AUTHORIZED DOCUMENTATION

Driver for Sentinel™ 6.1 and the Identity Vault Collector Implementation Guide

Novell®
Identity Manager

3.6

November 25, 2008

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/) for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

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/) 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).

Novell Trademarks

For Novell trademarks, see the Novell Trademark and Service Mark list (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

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

Audience

This guide is intended for administrators implementing Identity Manager and Sentinel.

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 3.6 and Sentinel 6.1 Integration Guide*, visit the XYZ Web site (http://www.novell.com/documentation/gw65).

Additional Documentation

For documentation on Identity Manager, see the Identity Manager Documentation Web site (http://www.novell.com/documentation/idm36).

For documentation on Sentinel, see the Novell Sentinel Documentation Web site (http://www.novell.com/documentation/sentinel61/index.html).

Documentation Conventions

In Novell documentation, a greater-than symbol (>) is used to separate actions within a step and items in a cross-reference path.

A trademark symbol (®, ™, etc.) denotes a Novell trademark. An asterisk (*) denotes a third-party trademark.

When a single pathname can be written with a backslash for some platforms or a forward slash for other platforms, the pathname is presented with a backslash. Users of platforms that require a forward slash, such as Linux* or UNIX*, should use forward slashes as required by your software.

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 that account. For more information, see Section 1.2, “How It Works,” on page 11.

The account tracking adds infrastructure that allows you to rapidly solve complex business problems. Some of the different business problems the account tracking solves are:

- How to validate which corporate policies are in place and are being enforced.
- How to monitor if terminated employees try to access corporate resources, and define what action is taken when this occurs.
- How to monitor for rogue administration and define what action is taken when this occurs.

The Sentinel driver and the Identity Vault Collector are used to solve these and many more problems your business is facing. For a complete list of solutions, see the *Novell Compliance Management Platform1.0 Integration Guide*.

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

1.1 Components of Account Tracking

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

1.1.1 DirXML-Accounts Attribute

The DirXML-Accounts attribute tracks and stores the different account identifiers. 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 database. 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 database 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 user's identities in the systems managed by Identity Manager. Business policies can be validated with this information. For more information, see Section 5.1, "Configuring Account Tracking," on page 25.

1.1.2 Sentinel Driver

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

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 a specific Collector designed to provide a 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 Identity Vault Collector. They are specific audit events coming directly from the drivers in your Identity Manager system; they do not come from the Sentinel driver.

The events sent by the Sentinel driver to the Identity Vault collector are XDS events. The custom events are audit events that make Sentinel aware of the business logic part of the Identity Manager solution.

With this increased awareness, Sentinel can now produce compliance reports and send warnings when the business logic is violated. For example, in the past Sentinel could only understand that an Add event occurred. It did not know what that meant for the business logic. It did not know if that user was supposed to be added or not. It recorded that the Add occurred, but that was all. Now, if an Add occurs, Sentinel understands what business logic is in place and verifies if that user is entitled to be added or not. If the user is not entitled, Sentinel can then take action to let you know that the business policies are not being carried out.

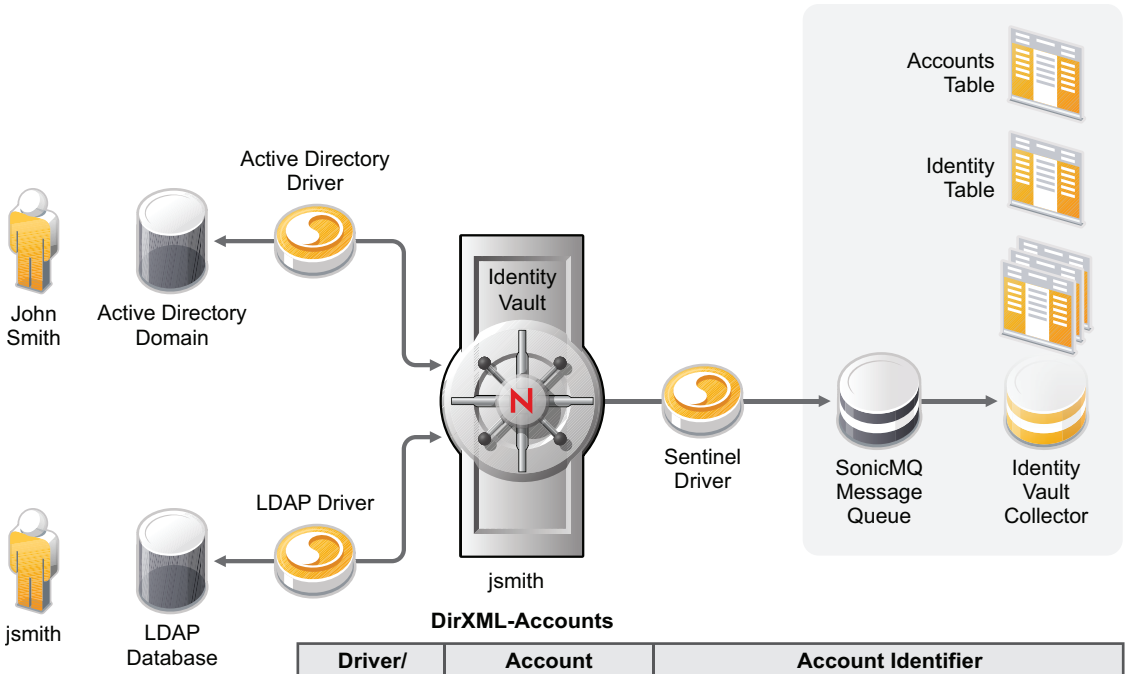
For more information, see Chapter 7, "Custom Audit Events," on page 35.

1.2 How It Works

Without the Sentinel driver or the Identity Vault collector, Sentinel receives information from each collector 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*



DirXML-Accounts

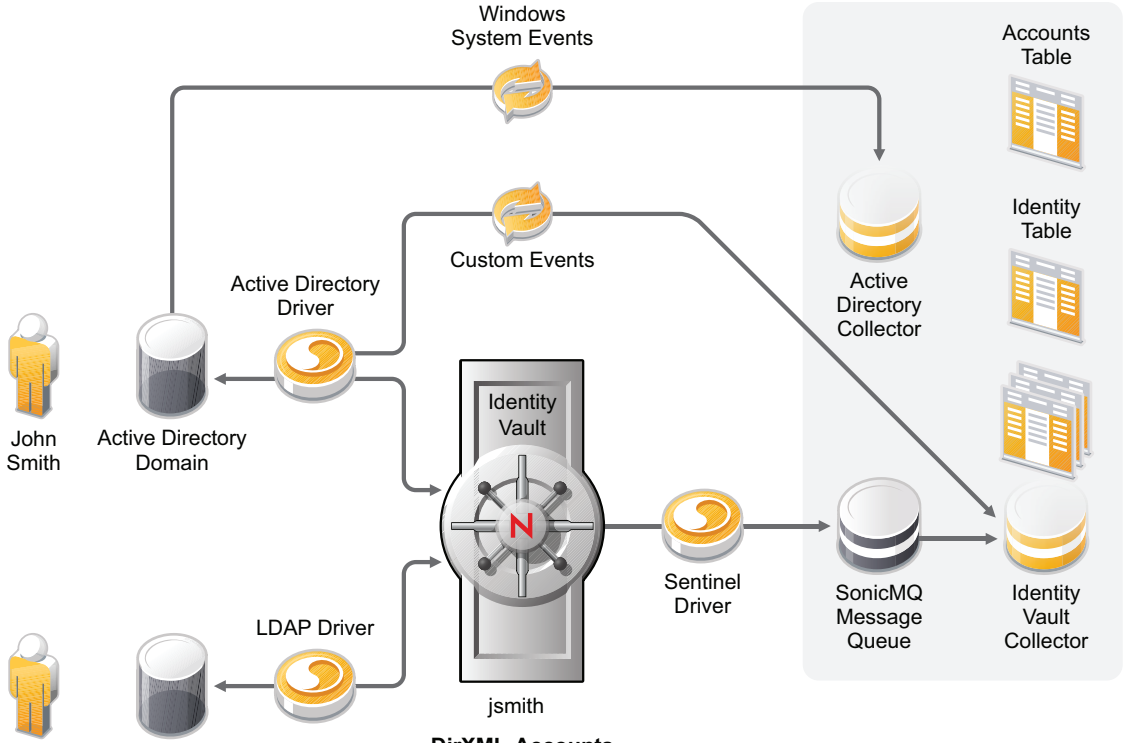
Driver/ Application	Account Identifier Type	Account Identifier Sample Data
AD	sAMAccountName	jsmith
AD	userPrincipalName	jsmith@company.com
AD	DN	cn=John Smith,cn=users,dc=company,dc=com
AD	association	5de77f84f3ab534babbf13edd6540d77
LDAP	DN	cn=jsmith,cn=users,dc=company,dc=com

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 SonicMQ* message queue that is part of Sentinel.
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 SonicMQ message queue.
8. The Identity Vault Collector reads the account data from the SonicMQ 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 other Sentinel Collectors use the information in the Identity table to verify that business policies are being enforced.

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*



DirXML-Accounts

Driver/ Application	Account Identifier Type	Account Identifier Sample Data
AD	sAMAccountName	jsmith
AD	userPrincipalName	jsmith@company.com
AD	DN	cn=John Smith,cn=users,dc=company,dc=com
AD	association	5de77f84f3ab534babbf13edd6540d77
LDAP	DN	cn=jsmith,cn=users,dc=company,dc=com

1. John Smith logs in to Active Directory, and that information is sent to Sentinel through the Active Directory Collector and through the Sentinel driver to the Identity Vault Collector.
2. 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.
3. 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.

4. The DirXML-Accounts attribute is updated with the new information, and the Sentinel driver detects this change.
5. The Sentinel driver synchronizes the new account information to the SonicMQ message queue.
6. The Identity Vault collector reads the new account information and writes it to the Identity table.
7. When John Smith logs in again to Active Directory, the Active Directory collector records the login information.
8. 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.
9. 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 7, "Custom Audit Events," on page 35.

   These policies are part of each driver that ships with Identity Manager 3.6.
10. When the custom events occur, they are sent directly to the Identity Vault collector. Sentinel records these events and then takes additional actions that are defined in the use cases.

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.

Checklist for the Complete Solution

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 3.6 and Sentinel™ 6.1 must be installed and configured before you proceed with any steps. See the *Identity Manager 3.6 Installation Guide* and the *Sentinel Installation Guide* (http://www.novell.com/documentation/sentinel61/index.html).

- ❑ Install the Sentinel driver. For instructions, see Chapter 3, "Installing the Sentinel Driver," on page 17.
- ❑ Copy the prerequisite `.jar` files to the Metadirectory engine or the Remote Loader. For instructions, see Section 3.1.3, "Placing Prerequisite Files," on page 18.
- ❑ Create and configure the Sentinel driver. For instructions, see Chapter 4, "Creating a New Driver," on page 19.
- ❑ Configure account tracking for the Active Directory driver and the LDAP driver. For instructions, see Section 5.1, "Configuring Account Tracking," on page 25.
- ❑ Verify that the Sentinel server is running. For instructions, see Section 5.2, "Starting the Sentinel Server," on page 27.
- ❑ Create the connection factories. For instructions see, Section 5.3, "Creating the Connection Factories," on page 27.
- ❑ Create message queues for the Sentinel driver. For instructions, see Section 5.4, "Creating Queues," on page 28.
- ❑ Create and configure the Identity Vault collector. For instructions, see Chapter 6, "Installing and Configuring the Identity Vault Collector," on page 31.
- ❑ Start the Sentinel driver. For instructions, see Section 4.1.4, "Using Designer to Start the Driver," on page 21 or Section 4.2.3, "Using iManager to Start the Driver," on page 24.

Installing the Sentinel Driver

3

The Sentinel™ driver is not included with base Identity Manager product, and therefore has a separate installation program. The following sections explain how to install one or more drivers.

3.1 Installing the Driver

Installing the driver requires three steps:

- Section 3.1.1, “Installing the Driver Files on the Metadirectory Engine,” on page 17
- Section 3.1.2, “Installing the Driver Files on the Remote Loader,” on page 17
- Section 3.1.3, “Placing Prerequisite Files,” on page 18

3.1.1 Installing the Driver Files on the Metadirectory Engine

Use the correct installation program for your platform:

Table 3-1 *Installation Programs*

Platform	File
Windows	`sentinel_driver_install.exe`
Linux	`./sentinel_driver_install_linux.bin`
Solaris*	`./sentinel_driver_install_solaris.bin`
AIX*	`./sentinel_driver_install_aix.bin`

The installation program detects the installation location of Metadirectory engine and places the driver files in this location. No information is required during the installation.

3.1.2 Installing the Driver Files on the Remote Loader

1 Use the correct installation program for your platform:

Platform	File
Windows	`sentinel_driver_install.exe`
Linux	`./sentinel_driver_install_linux.bin`
Solaris	`./sentinel_driver_install_solaris.bin`
AIX	`./sentinel_driver_install_aix.bin`

2 Specify the path to the `lib` directory. The default location is:

Platform	Location
Windows	`c:\Novell\RemoteLoader\lib`
Linux/UNIX	`/opt/novell/eDirectory/lib/dirxml`

3.1.3 Placing Prerequisite Files

The following the files must be copied into the correct directory for the driver to start.

1. On the Sentinel server, locate the following files:
   - `mfcontext.jar`
   - `sonic_Client.jar`

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

2. Copy these files to the Identity Manager server.

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

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

4

After the Sentinel™ driver files are installed on the server where you want to run the driver (see Chapter 3, "Installing the Sentinel Driver," on page 17), you can create the driver in the Identity Vault. You do so by importing the basic driver configuration file and then modifying the driver configuration to suite your environment. The following sections provide instructions:

- Section 4.1, "Using Designer to Create and Configure the Driver," on page 19
- Section 4.2, "Using iManager to Create and Configure the Driver," on page 21

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. For information about using iManager to accomplish these tasks, see Section 4.2, "Using iManager to Create and Configure the Driver," on page 21.

- Section 4.1.1, "Using Designer to Import the Driver Configuration File," on page 19
- Section 4.1.2, "Using Designer to Configure the Driver Settings," on page 20
- Section 4.1.3, "Using Designer to Deploy the Driver," on page 20
- Section 4.1.4, "Using Designer to Start the Driver," on page 21

4.1.1 Using Designer to Import the Driver Configuration File

Importing the Sentinel driver configuration file creates the driver in the Identity Vault and adds the policies needed to make the driver work properly.

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* to display the Driver Configuration Wizard.
3. In the Driver Configuration list, select *Sentinel*, then click *Run*.
4. On the Import Information Requested page, fill in the following fields:
   - **Driver Name:** Specify a name that is unique within the driver set.
   - **Broker URL:** Specify the IP address of the SonicMQ message queue with the default port of 10012. For example:

     ```
     tcp://localhost:10012
     ```
5. Click *Next* to import the driver configuration.
6. Click *Configure* to make additional configuration changes, or click *Close* to finish.

   If you require additional configuration for the driver, click *Configuration* to open the properties page of the driver. This is where the driver parameters are stored. For detailed information about all driver parameters, see Appendix A, "Driver Properties," on page 45.

4.1.2 Using Designer to Configure the Driver Settings

The information specified on the Import Information Requested page is the minimum information required to import the driver. However, the base configuration might not meet your needs.

- You might need to change whether the driver is running locally or remotely.
- You might need to change which broker the driver connects to.

The driver configuration settings are explained in Appendix A, "Driver Properties," on page 45.

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 and additional configuration procedures must be completed.

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.

- The connection factories must be created.
- The SonicMQ message queues must be created.
- The Identity Vault Collector must installed and configured.

See Chapter 5, "Additional Configuration," on page 25 for instructions on how to create the connection factories and message queues. For the Identity Vault Collector installation instructions, see Chapter 6, "Installing and Configuring the Identity Vault Collector," on page 31.

4.1.4 Using Designer to Start 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.

WARNING: 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 6.4, "Starting the Collector," on page 33.

To start the driver after the additional configuration is completed and the Identity Vault 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 8, "Managing the Driver," on page 41.

4.2 Using iManager to Create and Configure the Driver

The following sections provide steps for using iManager to create and configure a new Sentinel driver. For information about using Designer to accomplish these tasks, see Section 4.1, "Using Designer to Create and Configure the Driver," on page 19.

- Section 4.2.1, "Using iManager to Import the Driver Configuration File," on page 22
- Section 4.2.2, "Using iManager to Configure the Driver Settings," on page 23
- Section 4.2.3, "Using iManager to Start the Driver," on page 24

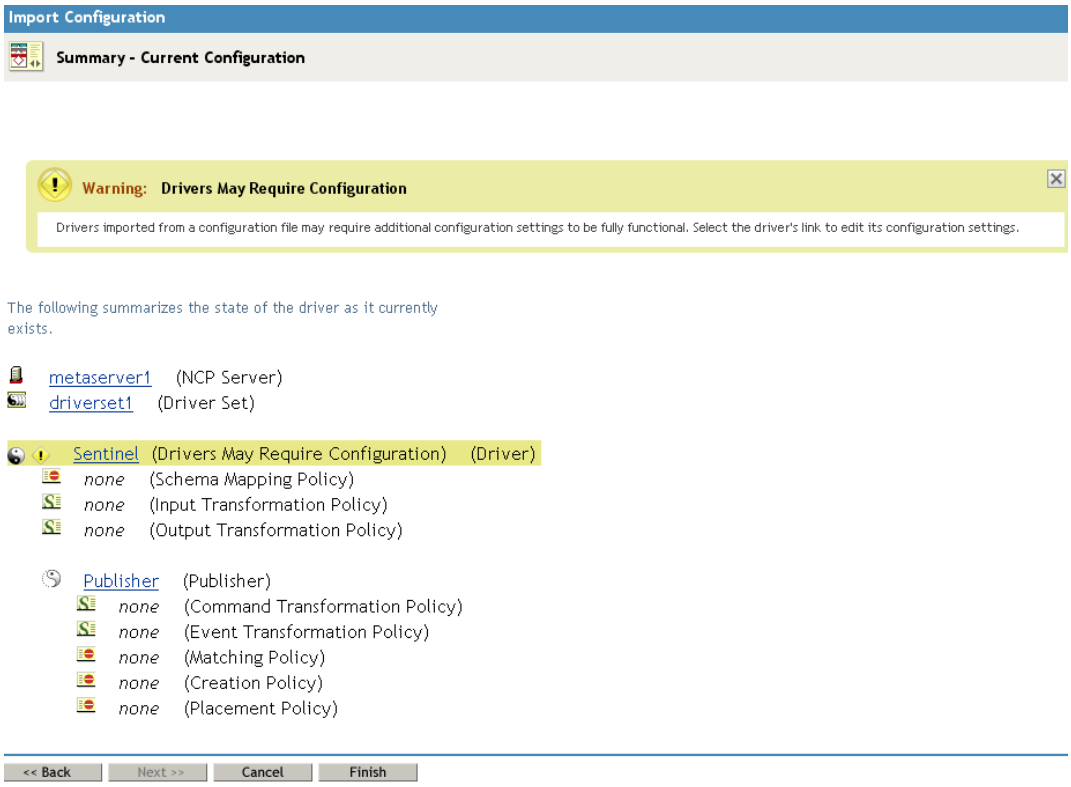
4.2.1 Using iManager to Import the Driver Configuration File

Importing the Sentinel driver configuration file creates the driver in the Identity Vault and adds the policies needed to make the driver work properly.

1. In iManager, click  to display the Identity Manager Administration page.
2. In the Administration list, click *Utilities* > *Import Configuration* to launch the Import Configuration Wizard.
3. Use the following information to complete the wizard and create the driver.

Prompt	Description
Where do you want to place the imported configuration?	You can add the driver to an existing driver set, or you can create a new driver set and add the driver to the new set. If you choose to create a new driver set, you'll be prompted to specify the name, context, and server for the driver set.
Import a configuration into this driver set	Use the default option, *Import a configuration from the server (.XML file)*. In the *Show* field, select *Identity Manager 3.6 configurations*. In the *Configurations* field, select the `Sentinel-IDM3_6_0-V1.xml` file.
Driver name	Specify a name that is unique within the driver set.
Broker URL	Specify the IP address of the SonicMQ message queue with the default port of 10012. For example: `tcp://localhost:10012`
Define Security Equivalences	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.
Exclude Administrative Roles	You should exclude any administrative User objects (for example, Admin and DriversUser) from synchronization.

When you finish providing the information required by the wizard, a Summary page similar to the following is displayed.



At this point, the driver is created from the basic configuration file. To ensure that the driver works the way you want it to for your environment, you must review and modify (if necessary) the driver's default configuration settings.

4 To modify the default configuration settings, click the linked driver name, then continue with the next section, Using iManager to Configure the Driver Settings.

or

To skip the configuration settings at this time, click *Finish*. When you are ready to configure the settings, continue with the next section, Using iManager to Configure the Driver Settings.

WARNING: Do not click Cancel on the Summary page. This removes the driver from the Identity Vault and results in the loss of your work.

4.2.2 Using iManager to Configure the Driver Settings

The information specified during the creation of the driver is the minimum information required to import the driver. However, the base configuration might not meet your needs.

- You might need to change whether the driver is running locally or remotely.
- You might need to change which broker the driver connects to.

The driver configuration settings are explained in Appendix A, "Driver Properties," on page 45.

To configure the settings:

1 Make sure the Modify Object page for the Sentinel driver is displayed in iManager. If it is not:

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

1b Click *Identity Manager Overview*.

1c Browse to and select the driver set object that contains the new Sentinel driver.

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

1e Click the upper right corner of the driver, then click *Edit properties*.

This displays the properties page of the driver.

2 Review the settings for the driver parameters, global configuration values, or engine control values. The configuration settings are explained in Appendix A, "Driver Properties," on page 45.

3 After modifying the settings, click *OK* to save the settings and close the Modify Object page.

4 (Conditional) If the Sentinel driver's Summary page for the Import Configuration Wizard is still displayed, click *Finish*.

WARNING: Do not click *Cancel* on the Summary page. This removes the driver from the Identity Vault and results in the loss of your work.

4.2.3 Using iManager to Start 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.

See Chapter 5, "Additional Configuration," on page 25 for instructions on how to create the connections factories and message queues. For the Identity Vault installation instructions, see Chapter 6, "Installing and Configuring the Identity Vault Collector," on page 31.

WARNING: 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 6.4, "Starting the Collector," on page 33.

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

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

2 Click *Identity Manager Overview*.

3 Browse to and select the driver object that contains the Sentinel driver you want to start.

4 Click the driver set name to access the Driver Set Overview page.

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

For information about management tasks with the driver, see Chapter 8, "Managing the Driver," on page 41.

Additional Configuration

5

After you create the driver and configure the basic settings, there is additional configuration that must be completed for the Sentinel™ driver to work. You must configure account tracking on the participating drivers, create connection factories, and specific queues for the Sentinel driver and the Identity Vault Collector to use. If you created the connection factories and queues when you installed and configured the Identity Vault Collector, complete the first section and skip the other sections.

5.1 Configuring Account Tracking

To configure account tracking, the schema must be extended with the DirXML-Accounts attribute and additional parameters configured for the drivers synchronizing the account information.

To enable account tracking, complete the following two tasks:

- Extend the schema by installing Identity Manager 3.6. If you have not installed Identity Manager 3.6, see the *Identity Manager 3.6 Installation Guide* for instructions.
- Enable account tracking on each driver used with the Sentinel driver. Currently, only the Active Directory driver and the LDAP driver are enabled to work with the Sentinel driver.

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

In Designer:

1. Open a project in the Modeler.
2. Right-click the driver icon  or line, then select click *Properties* > *GCVs*.
3. Set the *Account Tracking* > *Show Account Tracking Configuration* option to *show*.
4. Use the information in Table 5-1 to correctly enable account tracking.

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. To do so:
   - **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. Click the upper right corner of the driver icon, then click *Edit properties*.
4. Click the *Global Config Values* tab.

5 Set the *Account Tracking > Show Account Tracking Configuration* option to *show*.

6 Use the information in Table 5-1 to correctly enable account tracking.

Table 5-1 *Show Account Tracking Configuration Options*

Option	Description
Enable account tracking	Select *true* 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.
Identifiers	Specify each account identifier attribute. By default the attributes are prepopulated for each driver. Active Directory: ◆ sAMAccountName ◆ UserPrincipalName ◆ LDAPDN ◆ association LDAP: ◆ LDAPDN ◆ association
Status attribute	Specify the name of the attribute in the application namespace that represents the account status. By default the attributes are: ◆ **Active Directory:** dirxml-uACAccountDisable ◆ **LDAP:** loginDisabled
Status active value	The value of the status attribute that represents an active state. By default, the value is *false*.
Status inactive value	The value of the status attribute that represents an inactive state. By default, the value is *true*.
Subscription default status	The default status 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 *Active*.
Publication default status	The default status 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 *Uninitialized*.

5.2 Starting the Sentinel Server

The Sentinel server must be running and you must be logged into the Sonic* Management Console to create the connection factories and message queues.

1. Start Sentinel by entering the following command at a command prompt:
   - **Linux:** `/etc/init.d/ sentinel start`
   - **Windows:** `Net start sentinel`
2. Connect to the Sentinel server and run 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 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. Click *OK*.
5. Continue with Section 5.3, “Creating the Connection Factories,” on page 27.

5.3 Creating the Connection Factories

To create the connection factories:

1. From the Sonic Management Console toolbar, click *Tools* > *JMS Administered Objects*.
2. 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 in 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.
3. Click *Connect*.
4. Select the localhost:10012 entry in the tree on the left, then select the *Connection Factories* tab.
5. Click *New*.
6. Specify `TopicConnectionFactory` in the *Lookup Name* field.

   The connection factory name must be the specified name.
7. Specify `ConnectionFactory` in the *Factory Type* field.
8. Specify `tcp://`*`ipaddress`*`:10012` in the *Connection URL* field.

   The *ipaddress* is the IP address of your Sentinel server.

9 Click *Update* to save the information.

10 Repeat Step 5 through Step 9, but use `QueueConnectionFactory` as the *Lookup Name*.

11 Close the JMS Administered Objects dialog box.

12 Continue with Section 5.4, "Creating Queues," on page 28.

5.4 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.

5.5 Additional Configuration for Multiple Sentinel Drivers

If you have multiple instances of the Sentinel driver, you must complete additional configuration. Each driver instance must have a unique identity.

1 Set a unique value in the Sentinel Driver Instance Identifier GCV.

It is recommended to use 1, 2, 3 etc for a unique identifier value.

2 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.

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.

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 6.3, “Configuring the Identity Vault Collector,” on page 31.

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.

Installing and Configuring the Identity Vault Collector

6

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

6.1 Prerequisites

- ❑ Install Identity Manager 3.6 on a server in your environment. Follow the “Basic Identity Manager System Checklist” in the *Identity Manager 3.6 Installation Guide* to install Identity Manager.
- ❑ Create and configure the Sentinel™ driver. For more information, see the Chapter 4, “Creating a New Driver,” on page 19 and Chapter 5, “Additional Configuration,” on page 25.
- ❑ Install and configure the different Sentinel components. For more information, see the *Novell Sentinel Installation Guide* (http://www.novell.com/documentation/sentinel6/pdfdoc/sentinel60_installationguide.pdf).

6.2 Importing the Identity Vault Collector

The Identity Vault Collector must be added to the Event Source Manager to be installed. 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_6.1r1.clz.zip`) from the Customer Center Web site (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_6.1r1.clz.zip` file, then click *Next*.
5. Follow the remaining prompts, then click *Finish*.

6.3 Configuring the Identity Vault Collector

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 Value* in the *Name* column, then click *Next*.
4. In the *Installed Scripts* column, select *Novell_Identity_Manager_6.1r1*, then click *Next*.

5 Configure the Identity Vault Collector for your needs by using the following information:

Configuration Parameter	Default Value	Description
Event Source Time Zone	+0000	Sets the time zone offset UTC (+0000) of the event source data time stamps. This is used if the source data is reported only in local time with no time zone indicated. The format is + or - followed by a two-digit hour and minute offset.
Execution Mode	release	Sets the execution mode for the collector. There are three options: ◆ **release:** Use this mode for normal operation. ◆ **custom:** Use this mode if the Identity Manager Collector is customized. ◆ **debug:** Use this mode when troubleshooting issues. It generates debug trace files.
MSSP Customer Name		
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 with the collector script.
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.
iSCALE Connection URL	localhost:10012	The URL that the Identity Vault Collector uses to retrieve identity events stored in the SonicMQ message queue.

6 Click *Next*.

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

- **Name:** Specify a name for this connector.
- **Run:** Select whether the connector 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 Connector in the specified time period.
- **Limit Data Rate:** (Optional) Select this option to set a maximum limit on the rate of data the connector 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 connector.
- **Trust Event Source Time:** (Optional) Select this option if you trust the Event Source server's time.

8 Click *Finish*.

6.4 Starting the Collector

The collector must be started 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.

To start the Sentinel driver, see Section 4.1.4, "Using Designer to Start the Driver," on page 21 or Section 4.2.3, "Using iManager to Start the Driver," on page 24.

Custom Audit Events

7

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

These events are use to trace the business logic instead of the raw data events, so you can verify that your business policies and processes are being enforced.

For example, in the past Sentinel could only understand that an Add event occurred. It did not know what that meant for the business logic. It did not know if that user was supposed to be added or not. It recorded that the Add occurred, but that was all. Now, if an Add occurs, Sentinel understands what business logic is in place and verifies if that user is entitled to be added or not. If the user is not entitled, Sentinel can then take action to let you know that the business policies are not being carried out.

7.1 General Event Structure

Figure 7-1 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 7-1 *Components of the Event Structure*

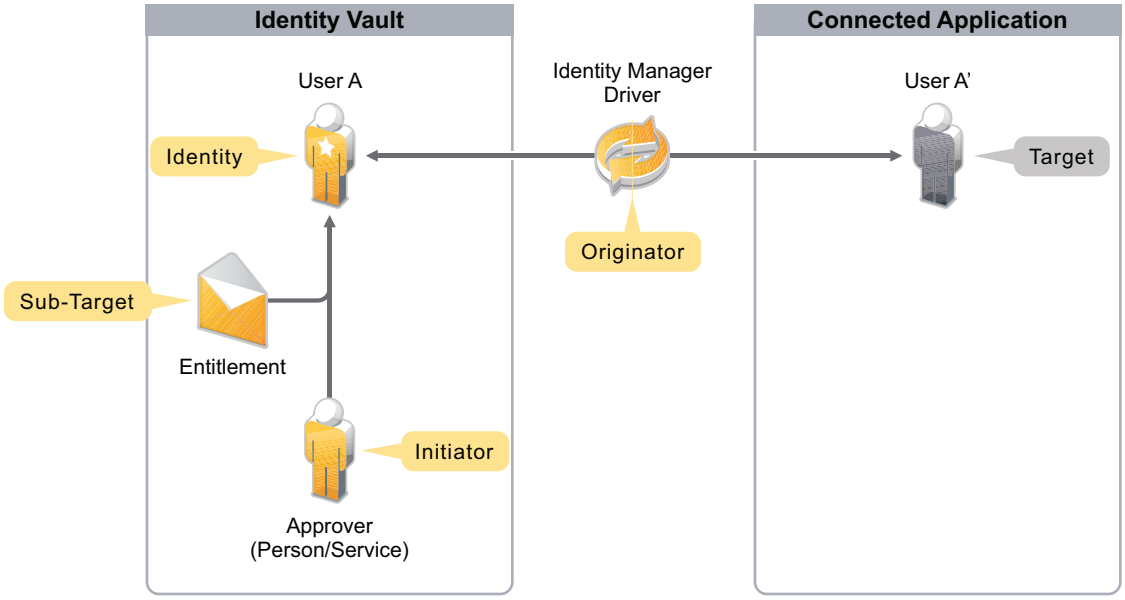


Table 7-1 contains the general event structure. The defined events are in the `dirxml_custom.lsc` file that is on the Identity Manager 3.6 media.

Table 7-1 *General Event Structure*

Descriptive Name	Description	Format	Audit Field Name	Sample Data
Audit Event ID	1200-1299	Int/Hex		
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	Int	targetType (V)	
	1=DN in Slash Notation			
	2=DN in Dot Notation			
	3=DN in LDAP Notation			
	4=Association			
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 00031200" on page 36
- "EventID 00031201" on page 37
- "EventID 00031202" on page 38
- "EventID 00031203" on page 39
- "EventID 00031230" on page 39
- "EventID 00031241" on page 40

EventID 00031200

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

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 00031201

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

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	

Fields	Values
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 00031202

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

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 disabled by entitlement $SV; Status:$N1 Driver:$SB from $iR\n

EventID 00031203

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

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 00031230

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

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

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

EventID 00031241

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

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
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

8

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 3.6 Common Driver Administration Guide*.

Troubleshooting

9

Use the following sections to troubleshoot the different components.

9.1 Troubleshooting the Sentinel Driver

Viewing driver processes is necessary to analyze unexpected behavior. To view the driver processing events, use DSTrace. You should only use it during testing and troubleshooting the driver. Running DSTrace while the drivers are in production increases the utilization on the Identity Manager server and can cause events to process very slowly. For more information, see “Viewing Identity Manager Processes” in the *Identity Manager 3.6 Common Driver Administration Guide*.

9.2 Troubleshooting the Identity Vault Collector

To verify that the Identity Vault Collector is working:

1. In the Event Source Management console, right-click an Event Source object, the select *Open Raw Data Tap*.
2. Log in to Active Directory as a user to generate events for the Active Directory Collector, the Sentinel driver, and the Identity Vault Collector.
3. Verify that the login information appears in the Raw Data Details window in the Event Source Management console.
4. Access the *Public:All Active View*:
   - **4a** Access the Sentinel Control Center.
   - **4b** Select *Active Views* in the toolbar, then click *Create Active View*.
   - **4c** In the Event Attribute drop-down list, select *Severity*.
   - **4d** In the Filter drop-down list, select *Public:ALL*, then click *Select*.
   - **4e** In the Display Events drop-down list, select *Yes*.
   - **4f** Click *Finish*.
5. Verify that the login information is parsed and is displayed in the Public:All Active View.

9.3 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 very 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 not Identities are sent or processed on the Sentinel server.

The Sentinel driver reports the error `javax.jms.JMSException: java.netUnknownHostException` 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.

9.4 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 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 Section 5.5, “Additional Configuration for Multiple Sentinel Drivers,” on page 28.

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 "Driver Properties" in the *Identity Manager 3.6 Common Driver Administration Guide* 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.

A.1 Driver Configuration

In Designer:

1. Open a project in the Modeler.
2. Right-click the driver icon or line, then select 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. To do so:
   - **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:

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
Java	Used to specify the name of the Java* class that is instantiated for the shim component of the driver. This class can be located in the `classes` directory as a class file, or in the `lib` directory as a `.jar` file. If this option is selected, the driver is running locally. The name of the Java class is: `com.novell.nds.dirxml.driver.sentinel.SentinelShim`
Connect to Remote Loader	Used when the driver is connecting remotely to the connected system. Designer includes two suboptions: ◆ *Driver Object Password*: Specifies a password for the Driver object. If you are using the Remote Loader, you must enter a password on this page. Otherwise, the remote driver does not run. The Remote Loader uses this password to authenticate itself to the remote driver shim. ◆ *Remote Loader Client Configuration for Documentation*: 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
Driver Object Password	Use this option to set a password for the driver object. If you are using the Remote Loader, you must enter a password on this page or the remote driver does not run. This password is used by the Remote Loader to authenticate itself to the remote driver shim.

A.1.3 Authentication

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

Table A-3 *Authentication Options*

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

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

Option	Description
Do not automatically synchronize the driver	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	The actual name you want to use for the driver. This parameter is only available during the import of the driver configuration file. 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.
Broker URL	The URL (Uniform Resource Locator) for the SonicMQ broker with the default port of 10012. For example: `tcp://10.0.0.2:10012`
Broker Name	The name of the user used to authenticate to the Sonic MQ broker. The default value is broker0.connection.Username. This field must not be blank.
Broker Password	The password of the authentication user for the SonicMQ broker.
Sentinel Driver Instance Identifier	Only change this field on the GCVs page. If the change is made here, the change is not persistent.
Default message expiration (milliseconds)	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.
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. To do so:
   - **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	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. 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. Only change this parameter here. If it is changed in any other location, the change is not persistent.