NetIQ® eDirectory™ 8.8 SP8

XDASv2 Administration Guide

September 2013

Legal Notice

Contents

About this Book and the Library 5
About NetIQ Corporation 7

1 Overview 9

1.1 Key Benefits . . . 9
1.2 XDASv2 Server Architecture . . . 9

2 Configuring XDASv2 11

2.1 Installing eDirectory and XDASv2 . . . 11
2.1.1 XDASv2 Files Installed with eDirectory . . . 11
2.1.2 Upgrading iManager Plugins For XDASv2 . . . 11
2.1.3 Configuring the XDASv2 Property File . . . 12
2.2 Configuring XDAS Events . . . 18
2.2.1 Loading and Unloading the Modules . . . 26
2.3 Enabling XDAS Event Caching . . . 27
2.4 Using Collectors for XDAS Events . . . 27

3 Understanding XDASv2 Auditing Event Filtering 29

3.1 Filtering XDASv2 Accounts . . . 29
3.2 Filtering XDASv2 Roles . . . 30

4 iManager Plug-In for XDASv2 31

4.1 System Requirements . . . 31
4.2 Installing the iManager Plug-In for XDASv2 . . . 31
4.3 Using the iManager Plug-In Console for XDASv2 . . . 31
4.4 Configuring XDASv2 Events for Auditing . . . 32
4.4.1 Configuring Events. . . . 32

5 XDASv2 Events 35

5.1 Account Management Events . . . 35
5.1.1 Examples for Account Management Events . . . 37
5.2 Session Management Events. . . . 40
5.2.1 Examples for Session Management Events . . . 41
5.3 Data Item and Resource Element Management Events . . . 41
5.3.1 Examples for Data Item and Resource Element Management Events . . . 44
5.4 Service or Application Management Events. . . . 46
5.4.1 Examples for Service or Application Management Events . . . 46
5.5 Service or Application Utilization Events . . . 47
5.5.1 Examples for Service or Application Utilization Events . . . 47
5.6 Trust Management Events . . . 48
5.6.1 Examples for Trust Management Events . . . 49
5.7 Peer Association Management Events . . . 50
5.7.1 Examples for Peer Association Management Events . . . 50
5.8 Data Item or Resource Element Content Access Events . . . 51
5.8.1 Examples for Data Item and Resource Element Management Events . . . 52

5.9 Role Management Events 53
5.9.1 Examples for Role Management Events 53
5.10 Exceptional Events. 55
5.10.1 Examples for Exceptional Events 55
5.11 Authentication Management Events. 56
5.11.1 Examples for Authentication Event 58
5.12 Operational Events. 59

6 Troubleshooting 61

A XDASv2 Schema 63

A.1 XDAS V2 JSON Schema 63
A.2 XDAS Field Definitions. 66
A.3 Outcome Codes 69
A.4 Example of an Event 69

About this Book and the Library

The *XDASv2 Administration Guide* describes how to configure and use XDASv2 to audit the NetIQ eDirectory and NetIQ Identity Manager.

For the most recent version of the *NetIQ XDASv2 Administration Guide*, see the NetIQ eDirectory 8.8 online documentation (https://www.netiq.com/documentation/edir88/) Web site.

Intended Audience

The guide is intended for network administrators.

Other Information in the Library

The library provides the following information resources:

Administration Guide

Describes how to manage and configure eDirectory.

Installation Guide

Describes how to install eDirectory. It is intended for network administrators.

Troubleshooting Guide

Describes how to resolve eDirectory issues.

Tuning Guide for Linux Platforms

Describes how to analyze and tune eDirectory on Linux platforms to yield superior performance in all deployments.

What’s New Guide

Describes the new features of eDirectory.

These guides are available at NetIQ eDirectory 8.8 documentation Web site (https://www.netiq.com/documentation/edir88/).

For information about the eDirectory management utility, see the *NetIQ iManager 2.7 Administration Guide* (https://www.netiq.com/documentation/imanager/).

About NetIQ Corporation

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

Our Viewpoint

Adapting to change and managing complexity and risk are nothing new

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

Enabling critical business services, better and faster

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

Our Philosophy

Selling intelligent solutions, not just software

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

Driving your success is our passion

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

Our Solutions

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

Contacting Sales Support

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

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

Contacting Technical Support

For specific product issues, contact our Technical Support team.

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

Contacting Documentation Support

Our goal is to provide documentation that meets your needs. If you have suggestions for improvements, click **Add Comment** at the bottom of any page in the HTML versions of the documentation posted at www.netiq.com/documentation. You can also email Documentation-Feedback@netiq.com. We value your input and look forward to hearing from you.

Contacting the Online User Community

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

1 Overview

The XDASv2 specification provides a standardized classification for audit events. It defines a set of generic events at a global distributed system level. XDASv2 provides a common portable audit record format to facilitate the merging and analysis of audit information from multiple components at the distributed system level. The XDASv2 events are encapsulated within a hierarchical notational system that helps to extend the standard or existing event identifier set.The XDASv2 taxonomy defines a set of fields, of these the primary fields are observer, initiator and target. XDASv2 events helps you easily understand the audit trails of heterogeneous applications

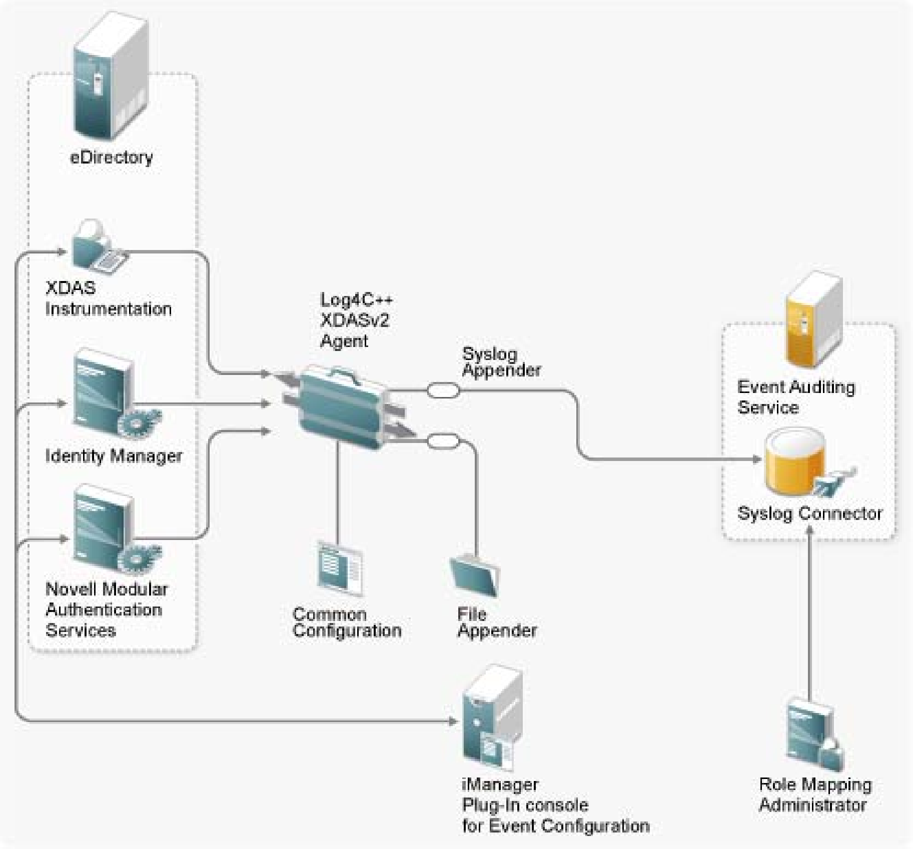
- Section 1.1, “Key Benefits,” on page 9
- Section 1.2, “XDASv2 Server Architecture,” on page 9

1.1 Key Benefits

- Provides secured audit services for a distributed system.
- Defines a set of generic events at a global distributed system level.
- Defines a common portable audit record format to help merge and analyze the audit information from multiple components of a distributed system.
- Defines a common format for audit events that analysis applications can use.
- Records XDASv2 audit trail.
- Configures event preselection criteria and event disposition actions.
- Provides a common audit format regardless of the platform on which the XDASv2 service is running.
- Supports heterogeneous environments without the necessity to re-engineer the current operating system or application-specific audit service implementations.
- Supports adequate separation of duties for users.
- Protects the audit log by making it accessible only to principals acting in specific administrative or security roles.
- Optionally caches audit events locally on the agent in case of communication failure between the agent and the auditing server and re-sends events when communication is re-established.

1.2 XDASv2 Server Architecture

Figure 1-1 XDASv2 Server Architecture



2 Configuring XDASv2

This chapter contains the following information:

- Section 2.1, “Installing eDirectory and XDASv2,” on page 11
- Section 2.2, “Configuring XDAS Events,” on page 18
- Section 2.3, “Enabling XDAS Event Caching,” on page 27
- Section 2.4, “Using Collectors for XDAS Events,” on page 27

2.1 Installing eDirectory and XDASv2

2.1.1 XDASv2 Files Installed with eDirectory

The following eDirectory XDASv2 files are, by default, installed as part of eDirectory.

- Linux
  - `novell-edirectory-xdaslog`
  - `novell-edirectory-xdaslog-conf`
  - `novell-edirectory-xdasinstrument`
- Windows
  - `xdasauditds.dlm`
  - `xdaslog.dll`

NOTE: From the OES 11 SP2 release, the XDAS RPMs are bundled with the Open Enterprise Server.

2.1.2 Upgrading iManager Plugins For XDASv2

You can upgrade the iManager Audit plugins to latest version.

1 Log in to the iManager console.

1a Open iManager from a Web browser, using the following URL:

```
https://ip_address_or_DNS/nps/iManager.html
```

where *ip_address_or_DNS* is the IP address or DNS name of your iManager server.

For example:

```
http://192.168.0.5/nps/iManager.html
```

1b Log in using your username and password.

In iManager, you have access only to those roles for which you have assigned rights. To have full access to all NetIQ iManager features, you must log in as a user with Admin rights to the tree.

For more information, see "Accessing iManager" in the *NetIQ iManager 2.7 Administration Guide*.

2 Select **Audit Configuration** from **Roles and Tasks**.

3 Click the **Upgrade XDAS Configuration** link.

An alert message about the upgrade process is displayed.

4 Click **Ok**.

During upgrade, new iManager files are installed and they cause configuration changes. After the upgrade completes, a message is displayed stating the success or failure status of the installation.

2.1.3 Configuring the XDASv2 Property File

When you install eDirectory, the installer lay down the `xdasconfig.properties.template` file in the `configdir` (`n4u.server.configdir`) directory.

Table 2-1 lists the default location of the `xdasconfig.properties` file in different operating systems.

***Table 2-1** XDAS Configuration File*

Operating System	File
Linux	`/etc/opt/novell/eDirectory/conf/xdasconfig.properties` For non-root installations, the XDASv2 property file is located in the `conf` directory.
Windows	`<Install Path>/novell/nds/xdasconfig` The property file is usually in the eDirectory installation directory.

If you configure the property file and then upgrade your environment to eDirectory 8.8 SP7, the installer does not replace it. Instead, it updates the file `(xdasconfig.properties.template`) to retain customization.

After you install iManager, you can configure XDAS. The XDAS configuration settings are stored in a simple text-based `xdasconfig.properties` configuration file. You can customize the file according to your requirements.

The following is the content of the XDASv2 property file:

Linux

```
# Set the level of the root logger to DEBUG and attach appenders.
#log4j.rootLogger=debug, S, R

# Defines appender S to be a SyslogAppender.
#log4j.appender.S=org.apache.log4j.net.SyslogAppender
```

```
# Defines location of Syslog server.
#log4j.appender.S.Host=localhost
#log4j.appender.S.Port=port

# Specify protocol to be used (UDP/TCP/SSL)
#log4j.appender.S.Protocol=UDP

# Specify SSL certificate file for SSL connection.
# File path should be given with double backslash.
#log4j.appender.S.SSLCertFile=/etc/opt/novell/mycert.pem

# Minimum log-level allowed in syslog.
#log4j.appender.S.Threshold=INFO

# Defines the type of facility.
#log4j.appender.S.Facility=USER

# Defines caching for SyslogAppender.
# Inputs should be yes/no
#log4j.appender.S.CacheEnabled=no

# Cache location directory
# Directory should be available for creating cache files
#log4j.appender.S.CacheDir=/var/opt/novell/eDirectory

# Cache File Size
# Cache File Size should be in the range of 50MB to 4000MB
#log4j.appender.S.CacheMaxFileSize=500MB

# Layout definition for appender Syslog S.
#log4j.appender.S.layout=org.apache.log4j.PatternLayout
#log4j.appender.S.layout.ConversionPattern=%c : %p%m%n

# Defines appender R to be a Rolling File Appender.
#log4j.appender.R=org.apache.log4j.RollingFileAppender

# Log file for appender R.
#log4j.appender.R.File=/var/opt/novell/eDirectory/log/xdas-events.log

# Max size of log file for appender R.
#log4j.appender.R.MaxFileSize=100MB

# Set the maximum number of backup files to keep for appender R.
# Max can be 13. If set to zero, then there will be no backup files.
#log4j.appender.R.MaxBackupIndex=10

# Layout definition for appender Rolling log file R.
#log4j.appender.R.layout=org.apache.log4j.PatternLayout
#log4j.appender.R.layout.ConversionPattern=%d{MMM dd HH:mm:ss} %c : %p%m%n
```

Windows

```
# Set the level of the root logger to DEBUG and attach appenders.
#log4j.rootLogger=debug, S, R

# Defines appender S to be a SyslogAppender.
#log4j.appender.S=org.apache.log4j.net.SyslogAppender

# Defines location of Syslog server.
#log4j.appender.S.Host=localhost
#log4j.appender.S.Port=port

# Specify protocol to be used (UDP/TCP/SSL)
#log4j.appender.S.Protocol=UDP
```

```
# Specify SSL certificate file for SSL connection.
# File path should be given with double backslash.
#log4j.appender.S.SSLCertFile=C:\\Novell\\mycert.pem

# Minimum log-level allowed in syslog.
#log4j.appender.S.Threshold=INFO

# Defines the type of facility.
#log4j.appender.S.Facility=USER

# Defines caching for SyslogAppender.
# Inputs should be yes/no
#log4j.appender.S.CacheEnabled=no

# Cache location directory
# Directory should be available for creating cache files
#log4j.appender.S.CacheDir=C:\\Novell\\NDS

# Cache File Size
# Cache File Size should be in the range of 50MB to 4000MB
#log4j.appender.S.CacheMaxFileSize=500MB

# Layout definition for appender Syslog S.
#log4j.appender.S.layout=org.apache.log4j.PatternLayout
#log4j.appender.S.layout.ConversionPattern=%c : %p%m%n

# Defines appender R to be a Rolling File Appender.
#log4j.appender.R=org.apache.log4j.RollingFileAppender

# Log file for appender R.
#log4j.appender.R.File=/var/opt/novell/eDirectory/log/xdas-events.log

# Max size of log file for appender R.
#log4j.appender.R.MaxFileSize=100MB

# Set the maximum number of backup files to keep for appender R.
# Max can be 13. If set to zero, then there will be no backup files.
#log4j.appender.R.MaxBackupIndex=10

# Layout definition for appender Rolling log file R.
#log4j.appender.R.layout=org.apache.log4j.PatternLayout
#log4j.appender.R.layout.ConversionPattern=%d{MMM dd HH:mm:ss} %c : %p%m%n
```

Table 2-2 *XDASv2 Property File*

Options	ID
Syslog Appender	S
Rolling File Appender	R

The entries in the `xdasconfig.properties` file are not case sensitive, entries can appear in any order, empty lines are valid, and any line that starts with a hash (#) is commented out.

The following table provides an explanation of each setting in the `xdasconfig.properties` file.

IMPORTANT: You must restart eDirectory any time you make a change to the configuration.

Table 2-3 XDAS Settings

Setting	Description
log4j.rootLogger=**debug, S, R**	Sets the level of the root logger to debug and attaches an appender named R or S, where S specifies a Syslog appender and R specifies a Rolling File appender.
log4j.appender.S=**org.apache.log4j.net.SyslogAppender**	Specifies the appender S to be a Syslog appender.
log4j.appender.S.Host=**localhost**	Specifies the location of the Syslog server where XDAS events are logged. IFor example, log4j.appender.S.Host=192.168.0.1
log4j.appender.S.Port=**port**	The port at which the XDAS connects to the Syslog server. The port supports values from 1 to 65535. If you specify an invalid value, the port defaults to 514. If the connection between XDAS and the Syslog server fails, Identity Manager cannot log events until the connection is restored.
log4j.appender.S.Protocol=UDP	Specifies the protocol to use. For example, UDP, TCP, or SSL.
log4j.appender.S.SSLCertFile=**/etc/opt/novell/mycert.pem**	Specifies the SSL certificate file for the SSL connection. Use double backslashes to specify the path of the file. This is an optional setting.
log4j.appender.S.Threshold=INFO	Specifies the minimum log level allowed in the Syslog appender. Currently, the INFO log level is supported.
log4j.appender.S.Facility=USER	Specifies the type of facility. The facility is used to try to classify the message.Currently, USER facility is supported. These values may be specified as upper or lower case characters.
log4j.appender.S.layout=**org.apache.log4j.PatternLayout**	Layout setting for Syslog appender.
log4j.appender.S.layout.ConversionPattern=**%c : %p%m%n**	Layout setting for Syslog appender. For information about the conversion patters and their descriptions, see logging.apache.org.
log4j.appender.R=**org.apache.log4j.RollingFileAppender**	Specifies appender R to be a Rolling File appender.
log4j.appender.R.File=**/var/opt/novell/eDirectory/log/xdas-events.log**	The location of the log file for a Rolling File appender.
log4j.appender.R.MaxFileSize=**100MB**	The maximum size, in MBs, of the log file for a Rolling File appender. Set this value to the maximum size that the client allows.

Setting	Description
log4j.appender.R.MaxBackupIndex=**10**	Specify the maximum number of backup files for a Rolling File appender. The maximum number of the backup files can be 10. A zero value means no backup files.
log4j.appender.R.layout=**org.apache.log4j.PatternLayout**	Layout setting for Rolling File appender.
log4j.appender.R.layout.ConversionPattern=**%d{MMM dd HH:mm:ss} %c : %p%m%n**	Layout setting for Rolling File appender. See Table 2-4 on page 16 for simple date format patterns. For information about the conversion patters and their descriptions, see logging.apache.org

The following examples illustrate the date and time patterns interpreted in the U.S. The given date and time are 2012-07-04 12:08:56 local time in the U.S. Pacific Time time zone.

Table 2-4 *Date and Time Pattern Example*

Date and Time Pattern	Result
"yyyy.MM.dd G 'at' HH:mm:ss z"	2012.07.04 AD at 12:08:56 PDT
"EEE, MMM d, "yy"	Wed, Jul 4, '01
"h:mm a"	12:08 PM
"hh 'o"clock' a, zzzz"	12 o'clock PM, Pacific Daylight Time
"K:mm a, z"	0:08 PM, PDT
"yyyyy.MMMMM.dd GGG hh:mm aaa"	02012.July.24 AD 12:08 PM
"EEE, d MMM yyyy HH:mm:ss Z"	Wed, 24 Jul 2012 12:08:56 -0700
"yyMMddHHmmssZ"	120724120856-0700
"yyyy-MM-dd'T'HH:mm:ss.SSSZ"	2012-07-04T12:08:56.235-0700

Enabling Syslog Appender

You can use the Syslog appender, if you want centralize the auditing messages at one place. Additionally, a Syslog server offers better backup support in the event of a disaster.

To enable the Syslog appender, make the following changes in the `xdasxconfig.properties` file:

1. Change the following entry to S to attach a Syslog appender:

   ```
   log4j.rootLogger=debug, S
   ```

2. Uncomment the following entries:

   ```
   log4j.appender.S=org.apache.log4j.net.SyslogAppender

   log4j.appender.S.Host=localhost

   log4j.appender.S.Port=port

   log4j.appender.S.Protocol=UDP
   ```

```
log4j.appender.S.SSLCertFile=/etc/opt/novell/mycert.pem

#log4j.appender.S.Threshold=INFO

#log4j.appender.S.Facility=USER

#log4j.appender.S.layout=org.apache.log4j.PatternLayout

#log4j.appender.S.layout.ConversionPattern=%c : %p%m%n
```

3 Log into iManager and change the log events. For information about configuring XDAS Events, see Section 2.2, “Configuring XDAS Events,” on page 18.

Generating Certificate for Syslog SSL Connection

To generate a certificate for syslog connection:

1. Create the certificate by using the following OpenSSL command:

   ```
   openssl s_client -host LOG_SERVER  -port 1443 -showcerts
   ```

2. Copy the certificate you created to the `/etc/opt/novell/eDirectory/conf/xdasconfig.properties` file.

Enabling Rolling File Appender

The File appender is preferred, if the auditing solution is limited to an individual server. Also, it is easy to bring up this solution because the number of components to be setup are few and thus, is more suited for demonstrations.

To enable the Rolling File appender, make the following changes in the `xdasxconfig.properties` file:

1 Change the following entry to R to attach a Rolling File appender.

```
log4j.rootLogger=debug, R
```

2 Uncomment the following entries:

```
log4j.appender.R=org.apache.log4j.RollingFileAppender

log4j.appender.R.File=/var/opt/novell/eDirectory/log/xdas-events.log

log4j.appender.R.MaxFileSize=100MB

log4j.appender.R.MaxBackupIndex=10

log4j.appender.R.layout=org.apache.log4j.PatternLayout

log4j.appender.R.layout.ConversionPattern=%d{MMM dd HH:mm:ss} %c : %p%m%n
```

3 Select the desired event from iManager.

For information about configuring XDAS Events, see Section 2.2, “Configuring XDAS Events,” on page 18.

2.2 Configuring XDAS Events

For information about configuring XDASv2 events for eDirectory, see Section 4.4, “Configuring XDASv2 Events for Auditing,” on page 32.

Table 2-5 on page 18 lists how eDirectory internal events are mapped to XDAS events.

NOTE: For information about eDirectory events and their description, see (http://support.novell.com/techcenter/articles/dnd19970708.html)

Table 2-5 *Mapping XDAS Events with eDirectory Events*

XDAS Event	eDirectory Event
CREATE_ACCOUNT	DSE_CREATE_ENTRY
For an example of this event, see “Create Account” on page 37.	DSE_LDAP_ADD
	DSE_LDAP_ADDRESPONSE
	DSE_NAME_COLLISION
DELETE_ACCOUNT	DSE_DELETE_ENTRY
For an example of this event, see “Delete Account” on page 38.	DSE_LDAP_DELETE
	DSE_LDAP_DELETERESPONSE
	DSE_MOVE_SOURCE_ENTRY
	DSE_REMOVE_ENTRY
ENABLE_ACCOUNT	DSE_ADD_VALUE
For an example of this event, see “Enable Account” on page 39.	
DISABLE_ACCOUNT	DSE_ADD_VALUE
For an example of this event, see “Disable Account” on page 39.	
QUERY_ACCOUNT	DSE_DSA_READ
For an example of this event, see “Query Account” on page 39.	DSE_INSPECT_ENTRY
	DSE_LDAP_SEARCH
	DSE_LDAP_SEARCHENTRYRESPONSE
	DSE_SEARCH
	DSE_LDAP_COMPARE

XDAS Event	eDirectory Event
MODIFY_ACCOUNT For an example of this event, see “Modify Account” on page 39.	DSE_ADD_VALUE DSE_DELETE_ATTRIBUTE DSE_DELETE_VALUE DSE_LDAP_MODDN DSE_LDAP_MODDNRESPONSE DSE_LDAP_MODIFY DSE_LDAP_MODIFYRESPONSE DSE_MERGE_ENTRIES DSE_MODIFY_ENTRY DSE_MODIFY_RDN DSE_RENAME_ENTRY
MODIFY_ACCOUNT_SECURITY_TOKEN For an example of this event, see “Modify Account Security Token” on page 40.	DSE_CHGPASS
CREATE_SESSION For an example of this event, see “Create Session” on page 41. To monitor the Authenticate Session event, you need to enable both the XDAS and NMAS Auditing. For more information, see Auditing with XDASv2 and Using XDASv2 for Auditing NMAS Events respectively. **NOTE:** Prior to eDirectory 8.8.8 P9, `DSE_LDAP_CONNECTION` event is used to monitor the Create Session event.	DSE_LOGIN_EX
TERMINATE_SESSION	DSE_LOGOUT
MODIFY_SESSION For an example of this event, see “Modify Session” on page 41.	DSE_CHANGE_CONN_STATE
CREATE_DATA_ITEM For an example of this event, see “Create Data Item” on page 44.	DSE_CREATE_BACKLINK DSE_CREATE_ENTRY DSE_CREATE_SUBREF DSE_LDAP_ADD DSE_LDAP_ADDRESPONSE DSE_NAME_COLLISION DSE_SPLIT_DONE DSE_SPLIT_PARTITION

XDAS Event	eDirectory Event
DELETE_DATA_ITEM For an example of this event, see "Delete Data Item" on page 45.	DSE_DELETE_ENTRY DSE_JOIN_PARTITIONS DSE_LDAP_DELETE DSE_LDAP_DELETERESPONSE DSE_MOVE_SOURCE_ENTRY DSE_REMOVE_ENTRY DSE_REMOVE_ENTRY_DIR DSE_REMOTE_SERVER_DOWN

XDAS Event	eDirectory Event
MODIFY_DATA_ITEM_ATTRIBUTE	DSE_ABORT_PARTITION_OP
For an example of this event, see “Modify Data Item Attribute” on page 45.	DSE_ADD_PROPERTY
	DSE_ADD_REPLICA
	DSE_ADD_VALUE
	DSE_CHANGE_REPLICA_TYPE
	DSE_CHECK_SEV
	DSE_DEFINE_ATTR_DEF
	DSE_DEFINE_CLASS_DEF
	DSE_DELETE_ATTRIBUTE
	DSE_DELETE_PROPERTY
	DSE_DELETE_VALUE
	DSE_LDAP_MODDN
	DSE_LDAP_MODDNRESPONSE
	DSE_GEN_CA_KEYS
	DSE_LDAP_MODIFY
	DSE_LDAP_MODIFYRESPONSE
	DSE_LDAP_PASSWDMODIFY
	DSE_MERGE_ENTRIES
	DSE_MODIFY_CLASS_DEF
	DSE_MODIFY_ENTRY
	DSE_MODIFY_RDN
	DSE_MOVE_SUBTREE
	DSE_MOVE_TREE
	DSE_MUTATE_ENTRY
	DSE_PARTITION_STATE_CHG
	DSE_PARTITION_EVENT
	DSE_RECERT_PUB_KEY
	DSE_REMOVE_ATTR_DEF
	DSE_REMOVE_BACKLINK
	DSE_REMOVE_CLASS_DEF
	DSE_REMOVE_REPLICA
	DSE_RENAME_ENTRY
	DSE_STREAM
	DSE_UPDATE_ATTR_DEF
	DSE_UPDATE_CLASS_DEF
	DSE_UPDATE_REPLICA
	DSE_UPDATE_SCHEMA
	DSE_UPDATE_SEV

XDAS Event	eDirectory Event
QUERY_DATA_ITEM_ATTRIBUTE For an example of this event, see “Query Data Item Attribute” on page 45.	DSE_CHECK_SEV DSE_COMPARE_ATTR_VALUE DSE_DSA_READ DSE_INSPECT_ENTRY DSE_LDAP_COMPARE DSE_LDAP_COMPARERESPONSE DSE_LDAP_SEARCH DSE_LDAP_SEARCHENTRYRESPONSE DSE_LDAP_SEARCHRESPONSE DSE_LIST_CONT_CLASSES DSE_LIST_PARTITIONS DSE_LIST_SUBORDINATES DSE_READ_ATTR DSE_READ_REFERENCES DSE_REFERRAL DSE_SEARCH DSE_STREAM DSE_VERIFY_PASS DSE_LOW_LEVEL_JOIN
ENABLE_SERVICE For an example of this event, see “Enable Service” on page 46.	DSE_CHANGE_MODULE_STATE
DISABLE_SERVICE For an example of this event, see “Disable Service” on page 46.	DSE_CHANGE_MODULE_STATE

XDAS Event	eDirectory Event
INVOKE_SERVICE	DSE_BACKLINK_PROC_DONE
For an example of this event, see “Invoke Service” on page 48.	DSE_LIMBER_DONE
	DSE_LUMBER_DONE
	DSE_MOVE_TREE_START
	DSE_PURGE_START
	DSE_RECV_REPLICA_UPDATES
	DSE_SEND_REPLICA_UPDATES
	DSE_START_JOIN
	DSE_START_UPDATE_REPLICA
	DSE_START_UPDATE_SCHEMA
	DSE_SYNC_PARTITION
	DSE_SYNC_PART_START
	DSE_SYNC_SCHEMA
	DSE_SYNC_SVR_OUT_START
TERMINATE_SERVICE	DSE_ABORT_JOIN
For an example of this event, see “Terminate Service” on page 48.	DSE_END_UPDATE_REPLICA
	DSE_END_UPDATE_SCHEMA
	DSE_JOIN_DONE
	DSE_MOVE_TREE_END
	DSE_PURGE_END
	DSE_SCHEMA_SYNC
	DSE_SYNC_PART_END
	DSE_SYNC_SVR_OUT_END
MODIFY_PROCESS_CONTEXT	DSE_CHANGE_TREE_NAME
For an example of this event, see “Modify Process Context” on page 48.	DSE_LDAP_MODLDAPSERVER
	DSE_MERGE_TREE
	DSE_PART_STATE_CHG_REQ
	DSE_REPAIR_TIME_STAMPS
	DSE_RESET_DS_COUNTERS
	DSE_SERVER_ADDRESS_CHANGE
	DSE_SERVER_RENAME
	DSE_SET_NEW_MASTER
	DSE_SYNTHETIC_TIME

XDAS Event	eDirectory Event
CREATE_PEER_ASSOCIATION	DSE_ADD_MEMBER
For an example of this event, see “Create Peer Association” on page 50.	DSE_ADD_VALUE
TERMINATE_PEER_ASSOCIATION	DSE_DELETE_MEMBER
For an example of this event, see “Terminate Peer Association” on page 51.	DSE_DELETE_VALUE
CREATE_DATA_ITEM_ASSOCIATION	DSE_ADD_VALUE
For an example of this event, see “Create Data Item Association” on page 52.	
TERMINATE_DATA_ITEM_ASSOCIATION	DSE_DELETE_ATTRIBUTE
For an example of this event, see “Terminate Data Item Association” on page 52.	DSE_DELETE_VALUE
MODIFY_DATA_ITEM_ASSOCIATION	DSE_BKLINK_OPERATOR
	DSE_BKLINK_SEV
	DSE_CHANGE_OBJ_SECURITY
	DSE_CHANGE_PROP_SECURITY
	DSE_CHANGE_SECURITY_EQUALS
CREATE_ROLE	DSE_CREATE_ENTRY
For an example of this event, see “Create Role” on page 54.	DSE_LDAP_ADD
	DSE_LDAP_ADDRESPONSE
	DSE_NAME_COLLISION
	DSE_ADD_ENTRY
DELETE_ROLE	DSE_DELETE_ENTRY
For an example of this event, see “Delete Role” on page 54.	DSE_DELETE_VALUE
	DSE_LDAP_DELETE
	DSE_LDAP_DELETERESPONSE
	DSE_MOVE_SOURCE_ENTRY
	DSE_REMOVE_ENTRY

XDAS Event	eDirectory Event
MODIFY_ROLE For an example of this event, see “Modify Role” on page 54.	DSE_ADD_VALUE DSE_DELETE_ATTRIBUTE DSE_DELETE_VALUE DSE_LDAP_MODIFY DSE_LDAP_MODIFYRESPONSE DSE_MERGE_ENTRIES DSE_MODIFY_ENTRY DSE_MODIFY_RDN DSE_RENAME_ENTRY
QUERY_ROLE For an example of this event, see “Query Role” on page 54.	DSE_LDAP_SEARCH DSE_LDAP_COMPARE
START_SYSTEM For an example of this event, see “Start System” on page 55.	DSE_AGENT_OPEN_LOCAL DSE_RELOAD_DS
SHUTDOWN_SYSTEM For an example of this event, see “Shutdown System” on page 56.	DSE_AGENT_CLOSE_LOCAL
BACKUP_DATA_STORE	DSE_BACKUP_ENTRY
RECOVER_DATA_STORE For an example of this event, see “Recover Data Store” on page 56.	DSE_RESTORE_ENTRY
AUTHENTICATE_SESSION For an example of this event, see “Authenticate Session” on page 58. To monitor the Authenticate Session event, you need to enable both the XDAS and NMAS Auditing. For more information, see Auditing with XDASv2 and Using XDASv2 for Auditing NMAS Events respectively. **NOTE:** Prior to eDirectory 8.8.8 P9, `DSE_LDAP_BIND`, `DSE_LDAP_BINDRESPONSE` and `DSE_LOGIN` events are used to monitor the Authenticate Session event.	DSE_AUTHENTICATE
UNAUTHENTICATE_SESSION For an example of this event, see “Unauthenticate Session” on page 58.	DSE_LDAP_UNBIND
CREATE_ACCESS_TOKEN For an example of this event, see “Create Access Token” on page 58.	DSE_ALLOW_LOGIN DSE_GEN_CA_KEYS DSE_RECERT_PUB_KEY

XDAS Event	eDirectory Event
EDIR_OPERATIONAL_ID	DSE_CRC_FAILURE
	DSE_DELETE_SUBTREE
	DSE_DELETE_UNUSED_EXTREF
	DSE_DSA_BAD_VERB
	DSE_LDAP_UNKNOWNOP
	DSE_LOST_ENTRY
	DSE_NEW_SCHEMA_EPOCH
	DSE_NO_REPLICA_PTR
	DSE_PURGE_ENTRY_FAIL
	DSE_RESEND_ENTRY

NOTE: After you select the event, it takes up to 3 minutes for the configuration changes to take effect on the NCP Server. If you want the configuration changes to be implemented immediately on the NCP server, you can unload and load the `xdasauditds` module.

2.2.1 Loading and Unloading the Modules

After you have configured the XDASv2 events, run the following commands to load and unload the XDASv2 modules:

To automatically load the xdasauditds module whenever the ndsd server is started:

- **Linux**

  Add `xdasauditds` to the `/etc/opt/novell/eDirectory/conf/ndsmodules.conf` file.
- **Windows**

  Run `ndscons.exe`, select `xdasauditds` from the list of available modules, click **Startup**, and then select **Automatic** for Startup Type.

To manually load and unload the `xdasauditds` module:

- **Linux**

  To load, run `ndstrace -c "load xdasauditds"`.

  To unload, run `ndstrace -c "unload xdasauditds"`.
- **Windows**

  To load, run `ndscons.exe`, select **xdasauditds** from the list of available modules, then click **Start**.

  To unload, run `ndscons.exe`, select **xdasauditds** from the list of available modules, then click **Stop**.

If you have installed Novell Modular Authentication Service (NMAS) and enabled NMAS auditing, the NMAS server automatically loads the XDASv2 library.

2.3 Enabling XDAS Event Caching

eDirectory 8.8 SP7 allows you to optionally store XDAS events locally on the agent in a Syslog Appender cache. With events cached, if the agent cannot communicate with the auditing server, the audit events generated are retained, ensuring that audit data is not lost. The agent then attempts to re-send the cached events when the agent computer can once again communicate with the auditing server.

XDAS event caching is disabled by default. To enable event caching, complete the steps below.

1. On the agent computer, navigate to the location of the XDASv2 property file. The xdasconfig.properties file is located at `/etc/opt/novell/eDirectory/conf/xdasconfig.properties` by default. For non-root installations, the XDASv2 property file is located in the `conf` directory by default.
2. Use a text editor to open the `xdasconfig.properties` file.
3. Within the property file, navigate to the `log4j.appender.S.CacheEnabled` property and change the property value to `yes`.
4. If you want to cache events in a specific directory, modify the value of the `log4j.appender.S.CacheDir` property to specify the directory path. The default path is `/var/opt/novell/eDirectory`. If you specify a directory, ensure that the directory path is a valid location on the server. If the specified path does not exist, the Syslog Appender logs events to the default location.
5. If you want to specify a custom file size for the cache, modify the value of the `log4j.appender.S.CacheMaxFileSize` property. The default value is 500 MB. The minimum value should be 50 MB, with a maximum value of 4 GB.
6. Save and close the `xdasconfig.properties` file.

2.4 Using Collectors for XDAS Events

For more information about using collectors for XDAS events, visit the Sentinel Plug-ins page (http://support.novell.com/products/sentinel/secure/sentinelplugins.html).

3 Understanding XDASv2 Auditing Event Filtering

Using filters and event notifications, XDASv2 is capable of reporting when a specific type of event occurs, or when it does not occur. You can also filter events for one or more specific object classes or attributes, depending on the event type. XDASv2 evaluates all the generated events against the configured filters on the eDirectory server and logs only the events matching those filters.

Multiple filters filter XDASv2 events separately. For example, if you configure filtering on both a specific object class and one or more attributes, XDASv2 logs events matching only the filters that you configured on the object class. You cannot configure filtering so that XDASv2 sends only events of a certain object class and certain attributes to the client. You can select multiple object classes or attributes for which you want to filter XDASv2 events.

You can configure filters and events notifications for XDASv2 Accounts and XDASv2 Roles.

This chapter provides the information you need to configure your system filters and notifications.

- Section 3.1, “Filtering XDASv2 Accounts,” on page 29
- Section 3.2, “Filtering XDASv2 Roles,” on page 30

3.1 Filtering XDASv2 Accounts

You can configure filtering for Accounts to look for only a specific event or events. For example, if you want to be notified when someone creates a user account in eDirectory, you can create a filter to look for only Create Object events that create a user object and log those events.

To configure accounts filtering, click the Account Management Events link, select the class, and then click **OK** to exit the application.

Note that for Accounts Management Events, you can configure filter on object classes only.

To configure filters for events that create a user object and log those events:

1. In iManager, navigate to **Roles and Tasks** > **eDirectory Auditing** > **Audit Configuration**.
2. Select the NCP Server you want to monitor, and then click **OK**.

   By default, the **XDAS Events** tab is selected.
3. Click **Account Management Events**.

   The XDAS Accounts Configuration Filtering window appears.
4. In the **Available Classes** list, select **User**, then click the right arrow to move *User* to the **Selected Classes** list, and then click **OK**.

   The filter for the Create Account event is configured.

With the filter configured, XDASv2 checks all generated events for user-creation events and logs those events.

3.2 Filtering XDASv2 Roles

Click the **Role Management Events** link to configure filtering for the XDASv2 roles. You can configure XDASv2 roles for the objects for which you want to collect XDASv2 events. You can select object classes and set attributes for them.

To configure filtering for XDAS roles:

1. In iManager, navigate to **Roles and Tasks** > **eDirectory Auditing** > **Audit Configuration**.
2. Select the NCP Server you want to monitor, and then click **OK**.

   By default, the **XDAS Events** tab is selected.
3. Click **Role Management Events**.

   The XDAS Roles Configuration Filtering window appears.
4. In the **Available Classes** list, select object classes for which you want to collect events, then click the right arrow to move them to the **Selected Classes** list.
5. In the **Available Attributes(s)** list, select any number of attributes for the object classes you have selected. Select the attribute and click the arrow to add the attribute to the selected list of attributes.

   NOTE: If you select an object class, then all the Role Events for all attributes on that object class are selected even if you had selected just a few attributes. If you want to specify only certain attributes, then you must select only those attributes and not any object class. In this case, you will get all the Role Events for the selected attributes on all object classes.
6. Click **OK**.

With the filter configured, XDASv2 checks all generated events for the attributes and logs those events.

4 iManager Plug-In for XDASv2

This chapter contains the following information.

- Section 4.1, “System Requirements,” on page 31
- Section 4.2, “Installing the iManager Plug-In for XDASv2,” on page 31
- Section 4.3, “Using the iManager Plug-In Console for XDASv2,” on page 31
- Section 4.4, “Configuring XDASv2 Events for Auditing,” on page 32

4.1 System Requirements

Installing and using the NetIQ Audit iManager Plug-in requires iManager 2.7.4 or later. See NetIQ iManager Product Page (http://www.novell.com/products/consoles/imanager/overview.html) for requirements and download instructions.

4.2 Installing the iManager Plug-In for XDASv2

The iManager plug-in for XDASv2 is bundled with eDirectory plug-ins. eDirectory plug-ins can be downloaded from the Novell download site (http://download.novell.com/Download?buildid=G_8Eymx0QtI~).

4.3 Using the iManager Plug-In Console for XDASv2

1 Log in to the iManager console.

1a Open iManager from a Web browser, using the following URL:

```
https://ip_address_or_DNS/nps/iManager.html
```

where *ip_address_or_DNS* is the IP address or DNS name of your iManager server.

For example:

```
http://192.168.0.5/nps/iManager.html
```

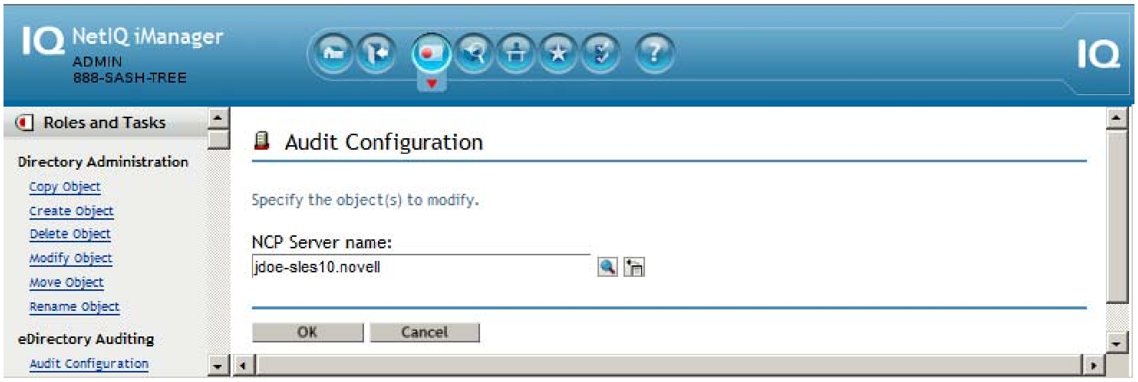
1b Log in using your username and password.

In iManager, you have access only to those roles for which you have assigned rights. To have full access to all NetIQ iManager features, you must log in as a user with Admin rights to the tree.

For more information, see “Accessing iManager” in the *NetIQ iManager 2.7 Administration Guide* (https://www.netiq.com/documentation/imanager/imanager_admin/data/bookinfo.html)

2 Select **Audit Configuration** from **Roles and Tasks**.

3 Specify the name of your eDirectory server in **NCP Server**.



Audit Configuration

Specify the object(s) to modify.

NCP Server name: jdoe-sles10.novell

Click the **Object Selector** icon to browse for the eDirectory server.

4 Click **OK**.

The XDASv2 Audit page is displayed. Continue with Section 4.4.1, “Configuring Events,” on page 32.

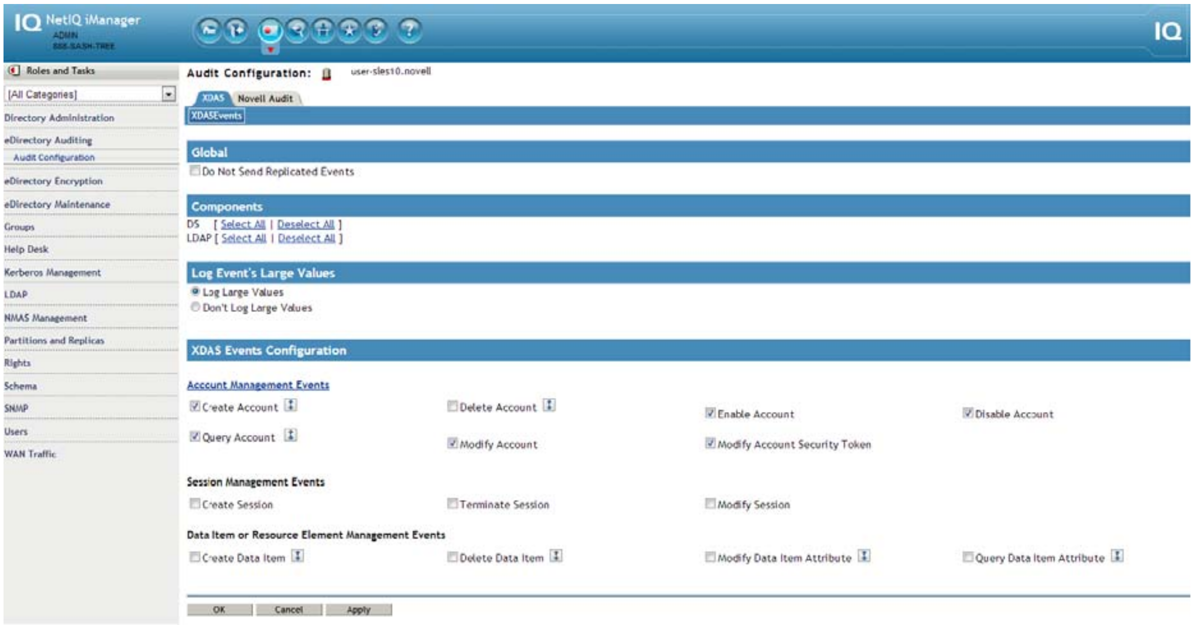
4.4 Configuring XDASv2 Events for Auditing

- Section 4.4.1, “Configuring Events,” on page 32

4.4.1 Configuring Events

Use this page to configure XDASv2 events.

***Figure 4-1** XDASv2 Events*



Audit Configuration: user-sles10.novell

Global

□ Do Not Send Replicated Events

Components

DS [Select All | Deselect All]
LDAP [Select All | Deselect All]

Log Event's Large Values

● Log Large Values
○ Don't Log Large Values

XDAS Events Configuration

Account Management Events

☑ Create Account □ Delete Account ☑ Enable Account ☑ Disable Account
☑ Query Account ☑ Modify Account ☑ Modify Account Security Token

Session Management Events

□ Create Session □ Terminate Session □ Modify Session

Data Item or Resource Element Management Events

□ Create Data Item □ Delete Data Item □ Modify Data Item Attribute □ Query Data Item Attribute

1 Global:

You can select or clear the global settings for duplicate entries.

Do Not Send Replicated Events: Select this option to stop receiving duplicate entries, such as logins, for eDirectory.

2 Log event values:

The events are logged into a text file. Event values with more than 768 bytes in size are considered "large values." You can log events of any size.

Log Large Values: Select this option to log events that are more than 768 bytes in size.

Don't Log Large Values: Select this option to log events that are less than 768 Byte in size. If the event size is more, the event value is truncated and saved to the log file.

3 You can select both or either of the following components for XDASv2 event settings:

DS: Specifies an eDirectory object. For each DS object, a corresponding LDAP object exists.

LDAP: Specifies an LDAP object.

NOTE: You can select the DS and LDAP components at the granular level for the XDAS events. Based on the event you select, the appropriate components that are supported for that event are selected. For example, if you select the **Delete Account** event, the **DS** and **LDAP** components are selected.

4 Specify the following based on your requirements:

Options	Description
Account Management Events	Select the account management events for which you want to log events. You can log events to create, delete, enable, disable, and query accounts, and also to modify account security token.
Session Management Events	Select the session management events for which you want to log events. You can log events to create, terminate, and modify sessions.
Data Item or Resource Element Management Events	Select the data item or resource element management events for which you want to log events. You can log events to create and delete data items and to modify and query data item attributes.
Service or Application Management Events	Select the service or application management events for which you want to log events. You can log events for enabling and disabling services.
Service or Application Utilization Events	Select the service or application utilization events for which you want to log events. You can log events to start and terminate services, and to modify process contexts.
Peer Association Management Events	Select the peer association events for which you want to log events. You can log events for creating and terminating peer associations.
Data Item or Resource Element Content Access Events	Select the data item or resource element content access events for which you want to log events. You can log events to create, terminate, and modify data item associations.
Role Management Events	Select the role management events for which you want to log events. You can log events to create, delete, query, and modify attributes or objects of eDirectory objects.
Exceptional Management Events	Select the exceptional management events for which you want to log events. You can log events to start and shut down systems and also to back up and recover data stores.

Options	Description
Authentication Management Events	Select the authentication management events for which you want to log events. You can log events to authenticate sessions and create access tokens.
Operational Events	Select the operational management events for which you want to log events. You can log events to generate eDirectory operation IDs.

5 XDASv2 Events

The XDASv2 events are classified into the following categories:

- Section 5.1, “Account Management Events,” on page 35
- Section 5.2, “Session Management Events,” on page 40
- Section 5.3, “Data Item and Resource Element Management Events,” on page 41
- Section 5.4, “Service or Application Management Events,” on page 46
- Section 5.5, “Service or Application Utilization Events,” on page 47
- Section 5.6, “Trust Management Events,” on page 48
- Section 5.7, “Peer Association Management Events,” on page 50
- Section 5.8, “Data Item or Resource Element Content Access Events,” on page 51
- Section 5.9, “Role Management Events,” on page 53
- Section 5.10, “Exceptional Events,” on page 55
- Section 5.11, “Authentication Management Events,” on page 56
- Section 5.12, “Operational Events,” on page 59

5.1 Account Management Events

An identity is a token used to represent a particular user or entity. The blame or credit for an action goes to the identity for a set of activities within a system. Accounts exist in the application domains to associate attributes with the set of identifiers typically associated with identities. Identities can be a a human being or an automated identity, such as another service, which is acting on behalf of a human or a regularly scheduled system activity. In both the cases, account management is considered as persistent account creation, wherein an identity with some limited or unlimited set of system rights is associated with attributes.

NOTE: The `Modify Account Security Token` event could have been defined in terms of `Modify Account`, but modification of account security tokens is considered critical to audit security, and is thus given its own event.

***Table 5-1** Account Management Event Taxonomy*

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Create Account	0.0.0.0	DSE_CREATE_ENTRY DSE_LDAP_ADD DSE_LDAP_ADDRESPONSE DSE_NAME_COLLISION	Create a new account	Consider this event as appropriate for any situation wherein an account, as defined above, is to be created.

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Delete Account	0.0.0.1	DSE_DELETE_ENTRY DSE_LDAP_DELETE DSE_LDAP_DELETERESPONSE DSE_MOVE_SOURCE_ENTRY DSE_REMOVE_ENTRY	Delete an existing account	This event has the opposite semantic meaning of account creation. Use this event wherever such an account, as described above, is to be deleted.
Disable Account	0.0.0.2	DSE_ADD_VALUE	Disable an existing account	Consider this event relevant for any situation where a particular record in an identifier database is disabled by an administrator or an automated security process such that it can no longer be used until it is re-enabled
Enable Account	0.0.0.3	DSE_ADD_VALUE	Enable an existing account	This is the counterpart event to the disable account event defined above.
Query Account	0.0.0.4	DSE_SEARCH DSE_DSA_READ DSE_INSPECT_ENTRY DSE_LDAP_SEARCH DSE_LDAP_SEARCHENTRYRESPONSE DSE_LDAP_COMPARE	Query an existing account	Consider the Query account events whenever a request for the attribute information of a particular account is made.
Modify Account	0.0.0.5	DSE_MERGE_ENTRIES DSE_ADD_VALUE DSE_DELETE_ATTRIBUTE DSE_DELETE_VALUE DSE_LDAP_MODDN DSE_LDAP_MODDNRESPONSE DSE_LDAP_MODIFY DSE_LDAP_MODIFYRESPONSE DSE_MODIFY_ENTRY DSE_MODIFY_RDN DSE_RENAME_ENTRY	Modify an existing account	Consider the Modify account events whenever a request to change attribute information of a particular account is made.

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Modify Account Security Token	0.0.0.6	DSE_CHGPASS	Modify an existing account security token	An account security token may be a password, or any other type of authentication materials associated with a user account. Here, a user account means any type of account by which a user, application, or system service may authenticate, and then act with the rights of that account.

5.1.1 Examples for Account Management Events

This section includes examples for the following Account Management events:

- "Create Account" on page 37
- "Delete Account" on page 38
- "Disable Account" on page 39
- "Enable Account" on page 39
- "Query Account" on page 39
- "Modify Account" on page 39
- "Modify Account Security Token" on page 40

NOTE: The examples provided in the following sections are for reference only.

Create Account

Click **Create Account** to generate an event for creating a user account. An output in JSON format, similar to the following is generated:

```
Jan 08 15:06:03 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SLES11-SP2,O=mycom"},"Entity" :
{"SysAddr" : "100.1.1.2","SysName" : "SLES11-SP2.my.com"}},"Initiator" :
{"Account" : {"Name" : "CN=admin,O=mycom","Id" : "32805"}},"Target" : {"Data" :
{"ClassName" : "User","Name" : "CN=USER,O=mycom"}},"Action" : {"Event" : {"Id" :
"0.0.2.0","Name" : "CREATE_ACCOUNT","CorrelationID" : "eDirectory#25#0ef05b4c-
e864-4d4c-f7a9-4c5bf00e64e8","SubEvent" : "DSE_CREATE_ENTRY"},"Time" : {"Offset" :
1389173763},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

The preceding example appears in XML format (when converted from JSON format), as follows:

```
<Source>eDirectory#DS</Source>
  <Observer>
    <Account>
      <Domain>MYTREE</Domain>
      <Name>CN=SLES11-SP2,O=mycom</Name>
    </Account>
    <Entity>
      <SysAddr>100.1.1.2</SysAddr>
      <SysName>SLES11-SP2.my.com</SysName>
    </Entity>
  </Observer>
  <Initiator>
    <Account>
      <Name>CN=admin,O=mycom</Name>
      <Id>32805</Id>
    </Account>
  </Initiator>
  <Target>
    <Data>
      <ClassName>User</ClassName>
      <Name>CN=USER,O=mycom</Name>
    </Data>
  </Target>
  <Action>
    <Event>
      <Id>0.0.2.0</Id>
      <Name>CREATE_ACCOUNT</Name>
      <CorrelationID>eDirectory#25#0ef05b4c-e864-4d4c-f7a9-4c5bf00e64e8</
CorrelationID>
      <SubEvent>DSE_CREATE_ENTRY</SubEvent>
    </Event>
    <Time>
      <Offset>1389173763</Offset>
    </Time>
    <Log>
      <Severity>7</Severity>
    </Log>
    <Outcome>0</Outcome>
    <ExtendedOutcome>0</ExtendedOutcome>
  </Action>
```

Delete Account

Click **Delete Account** to generate an event for creating a user account, as shown in the following example:

```
Jan 08 15:17:10 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SLES11-SP2,O=mycom"},"Entity" :
{"SysAddr" : "100.1.1.2","SysName" : "SLES11-SP2-164.my.com"}},"Initiator" :
{"Account" : {"Name" : "CN=admin,O=mycom","Id" : "32805"}},"Target" : {"Data" :
{"Name" : "CN=USER,O=mycom"}},"Action" : {"Event" : {"Id" : "0.0.0.1","Name" :
"DELETE_ACCOUNT","CorrelationID" : "eDirectory#25#bc9563e5-d322-43c5-fb91-
e56395bc22d3","SubEvent" : "DSE_REMOVE_ENTRY"},"Time" : {"Offset" :
1389174430},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Disable Account

Click **Disable Account** to generate an event for disabling a user account, as shown in the following example:

```
Jan 08 10:18:34 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"CN=admin,O=mycom","Id" : "32870"},"Entity" : {"SysAddr" :
"164.99.179.107:20366"}},"Target" : {"Data" : {"Attribute Name" : "Login
Disabled","Attribute Value" : "True","ClassName" : "User","Syntax" : "7"},"Account"
: {"Domain" : "MYTREE","Name" : "CN=user1,O=mycom","Id" : "32911"}},"Action" :
{"Event" : {"Id" : "0.0.0.2","Name" : "DISABLE_ACCOUNT","CorrelationID" :
"eDirectory#20#a7daeee2-990b-4203-1793-e2eedaa70b99","SubEvent" :
"DSE_ADD_VALUE"},"Time" : {"Offset" : 1389847714},"Log" : {"Severity" :
7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Enable Account

Click **Enable Account** to generate an event for enabling a user account, as shown in the following example:

```
Jan 08 10:18:34 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"CN=admin,O=mycom","Id" : "32809"},"Entity" : {"SysAddr" :
"100.1.2.142:40645"}},"Target" : {"Data" : {"Attribute Name" : "Object
Class","Attribute Value" : "ndsLoginProperties","Name" :
"dc=LDAPValidate","Syntax" : "20"}},"Action" : {"Event" : {"Id" : "0.0.0.3","Name"
: "ENABLE_ACCOUNT","CorrelationID" : "eDirectory#41#4477577d-b132-4d62-9e89-
7d57774432b1","SubEvent" : "DSE_ADD_VALUE"},"Time" : {"Offset" : 1389847714},"Log"
: {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Query Account

Click **Query Account** to generate an event for querying a user account, as shown in the following example:

```
Jan 08 10:18:34 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Domain"
: "MYTREE"},"Entity" : {"SysAddr" : "0.0.0.0:0"}},"Target" : {"Data" : {"Name" :
"CN=Test User1,dc=LDAPValidate"}},"Action" : {"Event" : {"Id" : "0.0.0.4","Name" :
"QUERY_ACCOUNT","CorrelationID" : "eDirectory#0#","SubEvent" :
"DSE_DSA_READ"},"Time" : {"Offset" : 1389847714},"Log" : {"Severity" : 7},"Outcome"
: "1","ExtendedOutcome" : "-603"}}
```

Modify Account

Click **Modify Account** to generate an event for querying a user account, as shown in the following example:

```
Jan 08 10:18:34 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Domain"
: "MYTREE"},"Entity" : {"SysAddr" : "0.0.0.0:0"}},"Target" : {"Data" : {"Attribute
Flag" : "2","Name" : "CN=Test User1,dc=LDAPValidate"}},"Action" : {"Event" : {"Id"
: "0.0.0.5","Name" : "MODIFY_ACCOUNT","CorrelationID" : "eDirectory#0#fa79e19c-
034a-445b-6292-9ce179fa4a03","SubEvent" : "DSE_MODIFY_ENTRY"},"Time" : {"Offset" :
1389847714},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Modify Account Security Token

Click **Modify Account Security Token** to generate an event for querying a user account, as shown in the following example:

```
Jan 08 10:18:34 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"CN=admin,O=mycom","Id" : "32809"},"Entity" : {"SysAddr" :
"100.1.2.142:40645"}},"Target" : {"Data" : {"Name" : "CN=Test
User1,dc=LDAPValidate"}},"Action" : {"Event" : {"Id" : "0.0.0.6","Name" :
"MODIFY_ACCOUNT_SECURITY_TOKEN","CorrelationID" : "eDirectory#41#d0f97989-ac20-
401f-03ab-8979f9d020ac","SubEvent" : "DSE_CHGPASS"},"Time" : {"Offset" :
1389847714},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

5.2 Session Management Events

A session is the association of an initiator with a stream of communication. A session may represent a user's connection to server, as in the case of logging into a Linux or Windows host, or a set of related transactions in a connection-less environment, as in the case of using a cookie to maintain persistent transactions between a browser client and a Web server.

Table 5-2 *Session Management Event Taxonomy*

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Create Session	0.0.1.0	DSE_LOGIN_EX	Create a new session	This event should be reported whenever a new session (as defined above) is created. **NOTE:** Prior to eDirectory 8.8.8 P9, `DSE_LDAP_CONNECTION` event is used to monitor the Create Session event.
Terminate Session	0.0.1.1	DSE_LOGOUT	Terminate an existing session	This event should be reported whenever an existing session (as defined above) is terminated.

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Modify Session	0.0.1.3	DSE_CHANGE_CONN_STATE	Modify user session attributes	This event should be reported whenever attribute information is modified on an existing session.

5.2.1 Examples for Session Management Events

The following sections are examples for Session Management events.

- "Create Session" on page 41
- "Modify Session" on page 41

Create Session

Click **Create Session** to generate an event for creating a session, as shown in the following example:

```
Nov 03 14:36:17 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "TP9","Name" : "CN=SLES11-SP3-164,O=novell"},"Entity" :
{"SysAddr" : "164.99.179.164","SysName" : "SLES11-SP3-164"}},"Initiator" :
{"Account" : {"Name" : "CN=admin,O=novell","Id" : "32816"},"Entity" : {"SysAddr" :
"164.99.179.165:15054"},"Assertions" : {"NetAddress" :
"164.99.179.165","NullPassword" : "FALSE","bindery login" : "FALSE"}},"Target" :
{"Data" : {"Name" : "CN=SLES11-SP3-164,O=novell"}},"Action" : {"Event" : {"Id" :
"0.0.1.0","Name" : "CREATE_SESSION","CorrelationID" : "eDirectory#11#","SubEvent"
: "DSE_LOGIN_EX"},"Time" : {"Offset" : 1478163977},"Log" : {"Severity" :
7},"Outcome" : "2.3","ExtendedOutcome" : "-669"}}
```

Modify Session

Click **Modify Session** to generate an event for modifying a session, as shown in the following example:

```
Jan 08 10:19:34 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Domain"
: "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr" :
"0.0.0.0:0"},"Assertions" : {"NetAddress" : "164.99.136.142"}},"Target" : {"Data" :
{"Name" : "CN=SRV1,O=mycom","newFlags" : "1","oldFlags" : "0"}},"Action" : {"Event"
: {"Id" : "0.0.1.3","Name" : "MODIFY_SESSION","CorrelationID" :
"eDirectory#0#","SubEvent" : "DSE_CHANGE_CONN_STATE"},"Time" : {"Offset" :
1389847774},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

5.3 Data Item and Resource Element Management Events

This set of events relate to the creation and management of data items and resource elements within a domain. The type of data item or resource element is dependent upon the domain. For example, files and directories, device special files, and shared memory segments within an operating system, tables and records within a database, messages within an e-mail system. The term data item is used in this context to refer to any type of resource element.

Table 5-3 Data Item and Resource Element Management Event Taxonomy

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Create Data Item	0.0.2.0	DSE_CREATE_BACKLINK DSE_CREATE_ENTRY DSE_CREATE_SUBREF DSE_LDAP_ADD DSE_LDAP_ADDRESPONSE DSE_NAME_COLLISION DSE_SPLIT_DONE DSE_SPLIT_PARTITION	Create a data item	This event is reported whenever a security-relevant data item or resource element is created.
Delete Data Item	0.0.2.1	DSE_DELETE_ENTRY DSE_JOIN_PARTITIONS DSE_LDAP_DELETE DSE_LDAP_DELETERESPONSE DSE_MOVE_SOURCE_ENTRY DSE_REMOVE_ENTRY DSE_REMOVE_ENTRY_DIR DSE_REMOTE_SERVER_DOWN	Delete a data item	This event is reported whenever a security-relevant data item or resource element is deleted

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Modify Data Item Attribute	0.0.2.3	DSE_ABORT_PARTITION_OP DSE_ADD_PROPERTY DSE_ADD_REPLICA DSE_ADD_VALUE DSE_CHANGE_REPLICA_TYPE DSE_CHECK_SEV DSE_DEFINE_ATTR_DEF DSE_DEFINE_CLASS_DEF DSE_DELETE_ATTRIBUTE DSE_DELETE_PROPERTY DSE_DELETE_VALUE DSE_LDAP_MODDN DSE_LDAP_MODDNRESPONSE DSE_GEN_CA_KEYS DSE_LDAP_MODIFY DSE_LDAP_MODIFYRESPONSE DSE_LDAP_PASSWDMODIFY DSE_MERGE_ENTRIES DSE_MODIFY_CLASS_DEF DSE_MODIFY_ENTRY DSE_MODIFY_RDN DSE_MOVE_SUBTREE DSE_MOVE_TREE DSE_MUTATE_ENTRY DSE_PARTITION_STATE_CHG DSE_PARTITION_EVENT DSE_RECERT_PUB_KEY DSE_REMOVE_ATTR_DEF DSE_REMOVE_BACKLINK DSE_REMOVE_CLASS_DEF DSE_REMOVE_REPLICA DSE_RENAME_ENTRY DSE_STREAM DSE_UPDATE_ATTR_DEF DSE_UPDATE_CLASS_DEF DSE_UPDATE_REPLICA DSE_UPDATE_SCHEMA	Modify data item attributes	This event is reported whenever a security-relevant data item or resource element is modified – either the value, or an attribute of the data item

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Query Data Item Attribute	0.0.2.2	DSE_CHECK_SEV DSE_COMPARE_ATTR_VALUE DSE_DSA_READ DSE_INSPECT_ENTRY DSE_LDAP_COMPARE DSE_LDAP_COMPARERESPONSE DSE_LDAP_SEARCH DSE_LDAP_SEARCHENTRYRESPONSE DSE_LDAP_SEARCHRESPONSE DSE_LIST_CONT_CLASSES DSE_LIST_PARTITIONS DSE_LIST_SUBORDINATES DSE_READ_ATTR DSE_READ_REFERENCES DSE_REFERRAL DSE_SEARCH DSE_STREAM DSE_VERIFY_PASS DSE_LOW_LEVEL_JOIN	Query data item attributes	This event is reported whenever a security-relevant data item or resource element is queried – either for value, or for an attribute of the data item.

5.3.1 Examples for Data Item and Resource Element Management Events

The following sections are some examples to generate Data Item and Resource Element Management events.

- “Create Data Item” on page 44
- “Delete Data Item” on page 45
- “Modify Data Item Attribute” on page 45
- “Query Data Item Attribute” on page 45

Create Data Item

Click **Create Data Item** to generate an event for creating a data item, as shown in the following example:

```
Jan 17 12:15:31 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "DYN_MARA","Name" : "CN=SLES11-SP2-
164,O=novell"},"Entity" : {"SysAddr" : "164.99.179.164","SysName" : "SLES11-SP2-
164.labs.blr.novell.com"}},"Initiator" : {"Account" : {"Name" :
"CN=admin,O=novell","Id" : "32797"}},"Target" : {"Data" : {"ClassName" :
"Computer","Name" : "CN=TEST-COM,O=novell"}},"Action" : {"Event" : {"Id" :
"0.0.6.0","Name" : "CREATE_DATA_ITEM","CorrelationID" : "eDirectory#15#d40ca920-
e43e-4ecc-79b4-20a90cd43ee4","SubEvent" : "DSE_CREATE_ENTRY"},"Time" : {"Offset" :
1389941131},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Delete Data Item

Click **Delete Data Item** to generate an event for deleting a data item, as shown in the following example:

```
Jan 08 10:18:35 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"CN=admin,O=mycom","Id" : "32809"},"Entity" : {"SysAddr" :
"164.99.136.142:40645"}},"Target" : {"Data" : {"ClassName" : "User","Name" :
"CN=NewTest User1,dc=LDAPValidate","newRDN" : "á°¸à¶\u0092"}},"Action" : {"Event"
: {"Id" : "0.0.2.1","Name" : "DELETE_DATA_ITEM","CorrelationID" :
"eDirectory#41#7ba31085-4e90-47fd-0aa6-8510a37b904e","SubEvent" :
"DSE_MOVE_SOURCE_ENTRY"},"Time" : {"Offset" : 1389847715},"Log" : {"Severity" :
7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Modify Data Item Attribute

Click **Modify Data Item Attribute** to generate an event for modifying a data item attribute, as shown in the following example:

```
Jan 08 10:18:36 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"CN=admin,O=mycom","Id" : "32809"},"Entity" : {"SysAddr" :
"100.1.2.164:40645"}},"Target" : {"Data" : {"Attribute Name" :
"modifiersName","Attribute Value" : "CN=admin,O=mycom","ClassName" : "User","Name"
: "CN=NewTest User2,OU=tmp,dc=LDAPValidate","Syntax" : "3"}},"Action" : {"Event" :
{"Id" : "0.0.2.3","Name" : "MODIFY_DATA_ITEM_ATTRIBUTE","CorrelationID" :
"eDirectory#41#0bbad762-4cd7-4063-4091-62d7ba0bd74c","SubEvent" :
"DSE_DELETE_VALUE"},"Time" : {"Offset" : 1389847716},"Log" : {"Severity" :
7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Query Data Item Attribute

Click **Query Data Item Attribute** to generate an event for querying a data item attribute, as shown in the following example:

```
Jan 08 10:18:36 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Id" :
"4278190081"},"Entity" : {"SysAddr" : "100.1.2.164:35218"}},"Target" : {"Data" :
{"Name" : "CN=SRV1,O=mycom"}},"Action" : {"Event" : {"Id" : "0.0.2.2","Name" :
"QUERY_DATA_ITEM_ATTRIBUTE","CorrelationID" : "eDirectory#19#","SubEvent" :
"DSE_READ_ATTR"},"Time" : {"Offset" : 1389847716},"Log" : {"Severity" :
7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

5.4 Service or Application Management Events

This set of events relates to the management of services or applications. For example, the RPM package manager might throw these events as packages are installed or removed from a Linux system. Windows 32 Service Control Manager (SCM) events sent to the Windows 32 System Event Log may be translated into these events as they are imported into OpenXDASv2. This set of events could also be much more domain-specific, including concepts such as installing, removing, or configuring installable executable-modules within a single application domain. The key idea is to ensure that reported events have security significance.

Table 5-4 *Service or Application Management Event Taxonomy*

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Enable Service	0.0.3.5	DSE_CHANGE_MODULE_STATE	Enable a service or application	This event ise reported when a service, operation or function is enabled.
Disable Service	0.0.3.4	DSE_CHANGE_MODULE_STATE	Disable a service or application	This event is reported when a service, operation or function is disabled.

5.4.1 Examples for Service or Application Management Events

The following sections include examples of events related to the management of services or applications.

- "Enable Service" on page 46
- "Disable Service" on page 46

Enable Service

Click **Enable Service** to generate an event for enabling a service, as shown in the following example:

```
Jan 08 15:06:03 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "GMC1-OESMARA","Name" : "CN=SLES11-SP3-
191,O=novell"},"Entity" : {"SysAddr" : "164.99.179.191","SysName" : "sles11-sp3-
191"}},"Initiator" : {"Account" : {"Domain" : "GMC1-OESMARA","Name" : "CN=SLES11-
SP3-191,O=novell"}},"Target" : {"Data" : {"Module State" : "Loaded","Name" :
"libspmdclnt.so"}},"Action" : {"Event" : {"Id" : "0.0.3.5","Name" :
"ENABLE_SERVICE","CorrelationID" : "eDirectory#4294967295#","SubEvent" :
"DSE_CHANGE_MODULE_STATE"},"Time" : {"Offset" : 1390473064},"Log" : {"Severity" :
7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Disable Service

Click **Disable Service** to generate an event for disabling a service, as shown in the following example:

```
Jan 08 16:04:58 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "GMC1-OESMARA","Name" : "CN=SLES11-SP3-
191,O=novell"},"Entity" : {"SysAddr" : "164.99.179.191","SysName" : "sles11-sp3-
191"}},"Initiator" : {"Account" : {"Domain" : "GMC1-OESMARA","Name" : "CN=SLES11-
SP3-191,O=novell"}},"Target" : {"Data" : {"Module State" : "Unloaded","Name" :
"libssldp.so"}},"Action" : {"Event" : {"Id" : "0.0.3.4","Name" :
"DISABLE_SERVICE","CorrelationID" : "eDirectory#4294967295#","SubEvent" :
"DSE_CHANGE_MODULE_STATE"},"Time" : {"Offset" : 1390473298},"Log" : {"Severity" :
7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

5.5 Service or Application Utilization Events

This class of events relates to the use of services and applications. They typically map to the execution of a program or a procedure and manipulation of the processing environment.

***Table 5-5** Service or Application Utilization Events Taxonomy*

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Invoke Service	0.0.4.0	DSE_START_UPDATE_SCHEMA	Invoke a service or application	This event is reported when a security-relevant service is invoked.
Terminate Service	0.0.4.1	DSE_END_UPDATE_SCHEMA	Terminate a service or application	This event is reported when a service is terminated.
Modify Process Context	0.0.4.3	DSE_CHANGE_TREE_NAME DSE_LDAP_MODLDAPSERVER DSE_MERGE_TREE DSE_PART_STATE_CHG_REQ DSE_REPAIR_TIME_STAMPS DSE_RESET_DS_COUNTERS DSE_SERVER_ADDRESS_CHANGE DSE_SERVER_RENAME DSE_SET_NEW_MASTER DSE_SYNTHETIC_TIME	Modify processing context	This event is reported when any attributes of a process context are modified – this event is somewhat specific to operating systems, but some use can be found in other domain-specific applications.

5.5.1 Examples for Service or Application Utilization Events

The following sections include examples for service or application utilization events.

- “Invoke Service” on page 48
- “Terminate Service” on page 48
- “Modify Process Context” on page 48

Invoke Service

Click **Invoke Service** to generate an event for invoking a service, as shown in the following example:

```
Jan 08 10:18:37 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Domain"
: "MYTREE"},"Entity" : {"SysAddr" : "0.0.0.0:0"}},"Target" : {"Data" : {"Name" :
"dc=Events"}},"Action" : {"Event" : {"Id" : "0.0.4.0","Name" :
"INVOKE_SERVICE","CorrelationID" : "eDirectory#0#a23fbaea-c482-4d6b-a98c-
eaba3fa282c4","SubEvent" : "DSE_PURGE_START"},"Time" : {"Offset" :
1389847717},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Terminate Service

Click **Terminate Service** to generate an event for terminating a service, as shown in the following example:

```
Jan 08 10:18:37 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Domain"
: "MYTREE"},"Entity" : {"SysAddr" : "0.0.0.0:0"}},"Target" : {"Data" : {"Name" :
"CN=SLES11-SP2-164,O=mycom"}},"Action" : {"Event" : {"Id" : "0.0.4.1","Name" :
"TERMINATE_SERVICE","CorrelationID" : "eDirectory#0#","SubEvent" :
"DSE_SYNC_SVR_OUT_END"},"Time" : {"Offset" : 1389847717},"Log" : {"Severity" :
7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Modify Process Context

Click **Modify Process Context** to generate an event when any attributes of a process context are modified, as shown in the following example:

```
Jan 08 10:30:18 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Domain"
: "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr" :
"0.0.0.0:0"}},"Action" : {"Event" : {"Id" : "0.0.4.3","Name" :
"MODIFY_PROCESS_CONTEXT","CorrelationID" : "eDirectory#0#","SubEvent" :
"DSE_SET_BINDERY_CONTEXT"},"Time" : {"Offset" : 1389848418},"Log" : {"Severity" :
7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

5.6 Trust Management Events

Trust Management events relate to the trust association of a user or an identity with a group, or the trust association of two users in a domain-specific context. For example, adding an LDAP user to a group, or associating two users for a domain-specific purpose in an application's identity association database. These events also relate to the association of identities within disparate authentication domains for federation purpose.

For example, when an identity in Domain A makes a request to a service governed by Domain B, an association of trust is required between the two domains. This is called a trust relationship. You set up a trust relationship by establishing an identity in Domain B, which is used as a proxy for any request coming from any identity in Domain A. Trust relationships can be much more complex. However, individual identities in Domain A can have individual associations with specific Domain B identities

***Table 5-6** Trust Management Events Taxonomy*

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Associate Trust	0.0.1.2	DSE_ADD_MEMBER DSE_ADD_VALUE	An association of an account with the trust which confers trust permissions to the user.	This event is reported when a new trust association is created.
De-Associate Trust	0.0.1.4	DSE_DELETE_MEMBER DSE_DELETE_VALUE	Disassociation of an account with a trust.	This event is reported when an existing trust association is destroyed.

5.6.1 Examples for Trust Management Events

The following sections include examples for trust management events.

- "Associate Trust" on page 49
- "De-Associate Trust" on page 49

Associate Trust

Click **Associate Trust** to generate an event when a new trust association is created, as shown in the following example:

```
Apr 25 15:06:49 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "AUDITTREE","Name" : "CN=paradigm1,O=novell"},"Entity" :
{"SysAddr" : "164.99.90.123","SysName" : "paradigm1"}},"Initiator" : {"Account" :
{"Name" : "CN=admin,O=novell","Id" : "32870"},"Entity" : {"SysAddr" :
"164.99.90.123:34745"}},"Target" : {"Data" : {"Attribute Name" :
"Member","Attribute Value" : "CN=user1,O=novell","ClassName" :
"dynamicGroup","Name" : "CN=mygroup,O=novell","Syntax" : "1"}},"Action" : {"Event"
: {"Id" : "0.0.1.2","Name" : "ASSOCIATE_TRUST","CorrelationID" :
"eDirectory#30#7f2e38a0-36f2-43a9-9d8f-a0382e7ff236","SubEvent" :
"DSE_ADD_VALUE"},"Time" : {"Offset" : 1461577009},"Log" : {"Severity" :
7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

De-Associate Trust

Click **De-Associate Trust** to generate an event when an existing trust association is destroyed, as shown in the following example:

```
Jan 08 10:18:14 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"CN=admin,O=mycom","Id" : "32809"},"Entity" : {"SysAddr" :
"100.1.2.3:37573"}},"Target" : {"Data" : {"Attribute Name" : "Group
Membership","Attribute Value" : "CN=mygroup,O=novell","ClassName" : "User","Name"
: "CN=user1,O=novell","Syntax" : "1"}},"Action" : {"Event" : {"Id" :
"0.0.1.4","Name" : "DEASSOCIATE_TRUST","CorrelationID" : "eDirectory#38#c92dfc98-
2b8c-4116-0197-98fc2dc98c2b","SubEvent" : "DSE_DELETE_VALUE"},"Time" : {"Offset" :
1389847694},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

5.7 Peer Association Management Events

IMPORTANT: The Peer Association Management Events will be deprecated with eDirectory 8.8 SP8 Patch 8 onwards.

Peer association events relate to the association of a user or identity with a group, or the association of two users in some domain-specific context. For example, adding an LDAP user to a group, or associating two users for a domain-specific purpose in an application's identity association database. These events are also related to the association of identities within disparate authentication domains for purposes of federation.

For example, when an identity in Domain A makes a request to a service governed by Domain B, then a peer association is required between these domains – often this is called a trust relationship. From an implementation perspective, setting up a trust relationship is often done by establishing an identity in Domain B, which is used as a proxy for any request coming from any identity in Domain A. Trust relationships can be much more complex. However, as individual identities in Domain A can have individual associations with specific Domain B identities.

***Table 5-7** Peer Association Management Events Taxonomy*

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Create Peer Association	0.0.5.0	DSE_ADD_MEMBER DSE_ADD_VALUE	Create an association with a peer	This event is reported when a new peer association is created.
Terminate Peer Association	0.0.5.1	DSE_DELETE_MEMBER DSE_DELETE_VALUE	Terminate an association with a peer	This event is reported when an existing peer association is destroyed.

5.7.1 Examples for Peer Association Management Events

The following sections include examples for peer association management events.

- “Create Peer Association” on page 50
- “Terminate Peer Association” on page 51

Create Peer Association

Click **Create Peer Association** to generate an event when a new peer association is created, as shown in the following example:

```
Jan 08 10:18:14 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"CN=admin,O=mycom","Id" : "32809"},"Entity" : {"SysAddr" :
"100.1.2.3:37573"}},"Target" : {"Data" : {"Attribute Name" : "LDAP Screen
Level","Attribute Value" : "29257","ClassName" : "LDAP Server","Name" : "CN=LDAP
Server - SLES11-SP2-164,O=mycom","Syntax" : "8"}},"Action" : {"Event" : {"Id" :
"0.0.5.0","Name" : "CREATE_PEER_ASSOCIATION","CorrelationID" :
"eDirectory#38#c92dfc98-2b8c-4116-0197-98fc2dc98c2b","SubEvent" :
"DSE_ADD_VALUE"},"Time" : {"Offset" : 1389847694},"Log" : {"Severity" :
7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Terminate Peer Association

Click **Terminate Peer Association** to generate an event when an existing peer is destroyed, as shown in the following example:

```
Jan 08 10:18:14 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"CN=admin,O=mycom","Id" : "32809"},"Entity" : {"SysAddr" :
"100.1.2.3:37573"}},"Target" : {"Data" : {"Attribute Name" :
"modifiersName","Attribute Value" : "CN=admin,O=mycom","ClassName" : "LDAP
Server","Name" : "CN=LDAP Server - SLES11-SP2-164,O=mycom","Syntax" :
"3"}},"Action" : {"Event" : {"Id" : "0.0.5.1","Name" :
"TERMINATE_PEER_ASSOCIATION","CorrelationID" : "eDirectory#38#c92dfc98-2b8c-4116-
0197-98fc2dc98c2b","SubEvent" : "DSE_DELETE_VALUE"},"Time" : {"Offset" :
1389847694},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

5.8 Data Item or Resource Element Content Access Events

Resource content-access events are related to access of any data files protected by an authentication domain. This could be file system files, database records, Web pages etc. While instrumenting applications, consider securing access to the resources. Resource access can be a high-bandwidth process. Therefore, only security-relevant events should be reported. Such instrumentation should be configurable at the application level by the application administrator, thus must be policy driven. This implies that such applications add additional infrastructure and user interface to allow administrators to manage the resource-access events that has to be audited, and determine the unimportant events within the security context.

Table 5-8 *Data Item or Resource Element Content Access Events Taxonomy*

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Create Data Item Association	0.0.6.0	DSE_ADD_VALUE	Create association with a data item	This event is reported when rights are granted by an identity to a specific data item – when a trust relationship is established between an identity and a data item.
Terminate Data Item Association	0.0.6.1	DSE_DELETE_ATTRIBUTE DSE_DELETE_VALUE	Terminate association with a data item	This event is reported when rights are revoked from an identity to a specific data item – when a trust relationship is revoked between an identity and a data item. This event is also thrown when the last value of a multi valued attribute is deleted via LDAP.

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Modify Data Item Association	0.0.6.3	DSE_BKLINK_OPERATOR DSE_BKLINK_SEV DSE_CHANGE_OBJ_SECURITY DSE_CHANGE_PROP_SECURITY DSE_CHANGE_SECURITY_EQUALS	Modify context of association with data item	This event is reported when rights are modified on the previously established relationship between an identity and specific data item.

5.8.1 Examples for Data Item and Resource Element Management Events

The following sections include examples for data item and resource element management events.

- "Create Data Item Association" on page 52
- "Terminate Data Item Association" on page 52

Create Data Item Association

Click **Create Data Item Association** to generate an event when rights are granted by an identity to a specific data item, as shown in the following example:

```
Jan 08 10:20:18 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"CN=SLES11-SP2-164,O=mycom","Id" : "32833"},"Entity" : {"SysAddr" :
"100.1.2.164:39570"}},"Target" : {"Data" : {"Attribute Name" : "Local Received Up
To","Attribute Value" : "2918332558536081408","ClassName" : "Tree Root","Syntax" :
"9"}},"Action" : {"Event" : {"Id" : "0.0.0.0","Name" :
"CREATE_DATA_ITEM_ASSOCIATION","CorrelationID" : "eDirectory#21#bf97ffb6-91d0-
4019-6988-b6ff97bfd091","SubEvent" : "DSE_ADD_VALUE"},"Time" : {"Offset" :
1389847818},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Terminate Data Item Association

Click **Terminate Data Item Association** to generate an event when rights are revoked from an identity to a specific data item, as shown in the following example:

```
Jan 08 10:20:18 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"CN=SLES11-SP2-164,O=mycom","Id" : "32833"},"Entity" : {"SysAddr" :
"100.1.2.164:39570"}},"Target" : {"Data" : {"Attribute Name" :
"syncPanePoint","ClassName" : "Tree Root","Syntax" : "9"}},"Action" : {"Event" :
{"Id" : "0.0.6.1","Name" : "TERMINATE_DATA_ITEM_ASSOCIATION","CorrelationID" :
"eDirectory#21#bf97ffb6-91d0-4019-6988-b6ff97bfd091","SubEvent" :
"DSE_DELETE_ATTRIBUTE"},"Time" : {"Offset" : 1389847818},"Log" : {"Severity" :
7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

5.9 Role Management Events

Role management event may also be classified in terms of data items, but role management is key to systems that manage identity, so these were also given their own category within the XDASv2 taxonomy.

Table 5-9 Role Management Event Taxonomy

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Create Role	0.0.8.0	DSE_CREATE_ENTRY DSE_LDAP_ADD DSE_LDAP_ADDRESPONSE DSE_NAME_COLLISION DSE_ADD_ENTRY	Create a new role	Creates a new role, or an attempt is made to create a new role.
Delete Role	0.0.8.1	DSE_DELETE_ENTRY DSE_DELETE_VALUE DSE_LDAP_DELETE DSE_LDAP_DELETERESPONSE DSE_MOVE_SOURCE_ENTRY DSE_REMOVE_ENTRY	Delete an existing role	An existing role is deleted, or an attempt is made to delete an existing role.
Modify Role	0.0.8.5	DSE_ADD_VALUE DSE_DELETE_ATTRIBUTE DSE_DELETE_VALUE DSE_LDAP_MODIFY DSE_LDAP_MODIFYRESPONSE DSE_MERGE_ENTRIES DSE_MODIFY_ENTRY DSE_MODIFY_RDN DSE_RENAME_ENTRY	Modify a role attribute	Role attributes are modified, or an attempt is made to modify role attributes.
Query Role	0.0.8.4	DSE_LDAP_SEARCH DSE_LDAP_COMPARE	Query role attributes	Role attributes are queried, or an attempt is made to query role attributes.

5.9.1 Examples for Role Management Events

The following sections include examples for role management events.

- “Create Role” on page 54
- “Delete Role” on page 54

- “Modify Role” on page 54
- “Query Role” on page 54

Create Role

Click **Create Role** to generate an event when a new role is created or an attempt is made to create a new role, as shown in the following example:

```
Jan 08 10:18:34 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"CN=admin,O=mycom","Id" : "32809"},"Entity" : {"SysAddr" :
"164.99.136.142:40645"}},"Target" : {"Data" : {"Name" :
"dc=LDAPValidate"}},"Action" : {"Event" : {"Id" : "0.0.8.0","Name" :
"CREATE_ROLE","CorrelationID" : "eDirectory#41#4477577d-b132-4d62-9e89-
7d57774432b1","SubEvent" : "DSE_ADD_ENTRY"},"Time" : {"Offset" : 1389847714},"Log"
: {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Delete Role

Click **Delete Role** to generate an event when an existing role is deleted or an attempt is made to delete an existing role, as shown in the following example:

```
Jan 08 10:18:35 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"CN=admin,O=mycom","Id" : "32809"},"Entity" : {"SysAddr" :
"164.99.136.142:40645"}},"Target" : {"Data" : {"ClassName" : "User","Name" :
"CN=NewTest User1,dc=LDAPValidate","newRDN" : "á°¸à¶\u0092"}},"Action" : {"Event"
: {"Id" : "0.0.8.1","Name" : "DELETE_ROLE","CorrelationID" :
"eDirectory#41#7ba31085-4e90-47fd-0aa6-8510a37b904e","SubEvent" :
"DSE_MOVE_SOURCE_ENTRY"},"Time" : {"Offset" : 1389847715},"Log" : {"Severity" :
7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Modify Role

Click **Modify Role** to generate an event when role attributes are modified or an attempt is made to modify role attributes, as shown in the following example:

```
Jan 08 10:20:23 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"CN=SLES11-SP2-164,O=mycom","Id" : "32833"},"Entity" : {"SysAddr" :
"100.1.2.164:39570"}},"Target" : {"Data" : {"Attribute Name" :
"Convergence","ClassName" : "domain","Name" : "dc=Events","Syntax" :
"8"}},"Action" : {"Event" : {"Id" : "0.0.8.5","Name" :
"MODIFY_ROLE","CorrelationID" : "eDirectory#21#e01904e8-b3b2-4012-3c98-
e80419e0b2b3","SubEvent" : "DSE_DELETE_ATTRIBUTE"},"Time" : {"Offset" :
1389847823},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Query Role

Click **Query Role** to generate an event when role attributes are queried or an attempt is made to query role attributes, as shown in the following example:

```
Jan 08 10:19:35 eDirectory : INFO {"Source" : "eDirectory#LDAP","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"cn=admin,o=mycom"},"Entity" : {"SysAddr" : "164.99.136.142:42181"},"Assertions" :
{"msgID" : "14","netAddress" : "164.99.136.142:50596","operationTime" : "01/16/14
10:19:34"}},"Target" : {"Data" : {"Data" : ", search filter:
(objectclass=inetOrgPerson)","DataLen" : "44","Name" : "cn=Test
User1,dc=LDAPValidate","connection" : "231405696","searchScope" :
"base"}},"Action" : {"Event" : {"Id" : "0.0.8.4","Name" :
"QUERY_ROLE","CorrelationID" : "eDirectory#4294967295#","SubEvent" :
"DSE_LDAP_SEARCH"},"Time" : {"Offset" : 1389847775},"Log" : {"Severity" :
7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

5.10 Exceptional Events

Exceptional events are generated very rarely, and are considered important because they are generated. For instance, shutting down an enterprise-critical server is exceptional because it can’t happen without someone's permission.

Table 5-10 *Exceptional Event Taxonomy*

Event Name	Event Identifier	Corresponding eDir Event	Description	Use
Start System	0.0.9.0	DSE_AGENT_OPEN_LOCAL DSE_RELOAD_DS	Start a system	This event is reported when a server, system, or mission-critical application starts up.
Shutdown System	0.0.9.1	DSE_AGENT_CLOSE_LOCAL	Shutdown a system	This event is reported when a server, system, or mission-critical application shuts down.
Back up Data Store	0.0.9.6	DSE_BACKUP_ENTRY	Back up Data Store	This event is reported when a server, system, or mission critical application backs up a critical data store.
Recover Data Store	0.0.9.7	DSE_RESTORE_ENTRY	Recover Data Store	This event is reported when a server, system, or mission critical application restores a critical data store.

5.10.1 Examples for Exceptional Events

The following sections include example for exceptional events.

- “Start System” on page 55
- “Shutdown System” on page 56
- “Recover Data Store” on page 56

Start System

Click **Start System** to generate an event when a server, system, or mission-critical application starts, as shown in the following example:

```
Jan 08 16:18:58 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "GMC1-OESMARA","Name" : "CN=SLES11-SP3-
191,O=novell"},"Entity" : {"SysAddr" : "164.99.179.191","SysName" : "sles11-sp3-
191"}},"Initiator" : {"Account" : {"Domain" : "GMC1-OESMARA","Name" : "CN=SLES11-
SP3-191,O=novell"},"Entity" : {"SysAddr" : "0.0.0.0:0"}},"Action" : {"Event" :
{"Id" : "0.0.9.0","Name" : "START_SYSTEM","CorrelationID" :
"eDirectory#0#","SubEvent" : "DSE_AGENT_OPEN_LOCAL"},"Time" : {"Offset" :
1390474138},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Shutdown System

Click **Shutdown System** to generate an event when a server, system, or mission-critical application shuts down, as shown in the following example:

```
Jan 08 16:18:47 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "GMC1-OESMARA","Name" : "CN=SLES11-SP3-
191,O=novell"},"Entity" : {"SysAddr" : "164.99.179.191","SysName" : "sles11-sp3-
191"}},"Initiator" : {"Account" : {"Domain" : "GMC1-OESMARA","Name" : "CN=SLES11-
SP3-191,O=novell"},"Entity" : {"SysAddr" : "0.0.0.0:0"}},"Action" : {"Event" :
{"Id" : "0.0.9.1","Name" : "SHUTDOWN_SYSTEM","CorrelationID" :
"eDirectory#0#","SubEvent" : "DSE_AGENT_CLOSE_LOCAL"},"Time" : {"Offset" :
1390474127},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Recover Data Store

Click **Recover Data Store** to generate an event when a server, system, or mission-critical application recovers a data store, as shown in the following example:

```
Jan 08 10:18:35 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Domain"
: "MYTREE"},"Entity" : {"SysAddr" : "100.1.2.164:32146"}},"Target" : {"Data" :
{"Name" : "OU=tmp,dc=LDAPValidate"}},"Action" : {"Event" : {"Id" : "0.0.9.5","Name"
: "RECOVER_DATA_STORE","CorrelationID" : "eDirectory#12#a565474e-e320-4121-2e9a-
4e4765a520e3","SubEvent" : "DSE_RESTORE_ENTRY"},"Time" : {"Offset" :
1389847715},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

5.11 Authentication Management Events

XDASv1 specified authentication as a modification of session attributes. XDASv2 makes authentication a first class event because authentication is critical to an audit.

Table 5-11 Authentication Events Taxonomy

Event Names	Event Identifier	eDirectory Events	Description	Use
Authenticate Session	0.0.11.0	DSE_AUTHENTICATE	A new identity is associated with a session	When a user authenticates a session, a new identity is associated with that session. This identity is then used to authorize requests for protected resources. **NOTE:** Prior to eDirectory 8.8.8 P9, `DSE_LDAP_BIND`, `DSE_LDAP_BINDRESPONSE` and `DSE_LOGIN` events are used to monitor the Authenticate Session event.
Unauthenticate Session	0.0.11.1	DSE_LDAP_UNBIND	A user has actively disassociated his identity from an existing authenticate session.	When a user clicks the “Logout” button on his or her web browser, the previously authenticated identity is removed from an existing authenticated session.
Create Access Token	0.0.11.4	DSE_ALLOW_LOGIN DSE_GEN_CA_KEYS DSE_RECERT_PUB_KEY	A SAMLv2, WS-*, OAuth, or other access token was provided upon request.	A resource access token was created by a service (or identity) provider to send to a service consumer. Access is limited by time frame, specifically requested resources, or other limiting criteria, in terms of a contract specified by previously agreed upon name/value pairs in the token. The act of creating and sending an access token is the start of a new pseudo-identity with limited and specific rights to protected resources. This pseudo-identity can be used as a correlation identifier between this and future authorization events. The actually identity of the system user behind the access token may or may not be hidden from the consumer.

NOTE: To monitor the failed login events for those login happening through NMAS, you must see the **Authenticate Session** in the NMAS collector.

5.11.1 Examples for Authentication Event

The following sections include examples for authentication events.

- "Authenticate Session" on page 58
- "Unauthenticate Session" on page 58
- "Create Access Token" on page 58

Authenticate Session

Click **Authenticate Session** to generate an event when a user authenticates a session, a new identity is associated with that session, as shown in the following example:

```
Oct 28 14:36:22 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "TEST-902-28","Name" : "CN=SLES11SP4-
192,O=novell"},"Entity" : {"SysAddr" : "100.1.2.164","SysName" : "SLES11SP4-
192"}},"Initiator" : {"Account" : {"Name" : "CN=admin,O=novell","Id" :
"32847"},"Entity" : {"SysAddr" : "100.1.2.164:48588"},"Assertions" : {"NetAddress"
: "100.1.2.164","NullPassword" : "FALSE","bindery login" : "FALSE"}},"Target" :
{"Data" : {"ClassName" : "User","Name" : "CN=SLES11SP4-192,O=novell"}},"Action" :
{"Event" : {"Id" : "0.0.11.0","Name" : "AUTHENTICATE_SESSION","CorrelationID" :
"eDirectory#17#","SubEvent" : "DSE_AUTHENTICATE"},"Time" : {"Offset" :
1477645582},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Unauthenticate Session

Click **Unauthenticate Session** to generate an event when a user authenticates a session, a new identity is associated with that session, as shown in the following example:

```
Jan 08 10:20:26 eDirectory : INFO {"Source" : "eDirectory#LDAP","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Name" :
"cn=admin,o=mycom"},"Entity" : {"SysAddr" : "164.99.136.142:42181"},"Assertions" :
{"msgID" : "54","netAddress" : "164.99.136.142:50596","operationTime" : "01/16/14
10:20:26"}},"Target" : {"Data" : {"connection" : "231405696"}},"Action" : {"Event"
: {"Id" : "0.0.11.1","Name" : "UNAUTHENTICATE_SESSION","CorrelationID" :
"eDirectory#4294967295#","SubEvent" : "DSE_LDAP_UNBIND"},"Time" : {"Offset" :
1389847826},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

Create Access Token

Click **Create Access Token** to generate an event when a a resource access token is created by a service (or identity) provider to send to a service consumer, as shown in the following example:

```
Jan 08 10:18:34 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "MYTREE","Name" : "CN=SRV1,O=mycom"},"Entity" : {"SysAddr"
: "100.1.2.164","SysName" : "SLES11-SP2-164"}},"Initiator" : {"Account" : {"Domain"
: "MYTREE"},"Entity" : {"SysAddr" : "0.0.0.0:0"}},"Target" : {"Data" : {"ClassName"
: "NCP Server","Name" : "CN=SRV1,O=mycom"}},"Action" : {"Event" : {"Id" :
"0.0.11.4","Name" : "CREATE_ACCESS_TOKEN","CorrelationID" :
"eDirectory#0#","SubEvent" : "DSE_ALLOW_LOGIN"},"Time" : {"Offset" :
1389847714},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```

5.12 Operational Events

Operational events are related to the operations of services or applications. They typically map to the events related to the operations of a program or operations related to the modules of a application.

Table 5-12 Operational Events Taxonomy

Event Names	Event Identifier	eDirectory Events	Description	Use
eDir Operational ID	0.1.0.3.0.0	DSE_CRC_FAILURE DSE_DELETE_SUBTREE DSE_DELETE_UNUSED_EXTREF DSE_DSA_BAD_VERB DSE_LDAP_UNKNOWNOP DSE_LOST_ENTRY DSE_NEW_SCHEMA_EPOCH DSE_NO_REPLICA_PTR DSE_PURGE_ENTRY_FAIL DSE_RESEND_ENTRY	Event related to the operation of a service or application.	Used for logging events to generate eDirectory operation IDs.

6 Troubleshooting

Keep in mind the following information when you install Novell XDASv2:

Initializing XDAS module error

Possible Cause: You cannot connect to the server IP or the port number mentioned in `xdasconfig.properties` file when you initialize the XDASv2 module. It displays the following message:

```
log4cxx: Could not instantiate TCP Socket to <IP>. All logging
will FAIL.

log4cxx: IO Exception : status code = 111
```

Action: To work around this issue,

1. Check whether the sever IP or the port number given in the `xdasconfig.properties` file is correct.
2. Check whether the remote server is reachable and is accepting the connection on the given port.
3. Reload the xdasauditds module.

The TCP connection is lost

Possible Cause: If the remote server is not reachable or does not accept connection on the given port, the following error is displayed:

```
log4cxx: Detected problem with TCP connection to <IP>. All logging
will FAIL.

log4cxx: IO Exception : status code = 32
```

Action: To work around this issue:

1. Check whether remote server is reachable and is accepting the connection on the given port.
2. Reload the xdasauditds module.

The SSL certificate file issue

Possible Cause: The SSL certificate file is either not valid or not present at the given location in the `xdasconfig.properties` file. The following error is displayed:

```
log4cxx: could not load verify locations for SSL
```

Action: To work around this issue,

1. Specify the absolute path to a valid certificate file.
2. Reload the xdasauditds module.

The network connection to the remote server is lost

Source: The following error is displayed:

```
log4cxx: SSL write failed for <IP>. All logging will FAIL.
```

Action: To work around this issue,

1. Check whether remote server is reachable and is accepting the connection on the given port.
2. Reload the xdasauditds module.

The SSL connection has failed

Possible Cause: The SSL connection fails because either the TLS/SSL handshake fails or a connection failure occurs. The following error message is displayed:

```
log4cxx: SSL Connect Failed to <IP>
```

Action: To work around this issue,

1. Check whether remote server is reachable and is listening on the given port.
2. Check whether the certificate is valid.
3. Reload the xdasauditds module.

A XDASv2 Schema

- Section A.1, “XDAS V2 JSON Schema,” on page 63
- Section A.2, “XDAS Field Definitions,” on page 66
- Section A.3, “Outcome Codes,” on page 69
- Section A.4, “Example of an Event,” on page 69

The XDAS schema is defined as follows:

A.1 XDAS V2 JSON Schema

```
{
    "id":"XDASv2",
    "title":"XDAS Version 2 JSON Schema",
    "description":"A JSON representation of an XDASv2 event record.",
    "type":"objectr",
    "properties":{
      "Source":{
        "description":"The original source of the event, if applicable.",
        "type":"string",
        "optional":true
      },
      "Observer":{
        "description":"The recorder (ie., the XDASv2 service) of the event.",
        "type":"object",
        "optional":false,
        "properties":{
          "Account":{"$ref":"account"},
          "Entity":{"$ref":"entity"}
        }
      },
      "Initiator":{
        "description":"The authenticated entity or access token that causes an
event.",
        "type":"object",
        "optional":false,
        "properties":{
          "Account":{"$ref":"account","optional":true},
          "Entity":{"$ref":"entity"},
          "Assertions":{
            "description":"Attribute/value assertions about an identity.",
            "type":"object",
            "optional":true
          }
        }
      },
      "Target":{
        "description":"The target object, account, data item, etc of the event.",
        "type":"object",
        "optional":true,
        "properties":{
          "Account":{"$ref":"account"},
          "Entity":{"$ref":"entity"},
          "Data":{
            "description":"A set attribute/value pairs describing the target
object.",       *
            "type":"object",
            "optional":true
          }
        }
      },
      "Action":{
        "description":"The action describes the event in a uniform manner.",
        "type":"object",
        "optional":false,
        "properties":{
          "Event":{
            "description":"The event identifier in standard XDASv2 taxonomy.",
            "type":"object",
            "optional":false,
            "properties":{
              "Id":{
                "description":"The XDASv2 taxonomy event identifier.",
                "type":"string",
                "optional":false,
                "pattern":"/^[0-9]+(\.[0-9]+)*$/"
              },
              "Name":{
                "description":"A short descriptive name for the specific event.",
eg. a new replica is added
                "type":"string",
                "optional":true
              },
      "CorrelationID":{
          "description":"Correlation ID, source#uniqueID#connID",
                 "type":"string",
                 "optional":true
      }
     },
     "SubEvent":{
      "type":object
      "description": "Describes the actual domain specific event that has
occured.",
      "optional":true,
      "properties":{
        "Name"":{
                    "description":"A short descriptive name for this event.",
                    "type":"string",
                    "optional":true
                  },
      }
            }
          }
          "Log":{
            "description":"Client-specified logging attributes.",
            "optional":true,
            "properties":{
              "Severity":{"type":"integer", "optional":true},
              "Priority":{"type":"integer", "optional":true},
              "Facility":{"type":"integer", "optional":true}
            }
          }
        "Outcome":{
          "description":"The XDASv2 taxonomy outcome identifier.",
          "type":"string",
          "optional":false,
          "pattern":"/^[0-9]+(\.[0-9]+)*$/"
        }
        "Time":{
          "description":"The time the event occurred.",
          "type":"object",
          "optional":false,
          "properties":{
            "Offset":{
              "description":"Seconds since Jan 1, 1970.",
              "type":"integer"
            },
            "Sequence":{
              "description":"Milliseconds since last integral second.",
              "type":"integer",
              "optional":true
            },
            "Tolerance":{
              "description":"A tolerance value in milliseconds.",
              "type":"integer",
              "optional":true
            },
            "Certainty":{
              "description":"Percentage certainty of tolerance.",
              "type":"integer",
              "optional":true,
              "minimum":0,
              "maximum":100,
              "default":100,
            },
            "Source":{
              "description":"The time source (eg., ntp://time.nist.gov).",
              "type":"string",
              "optional":true
            },
            "Zone":{
              "description":"A valid timezone symbol (eg., MST/MDT).",
              "type":"string",
              "optional":true
            }
          }
    "ExtendedOutcome":{
          "description":"The XDASv2 taxonomy outcome identifier.",
          "type":"string",
          "optional":false,
          "pattern":"/^[0-9]+(\.[0-9]+)*$/"
         }
      }
    }
  }
},
{
  "id":"account",
  "description":"A representation of an XDAS account.",
  "type":"object",
  "properties":{
    "Domain":{
       "description":"A (URL) reference to the authority managing this account.",
/* lets take it as the partition?
        "type":"string"
      },
      "Name":{
        "description":"A human-readable account name.",        - DN
        "type":"string",
        "optional":true
      },
      "Id":{
        "description":"A machine-readable unique account identifier value.",  -
EntryID
        "type":"integer"
      }
    }
  },
  {
    "id":"entity",                    - Server details for Target, client address
details for the initiator
    "description":"A representation of an addressable entity.",
    "type":"object",
    "properties":{
      "SysAddr":{"type":"string","optional":true},
      "SysName":{"type":"string","optional":true},
      "SvcName":{"type":"string","optional":true},
      "SvcComp":{"type":"string","optional":true},
    }
  }
```

A.2 XDAS Field Definitions

These fields in the schema are the XDASv2 fields defined specifically for audit events. Some or all of these fields may also be relevant to other types of event, but information of this sort is required for auditing services. The XDASv2 JSON record format is open. By that, we mean that any additional fields may be added to the record at any place, as long as they don't conflict with the field values defined for audit by the XDASv2 standard.Thus, if there is a particular type of correlation data, such as a workflow identifier, or a session identifier that can be used as correlation data points between events within a particular workflow or client session, you may add these fields. Simply choose a non-conflicting name for your field.

Table A-1 *XDAS Field Definitions*

XDAS Field	Description
Source (Optional)	The source of an event identifies the event service of another system from which this event was originally defined and converted to an XDAS event. Since many events are generated directly by XDAS clients, the source field is optional.

XDAS Field	Description
Initiator	The initiator of an event is the authenticated entity that initially provoked creation of the event. Note that an initiator need not be identified. If the entity can't be identified - perhaps an entity is attempting to login, thus provoking the generation of a login event by an observer - then as much information about the origin of the event as possible should be specified. NOTE: In the special case of a login event, the authenticated identity of the initiator is not yet known until after the login attempt has succeeded. Therefore a failed login event should not give the identity of the target account as the identity of the initiator. An initiator is described in terms of an account and an entity (described below), as well as an optional set of assertions. These assertions describe, in terms of a set of name/value pairs, the attributes of the initiator identity. Some initiators are not known by a specific account, but are known only by a set of assertions (SAML2, for instance) that describe the rights of the actor. The schema is not defined for these assertions, as they will be different for each class and potentially for each individual object.
Action	The action identifies the event that is being recorded. This field provides the XDASv2 event identifier, as well as an outcome code (success, or failure class), and the time the event occurred, with as much accuracy as possible.
Event	The event field is the key to XDAS events. Event encapsulates a taxonomical identifier and a short descriptive name for human readability.
Id	The event Id code represents the event identifier, defined by the XDASv2 standard event taxonomy, and extensions defined by the Novell CSS product.
Name	The event name is a human readable representation of the event identifier. The event name is optional, but recommended for readability.
Data	The event data provides additional descriptive information about the event.
Log	The log field contains standard syslog-like log-level values, in terms of Severity and Facility numeric identifiers. The log field is optional, as well as every sub-field within the log field. These values should only be used when necessary, as they generally represent judgment calls on the part of the instrumentor. Such judgment calls are best left to analysis software or engineers once the event data is collected.
Outcome	For details on outcome codes, see Section A.3, “Outcome Codes,” on page 69.
Time	The event time is the time recorded by the observer at the point the event was committed to the event service. Time values are gathered by the XDAS client helper library. Thus, there is no reason to be concerned about values stored in this field, as the helper library will attempt to be as accurate as possible when generating time information.
Offset	The offset field contains a value representing the number of seconds since midnight, January 1, 1970 - otherwise known as the Linux epoch.
Sequence	The sequence field contains a unique numeric value identifying this event from another event which may have been recorded within the same second. For the most part, this value should be taken as a monotonically increasing numeric value that begins at zero and continues until the next second boundary, at which point, it begins again at zero.
Tolerance	The tolerance value is a value between 0 and 100, indicating the tolerance of the clock used to record the time in offset. Values of zero indicate the clock is very accurate. Values of 100 indicate that the clock should not be trusted.

XDAS Field	Description
Certainty	The certainty value is a value between 0 and 100, indicating the percentage certainty of the tolerance value. Zero means there is no certainty of the tolerance, and thus, it shouldn't be trusted to any degree of accuracy. A value of 100 indicates that the tolerance value is very accurate.
Source	The time source is information indicating the source of time for the observer system. This may be a URL for a time server, or simply a local time source, such as a hardware clock.
Zone	The time zone is the new time zone string representing the time zone of this clock.
Target (Optional)	The target of an event is the account or protected resource upon which the initiator is attempting to act, thereby provoking the generation of an event. A target is described in terms of an account and an entity (described below), as well as an optional and unspecified Data object. The Data object is a set of name/value pairs describing class-specific attributes of the actor. The schema does not define the actual fields, as different classes will have a unique set of data attributes (if any).
Observer	The observer of an event is the authenticated identity of an entity (service) that is monitoring the system, and generating events based on initiator actions. An observer is described in terms of an account and an entity (described below).
Referenced Classes	The observer, initiator, and target fields contain references to the account and entity classes defined separately within the schema. These other classes identify key attributes of the three primary actors within an audit event.
Account Class	The account class represents the identity of the actor. This identity is relative to an authentication realm or Domain. Both an account name and an account Id are provides, although only the Id is really required. The Name is for human readability.
Account Domain	The account Domain defines the authentication authority of the actor. Account identifiers mean very little without an authentication authority.
Account Name	The account Name is optional, providing human readability.
Account Id	The account Id is a unique identifier of the account within the authentication Domain.
Entity Class	The entity class describes the location of the actor. This location is defined in terms of a system access end point (IP network) address and a system access end point (host/domain) name. Additional fields are also available to describe the service and component names within the software that manages the above end points.
Entity SysAddr	An IP address describing the access end point of the software actor. The IP address is displayed as IP Address:Port. For example: ◆ IPv4: 194.99.188.103:34564 ◆ IPv6: [2015::15]:43333 Note that internal event IP address is displayed as 0.0.0.0:0.
Entity SysName	A host/domain name describing the access end point of the software actor.
Entity SvcName	A service name further describing the service that manages the above end point.
Entity SvcComp	A service component name describing the component within the above service.

A.3 Outcome Codes

The outcome code is a hierarchical numeric value much like the event code. Outcome codes indicate success or a failure class and reason. The success hierarchy is encapsulated by the 0.x sub-arc. Failure classes are represented by the 1.x hierarchy. Denial codes are represented by the 2.x hierarchy.

A.4 Example of an Event

An example event is given below:

```
Sep 12 14:51:26 eDirectory : INFO {"Source" : "eDirectory#DS","Observer" :
{"Account" : {"Domain" : "DEMOTREE","Name" : "CN=demo-host,O=org"},"Entity" :
{"SysAddr" : "192.168.0.15","SysName" : "demo-host"}},"Initiator" : {"Account" :
{"Domain" : "DEMOTREE"},"Entity" : {"SysAddr" : "192.168.0.10:18408"}},"Target" :
{"Data" : {"Name" : "CN=demo-host,O=org"}},"Action" : {"Event" : {"Id" :
"0.0.2.2","Name" : "QUERY_DATA_ITEM_ATTRIBUTE","CorrelationID" :
"eDirectory#5#","SubEvent" : "DSE_DSA_READ"},"Time" : {"Offset" :
1378977686},"Log" : {"Severity" : 7},"Outcome" : "0","ExtendedOutcome" : "0"}}
```