All data that is received is flagged with his value so that data segregation can be maintained.
Script Error Severity	Severe (5)	If an error is detected in the Collector script configuration, this parameter determines the severity applied to the resulting event.
Send Script Error Message	yes	Select whether an event is generated when an error is detected with the Collector script configuration.
Sentinel Driver Instance ID		If you have multiple Sentinel drivers, you must specify a unique instance ID for each Sentinel driver. This value must be the same as the value specific in the Sentinel driver configuration. For more information, see Chapter 8, "Configuring Multiple Instances of the Sentinel Driver," on page 33.

7 Click *Next*.

8 Complete the configuration of the Identity Manager Collector with the following information:

Name: Specify a name for this collector.

Run: Select whether the collector is started whenever the Collector Manager is started.

Alert if no data received in specified time period: (Optional) Select this option to send the No Data Alert event to Sentinel if data is not received by the collector in the specified time period.

Limit Data Rate: (Optional) Select this option to set a maximum limit on the rate of data the collector sends to Sentinel. If the data rate limit is reached, Sentinel throttles back on the source in order to limit the flow of data.

Set Filter: (Optional) Specify a filter on the raw data passing through the collector.

Trust Event Source Time: (Optional) Select this option if you trust the Event Source server's time.

9 Click *Finish*.

7.4 Configuring an SSL Connection

Sentinel RD only allows an SSL connection to the ActiveMQ JMS message bus. This requires an SSL connection for the Sentinel driver and the Identity Vault Collector. Complete the following steps only if you are using Sentinel RD.

- ♦ [Section 7.4.1, “Generating the Keystore File,” on page 29](#)
- ♦ [Section 7.4.2, “Moving the Keystore File,” on page 29](#)
- ♦ [Section 7.4.3, “Configuring the Remote Collector Manager Installation,” on page 30](#)

7.4.1 Generating the Keystore File

You must generate a keystore file that is used by the Sentinel driver and the Identity Vault Collector:

1 Access the `Sentinel_RD_installation_directory/config` directory.

2 Enter the following command to extract the trusted root certificate:

```
../jre64/bin/keytool -exportcert -alias broker -keystore  
.activemqclientkeystore.jks -storepass password -file broker.cert
```

3 Enter the following commands to import the trusted root certificate into a new keystore file named `jssecacerts`:

3a Enter the following:

```
../jre64/bin/keytool -importcert -alias broker -file broker.cert -  
keystore jssecacerts -storepass password
```

3b Enter `yes` to trust to the certificate.

4 Remove the `broker.cert` file by entering `rm broker.cert`.

After you have generated the keystore file, it must be moved to the correct location. Proceed with [Section 7.4.2, “Moving the Keystore File,” on page 29](#).

7.4.2 Moving the Keystore File

After you have generated the keystore `jssecacerts` file, it must be moved to the JRE* `security` directory in the Sentinel driver and the Identity Vault Collector. The Sentinel driver and the Identity Vault Collector each contain a JRE. You must establish an SSL connection for each JRE for Sentinel RD to work.

You have the option of installing the Sentinel driver and the Identity Vault Collector locally or remotely. The following contains the default installation directories for each option on Linux/UNIX:

Table 7-1 Location of the JRE Security Directories on Linux/UNIX

Product	JRE Security Directory
Sentinel Driver	<p>Local Installation: <code>/opt/novell/eDirectory/lib/nds-modules/jre/lib/security</code></p> <p>Remote Installation: <code>/opt/novell/eDirectory/lib/nds-modules/jre/lib/security</code></p> <p>If you are using a 64-bit platform, the directory is <code>lib64</code> instead of <code>lib</code>. On a 64-bit platform, you would use this directory:</p> <p><code>/opt/novell/eDirectory/lib64/nds-modules/jre/lib/security</code></p>
Identity Vault Collector	<p>Local Installation: <code>/opt/novell/sentinel6_rd_x86-64/jre64/lib/security</code></p> <p>Remote Installation: <code>/opt/novell/sentinel6_rd_x86-64/jre64/lib/security</code></p>

After the `jssecacerts` file is in the proper location, you must restart Identity Manager, the Remote Loader, and Sentinel RD for the applications to use the certificate.

Now you need to restart Sentinel RD and eDirectory. Since your driver may start automatically and since the ID Vault Collector must be running before the driver starts, then you should restart Sentinel RD before eDirectory.

7.4.3 Configuring the Remote Collector Manager Installation

If you are using the Remote Collector Manager, there are some additional steps that are required:

- 1 Copy the `config/activemqusers.properties` file from your Sentinel RD server into the `config` directory in your remote installation.
- 2 Change the localhost part of the *Broker URL* parameter for the Collector to the IP address of the Sentinel RD server.

7.5 Starting the Collector

You must start the collector before the driver is started. When the collector is started, the JNDI destinations are created. The driver looks for these JNDI destinations and if they do not exist the driver cannot start. Start the collector before starting the Sentinel driver.

To start the collector:


- 1 In the Event Source Management live view, right-click the Identity Vault collector.
- 2 Click *Start* to start the Collector.

7.6 Starting the Sentinel Driver

After the Sentinel driver has been created and the collector has been started, you must start the Sentinel driver. Identity Manager is an event-driven system, so after the driver is started, it waits for events for events to occur.

IMPORTANT: The Identity Vault collector must be started before the driver is started. When the collector starts, the JNDI destinations are created. The driver looks for the JNDI destinations when it starts and if they do not exist, the driver cannot start. To start the collector, see [Section 7.5, “Starting the Collector,”](#) on page 30.

To start the driver after the additional configuration is completed and the Identity Vault Collector is created:

- 1 In Designer, open your project.
- 2 In the Modeler, right-click the driver icon  or the driver line, then select *Live > Start Driver*.

For information about management tasks with the driver, see [Chapter 10, “Managing the Driver,”](#) on page 43.

To verify that the collector and the driver are working properly:

- 1 Create a user in iManager to use a test.
- 2 Run the Identity Browser in the Sentinel RD server (via Control Center) and confirm that your user is found.

If your user is not found, then turn on tracing on the Sentinel driver to see if there are any errors occurring, particularly at start time.
- 3 At this point, you should be able to test Identity Tracking for all the drivers for which you have Account Tracking turned on.

Configuring Multiple Instances of the Sentinel Driver

8

If you have more than one instance of the Sentinel driver, additional configuration is required. Each driver instance must have a unique identifier value.

- 1** Set a unique value in the Sentinel Driver Instance Identifier GCV. For more information, see [“Sentinel Driver Instance Identifier” on page 52](#).

We recommend using 1, 2, 3, etc. for the unique identifier value.

- 2** If you are using Sentinel RD, skip to [Step 3](#). Otherwise, create three new queues based on the Sentinel Driver Instance Identifier.

- 2a** In the Sonic Management Console, select *Configuration* tab, then expand the *Brokers* folder.

- 2b** Expand *esecBroker*, then select *Queues*.

- 2c** Right-click a queue, then select *New Queue*.

- 2d** Specify `pubReceiveEvent-n` in the *Name* field.

The *n* is the unique identifier value that you set in [Step 1](#).

- 2e** Click *OK* to create the new queue.

- 2f** Repeat [Step 2c](#) through [Step 2e](#) twice more. Use the names of `pubReceiveEventResponse-n` and `subReceiveResponse-n` for each of the new queues.

The *n* is the unique identifier value that you set in [Step 1](#).

- 2g** Close the Management Console.

- 3** Create an additional Identity Vault Collector.

The additional Identity Vault Collector must contain the same unique identifier specified in the Sentinel driver. For more information on creating an Identity Vault Collector, see [Section 7.3, “Configuring the Identity Vault Collector,” on page 26](#).

You do not need to create a new connection factory. The existing connection factory can share the connection with multiple queues, as long as the queues contain the unique identifier.

Custom Audit Events

9

This section contains a list of the custom audit events that are generated by policies in each driver. These events are sent to the Identity Manager Collector. It parses the events and stores this information in the Sentinel data store.

These events are used to inject business relevance instead of the sending raw data events. This allows you to verify that your business policies and processes are being enforced.

In the past, Sentinel tracked Add, Delete, and Modify events. Sentinel could report on how many events occurred, but not if that event was supposed to occur. The custom events track granting and revoking of entitlements. The entitlements generate Add, Delete, or Modify events. Sentinel tracks which entitlement generated the Add event, and the reports show when and why an Add event occurred, instead of just when an Add event occurred.

Figure 9-1 represents the common components that make up the event structure. Each item in the illustration is part of an event. The different items are tracked to verify the uniqueness of the event.

Figure 9-1 Components of the Event Structure

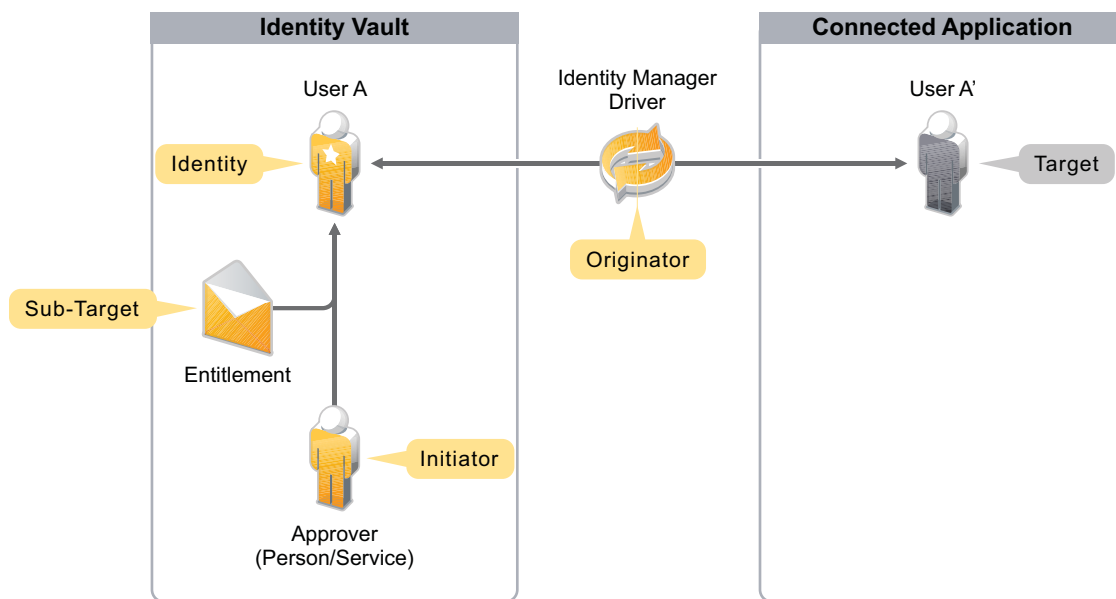


Table 9-1 contains the general event structure. The defined events are in the `dirxml_custom.lsc` file that is on the Identity Manager 3.6 media.

Table 9-1 General Event Structure

Descriptive Name	Description	Format	Audit Field Name	Sample Data
Audit Event ID	1200-1299	Int/Hex		

Descriptive Name	Description	Format	Audit Field Name	Sample Data
Version	Sequential number incremented by one whenever the event structure changes.	Int	Value 3 (3)	
Originator	Always the driver DN.	String	Originator (B)	
Target	Object (account) in the connected application.	String	Target (U)	
Target Type	0=None 1=DN in Slash Notation 2=DN in Dot Notation 3=DN in LDAP Notation 4=Association	Int	targetType (V)	
Sub Target	Entitlements/attribute name.	String	Sub-Target (Y)	
Status	Identity Manager status.	Int	value (1)	0=success 1=retry 2=warning 3=error 4=fatal
IDM Event ID	@event-id from XDS document	String	Text 3 (F)	
Identity	GUID	B64 encoded octet string value	Text 1 (S)	

The following events are defined:

- ♦ [“EventID 000304B0” on page 36](#)
- ♦ [“EventID 000304B1” on page 37](#)
- ♦ [“EventID 000303B2” on page 38](#)
- ♦ [“EventID 000304B3” on page 39](#)
- ♦ [“EventID 000304CE” on page 39](#)
- ♦ [“EventID 000304D9” on page 40](#)

EventID 000304B0

This is the Account Create By Entitlements Grant. The following table contains the fields of this EventID, with the proper values.

Table 9-2 Account Create By Entitlements Grant

Fields	Values
Originator (B) Title	Driver DN
Target (U) Title	Target account DN or the association
Subtarget (V) Title	Entitlement
Text1 (S) Title	Source Identity DN or GUID
Text2 (T) Title	Detail
Text3 (F) Title	Identity Manager EventID
Value1 (1) Title	Status
Value1 Type	N
Value2 (2) Title	
Value2 Type	
Value3 (3) Title	Version
Value3 Type	N
Group (G) Title	
Group Type	
Data (D) Title	XML Document
Data Type	S
Display Schema	[\$TC] \$SO: Account \$SU created by entitlement \$SV; Status:\$N1 Driver:\$SB from \$iR\n

EventID 000304B1

This is the Account Delete By Entitlements Revoke. The following table contains the fields of this EventID, with the proper values.

Table 9-3 Account Delete By Entitlements Revoke

Fields	Values
Originator (B) Title	Driver DN
Target (U) Title	Target account DN or the association
Subtarget (V) Title	Entitlement
Text1 (S) Title	Source Identity DN or GUID
Text2 (T) Title	Detail
Text3 (F) Title	Identity Manager EventID
Value1 (1) Title	Status

Fields	Values
Value1 Type	N
Value2 (2) Title	
Value2 Type	
Value3 (3) Title	Version
Value3 Type	N
Group (G) Title	
Group Type	
Data (D) Title	XML Document
Data Type	S
Display Schema	[\$TC] \$SO: Account \$SU deleted by entitlement \$SV; Status:\$N1 Driver:\$SB from \$iR\n

EventID 000303B2

This is the Account Disabled By Entitlements Revoke. The following table contains the fields of this EventID, with the proper values.

Table 9-4 Account Disabled By Entitlements Revoke

Fields	Values
Originator (B) Title	Driver DN
Target (U) Title	Target account DN or the association
Subtarget (V) Title	Entitlement
Text1 (S) Title	Source Identity DN or GUID
Text2 (T) Title	Detail
Text3 (F) Title	Identity Manager EventID
Value1 (1) Title	Status
Value1 Type	N
Value2 (2) Title	
Value2 Type	
Value3 (3) Title	Version
Value3 Type	N
Group (G) Title	
Group Type	
Data (D) Title	XML Document

Fields	Values
Data Type	S
Display Schema	[\$TC] \$SO: Account \$SU disabled by entitlement \$SV; Status:\$N1 Driver:\$SB from \$iR\n

EventID 000304B3

This is the Account Enable By Entitlements Grant. The following table contains the fields of this EventID with the proper values.

Table 9-5 Account Enable By Entitlements Grant

Fields	Values
Originator (B) Title	Driver DN
Target (U) Title	Target account DN or the association
Subtarget (V) Title	Entitlement
Text1 (S) Title	Source Identity DN or GUID
Text2 (T) Title	Detail
Text3 (F) Title	Identity Manager EventID
Value1 (1) Title	Status
Value1 Type	N
Value2 (2) Title	
Value2 Type	
Value3 (3) Title	Version
Value3 Type	N
Group (G) Title	
Group Type	
Data (D) Title	XML Document
Data Type	S
Display Schema	[\$TC] \$SO: Account \$SU enabled by entitlement \$SV; Status:\$N1 Driver:\$SB from \$iR\n

EventID 000304CE

This is the Driver Health State Change. The following table contains the fields of this EventID, with the proper values.

Table 9-6 *Driver Health State Change*

Fields	Values
Originator (B) Title	Driver DN
Target (U) Title	
Subtarget (V) Title	
Text1 (S) Title	
Text2 (T) Title	
Text3 (F) Title	
Value1 (1) Title	Status
Value1 Type	N
Value2 (2) Title	
Value2 Type	
Value3 (3) Title	Version
Value3 Type	N
Group (G) Title	
Group Type	
Data (D) Title	
Data Type	
Display Schema	[\$TC] \$SO: Account \$SU enabled by entitlement \$SV; Status:\$N1 Driver:\$SB from \$iR\n

EventID 000304D9

This is a Generic Event. The following table contains the fields of this EventID with the proper values.

Table 9-7 *Generic Event*

Fields	Values
Originator (B) Title	Driver DN
Target (U) Title	Target Object DN
Subtarget (V) Title	Object Class
Text1 (S) Title	Source Identity DN
Text2 (T) Title	Detail
Text3 (F) Title	Identity Manager EventID
Value1 (1) Title	Status

Fields	Values
Value1 Type	N
Value2 (2) Title	
Value2 Type	
Value3 (3) Title	Version
Value3 Type	N
Group (G) Title	
Group Type	
Data (D) Title	XML Document
Data Type	S
Display Schema	[\$TC] \$SO: Event: \$ST; Src DN: \$SS; Object: \$SU

Managing the Driver

10

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

- ◆ Starting and stopping the driver
- ◆ Viewing the driver versioning information
- ◆ Using Named Passwords to securely store passwords associated with the driver
- ◆ Monitoring the driver's health status
- ◆ Backing up the driver
- ◆ Inspecting the driver's cache files
- ◆ Viewing the driver's statistics
- ◆ Using the DirXML[®] Command Line utility to perform management tasks through scripts
- ◆ Securing the driver and its information

Because these tasks, as well as several others, are common to all Identity Manager drivers, they are included in one reference, the *Identity Manager Common Driver Administration Guide* (http://www.novell.com/documentation/idm401/idm_common_driver/?page=/documentation/idm401/idm_common_driver/data/front.html).

Use the following sections to troubleshoot the different components.

- ♦ [Section 11.1, “Troubleshooting the Sentinel Driver,” on page 45](#)
- ♦ [Section 11.2, “Troubleshooting the Identity Vault Collector,” on page 45](#)
- ♦ [Section 11.3, “Account Tracking Information Is Not Written to the Sentinel Server,” on page 45](#)
- ♦ [Section 11.4, “Error -9005 Sentinel Driver Does Not Start,” on page 46](#)
- ♦ [Section 11.5, “Error Occurs when Uninstalling the Driver,” on page 46](#)

11.1 Troubleshooting the Sentinel Driver

Viewing driver processes is necessary to analyze unexpected behavior. To view the processes, use DTrace. You should only use it during testing and troubleshooting the driver. Running DTrace while the drivers are in production increases the utilization on the Identity Manager server and can cause events to process very slowly. For more information, see the discussion on viewing Identity Manager processes in the *Identity Manager Common Driver Administration Guide* (http://www.novell.com/documentation/idm401/idm_common_driver/?page=/documentation/idm401/idm_common_driver/data/front.html).

11.2 Troubleshooting the Identity Vault Collector

To verify that the Identity Vault Collector is working:

- 1 Verify that the Identity Vault Collector is started.
- 2 Verify that the Sentinel driver is started.
- 3 Create an account in the Identity Vault.
- 4 Open the Identity Browser in Sentinel and verify that the account information is populated in the Identity Browser.

11.3 Account Tracking Information Is Not Written to the Sentinel Server

Some times the account tracking information is not written to the Sentinel™ server. When the Sentinel driver attempts to write messages to the JMS destination of the Sentinel server, it tries to verify the hostname of the target system. If the Sentinel driver cannot verify the hostname of the Sentinel server (either through regular DNS or an entry in the Identity Vault server `/etc/hosts` file), the Sentinel driver fails to write the account tracking information to the Sentinel server and no Identities are sent or processed on the Sentinel server.

The Sentinel driver reports the error `javax.jms.JMSEException: java.net.UnknownHostException` followed by the JMS connection information as seen through `ndstrace` or `iMonitor`. This error message contains the hostname of the Sentinel server to which the Sentinel driver is attempting to connect.

The solution to this problem is to enter a valid A record in the nameserver that your Identity Vault server is using, or make the appropriate address name entry in the Identity Vault `/etc/host` file.

11.4 Error -9005 Sentinel Driver Does Not Start

The -9005 error occurs when the driver Sentinel Driver Instance Identifier GCV value is different from the Sonic queue names. Each instance of the Sentinel driver must have a unique identifier that is tied to the Sonic queue names.

The error that is displayed in the driver trace log is:

```
DirXML Log Event -----
  Driver:   \novell\system\services\idm\driverset1\Sentinel
  Channel:  Publisher
  Status:   Fatal
  Message:  Code(-9005) The driver returned a "fatal" status indicating that
the driver should be shut down. Detail from driver:
<description>javax.naming.NameNotFoundException: /pubReceiveEventResponse-1
not
found in the specified context</description>
<exception class-name="javax.naming.NameNotFoundException">
  <message>/pubReceiveEventResponse-1 not found in the specified
context</message>
```

If you have more than one Sentinel driver, you must add a value to the Sentinel Driver Instance Identifier GCV. This makes each instance of the driver unique. The Sonic queue names must end in the number specified in the GCV for the connection between the driver and the Sonic queue to work. For more information, see [Chapter 8, “Configuring Multiple Instances of the Sentinel Driver,” on page 33](#).

11.5 Error Occurs when Uninstalling the Driver

If you have installed the Sentinel driver on a server that does not have a Java* Virtual Machine* (JVM*) installed on it, you receive the following error when trying to uninstall the driver.

```
No Java virtual machine could be found from your PATH
environment variable. You must install a VM prior to
running this program.
```

The problem only occurs if you install the Sentinel driver on a server that does not have Identity Manager or the Remote Loader installed on it.

The work around is to install the driver on a server with Identity Manager or the Remote Loader, or install the JVM and add the installation location to the PATH variable.

Linux/UNIX: To add the JVM to the PATH variable:

- 1 From a command line, enter `export PATH=<JAVA-HOME-PATH>/bin/:$PATH`.
- 2 Run the uninstall script for the Sentinel driver, where the JAVA-HOME-PATH is the Java or JRE installation location.


Windows: To add the JVM to the PATH variables, use the following command:

```
"Uninstall_NIDm_Integration_Module_for_Sentinel.exe" LAX_VM "<JAVA-HOME-
PATH>\bin\java.exe"
```

Driver Properties

A


This section provides information about the Driver Configuration and Global Configuration Values properties for the Sentinel™ driver. These are the only unique properties for the Sentinel driver. All other driver properties (Named Password, Engine Control Values, Log Level, and so forth) are common to all drivers. Refer to the section on driver properties in the *Identity Manager Common Driver Administration Guide* (http://www.novell.com/documentation/idm40/idm_common_driver/data/b94pq23.html) for information about the common properties.

The information is presented from the viewpoint of iManager. If a field is different in Designer, it is marked with an  icon.


- ♦ [Section A.1, “Driver Configuration,” on page 47](#)
- ♦ [Section A.2, “Global Configuration Values,” on page 51](#)

A.1 Driver Configuration

In Designer:

- 1 Open a project in the Modeler.
- 2 Right-click the driver icon  or line, then click *Properties > Driver Configuration*.

In iManager:

- 1 In iManager, click  to display the Identity Manager Administration page.
- 2 Open the driver set that contains the driver whose properties you want to edit:
 - 2a In the *Administration* list, click *Identity Manager Overview*.
 - 2b If the driver set is not listed on the *Driver Sets* tab, use the *Search In* field to search for and display the driver set.
 - 2c Click the driver set to open the Driver Set Overview page.
- 3 Locate the *Sentinel* driver icon, then click the upper right corner of the driver icon to display the *Actions* menu.
- 4 Click *Edit Properties* to display the driver’s properties page.



The Driver Configuration options are divided into the following sections:

- ♦ [Section A.1.1, “Driver Module,” on page 47](#)
- ♦ [Section A.1.2, “Driver Object Password \(iManager Only\),” on page 48](#)
- ♦ [Section A.1.3, “Authentication,” on page 48](#)
- ♦ [Section A.1.4, “Startup Options,” on page 49](#)
- ♦ [Section A.1.5, “Driver Parameters,” on page 50](#)
- ♦ [Section A.1.6, “ECMAScript \(Designer Only\),” on page 51](#)

A.1.1 Driver Module

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

Table A-1 *Driver Modules*

Option	Description
<i>Java</i>	<p>Used to specify the name of the Java class that is instantiated for the shim component of the driver. This class can be located in the <code>classes</code> directory as a class file, or in the <code>lib</code> directory as a <code>.jar</code> file. If this option is selected, the driver is running locally.</p> <p>The name of the Java class is: <code>com.novell.nds.dirxml.driver.sentinel.SentinelShim</code></p>
<i>Connect to Remote Loader</i>	<p>Used when the driver is connecting remotely to the connected system. Designer includes two suboptions:</p> <ul style="list-style-type: none">◆  <i>Driver Object Password</i>: Specifies a password for the Driver object. If you are using the Remote Loader, you must enter a password on this page. Otherwise, the remote driver does not run. The Remote Loader uses this password to authenticate itself to the remote driver shim.◆  <i>Remote Loader Client Configuration for Documentation</i>: Includes information on the Remote Loader client configuration when Designer generates documentation for the Sentinel driver.

A.1.2 Driver Object Password (iManager Only)










Table A-2 *Driver Object Password*

Option	Description
<i>Driver Object Password</i>	<p>Use this option to set a password for the driver object. If you are using the Remote Loader, you must enter a password on this page or the remote driver does not run. This password is used by the Remote Loader to authenticate itself to the remote driver shim.</p>

A.1.3 Authentication

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

Table A-3 Authentication Options


Option	Description
<i>Authentication ID</i>	Not used in this driver.
<i>Authentication Context</i>	Not used in this driver.
or	
 <i>Connection Information</i>	
<i>Remote Loader Connection Parameters</i>	Used only if the driver is connecting to the application through the Remote Loader. The parameter to enter is hostname=xxx.xxx.xxx.xxx port=xxxx kmo=certificatename, when the hostname is the IP address of the application server running the Remote Loader server and the port is the port the Remote Loader is listening on. The default port for the Remote Loader is 8090.
or	
 <i>Host name</i>	
 <i>Port</i>	
 <i>KMO</i>	The kmo entry is optional. It is only used when there is an SSL connection between the Remote Loader and the Metadirectory engine.
 <i>Other parameters</i>	Example: hostname=10.0.0.1 port=8090 kmo=IDMCertificate
<i>Driver Cache Limit (kilobytes)</i>	Specify the maximum event cache file size (in KB). If it is set to zero, the file size is unlimited.
or	
 <i>Cache limit (KB)</i>	 Click <i>Unlimited</i> to set the file size to unlimited in Designer.
<i>Application Password</i>	Not used in this driver.
or	
 <i>Set Password</i>	
<i>Remote Loader Password</i>	Used only if the driver is connecting to the application through the Remote Loader. The password is used to control access to the Remote Loader instance. It must be the same password specified during the configuration of the Remote Loader on the connected system.
or	
 <i>Set Password</i>	

A.1.4 Startup Options

The startup options allow you to set the driver state when the Identity Manager server is started.

Table A-4 Startup Options

Option	Description
<i>Auto start</i>	The driver starts every time the Identity Manager server is started.
<i>Manual</i>	The driver does not start when the Identity Manager server is started. The driver must be started through Designer or iManager.
<i>Disabled</i>	The driver has a cache file that stores all of the events. When the driver is set to <i>Disabled</i> , this file is deleted and no new events are stored in the file until the driver state is changed to <i>Manual</i> or <i>Auto Start</i> .

Option	Description
 <i>Do not automatically synchronize the driver</i>	This option only applies if the driver is deployed and was previously disabled. If this is not selected, the driver re-synchronizes the next time it is started.

A.1.5 Driver Parameters

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

Table A-5 *Driver Parameters*

Parameter Name	Parameter Descriptions
Driver Name	<p>The actual name you want to use for the driver. This parameter is only available during the import of the driver configuration file.</p> <p>The associations for this driver are based on the driver name. If the driver object is renamed, all of the associations on each object are also renamed, and this can take a long time.</p>
Broker Type	<p>Select the type of broker you are using. The broker type is determined by the version of Sentinel you are using.</p> <p>Sentinel: Sentinel/Sonic MQ</p> <p>Sentinel RD: Sentinel RD/ActiveMQ</p>
Broker URL	<p>The URL (Uniform Resource Locator) for the Sentinel broker. The following are examples for the different versions of Sentinel. The ports listed are the default ports for the brokers.</p> <p>Sentinel: <code>tcp://xxx.xxx.xxx.xxx:10012</code></p> <p>Sentinel RD: <code>ssl://xxx.xxx.xxx.xxx:61616</code></p>
Broker Name	<p>The name of the user used to authenticate to the Sentinel broker. If you are connecting to a Sentinel system, use a random username. If you are connecting to a Sentinel RD system, use a valid ActiveMQ username. This field must not be blank.</p>
Broker Password	<p>The password of the authentication user for the Sentinel broker.</p>
Default message expiration (milliseconds)	<p>Determines how long a message lives in the destination. This setting is global for all messages.</p> <p>The default value of 0 means that the message lives indefinitely in the destination.</p>

Parameter Name	Parameter Descriptions
Default message expiration (millisecond)	Determines how long a message lives in the destination. This setting is global for all messages. The default value of 0 means that the message lives indefinitely in the destination.
Heartbeat interval (minutes)	The number of minutes of inactivity that elapse before the Publisher channel sends a heartbeat document. More than the specified number of minutes can elapse, because this parameter defines the lower bound.


A.1.6 ECMAScript (Designer Only)

Enables you to add ECMAScript resource files. The resources extend the driver's functionality when Identity Manager starts the driver.

A.2 Global Configuration Values

Global configuration values (GCVs) allow you to specify settings for the Identity Manager features such as driver heartbeat, as well as settings that are specific to the function of an individual driver configuration.

In Designer:

- 1 Open a project in the Modeler.
- 2 Right-click the driver icon  or line, then select *Properties > Global Configuration Values*.

In iManager:


- 1 In iManager, click  to display the Identity Manager Administration page.
- 2 Open the driver set that contains the driver whose properties you want to edit:
 - 2a In the *Administration* list, click *Identity Manager Overview*.
 - 2b If the driver set is not listed on the *Driver Sets* tab, use the *Search In* field to search for and display the driver set.
 - 2c Click the driver set to open the Driver Set Overview page.
- 3 Locate the Sentinel driver icon, then click the upper right corner of the driver icon to display the *Actions* menu.
- 4 Click *Edit Properties* to display the driver's properties page.

Table A-6 *Global Configuration Values*

Global Configuration Values	Descriptions
Sentinel Driver Instance Identifier	<p>The unique identifier for each Sentinel Driver. When multiple Sentinel drivers are required, the instance identifier is appended to the queue names to guarantee uniqueness.</p> <p>If you have more than one driver, add a value here for the increased drivers. The Sonic queues names must end with the same number as specified in this field.</p> <p>Only change this parameter here. If it is changed in any other location, the change is not persistent.</p>