

Administration Console Guide

Novell[®] Access Manager

3.1 SP2

November 16, 2010

www.novell.com



Legal Notices

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

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

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

Copyright © 2006-2010 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

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

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

Novell Trademarks

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

Third-Party Materials

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

Contents

About This Guide	11
1 Administration Console	13
1.1 Security Considerations	13
1.1.1 Securing the Administration Console	13
1.1.2 Protecting the Configuration Store	14
1.1.3 Enabling Auditing and Event Notification	15
1.1.4 Forcing 128-Bit Encryption	16
1.2 Configuring the Administration Console	16
1.2.1 Configuring the Default View	17
1.2.2 Changing the Administration Console Session Timeout	19
1.2.3 Changing the Password for the Administration Console	19
1.2.4 Understanding Administration Console Conventions	20
1.3 Multiple Administrators, Multiple Sessions	20
1.3.1 Creating Multiple Admin Accounts	21
1.4 Managing Delegated Administrators	21
1.4.1 Access Gateway Administrators	23
1.4.2 Policy Container Administrators	24
1.4.3 Identity Server Administrators	24
1.4.4 SSL VPN Administrators	24
1.4.5 J2EE Agent Administrators	25
1.4.6 Activating eDirectory Auditing for LDAP Events	25
1.5 Enabling Auditing	26
1.5.1 Configuring Access Manager for Auditing	27
1.5.2 Querying Data and Generating Reports in Novell Audit	30
2 Backing Up and Restoring	33
2.1 How The Backup and Restore Process Works	33
2.1.1 Default Parameters	33
2.1.2 The Process	33
2.2 Backing Up the Access Manager Configuration	34
2.3 Restoring an Administration Console Configuration	35
2.3.1 Restoring the Configuration on a Standalone Administration Console or with a Traditional SSL VPN Server	36
2.3.2 Restoring the Configuration with an Identity Server on the Same Machine	37
2.3.3 Restoring the Configuration with an ESP-Enabled SSL VPN Server	39
2.4 Restoring an Identity Server	39
2.5 Restoring an Access Gateway	40
2.5.1 Clustered Access Gateway	40
2.5.2 Single Access Gateway	41
2.6 Running the Diagnostic Configuration Export	41
3 Security and Certificate Management	43
3.1 Understanding How Access Manager Uses Certificates	43
3.1.1 Process Flow	44
3.1.2 Access Manager Trust Stores	45
3.1.3 Access Manager Keystores	47
3.2 Creating Certificates	49

3.2.1	Creating a Locally Signed Certificate	51
3.2.2	Editing the Subject Name	54
3.2.3	Assigning Alternate Subject Names	56
3.2.4	Generating a Certificate Signing Request	57
3.2.5	Importing a Signed Certificate	58
3.3	Managing Certificates and Keystores	58
3.3.1	Viewing Certificate Details	59
3.3.2	Adding a Certificate to a Keystore	61
3.3.3	Renewing a Certificate	61
3.3.4	Exporting a Private/Public Key Pair	63
3.3.5	Exporting a Public Certificate	64
3.3.6	Importing a Private/Public Key Pair	65
3.3.7	Reviewing the Command Status for Certificates	65
3.3.8	Keystore Details	67
3.4	Managing Trusted Roots and Trust Stores	67
3.4.1	Importing Public Key Certificates (Trusted Roots)	68
3.4.2	Adding Trusted Roots to Trust Stores	68
3.4.3	Auto-Importing Certificates from Servers	69
3.4.4	Exporting the Public Certificate of a Trusted Root	69
3.4.5	Viewing Trust Store Details	69
3.4.6	Viewing Trusted Root Details	70
3.5	Security Considerations for Certificates	71
3.6	Assigning Certificates to Access Manager Devices	72
3.6.1	Importing a Trusted Root to the LDAP User Store	72
3.6.2	Managing Identity Server Certificates	73
3.6.3	Assigning Certificates to an Access Gateway	75
3.6.4	Assigning Certificates to J2EE Agents	75
3.6.5	Configuring SSL for Authentication between the Identity Server and Access Manager Components	76
3.6.6	Changing a Non-Secure (HTTP) Environment to a Secure (HTTPS) Environment	76
3.6.7	Creating Keystores and Trust Stores	77
4	Access Manager Logging	79
4.1	Understanding the Types of Logging	79
4.1.1	Component Logging for Troubleshooting Configuration or Network Problems	79
4.1.2	HTTP Transaction Logging for Proxy Services	80
4.2	Downloading the Log Files	80
4.2.1	Linux Administration Console Logs	81
4.2.2	Windows Server 2003 Administration Console Logs	82
4.2.3	Windows Server 2008 Administration Console Logs	82
4.2.4	Linux Identity Server Logs	83
4.2.5	Windows Server 2003 Identity Server Logs	83
4.2.6	Windows Server 2008 Identity Server Logs	83
4.2.7	Linux Access Gateway Appliance Logs	84
4.2.8	Linux Access Gateway Service Logs	85
4.2.9	Windows Access Gateway Service Logs	85
4.2.10	SSL VPN Server Logs	86
4.3	Using the Log Files for Troubleshooting	86
4.3.1	Enabling Logging	87
4.3.2	Understanding the Log Format	87
4.3.3	Sample Authentication Traces	90
5	Changing the IP Address of Access Manager Devices	95
5.1	Changing the IP Address of the Administration Console	95
5.2	Changing the IP Address of an Identity Server	95

5.3	Changing the IP Address of the Access Gateway Appliance	97
5.4	Changing the IP Address of the Access Gateway Service	98
5.5	Changing the IP Address of the Audit Server	99
6	Troubleshooting the Administration Console	101
6.1	Global Troubleshooting Options	101
6.1.1	Checking for Potential Configuration Problems	101
6.1.2	Checking for Version Conflicts	103
6.1.3	Checking for Invalid Policies	104
6.1.4	Viewing Device Health	104
6.1.5	Viewing Health by Using the Hardware IP Address	104
6.1.6	Using the Dashboard	105
6.1.7	Viewing System Alerts	108
6.2	Stopping Tomcat on Windows	108
6.3	Logging	108
6.4	Event Codes	109
6.5	Restoring a Failed Secondary Console	109
6.6	Moving the Primary Administration Console to New Hardware	109
6.7	Converting a Secondary Console into a Primary Console	110
6.7.1	Shutting Down the Administration Console	111
6.7.2	Changing the Master Replica	111
6.7.3	Restoring CA Certificates	112
6.7.4	Editing the vcdn.conf File	112
6.7.5	Deleting Objects from the eDirectory Configuration Store	113
6.7.6	Performing Component-Specific Procedures	113
6.7.7	Enabling Backup on the New Primary Administration Console	121
6.8	Orphaned Objects in the Trust/Configuration Store	122
6.9	Repairing the Configuration Datastore	123
6.10	Session Conflicts	123
6.11	Unable to Log In to the Administration Console	123
6.12	(Linux) Exception Processing IdentityService_ServerPage.JSP	124
6.13	Backup/Restore Failure Because of Special Characters in Passwords	124
6.14	Unable to Install NMAS SAML Method	125
7	Troubleshooting Certificate Issues	127
7.1	Resolving Certificate Import Issues	127
7.1.1	Importing an External Certificate Key Pair	127
7.1.2	Resolving a -1226 PKI Error	128
7.1.3	When the Full Certificate Chain Is Not Returned During an Automatic Import of the Trusted Root	128
7.1.4	Using Internet Explorer to Add a Trusted Root Chain	129
7.2	Mutual SSL with X.509 Produces Untrusted Chain Messages	129
7.3	Troubleshooting Options for Certificate Problems	129
7.4	Can't Log In with Certificate Error Messages	131
7.5	When a User Accesses a Resource, the Browser Displays Certificate Errors	131
7.6	Access Gateway Canceled Certificate Modifications	131
A	Certificates Terminology	133
B	Troubleshooting XML Validation Errors on the Access Gateway Appliance	135
B.1	Modifying a Configuration That References a Removed Object	135

B.2	Configuration UI Writes Incorrect Information to the Local Configuration Store.	137
-----	---	-----

C Access Manager Audit Events and Data 141

C.1	NIDS: Sent a Federate Request (002e0001)	143
C.2	NIDS: Received a Federate Request (002e0002)	144
C.3	NIDS: Sent a Defederate Request (002e0003)	144
C.4	NIDS: Received a Defederate Request (002e0004)	145
C.5	NIDS: Sent a Register Name Request (002e0005)	145
C.6	NIDS: Received a Register Name Request (002e0006)	146
C.7	NIDS: Logged Out an Authentication that Was Provided to a Remote Consumer (002e0007)	146
C.8	NIDS: Logged out a Local Authentication (002e0008)	147
C.9	NIDS: Provided an Authentication to a Remote Consumer (002e0009)	147
C.10	NIDS: User Session Was Authenticated (002e000a)	148
C.11	NIDS: Failed to Provide an Authentication to a Remote Consumer (002e000b)	149
C.12	NIDS: User Session Authentication Failed (002e000c)	149
C.13	NIDS: Received an Attribute Query Request (002e000d)	150
C.14	NIDS: User Account Provisioned (002e000e)	150
C.15	NIDS: Failed to Provision a User Account (002e000f)	151
C.16	NIDS: Web Service Query (002e0010)	152
C.17	NIDS: Web Service Modify (002e0011)	152
C.18	NIDS: Connection to User Store Replica Lost (002e0012)	153
C.19	NIDS: Connection to User Store Replica Reestablished (002e0013)	154
C.20	NIDS: Server Started (002e0014)	154
C.21	NIDS: Server Stopped (002e0015)	155
C.22	NIDS: Server Refreshed (002e0016)	155
C.23	NIDS: Intruder Lockout (002e0017)	156
C.24	NIDS: Severe Component Log Entry (002e0018)	156
C.25	NIDS: Warning Component Log Entry (002e0019)	157
C.26	NIDS: Roles PEP Configured (002e0300)	157
C.27	Access Gateway: PEP Configured (002e0301)	158
C.28	J2EE Agent: Web Service Authorization PEP Configured (002e0305)	158
C.29	J2EE Agent: JACC Authorization PEP Configured (002e0306)	159
C.30	Roles Assignment Policy Evaluation (002e0320)	160
C.31	Access Gateway: Authorization Policy Evaluation (002e0321)	160
C.32	Access Gateway: Form Fill Policy Evaluation (002e0322)	161
C.33	Access Gateway: Identity Injection Policy Evaluation (002e0323)	161
C.34	J2EE Agent: Web Service Authorization Policy Evaluation (002e0324)	162
C.35	J2EE Agent: Web Service SSL Required Policy Evaluation (002e0325)	162
C.36	J2EE Agent: Startup (002e0401)	163
C.37	J2EE Agent: Shutdown (002e0402)	163
C.38	J2EE Agent: Reconfigure (002e0403)	164
C.39	J2EE Agent: Authentication Successful (002e0404)	164
C.40	J2EE Agent: Authentication Failed (002e0405)	165
C.41	J2EE Agent: Web Resource Access Allowed (002e0406)	166
C.42	J2EE Agent: Clear Text Access Allowed (002e0407)	166
C.43	J2EE Agent: Clear Text Access Denied (002e0408)	167
C.44	J2EE Agent: Web Resource Access Denied (002e0409)	167
C.45	J2EE Agent: EJB Access Allowed (002e040a)	168
C.46	J2EE Agent: EJB Access Denied (002e040b)	169
C.47	Access Gateway: Access Denied (0x002e0505)	169

C.48	Access Gateway: URL Not Found (0x002e0508)	170
C.49	Access Gateway: System Started (0x002e0509)	171
C.50	Access Gateway: System Shutdown (0x002e050a)	171
C.51	Access Gateway: Identity Injection Parameters (0x002e050c)	172
C.52	Access Gateway: Identity Injection Failed (0x002e050d)	173
C.53	Access Gateway: Form Fill Authentication (0x002e050e)	173
C.54	Access Gateway: Form Fill Authentication Failed (0x002e050f)	174
C.55	Access Gateway: URL Accessed (0x002e0512)	175
C.56	Access Gateway: IP Access Attempted (0x002e0513)	176
C.57	Access Gateway: Webserver Down (0x002e0515)	176
C.58	Access Gateway: All WebServers for a Service is Down (0x002e0516)	177
C.59	Management Communication Channel: Health Change (0x002e0601)	178
C.60	Management Communication Channel: Device Imported (0x002e0602)	178
C.61	Management Communication Channel: Device Deleted (0x002e0603)	179
C.62	Management Communication Channel: Device Configuration Changed (0x002e0604)	180
C.63	Management Communication Channel: Device Alert (0x002e0605)	180

About This Guide

This guide describes the following features of the Novell Access Manager Administration Console that are not specific to an Access Manager device:

- ◆ Chapter 1, “Administration Console,” on page 13
- ◆ Chapter 2, “Backing Up and Restoring,” on page 33
- ◆ Chapter 3, “Security and Certificate Management,” on page 43
- ◆ Chapter 4, “Access Manager Logging,” on page 79
- ◆ Chapter 5, “Changing the IP Address of Access Manager Devices,” on page 95
- ◆ Chapter 6, “Troubleshooting the Administration Console,” on page 101
- ◆ Chapter 7, “Troubleshooting Certificate Issues,” on page 127
- ◆ Appendix A, “Certificates Terminology,” on page 133
- ◆ Appendix B, “Troubleshooting XML Validation Errors on the Access Gateway Appliance,” on page 135
- ◆ Appendix C, “Access Manager Audit Events and Data,” on page 141

Audience

This guide is intended for Access Manager administrators. It is assumed that you have knowledge of evolving Internet protocols, such as:

- ◆ Extensible Markup Language (XML)
- ◆ Simple Object Access Protocol (SOAP)
- ◆ Security Assertion Markup Language (SAML)
- ◆ Public Key Infrastructure (PKI) digital signature concepts and Internet security
- ◆ Secure Socket Layer/Transport Layer Security (SSL/TLS)
- ◆ Hypertext Transfer Protocol (HTTP and HTTPS)
- ◆ Uniform Resource Identifiers (URIs)
- ◆ Domain Name System (DNS)
- ◆ Web Services Description Language (WSDL)

Feedback

We want to hear your comments and suggestions about this guide and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation, or go to [Documentation Feedback \(http://www.novell.com/documentation/feedback.html\)](http://www.novell.com/documentation/feedback.html) at www.novell.com/documentation/feedback.html and enter your comments there.

Documentation Updates

For the most recent version of the *Access Manager Administration Console Guide*, visit the [Novell Access Manager Documentation Web site \(http://www.novell.com/documentation/novellaccessmanager31\)](http://www.novell.com/documentation/novellaccessmanager31).

Additional Documentation

Before proceeding, you should be familiar with the *Novell Access Manager 3.1 SP2 Installation Guide* and the *Novell Access Manager 3.1 SP2 Setup Guide*, which provides information about setting up the Access Manager system.

For information about the other Access Manager devices and features, see the following:

- ♦ *Novell Access Manager 3.1 SP2 Identity Server Guide*
- ♦ *Novell Access Manager 3.1 SP2 Access Gateway Guide*
- ♦ *Novell Access Manager 3.1 SP2 Policy Guide*
- ♦ *Novell Access Manager 3.1 SP2 J2EE Agent Guide*
- ♦ *Novell Access Manager 3.1 SP2 SSL VPN Server Guide*
- ♦ *Novell Access Manager 3.1 SP2 Event Codes*

Documentation Conventions

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

Administration Console

1

- ♦ [Section 1.1, “Security Considerations,” on page 13](#)
- ♦ [Section 1.2, “Configuring the Administration Console,” on page 16](#)
- ♦ [Section 1.3, “Multiple Administrators, Multiple Sessions,” on page 20](#)
- ♦ [Section 1.4, “Managing Delegated Administrators,” on page 21](#)
- ♦ [Section 1.5, “Enabling Auditing,” on page 26](#)

For information about installing secondary consoles for fault tolerance, see “[Clustering and Fault Tolerance](#)” in the *Novell Access Manager 3.1 SP2 Setup Guide*.

For troubleshooting information about converting a secondary console into a primary console, see [Section 6.7, “Converting a Secondary Console into a Primary Console,” on page 110](#).

1.1 Security Considerations

The Administration Console contains all the configuration information for all Access Manager components. If you federate your users with other servers, it stores configuration information about these users. You need to protect the Administration Console so that unauthorized users cannot change configuration settings or gain access to the information in the configuration store. When you develop a security plan for Access Manager, consider the following:

- ♦ [Section 1.1.1, “Securing the Administration Console,” on page 13](#)
- ♦ [Section 1.1.2, “Protecting the Configuration Store,” on page 14](#)
- ♦ [Section 1.1.3, “Enabling Auditing and Event Notification,” on page 15](#)
- ♦ [Section 1.1.4, “Forcing 128-Bit Encryption,” on page 16](#)

1.1.1 Securing the Administration Console

When you look for ways to secure the Administration Console from unauthorized access, consider the following:

Admin User: The admin user you create when you install the Administration Console has all rights to the Access Manager components. We recommend that you protect this account by configuring the following features:

- ♦ **Password Restrictions:** When the admin user is created, no password restrictions are set. To ensure that the password meets your minimum security requirements, you should configure the standard eDirectory password restrictions for this account. In the Administration Console, select the *Roles and Tasks* view in the iManager header, then click *Users*. Browse to the admin user (found in the novell container), then click *Restrictions*. For configuration help, use the *Help* button.
- ♦ **Intruder Detection:** The admin user is created in the novell container. You should set up an intruder detection policy for this container. In the Administration Console, select the *Roles and Tasks* view in the iManager header, then click *Directory Administration > Modify Object*. Select *novell*, then click *OK*. Click *Intruder Detection*. For configuration help, use the *Help* button.

- ♦ **Multiple Administrator Accounts:** Only one admin user is created when you install Access Manager. If something happens to the user who knows the name of this user and password or if the user forgets the password, you cannot access the Administration Console. Novell recommends that you create at least one backup user and make that user security equivalent to the admin user. For instructions, see [Section 1.3.1, “Creating Multiple Admin Accounts,” on page 21](#). For other considerations when you have multiple administrators, see [Section 1.3, “Multiple Administrators, Multiple Sessions,” on page 20](#).

Network Configuration: You need to protect the Administration Console from Internet attacks. It should be installed behind your firewall.

If you are installing the Administration Console on its own machine, ensure that the DNS names resolve between the Identity Server and the Administration Console. This ensures that SSL security functions correctly between the Identity Server and the configuration store in the Administration Console.

Delegated Administrators: If you create delegated administrators for policy containers (see [Section 1.4, “Managing Delegated Administrators,” on page 21](#)), be aware that they have sufficient rights to implement a cross-site scripting attack using the Deny Message in an Access Gateway Authorization policy.

They are also granted rights to the LDAP server, which gives them sufficient rights to access the configuration datastore with an LDAP browser. Modifications done with an LDAP browser are not logged by Access Manager. To enable the auditing of these events, see [“Activating eDirectory Auditing for LDAP Events” on page 25](#).

Test Certificates: When you install the Administration Console, the following test certificates are automatically generated:

test-signing
test-encryption
test-connector
test-provider
test-consumer
test-stunnel

For tight security, we recommend that you replace these certificates, except the test-stunnel certificate, with certificates from a well-known certificate authority.

Two years after you install the Administration Console, new versions of these certificates are automatically generated as the old certificates expire. If you are using any of the test certificates in your configuration, the Administration Console cannot use the new version until you reboot the machine.

1.1.2 Protecting the Configuration Store

The configuration store is an embedded, modified version of eDirectory. It is backed up and restored with command line options, which back up and restore the Access Manager configuration objects in the ou=accessManagerContainer.o=novell object.

You should back up the configuration store on a regular schedule, and the ZIP file created should be stored in a secure place. See [Section 2, “Backing Up and Restoring,” on page 33](#).

In addition to backing up the configuration store, you should also install at least two Administration Consoles (a primary and a secondary). If the primary console goes down, the secondary console can keep the communication channels open between the various components. You can install up to three Administration Consoles. For installation information, see “[Installing Secondary Versions of the Administration Console](#)” in the *Novell Access Manager 3.1 SP2 Setup Guide*.

The configuration store should not be used for a user store.

1.1.3 Enabling Auditing and Event Notification

For a secure system, you need to set up either auditing or syslogging to notify the system administrator when certain events occur. The most important audit events to monitor are the following:

- ◆ Configuration changes
- ◆ System shutdowns and startups
- ◆ Server imports and deletes
- ◆ Intruder lockout detection (available only for eDirectory user stores)
- ◆ User account provisioning

Audit events are device-specific. You can select events for the following devices:

- ◆ **Administration Console:** In the Administration Console, click *Auditing > Novell Auditing*.
- ◆ **Identity Server:** In the Administration Console, click *Devices > Identity Servers > Edit > Logging*.
- ◆ **Access Gateway:** In the Administration Console, click *Devices > Access Gateways > Edit > Novell Audit*.
- ◆ **J2EE Agent:** In the Administration Console, click *Devices > J2EE Agents > Edit*.
- ◆ **SSL VPN:** In the Administration Console, click *Devices > SSL VPNs > Edit > Novell Audit Settings*.

You can configure Access Manager to send audit events to a Novell Audit Server, a Sentinel server, or a Sentinel Log Manager. For configuration information, see [Section 1.5, “Enabling Auditing,” on page 26](#).

In addition to the selectable events, device-generated alerts are automatically sent to the audit server. These Management Communication Channel events have an ID of 002e0605. All Access Manager events begin with 002e except for SSL VPN events, which start with 0031. You can set up Novell Auditing to send e-mail whenever these events or your selected audit events occur. See “[Configuring System Channels](#)” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/al6t4sd.html>) in the *Novell Audit 2.0 Guide* (<http://www.novell.com/documentation/novellaudit20/treetitl.html>).

For information about audit event IDs and field data, see [Appendix C, “Access Manager Audit Events and Data,” on page 141](#).

The Access Gateway also supports a syslog that allows you to send e-mail notification to system administrators. To configure this system in the Administration Console, click *Devices > Access Gateways > Edit > Alerts*.

1.1.4 Forcing 128-Bit Encryption

You can force all client communication with the Administration Console to use 128-bit encryption by modifying the `server.xml` file used by Tomcat. If the browser is unable to supported the encryption level specified in this file, the user is not allowed to authenticate. If the Identity Server is installed on the same machine as the Administration Console, the following procedure forces all client communication with the Identity Server to use 128-bit encryption.

- 1 At a command prompt, change to the Tomcat configuration directory:

Linux: `/var/opt/novell/tomcat5/conf`

Windows Server 2003: `\Program Files\Novell\Tomcat\conf`

Windows Server 2008: `\Program Files (x86)\Novell\Tomcat\conf`

- 2 To the `server.xml` file, add the cipher suites you want to support. For 128-bit encryption, add the following line:

```
ciphers="TLS_RSA_WITH_AES_128_CBC_SHA,TLS_DHE_RSA_WITH_AES_128_CBC_SHA,
TLS_DHE_DSS_WITH_AES_128_CBC_SHA"
```

This is a comma separated list of the JSSE names for the TLS cipher suites.

IMPORTANT: If you enter a cipher name incorrectly, Tomcat reverts to the default values, which allow the weak ciphers to be used.

If you want to allow the SSL cipher suites, the following JSSE names can be added to the list:

```
SSL_RSA_WITH_RC4_128_MD5
```

```
SSL_RSA_WITH_RC4_128_SHA
```

For a complete list of supported cipher suites and their requirements, see “[The SunJSSE Provider](http://java.sun.com/javase/6/docs/technotes/guides/security/SunProviders.html#SunJSSEProvider)” (<http://java.sun.com/javase/6/docs/technotes/guides/security/SunProviders.html#SunJSSEProvider>).

- 3 To activate the cipher list, restart Tomcat.

Linux: Enter the following command:

```
/etc/init.d/novell-tomcat5 restart
```

Windows: Enter the following commands:

```
net stop Tomcat5
```

```
net start Tomcat5
```

- 4 (Conditional) If you have multiple Identity Servers in your cluster configuration, repeat these steps on each Identity Server.

1.2 Configuring the Administration Console

- ♦ [Section 1.2.1, “Configuring the Default View,”](#) on page 17
- ♦ [Section 1.2.2, “Changing the Administration Console Session Timeout,”](#) on page 19
- ♦ [Section 1.2.3, “Changing the Password for the Administration Console,”](#) on page 19
- ♦ [Section 1.2.4, “Understanding Administration Console Conventions,”](#) on page 20

1.2.1 Configuring the Default View

Access Manager has two views in the Administration Console. Access Manager 3.0 and its Support Packs used the *Roles and Tasks* view, with Access Manager as the first listed task in the left hand navigation frame. It looks similar to the following:

Figure 1-1 Access Manager Roles and Tasks View

Novell iManager
ADMIN
JWILSON_TREE

Roles and Tasks
[All Categories]

Access Manager
Auditing and Logging
Directory Administration
Groups
Help Desk
NMAS
Partitions and Replicas
Rights
Schema
Users

Novell iManager
VERSION 2.7.2

You are currently logged in to JWILSON_TREE as admin.novell with Unrestricted Access.

Notice: Some of the roles and tasks are not available.
To see the list of Roles and Tasks not displayed and troubleshooting information go to the [View Details](#) page.

Web-based Administration	iManager Access Modes
<p>Novell iManager is a state-of-the-art Web-based administration console that provides customized secure access to network administration utilities and content from any location in the world. With a global view of your network from a browser-based tool, you can proactively assess and respond to changing network demands. Using a roles and tasks framework, decentralized administration has never been easier.</p> <p>The roles and tasks available depend upon the access mode and the rights granted to you.</p>	<p>Unrestricted Access This mode displays all of the roles and tasks installed. Although all roles and tasks are visible, the authenticated user will still need the necessary rights to use the tasks.</p> <p>Assigned Access This mode displays only the roles and tasks assigned to the authenticated user. This mode takes full advantage of the Role Based Services (RBS) technology.</p> <p>Collection Owner Access This mode displays the roles and tasks in any and all collections for which the authenticated user is an owner. It allows user Admin to use all of the roles and tasks in the collections, even if specific rights have not been assigned. Role Based Services (RBS) must be installed in order to use this mode.</p>

This view has the following advantages:

- ◆ Other tasks that you occasionally need to manage the configuration datastore are visible.
- ◆ If you are familiar with 3.0, you do not need to learn new ways to navigate to configure options.

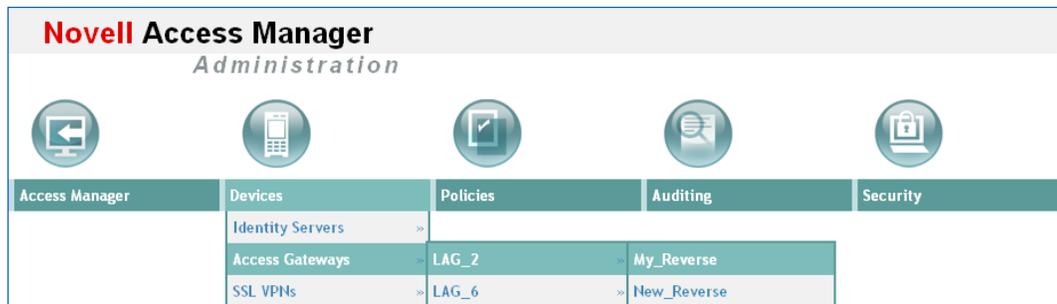
Access Manager 3.1 introduced a new view, the Access Manager view. It looks similar to the following:

Figure 1-2 Access Manager View



This view has the following advantages:

- ◆ You can follow a path to a Identity Server cluster configuration or an Access Gateway proxy service with one click. The following image shows the path to the My_Reverse proxy service of the LAG_2 Access Gateway.



- ◆ It can remember where you have been. For example, if you are configuring the Access Gateway and need to check a setting for a Role policy, you can view that setting. If you click the *Devices* tab, the Administration Console remembers where you were in the Access Gateway configuration. If you click *Access Gateways*, it resets to that view.
- ◆ With the navigation moved to the top of the page, the wider configuration pages no longer require a scroll bar to see all of the options.
- ◆ Navigation is faster.

When you install or upgrade to Access Manager 3.1 or above and log in to the Administration Console, the default view is set to the Access Manager view.

Changing the View

- 1 Locate the Header frame.



- 2 Click either the Roles and Tasks view  or the Access Manager view .

Setting a Permanent Default View

- 1 In the iManager Header frame, click the Preferences view.
- 2 In the left navigation frame, click *Set Initial View*.
- 3 Select your preferred view, then click *OK*.

1.2.2 Changing the Administration Console Session Timeout

The `web.xml` file for Tomcat specifies how long an Administration Console session can remain inactive before the session times out and the administrator must authenticate again. The default value is 30 minutes.

To change this value:

- 1 Change to the Tomcat configuration directory:
Linux: `/etc/opt/novell/tomcat5/web.xml`
Windows Server 2003: `\Program Files\Novell\Tomcat\conf`
Windows Server 2008: `\Program Files (x86)\Novell\Tomcat\conf`
- 2 Open the `web.xml` file in a text editor and search for the `<session-timeout>` parameter.
- 3 Modify the value and save the file.
- 4 Restart Tomcat:
Linux: `/etc/init.d/novell-tomcat5 restart`
Windows: `net stop Tomcat5`
`net start Tomcat5`

1.2.3 Changing the Password for the Administration Console

The admin of the Administration Console is a user created in the novell container of the configuration store. To change the password:

- 1 In the Administration Console, click *Users > Modify User*.
- 2 Click the *Object Selector* icon.
- 3 Browse the novell container and select the name of the admin user, then click *OK*.
- 4 Click *Restrictions > Set Password*.
- 5 Enter a password in the *New password* text box.
- 6 Confirm the password in the *Retype new password* text box.
- 7 Click *OK* twice.

1.2.4 Understanding Administration Console Conventions

- ◆ The required fields on a configuration page contain an asterisk by the field name.
- ◆ All actions such as delete, stop, and purge, require verification before they are executed.
- ◆ Changes are not applied to a server until you update the server.
- ◆ Sessions are monitored for activity. If your session becomes inactive, you are asked to log in again and unsaved changes are lost.
- ◆ Do not use the browser Back button. If you need to move back, use one of the following:
 - ◆ Click the *Cancel* button.
 - ◆ Click a link in the breadcrumb path that is displayed under the menu bar.
 - ◆ Use the menu bar to select a location.
- ◆ Right-clicking links in the interface, then selecting to open the link in a new tab or window is not supported.
- ◆ If you are in the Roles and Task view and the left navigation panel is not present in the window or tab, close the session and start a new one.
- ◆ The Administration Console uses a modified version of iManager. You cannot use standard iManager features or plug-ins with the Access Manager version of the product.
- ◆ If you access the Administration Console as a protected Access Gateway resource, you cannot configure it for single sign-on. The version of iManager used for the Administration Console is not compatible with either Identity Injection or Form Fill for single sign-on.

1.3 Multiple Administrators, Multiple Sessions

The Administration Console has been designed to warn you when another administrator is making changes to a policy container or to an Access Manager device (such as an Access Gateway, SSL VPN, or J2EE Agent). The person who is currently editing the configuration is listed at the top of the page with an option to unlock and with the person's distinguished name and IP address. If you select to unlock, you destroy all changes the other administrator is currently working on.

WARNING: Currently, locking has not been implemented on the pages for modifying the Identity Server. If you have multiple administrators, they need to coordinate with each other so that only one administrator is modifying an Identity Server cluster at any given time.

Multiple Sessions: You should not start multiple sessions to the Administration Console with the same browser on a workstation. Browser sessions share settings that can result in problems when you apply changes to configuration settings. However, if you are using two different brands of browsers simultaneously, such as Internet Explorer and Firefox, it is possible to avoid the session conflicts.

Multiple Administration Consoles: As long as the primary console is running, all configuration changes should be made at the primary console. If you make changes at both a primary console and a secondary console, browser caching can cause you to create an invalid configuration.

The following sections explain how to create additional administrator accounts and how to delegate rights to administrators:

- ◆ [“Creating Multiple Admin Accounts” on page 21](#)
- ◆ [“Managing Delegated Administrators” on page 21](#)

1.3.1 Creating Multiple Admin Accounts

The Administration Console is installed with one admin user account. If you have multiple administrators, you might want to create a user account for each one so that log files reflect the modifications of each administrator. The easiest way to do this is to create an account for each administrator and make the user security equivalent to the admin user. This also ensures that you have more than one user who has full access to the Administration Console. If you have only one administrator and something happens to the user who knows the name and password of admin account or if the user forgets the password, you cannot access the Administration Console.

To create a user who is security equivalent to the admin user:

- 1 In the Administration Console, select the *Roles and Tasks* view in the iManager header.
- 2 Click *Users > Create User*.
- 3 Create a user account for each administrator.
- 4 Click *Modify User*, then select the created user.
- 5 Click *Security > Security Equal To*.
- 6 Select the admin user, then click *Apply > OK*.
- 7 Repeat [Step 4](#) through [Step 6](#) for each user you want to make security equivalent to the admin user.

You can also create delegated administrators and configure them to have rights to specific components of Access Manager. For configuration information for this type of user, see [Section 1.4, “Managing Delegated Administrators,”](#) on page 21.

1.4 Managing Delegated Administrators

As the Access Manager admin user, you can create delegated administrators to manage the following Access Manager components.

- ♦ Individual Access Gateways or an Access Gateway cluster
- ♦ Identity Server clusters
- ♦ Individual J2EE agents or a J2EE agent cluster
- ♦ Individual SSL VPN servers or an SSL VPN cluster
- ♦ Policy containers

IMPORTANT: You need to trust the users you assign as delegated administrators. They are granted sufficient rights that they can compromise the security of the system. For example if you create delegated administrators with View/Modify rights to policy containers, they have sufficient rights to implement a cross-site scripting attack using the Deny Message in an Access Gateway Authorization policy.

Delegated administrators are also granted rights to the LDAP server, which means they can access the configuration datastore with an LDAP browser. Any modifications made with the LDAP browser are not logged by Access Manager. To log LDAP events, you need to turn on eDirectory auditing. For configuration information, see [“Activating eDirectory Auditing for LDAP Events”](#) on page 25.

By default, all users except the admin user are assigned no rights to the policy containers and the devices. The admin user has all rights and cannot be configured to have less than all rights. The admin user is the only user who has the rights to delegate rights to other users, and the only user with sufficient rights to modify keystores, create certificates, and import certificates.

The configuration pages for delegated administrators control access to the Access Manager pages. They do not control access to the tasks available for the *Roles and Tasks* view in iManager. If you want your delegated administrators to have rights to any of these tasks such as Directory Administration or Groups, you must use eDirectory methods to grant the user rights to these tasks or enable and configure Role-Based Services in iManager.

To create a delegated administrator, you must first create the user accounts, then assign them rights to the Access Manager components.

- 1 In the Administration Console, select the Roles and Tasks view from the iManager view bar.
- 2 (Optional) If you want to create a container for your delegated administrators, click *Directory Administration > Create Object*, then create a container for the administrators.
- 3 To create the users, click *Users > Create User* and create user accounts for your delegated administrators.
- 4 Return to the Access Manager view, then click *Administrators* in the *Access Manager* menu.
- 5 Select the component you want to assign a user to manage.

For more information about the types of rights you might want to assign for each component, see the following

- ♦ [“Access Gateway Administrators” on page 23](#)
 - ♦ [“Policy Container Administrators” on page 24](#)
 - ♦ [“Identity Server Administrators” on page 24](#)
 - ♦ [“SSL VPN Administrators” on page 24](#)
 - ♦ [“J2EE Agent Administrators” on page 25](#)
- 6 To assign all delegated administrators the same rights to a component, configure *All Users* by using the drop-down menu and selecting *None*, *View Only*, or *View/Modify*.

By default, *All Users* is configured for *None*. *All Users* is a quick way to assign everyone View Only rights to a component when you want your delegated administrators to have the rights to view the configuration but not change it.

- 7 To select one or more users to assign rights, click *Add*, then fill in the following fields:

Name filter: Specify a string that you want the user’s cn attribute to match. The default value is an asterisk, which matches all cn values.

Search from context: Specify the context you want used for the search. Click the down-arrow to select from a list of available contexts.

Include subcontainers: Specifies whether subcontainers should be searched for users.

- 8 Click *Query*, and the *User* section is populated with the users that match the query.
- 9 In the *User* section, select one or more users to whom you want to grant the same rights.
- 10 For the *Access* option, click the down-arrow and select one of the following values:

View/Modify: Grants full configuration rights to the device. View/Modify rights do not grant the rights to manage keystores, to create certificates, or to import certificates from other servers or certificate authorities. View/Modify rights allow the delegated administrator to perform actions such as stop, start, and update the device.

If the assignment is to a policy container, this option grants the rights to create policies of any type and to modify any existing policies in the container

View Only: Grants the rights to view all the configuration options of the device or all rules and conditions of the policies in a container.

None: Prevents the user from seeing the device or the policy container.

11 In the *Device* or *Policy Containers* section, select the devices, the clusters, or policy containers that you want to assign for delegated administration.

12 Click *Apply*.

The rights are immediately assigned to the selected users. If the user already had a rights assignment to the device or policy container, this new assignment overwrites any previous assignments.

13 After assigning a user rights, check the user's effective rights.

A user's effective rights and assigned rights do not always match. For example, if Kim is granted View Only rights but All Users have been granted View/Modify rights, Kim's effective rights are View/Modify.

1.4.1 Access Gateway Administrators

You can assign a user to be a delegated administrator of an Access Gateway cluster or a single Access Gateway that does not belong to a cluster. You cannot assign a user to manage a single member of a cluster.

When a delegated administrator of an Access Gateway cluster is granted View/Modify rights, the administrator has sufficient rights to change the cluster configuration, to stop and start (or reboot and shut down), and to update the Access Gateways in the cluster. However, to configure the Access Gateway to use SSL, you need to be the admin user, rather than a delegated administrator.

When the user is assigned View/Modify rights to manage a cluster or an Access Gateway, the user is automatically granted View Only rights to the master policy container. If you have created other policy containers, these containers are hidden until you grant the delegated administrator rights to them. View Only rights allows the delegated administrator to view the policies and assign them to protected resources. It does not allow them to modify the policies. If you want the delegated administrator to modify or create policies, you need to grant View/Modify rights to a policy container.

View/Modify rights to an Access Gateway or a cluster allows the delegated administrator to modify which Identity Server cluster the Access Gateway uses for authentication. It does not allow delegated administrators to update the Identity Server configuration, which is required whenever the Access Gateway is configured to trust an Identity Server. To update the Identity Server, the delegated administrator needs View/Modify rights to the Identity Server configuration.

1.4.2 Policy Container Administrators

All delegated administrators with View/Modify rights to a device have read rights to the master policy container. To create or modify policies, a delegated administrator needs View/Modify rights to a policy container. When a delegated administrator has View/Modify rights to any policy container, the delegated administrator is also granted enough rights to allow the administrator to select shared secret values, attributes, LDAP groups, and LDAP OUs to policies.

If you want your delegated administrators to have full control over a device and its policies, you might want to create a separate policy container for each delegated administrator or for each device that is managed by a group of delegated administrators.

1.4.3 Identity Server Administrators

You cannot assign a delegated administrator to an individual Identity Server. You can only assign a delegated administrator to a cluster configuration, which gives the delegated administrator rights to all the cluster members.

When a delegated administrator of an Identity Server cluster is granted View/Modify rights, the administrator has sufficient rights to change the cluster configuration and to stop, start, and update the Identity Servers in the cluster. The administrator is granted view rights to the keystores for each Identity Server in the cluster. To change any of the certificates, the administrator needs to be the admin user rather than a delegated administrator.

The delegated administrator of an Identity Server cluster is granted View Only rights to the master policy container. If you want the delegated administrator with View/Modify rights to have sufficient rights to manage policies, grant the following rights:

- ♦ To have sufficient rights to create Role policies, grant View/Modify rights to a policy container.
- ♦ To have sufficient rights to enable Role policies, grant View Only rights to the policy containers with Role policies.

1.4.4 SSL VPN Administrators

If the SSL VPN has an Embedded Service Provider and you grant the delegated administrator View/Modify rights to the SSL VPN or its cluster, the delegated administrator is granted sufficient rights to modify which Identity Server the SSL VPN or cluster uses for authentication. It does not allow them to update the Identity Server configuration, which is required for this type of modification. To update the Identity Server, the delegated administrator needs View/Modify rights to the Identity Server configuration.

If the SSL VPN is a protected resource of an Access Gateway and you want the delegated administrator to have rights to the Access Gateway and the SSL VPN policy, you need to also grant the user View/Modify rights to the Access Gateway and the SSL VPN policy container.

When a delegated administrator of an SSL VPN is granted View/Modify rights, the administrator has sufficient rights to change the configuration, to stop and start the service, and to update the server's configuration.

To set up the secure tunnel certificate, the SSL VPN administrator also needs to be a certificate administrator with View/Modify rights.

1.4.5 J2EE Agent Administrators

You can assign a user to be a delegated administrator of a J2EE Agent cluster or a single J2EE Agent that does not belong to a cluster. When a user is assigned View/Modify rights to manage an agent, the user is automatically assigned View Only rights to the master policy container. If you want the delegated administrator to create or modify J2EE Agent Authorization policies, you need to grant the delegated administrator View/Modify rights to a policy container.

View/Modify rights to an agent also allows the delegated administrator to modify which Identity Server the agent uses for authentication. It does not allow them to update the Identity Server configuration, which is required for this configuration change. To update the Identity Server, the delegated administrator needs View/Modify rights to the Identity Server configuration.

View/Modify rights allows the administrator rights to change the configuration, to stop and start the agent, and to update the agent's configuration.

To configure certificates for the agent, the J2EE agent administrator also needs to be a certificate administrator with View/Modify rights.

1.4.6 Activating eDirectory Auditing for LDAP Events

If you are concerned that your delegated administrators might use an LDAP browser to access the configuration datastore, you can configure eDirectory to audit events that come from LDAP connections to the LDAP server.

- 1 In the Administration Console, click *Auditing > Auditing*.
- 2 Make sure you have configured the IP address and port to use for your Secure Logging Server. The server can be a Novell Audit server, a Sentinel server, or a Sentinel Log Manager. For more information about this process, see [Section 1.5, "Enabling Auditing," on page 26](#).

WARNING: Whenever you change the port or address of the Secure Logging Server, all Access Gateways must be updated, then every Access Manager device (Identity Server, Administration Console, Access Gateways, SSL VPN servers, and J2EE Agents) must be rebooted (not just the module stopped and started) before the configuration change takes affect.

- 3 From the iManager view bar, select the Roles and Tasks view.
- 4 Click *Directory Administration > Modify Object*.
- 5 Click the *Object Selector* icon, expand the *novell* container, then select the eDirectory server. The eDirectory server uses the tree name, without the *_TREE* suffix, for its name. The tree name is displayed in the iManager view bar.
- 6 Click *OK > Novell Audit > eDirectory*.
- 7 From the *Meta*, *Objects*, and *Attributes* sections, select the events that you want to monitor for potential security problems.
 - ♦ In the *Meta* section, you probably want to monitor changes made to groups and ACLs.
 - ♦ In the *Objects* section, you probably want to monitor who is logging in and out and if objects are being created or deleted.
 - ♦ In the *Attributes* section, you probably want to monitor when attribute values are added or deleted.
- 8 Click *Apply*.

9 (Linux) Restart eDirectory and the Audit Server. Enter the following commands:

```
/etc/init.d/ndsd restart  
/etc/init.d/novell-naudit restart
```

10 (Windows) Restart eDirectory and the Audit Server:

10a Click *Control Panel > Administrative Tools > Services*.

10b Right click *NDS Server*, then select *Stop*.

10c Answer *Yes* to the prompt to stop the *Novell Audit Log Server*.

10d Right click *NDS Server*, then select *Start*.

10e Right click *Novell Audit Log Server*, then select *Start*.

1.5 Enabling Auditing

Access Manager includes a licensed version of Novell Audit to provide compliance assurance logging and to maintain audit log entries that can be subsequently included in reports. In addition to selectable events, device-generated alerts are automatically sent to the audit server. Access Manager comes preconfigured to use the Novell Audit server, but you can configure Access Manager to use an already existing Novell Audit server, a Sentinel server, or a Sentinel Log Manager server.

The audit logs record events that have occurred in the identity and access management system and are primarily intended for auditing and compliance purposes. You can configure the following types of events for logging:

- ◆ Starting, stopping, and configuring a component
- ◆ Success or failure of user authentication
- ◆ Role assignment
- ◆ Allowed or denied access to a protected resource
- ◆ Error events
- ◆ Denial of service attacks
- ◆ Security violations and other events necessary for verifying the correct and expected operation of the identity and access management system.

Audit logging does not track the operational processing of the Access Manager components; that is, the processing and interactions between the Access Manager components required to fulfill a user request. (For this type of logging, see “[Configuring Component Logging](#)” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.) Audit logs record the results of user and administrator requests and other system events. Although the primary purpose for audit logging is for auditing and compliance, the types of events logged can also be useful for detecting abnormal and error conditions and can be used as a first alert mechanism for system support. You can configure the audit log entries to generate alerts by leveraging the Novell Audit Notification feature. You can select to generate e-mail, syslog, and SNMP notifications.

Access Manager has been assigned the Novell Audit server-alert event code 0x002E0605. The Novell Audit Platform Agent is responsible for packaging and forwarding the audit log entries to the configured audit server. If the audit server is not available, the Platform Agent caches log entries until the server is operational and can accept audit log data.

- ♦ [Section 1.5.1, “Configuring Access Manager for Auditing,” on page 27](#)
- ♦ [Section 1.5.2, “Querying Data and Generating Reports in Novell Audit,” on page 30](#)

1.5.1 Configuring Access Manager for Auditing

By default, Access Manager is preconfigured to use the Novell Audit server it installs on the first instance of the Administration Console. If you install more than one instance of the Administration Console for failover, Novell Audit is installed with each instance. However, if you already use Novell Audit, you can configure Access Manager to use your audit server. You also need to register the Access Manager with your audit servers by importing the `nids_en.lsc` and `sslvpn_en.lsc` files. If you have a Sentinel server or a Sentinel Log Manager server, you can configure Access Manager to send the events to them.

Access Manager allows you to specify only one audit server. You still have failover if the audit server goes down. The auditing clients on the Novell Access Manager components go into caching mode when the audit server is not available. They save all events until the entries can be sent to the audit server.

This section includes the following topics:

- ♦ [“Specifying the Logging Server and the Console Events” on page 27](#)
- ♦ [“Configuring the Platform Agent” on page 28](#)
- ♦ [“Configuring the Devices for Auditing” on page 29](#)

Specifying the Logging Server and the Console Events

The Secure Logging Server manages the flow of information to and from the auditing system. It receives incoming events and requests from the Platform Agents, logs information to the data store, monitors designated events, and provides filtering and notification services. It can also be configured to automatically reset critical system attributes according to a specified policy.

1 To specify the logging server, click *Auditing > Novell Auditing*.

2 Fill in the following fields:

Server: Specify the IP address or DNS name of the audit logging server you want to use. By default, the system uses the primary Administration Console IP address. If you want to use a different Secure Logging Server, specify that server here.

To use a Sentinel server or a Sentinel Log Manager instead of Novell Audit, specify the IP address or DNS name of your Collector.

- ♦ For more information on Sentinel, see [Sentinel 6.1 \(http://www.novell.com/documentation/sentinel61/index.html\)](http://www.novell.com/documentation/sentinel61/index.html).
- ♦ For more information on Sentinel Log Manager, see [Sentinel Log Manager 1.0 \(http://www.novell.com/documentation/novelllogmanager10/\)](http://www.novell.com/documentation/novelllogmanager10/).

Port: Specify the port that the Platform Agents use to connect to the Secure Logging Server.

To use a Sentinel server or Sentinel Log Manager instead of Novell Audit, specify the port of your Collector.

IMPORTANT: Whenever you change the port or address of the Secure Logging Server, all Access Gateways must be updated, then every Access Manager device (Identity Server, Administration Console, Access Gateways, SSL VPN servers, and J2EE Agents) must be rebooted (not just stopping and starting the module) before the configuration change takes affect.

- 3** Under *Management Console Audit Events*, specify the system-wide events you want to audit:

Select All: Selects all of the audit events.

Health Changes: Generated whenever the health of a server changes.

Server Imports: Generated whenever a server is imported into the Administration Console.

Server Deletes: Generated whenever a server is deleted from the Administration Console.

Configuration Changes: Generated whenever you change a server configuration.

- 4** Click *OK*.

If you did not change the address or port of the Secure Logging Server, this completes the process. It might take up to fifteen minutes for the events you selected to start appearing in the audit files.

- 5** (Conditional) If the Administration Console is the only Access Manager component installed on the machine and you have changed the address or port of the Secure Logging Server, complete the following steps:

For security reasons, the Novell Audit Configuration file cannot be edited by the Administration Console when it is the only Access Manager component on the machine. It can only be edited by a system administrator.

- 5a** Open the `logevent.conf` file.

Linux: Located in the `etc` directory

Windows: Located in the `Windows` directory.

- 5b** Specify the new address and port of the Secure Logging Server, then save the file.

- 6** Restart the Administration Console. Open a terminal window, then enter the command for your platform:

Linux: `/etc/init.d/novell-tomcat5 restart`

Windows: `net stop Tomcat5`
`net start Tomcat5`

- 7** Restart every device imported into the Administration Console.

The devices (Identity Server, Access Gateway, SSL VPN, J2EE Agents) do not start reporting events until they have been restarted.

Configuring the Platform Agent

The Platform Agents installed with the Access Manager components use an embedded certificate. Access Manager does not currently support the use of custom application certificates. For information on this Novell Audit feature, see “[Authenticating Logging Applications](http://www.novell.com/documentation/novellaudit20/novellaudit20/data/am8ewv2.html)” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/am8ewv2.html>) in the *Novell Audit Administration Guide* (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/bookinfo.html>).

The Platform Agents that are installed on each Access Manager component can be configured by modifying the `logevent` file. For the location of this file and its parameters, see “[Logevent](http://www.novell.com/documentation/novellaudit20/novellaudit20/data/al36zjk.html#alibmyw)” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/al36zjk.html#alibmyw>) in the *Novell Audit Administration Guide* (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/bookinfo.html>).

IMPORTANT: Do not use this file to modify the IP address of the Secure Audit Server. Use the Administration Console for this task (see “[Specifying the Logging Server and the Console Events](#)” on page 27).

If you are using Sentinel, most of the parameters in this file should be set on the collector.

When the Platform Agent loses its connection to the audit server, it enters caching mode. The default size of the audit cache file is unlimited. This means that if the connection is broken for long and traffic is high, the cache file can become quite large. When the connection to the audit server is re-established, the Platform Agent becomes very busy while it tries to upload the cached events to the audit server and still process new events. When it comes out of caching mode, the Platform Agent appears unresponsive because it is so busy and because it holds application threads that are logging new events for a long period of time. If it holds too many threads, the whole system can appear to be hung. You can minimize the effects of this scenario by configuring the following two parameters in the `logevent` file.

Parameter	Description
LogMaxCacheSize	Sets a limit to the amount of cache the Platform Agent can consume to log events when the audit server is unreachable. The default is unlimited.
LogCacheLimitAction	Specifies what the Platform Agent should do with incoming events when the maximum cache size limit is reached. You can select one of the following actions: Delete the current cache file and start logging events in a new cache file. Stop logging, which preserves all entries in cache and stops collecting new events.

When you set a finite cache file size, it limits the number of events that must be uploaded to the audit server when caching mode is terminated and keeps the Platform Agent responsive to new audit events that are registered. If you have many users and are logging many events, you might need to configure these parameters.

For more information about these parameters, see “[Logevent](http://www.novell.com/documentation/novellaudit20/novellaudit20/data/al36zjk.html#alibmyw)” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/al36zjk.html#alibmyw>) in the *Novell Audit Administration Guide* (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/bookinfo.html>).

Configuring the Devices for Auditing

Each device defines the events that can be enabled for auditing. For information on enabling these events, see the following:

- ♦ “[Enabling Access Gateway Audit Events](#)” in the *Novell Access Manager 3.1 SP2 Access Gateway Guide*

- ◆ “Enabling Identity Server Audit Events” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*
- ◆ “Enabling SSL VPN Audit Events” in the *Novell Access Manager 3.1 SP2 SSL VPN Server Guide*
- ◆ “Enabling Tracing and Auditing of Events” in the *Novell Access Manager 3.1 SP2 J2EE Agent Guide*

For a listing of all Novell Audit events logged by Access Manager, see [Appendix C, “Access Manager Audit Events and Data,”](#) on page 141.

1.5.2 Querying Data and Generating Reports in Novell Audit

Queries let you create, run, edit, and delete queries and event verifications. You can create two kinds of queries in Access Manager: manual queries and saved queries. Manual queries are simply queries that are not saved; they only run one time. All verification queries are saved. Saved queries and verifications are listed in the Queries list and can be run again and again against different databases.

Access Manager uses queries to request information from MySQL and Oracle databases. All queries are defined in SQL. Although you must be familiar with the SQL language to create SQL query statements, this is the most powerful and flexible query method.

Novell Audit provides two tools to query events and generate reports: the Novell Audit iManager plug-in and Novell Audit Report (LReport).

The following sections provide more information on these tools:

- ◆ “The Novell Audit iManager Plug-In” on page 30
- ◆ “Novell Audit Report” on page 31

The Novell Audit iManager Plug-In

The Novell Audit iManager plug-in is a Web-based JDBC application that enables you to query MySQL and Oracle databases. All queries are defined in SQL.

iManager includes several predefined queries and it includes a Query Builder to help you define basic query statements. Of course, you can also build your own SQL query statements.

For complete information on defining and running queries in iManager, see the following sections in the *Novell Audit 2.0 Administration Guide* (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/bookinfo.html>).

- ◆ “Defining Your Query Databases in iManager” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/alorpq2.html#alost1z>)
- ◆ “Defining Queries in iManager” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/alorpq2.html#alpvc0a>)
- ◆ “Running Queries in iManager” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/alorpq2.html#alpv7ft>)
- ◆ “Verifying Event Authenticity in iManager” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/alorpq2.html#b34tzvi>)

- ◆ “Exporting Query Results in iManager” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/alorpq2.html#alqvrze>)
- ◆ “Printing Query Results in iManager” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/alorpq2.html#alqvzva>)

Novell Audit Report

Novell Audit Report is a Windows-based, ODBC-compliant application that can use SQL query statements or Crystal Decisions Reports to query Oracle and MySQL data stores (or any other database that has ODBC driver support). You can define your own SQL query statements or import existing query statements and reports. Query results are returned in simple data tables; rows represent individual records and columns represent fields within those records.

For complete information on defining and running queries in Novell Audit Report, see the following sections in the *Novell Audit 2.0 Administration Guide* (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/bookinfo.html>).

- ◆ “Novell Audit Report Interface” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/alorpgw.html#als9vcm>)
- ◆ “Defining Your Databases in Novell Audit Report” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/alorpgw.html#als94w4>)
- ◆ “Verifying Event Authenticity in Novell Audit Report” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/alorpgw.html#am9dbll>)
- ◆ “Working with Reports in Novell Audit Report” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/alorpgw.html#alsn2fj>)
- ◆ “Working with Queries in Novell Audit Report” (<http://www.novell.com/documentation/novellaudit20/novellaudit20/data/alorpgw.html#alshpuw>)

Backing Up and Restoring

2

The backup and restore utilities are scripts that are run from the command line, and they allow you to back up and restore your Access Manager configuration. An additional script allows you to export your configuration so Novell Support can help diagnose possible configuration problems.

IMPORTANT: You cannot restore data from a previous version of Access Manager to a new version. You should create a new configuration backup whenever you upgrade to a newer version of Access Manager.

The following sections describe how to back up and restore your Access Manager configuration, how to export your configuration for Novell Support, and how to restore the configuration to Identity Servers and Access Gateways:

- ♦ [Section 2.1, “How The Backup and Restore Process Works,” on page 33](#)
- ♦ [Section 2.2, “Backing Up the Access Manager Configuration,” on page 34](#)
- ♦ [Section 2.3, “Restoring an Administration Console Configuration,” on page 35](#)
- ♦ [Section 2.4, “Restoring an Identity Server,” on page 39](#)
- ♦ [Section 2.5, “Restoring an Access Gateway,” on page 40](#)
- ♦ [Section 2.6, “Running the Diagnostic Configuration Export,” on page 41](#)

2.1 How The Backup and Restore Process Works

- ♦ [Section 2.1.1, “Default Parameters,” on page 33](#)
- ♦ [Section 2.1.2, “The Process,” on page 33](#)

2.1.1 Default Parameters

Linux: All of the scripts call the `getparams.sh` script to request the parameters from the user. The `defbkparm.sh` script is created by the Access Manager installation. It contains the default parameters for several of options required by the underlying backup and restore utilities. If the entries in this file are commented out, the user is prompted for the additional parameters.

Windows: The default parameters are specified in the `defbkparm.properties` file. It contains the default parameters for several of options required by the underlying backup and restore utilities. If the entries in this file are commented out, the user is prompted for the additional parameters.

2.1.2 The Process

The backup script must be run on the primary Administration Console. It creates a ZIP file that contains all the certificates that the various devices are using and an encrypted LDIF file that contains the configuration parameters for all imported devices. This means that you do not need to back up the configuration of individual devices. By backing up the primary Administration Console, you back up the configuration of all Access Manager devices.

The backup script backs up the objects in the `ou=accessManagerContainer.o=novell` container. It does not back up the following:

- ♦ Admin user account and password
- ♦ Delegated administrator accounts, their passwords, or rights
- ♦ Role Based Services (RBS) configuration
- ♦ Modified configuration files on the devices such as the `web.xml` file
- ♦ Local files installed on devices such as touch files or log files
- ♦ Custom login pages, custom error pages, or custom messages

You need to perform your own backup of custom or modified configuration files.

For information on how to perform a configuration backup, see [Section 2.2, “Backing Up the Access Manager Configuration,”](#) on page 34.

The only time you need to restore a backup is when the Administration Console fails. If another device fails, you simply replace the hardware, reinstall the device, using the same IP address as the failed device, and the device imports into the Administration Console and acquires the configuration of the failed device. For the details of this process, see [Section 2.4, “Restoring an Identity Server,”](#) on page 39 and [Section 2.5, “Restoring an Access Gateway,”](#) on page 40.

If the Administration Console fails, you need to restore the files you backed up. In this case, you replace the hardware and reinstall the Administration Console using the same DNS name and IP address as the failed console. You then use the restore utility to restore the certificates and the device configuration. The Administration Console notifies all the devices that it is online, and they resume communicating with it rather than a secondary console. For details of this process, see [Section 2.3.1, “Restoring the Configuration on a Standalone Administration Console or with a Traditional SSL VPN Server,”](#) on page 36.

If the Identity Server was installed with the Administration Console, you need to be aware that the backup file contains only the Tomcat configuration information for the Administration Console. After you have installed the Administration Console and restored the configuration, you then need to install the Identity Server software. It will acquire its configuration parameters from the Administration Console. For details of this process, see [Section 2.3.2, “Restoring the Configuration with an Identity Server on the Same Machine,”](#) on page 37.

2.2 Backing Up the Access Manager Configuration

- 1 On the primary Administration Console, change to the utility directory.

Linux: `/opt/novell/devman/bin`

Windows Server 2003: `\Program Files\Novell\bin`

Windows Server 2008: `\Program Files (x86)\Novell\bin`

- 2 Run the following command:

Linux: `./ambkup.sh`

Windows: `ambkup.bat`

- 3 Enter the Access Manager administration password.
- 4 Re-enter the password for verification.

- 5 Specify a path for where you want the backup files stored. Press Enter to use the default location.
- 6 (Windows) Specify the name for the ZIP file.
- 7 Enter a password for encrypting and decrypting private keys, then re-enter it for verification. You must use the same password for both backup and restore.
- 8 Press Enter.

The backup script creates a ZIP file containing several files, including the certificate information. This file contains the following:

- ♦ The configurations store's CA key.
- ♦ The certificates contained in the configuration store.
- ♦ The trusted roots in the trustedRoots container of the accessManagerContainer object.
- ♦ An encrypted LDIF file, containing everything found in the OU=accessManagerContainer,O=novell container.
- ♦ A `server.xml` file containing the Tomcat configuration information for the Administration Console.

The trusted roots are backed up in both the LDIF file and the ZIP file. They are added to the ZIP file so that the ZIP file has the complete certificate-related configuration.

IMPORTANT: The backup utility prompts you for a location to store the backup file, so that it is not erased if you uninstall the product. The default location for Linux is `/root/nambkup` and for Windows it is `C:/nambkup`.

2.3 Restoring an Administration Console Configuration

The restore script replaces the configuration records in the configuration database with the records in the backup of the configuration store. It should be used to restore configuration data for one of the following types of scenarios:

- ♦ An upgrade failed and you need to return to the configuration before the upgrade.
- ♦ You want to return to the backed up configuration because the current modified configuration does not meet your needs.

The restoration steps are dependent upon whether the Administration Console is installed on its own machine or with other Access Manager components:

- ♦ [Section 2.3.1, “Restoring the Configuration on a Standalone Administration Console or with a Traditional SSL VPN Server,” on page 36](#)
- ♦ [Section 2.3.2, “Restoring the Configuration with an Identity Server on the Same Machine,” on page 37](#)
- ♦ [Section 2.3.3, “Restoring the Configuration with an ESP-Enabled SSL VPN Server,” on page 39](#)

If the primary Administration Console machine has failed, you have lost both the configuration and the configuration database. To recover from this scenario, you need to do more than restore the configuration. For instructions, see [Section 6.6, “Moving the Primary Administration Console to New Hardware,”](#) on page 109.

The restore script cannot be used to move the Administration Console to a different platform, even if the new machine is configured to use the same IP address and DNS name. The backup files contains path information that is specific to the operating system. To move the Administration Console from Linux to Windows or Windows to Linux, you need to install a secondary Administration Console on the desired platform, then promote it to being the primary Administration Console. For instructions on this process, see [Section 6.7, “Converting a Secondary Console into a Primary Console,”](#) on page 110.

2.3.1 Restoring the Configuration on a Standalone Administration Console or with a Traditional SSL VPN Server

- 1 Ensure that the .zip file created during the backup process is accessible.
- 2 Log in as root.
- 3 (Conditional) If you have modified the Tomcat password in the `server.xml` file on a Linux Administration Console, back up this file. This file is located in the following directory:

```
/etc/opt/novell/tomcat5
```

The feature to modify this password was removed in Access Manager 3.0 SP3.

- 4 Change to the utility directory.

Linux: `/opt/novell/devman/bin`

Windows Server 2003: `\Program Files\Novell\bin`

Windows Server 2008: `\Program Files (x86)\Novell\bin`

- 5 Run the following command:

Linux: `./amrestore.sh`

Windows: `amrestore.bat`

- 6 Enter and re-enter the Access Manager administration password.
- 7 (Windows) Enter the path to where the backup file is stored.
- 8 Enter the name of the backup file. Do not include the .zip extension.
- 9 Enter the private key encryption password, then press Enter.
- 10 Re-enter the private key encryption password, then press Enter.
- 11 (Conditional) If you modified the Tomcat password on the Linux machine:
 - 11a Restore the backup you made of the `server.xml` file to the Tomcat directory.

```
/etc/opt/novell/tomcat5
```
 - 11b Restart Tomcat with the following command:

```
/etc/init.d/novell-tomcat5 restart
```
- 12 (Windows) Reboot the machine.
- 13 (Conditional) If you have a secondary Administration Console installed, reboot the machines.

- 14** (Conditional) If any devices report certificate errors, you need to re-push the certificates.
- 14a** Click *Auditing > Troubleshooting > Certificates*.
 - 14b** Select the store that is reporting errors, then click *Re-push certificates*.
You can select multiple stores at the same time.
 - 14c** (Optional) To verify that the re-push of the certificates was successful, click *Security > Command Status*.

If you are restoring only the Administration Console, other components should still function properly after the restore.

2.3.2 Restoring the Configuration with an Identity Server on the Same Machine

Select the type of machine the Administration Console is installed on:

- ♦ [“Linux” on page 37](#)
- ♦ [“Windows” on page 38](#)

Linux

Whenever you run the `amrestore.sh` script, the Administration Console is restored as a standalone Administration Console. You must perform the steps described in [Step 10](#) to restore your Identity Server into the configuration.

- 1** Ensure that the `.zip` file created during the backup process is accessible.
- 2** Log in as `root`.
- 3** Change to the `/opt/novell/devman/bin` directory.
- 4** Run the following command:

```
./amrestore.sh
```
- 5** Enter the Access Manager administration user ID.
- 6** Enter the Access Manager administration password.
- 7** Enter the name of the backup file. Do not include the `.zip` extension.
- 8** Enter the private key encryption password, then press Enter.
- 9** Re-enter the private key encryption password, then press Enter.
- 10** For the Identity Server, complete the following steps after the restore has finished:
 - 10a** Remove the Identity Server from the cluster configuration. (See [“Removing a Server from a Cluster Configuration”](#) in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.)
 - 10b** Delete the Identity Server from the Administration Console. (See [“Managing an Identity Server”](#) in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.)
 - 10c** Uninstall the Identity Server. (See [“Uninstalling the Identity Server”](#) in the *Novell Access Manager 3.1 SP2 Installation Guide*.)

This is required if the Identity Server is installed on the machine. If you installed the Identity Server before running the `amrestore.sh` script, you need to uninstall the Identity Server.

- 10d** Install the Identity Server. (See “[Installing the Novell Identity Server](#)” in the *Novell Access Manager 3.1 SP2 Installation Guide*.)
- 10e** If you have customized login pages, error pages, messages, or configuration files, copy these files to the Identity Server.
- 10f** Reassign the Identity Server to the cluster configuration that it was removed from. (See “[Assigning an Identity Server to a Cluster Configuration](#)” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.)
- 10g** Update the Identity Server.
- 11** (Conditional) If any devices report certificate errors, you need to re-push the certificates.
 - 11a** Click *Auditing > Troubleshooting > Certificates*.
 - 11b** Select the store that is reporting errors, then click *Re-push certificates*.
You can select multiple stores at the same time.
 - 11c** (Optional) To verify that the re-push of the certificates was successful, click *Security > Command Status*.

Windows

To perform a restore when a Windows Administration Console and an Identity Server are installed on the same machine:

- 1** Log in as the administrator user.
- 2** Run the Access Manager Restore utility.
 - 2a** From a command line, change to the utility directory:
 - Windows Server 2003:** \Program Files\Novell\bin directory.
 - Windows Server 2008:** \Program Files (x86)\Novell\bin directory.
 - 2b** Specify *amrestore.bat*.
 - 2c** Answer the prompts.
- 3** Remove the Identity Server from the cluster configuration. (See “[Removing a Server from a Cluster Configuration](#)” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.)
- 4** Delete the Identity Server from the Administration Console. (See “[Managing an Identity Server](#)” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.)
- 5** Install the Identity Server on the Administration Console. (See “[Installing the Novell Identity Server](#)” in the *Novell Access Manager 3.1 SP2 Installation Guide*.)
- 6** If you have customized login pages, error pages, messages, or configuration files, copy these files to the Identity Server.
- 7** Reassign the Identity Server to the cluster configuration that it was removed from. (See “[Assigning an Identity Server to a Cluster Configuration](#)” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.)
- 8** Update the Identity Server.

2.3.3 Restoring the Configuration with an ESP-Enabled SSL VPN Server

Whenever you run the `amrestore.sh` script, the Administration Console is restored as a standalone Administration Console. You must perform the steps described in [Step 10](#) to restore your ESP-enabled SSL VPN server into the configuration.

- 1 Ensure that the `.zip` file created during the backup process is accessible.
- 2 Log in as `root`.
- 3 Change to the `/opt/novell/devman/bin` directory.
- 4 Run the following command:

```
./amrestore.sh
```
- 5 Enter the Access Manager administration user ID.
- 6 Enter the Access Manager administration password.
- 7 Enter the name of the backup file. Do not include the `.zip` extension.
- 8 Enter the private key encryption password, then press Enter.
- 9 Re-enter the private key encryption password, then press Enter.
- 10 For the SSL VPN Server, complete the following steps after the restore has finished:
 - 10a Remove the SSL VPN Server from the cluster configuration.
 - 10b Delete the SSL VPN Server from the Administration Console.
 - 10c Uninstall the SSL VPN server.
 - 10d Install the SSL VPN server.
 - 10e Reassign the SSL VPN server to the cluster configuration that it was removed from.
 - 10f Update the SSL VPN server.
- 11 (Conditional) If any devices report certificate errors, you need to re-push the certificates.
 - 11a Click *Auditing > Troubleshooting > Certificates*.
 - 11b Select the store that is reporting errors, then click *Re-push certificates*.

You can select multiple stores at the same time.
 - 11c (Optional) To verify that the re-push of the certificates was successful, click *Security > Command Status*.

2.4 Restoring an Identity Server

If an Identity Server machine experiences a hardware failure, such as a failed hard disk, you can preserve its configuration and apply it to the replacement machine

- 1 Remove the Identity Server from the Identity Server cluster configuration. (See “[Removing a Server from a Cluster Configuration](#)” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.)
- 2 Delete the Identity Server from the Administration Console. (See “[Managing an Identity Server](#)” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.)
- 3 Uninstall the Identity Server. (See “[Uninstalling the Identity Server](#)” in the *Novell Access Manager 3.1 SP2 Installation Guide*.)

This might not be necessary, if you are using a new machine for the restored Identity Server.

- 4 Install the new Identity Server, which imports it into the Administration Console. (See “[Installing the Novell Identity Server](#)” in the *Novell Access Manager 3.1 SP2 Installation Guide*.)
- 5 If you have customized login pages, error pages, messages, or configuration files, copy these files to the Identity Server.
- 6 Assign the new server to the Identity Server cluster configuration. (See “[Assigning an Identity Server to a Cluster Configuration](#)” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.)

2.5 Restoring an Access Gateway

If an Access Gateway machine experiences a hardware failure, such as a failed hard disk, you can preserve its configuration and have it applied to the replacement machine.

- ♦ [Section 2.5.1, “Clustered Access Gateway,” on page 40](#)
- ♦ [Section 2.5.2, “Single Access Gateway,” on page 41](#)

2.5.1 Clustered Access Gateway

If the hardware fails on an Access Gateway machine that belongs to a cluster:

- 1 In the Administration Console, view the configuration of the cluster. Click *Devices > Access Gateways*.
- 2 (Conditional) If the failed Access Gateway is the primary server, assign another server to be the primary server:
 - 2a On the Access Gateways page, click *[Name of Cluster] > Edit*.
 - 2b For the *Primary Server* field, select another server to be the primary server, then click *OK > Close*.
 - 2c Click *Identity Servers > Update*.
- 3 Delete the failed Access Gateway from the cluster. Click *Access Gateways*, select the failed Access Gateway, then click *Actions > Remove from Cluster*.

IMPORTANT: Do not delete the Access Gateway from the Administration Console.

- 4 On the new machine, install the Access Gateway, specifying the same Administration Console, IP address, host name, and domain name as the failed machine.
- 5 (Conditional) If you have customized error messages, copy the message files to the Access Gateway.
- 6 (Conditional) If you have configured the Access Gateway Alliance to use touch files, re-create the touch files on the Access Gateway Appliance. For a list of touch files, see “[Using Touch Files](#)” in the *Novell Access Manager 3.1 SP2 Access Gateway Guide*.
- 7 When the machine imports into the Administration Console, add the machine to the Access Gateway cluster:
 - 7a In the Administration Console, click *Devices > Access Gateways*.

- 7b** Select the name of the Access Gateway, then click *Actions > Assign to Cluster > [Name of Cluster]*.
- 7c** Update the Access Gateway.

2.5.2 Single Access Gateway

If the failed Access Gateway is a single machine and you want to preserve its configuration:

- 1** Do not delete the Access Gateway from the Administration Console.
If you delete the Access Gateway from the Administration Console, the configuration information is deleted.
- 2** On the new machine, install the Access Gateway software, using the same IP address, host name, and domain name as the failed device and specifying the same Administration Console.
- 3** (Conditional) If you have customized error messages, copy the message files to the Access Gateway.
- 4** (Conditional) If you have configured the Access Gateway Appliance to use touch files, re-create the touch files on the Access Gateway Appliance. For a list of touch files, see “[Using Touch Files](#)” in the *Novell Access Manager 3.1 SP2 Access Gateway Guide*.
- 5** When the installation has completed and the device has been imported in the Administration Console, verify the following:
 - 5a** Check its trusted relationship with the Identity Server. Click *Devices > Access Gateways > Edit > Reverse Proxy / Authentication*.
 - 5b** If you have configured the Access Gateway to use SSL, reconfigure the certificates for the listener. Click *Devices > Access Gateways > Edit > [Name of Reverse Proxy]*.
 - 5c** Save any changes, and update the Access Gateway.

2.6 Running the Diagnostic Configuration Export

On a Linux Administration Console, you can create an `.ldif` file that you can export for diagnostic purposes:

- 1** Log in as `root`.
- 2** Change to the `/opt/novell/devman/bin` directory.
- 3** Run the following command:

```
./amdiagcfg.sh
```
- 4** Enter the Access Manager administration user ID.
- 5** Enter the Access Manager password.
- 6** Re-enter the password for verification.
- 7** Press Enter.

The diagnostic configuration export utility is almost identical to the backup utility with two differences: the ZIP file is not created, and the final LDIF file is scanned to have passwords removed. Passwords are blanked out by a program called `Strippasswd`.

Strippasswd removes occurrences of passwords in the LDIF file, replacing them with empty strings. If you look at the LDIF file, you will see that password strings are blank. You might see occurrences within the file or text that looks similar to password="String". These are not instances of passwords, but rather definitions that describe passwords as string types.

The LDIF file can then be sent to Novell Support for help in diagnosing configuration problems.

Security and Certificate Management

3

- ♦ [Section 3.1, “Understanding How Access Manager Uses Certificates,” on page 43](#)
- ♦ [Section 3.2, “Creating Certificates,” on page 49](#)
- ♦ [Section 3.3, “Managing Certificates and Keystores,” on page 58](#)
- ♦ [Section 3.4, “Managing Trusted Roots and Trust Stores,” on page 67](#)
- ♦ [Section 3.5, “Security Considerations for Certificates,” on page 71](#)
- ♦ [Section 3.6, “Assigning Certificates to Access Manager Devices,” on page 72](#)

3.1 Understanding How Access Manager Uses Certificates

Access Manager allows you to manage centrally stored certificates used for digital signatures and data encryption. eDirectory resides on the Administration Console and is the main certificate store for all of the Access Manager components. If you use a Novell Certificate Server, you can create certificates there and import them into Access Manager.

By default, all Access Manager components (Identity Server, Access Gateway, SSL VPN, and J2EE agents) trust the local Access Manager certificate authority (CA). However, if the Identity Server is configured to use an SSL certificate signed externally, the trust store of the Embedded Service Provider for each component must be configured to trust this new CA.

Certificate management commands issued from a secondary Administration Console can work only if the primary console is also running properly. Other commands can work independently of the primary console.

You can create and distribute certificates to the following components:

- ♦ **Identity Server:** Uses certificates and trust stores to provide secure authentication to the Identity Server and enable encrypted content from the Identity Server portal, via HTTPS. Certificates also provide secure communications between trusted Identity Servers and user stores.

Liberty and SAML 2.0 protocol messages that are exchanged between identity and service providers often need to be digitally signed. The Identity Server uses the signing certificate included with the metadata of a trusted provider to validate signed messages from the trusted provider. For protocol messages to be exchanged between providers through SSL, each provider must trust the CA of the other provider. You must import the public key of the CA used by the other provider.

The Identity Server also has a trust store for OCSP (Online Certificate Status Protocol) certificates, which is used to check the revocation status of a certificate.

- ♦ **Access Gateway:** Uses server certificates and trusted roots to protect Web servers, provide single sign-on, and enable the product’s data confidentiality features, such as encryption. They are used for background communication with the Identity Server and policy engine and to establish trust between the Identity Server and the Access Gateway.
- ♦ **SSL VPN:** Uses server certificates and trusted roots to secure access to non-HTTP applications.
- ♦ **J2EE Agent:** Uses certificates and trust stores to establish trust between the J2EE Agent and the Identity Server, and for SSL between the J2EE server and the Identity Server.

To ensure the validity of X.509 certificates, Access Manager supports both Certificate Revocation Lists (CRLs) and Online Certificate Status Protocol (OCSP) methods of verification.

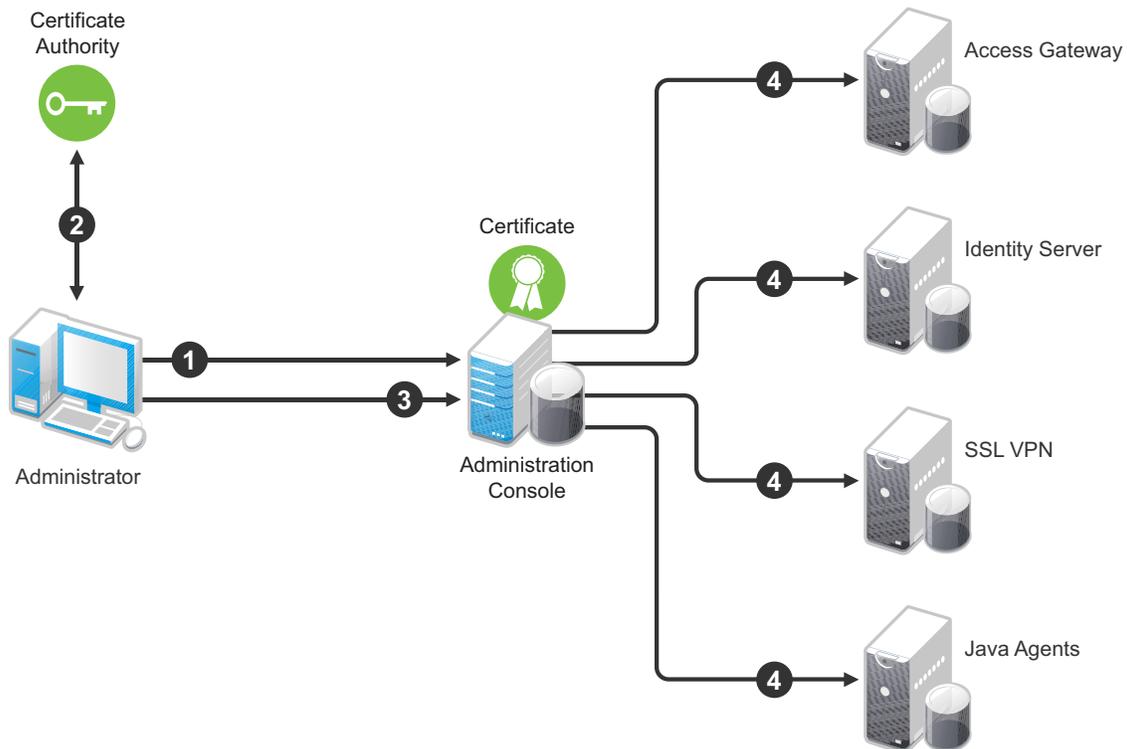
Access Manager stores the certificates that a device has been configured to use in trust stores and keystores. This section describes the following certificate features:

- ♦ [Section 3.1.1, “Process Flow,” on page 44](#)
- ♦ [Section 3.1.2, “Access Manager Trust Stores,” on page 45](#)
- ♦ [Section 3.1.3, “Access Manager Keystores,” on page 47](#)

3.1.1 Process Flow

You can install and distribute certificates to the Access Manager product components and configure how the components use certificates. This includes central storage, distribution, and expired certificate renewal. [Figure 3-1](#) illustrates the primary administrative actions for certificate management in Access Manager:

Figure 3-1 Certificate Management



1. Generate a certificate signing request (CSR). See [Section 3.2.4, “Generating a Certificate Signing Request,” on page 57.](#)
2. Send the CSR to the external certificate authority (CA) for signing.
A CA is a third-party or network authority that issues and manages security credentials and public keys for message encryption. The CA’s certificate is held in the configuration store of the computers that trust the CA.
3. Import the signed certificate and CA chain into the configuration store. See [“Importing Public Key Certificates \(Trusted Roots\)” on page 68.](#)
4. Assign certificates to devices. See [“Assigning Certificates to Access Manager Devices” on page 72.](#)

If you are unfamiliar with public key cryptography concepts, see [“Public Key Cryptography Basics” \(http://www.novell.com/documentation/crt311/crtadmin/data/a2uqry.html#a2uqry\)](#) in the *Novell Certificate Server 3.1.1 Guide* ([http://www.novell.com/documentation/crt33/crtadmin/data/a2ebomw.html](#)).

See [Appendix A, “Certificates Terminology,” on page 133](#) for information about certificate terminology.

3.1.2 Access Manager Trust Stores

A trust store contains trusted roots, which are public certificates of known, trusted certificate authorities. Access Manager creates the trust stores listed below for the devices that it manages. The trust stores are created when you import a device into the Administration Console. If you have not imported a particular device type, the trust store for that device type does not exist. If you have imported multiple devices of the same type, the Administration Console creates an instance of the trust store for each device.

When a certificate has been created by a root CA, the trust store needs to contain only the public certificate of the CA. However, some certificates are created by an intermediate CA, which has been issued by a root CA. When intermediate CAs are involved, all the public certificates of the CAs in the chain need to be included in the trust store.

The Administration Console creates a trust store in the file system of the device that is assigned to the trust store.

- ◆ **Linux:** /opt/novell/devman/jcc/certs/<device>
- ◆ **Windows Server 2003:** \Program Files\novell\devman\jcc\certs\<device>
- ◆ **Windows Server 2008 Identity Server:** \Program Files (x86)\novell\devman\jcc\certs\ <device>
- ◆ **Windows Server 2008 Access Gateway Service:** \Program Files\novell\devman\jcc\certs\<device>

The <device> can be idp (for the Identity Server), esp (for the Embedded Service Providers, including Access Gateways, J2EE agents, and SSL VPN servers), or sslvpn (for the SSL VPN server).

To view the trust stores:

- 1 In the Administration Console, click *Security > Trusted Roots*.

- 2 Select a trusted root, then click *Add Trusted Root to Trust Stores*.
- 3 Click the *Select Keystore* icon.

The list can include the following trust stores:

Trust Store: This Identity Server trust store contains the trusted root certificates of all the providers that it trusts. Liberty and SAML 2.0 protocol messages that are exchanged between identity and service providers often need to be digitally signed. A provider uses the signing certificate included with the metadata of a trusted provider to validate signed messages from the trusted provider. The trusted root of the CA that created the signing certificate for the provider needs to be in this trust store.

To use SSL for exchanging messages between providers, each provider must trust the SSL certificate authority of the other provider. You must import the root certificate chain for the other provider. Failure to do so causes numerous system errors.

This trust store is also used to store the trusted root certificates of the user stores that it has been configured to use.

OCSP Trust Store: The Identity Server uses this trust store for OCSP (Online Certificate Status Protocol) certificates. OCSP is a method used for checking the revocation status of a certificate. To use this feature, you must set up an OCSP server. The Identity Server sends an OCSP request to the OCSP server to determine if a certain certificate has been revoked. The OCSP server replies with the revocation status. If this revocation checking protocol is used, the Identity Server does not cache or store the information in the reply, but sends a request every time it needs to check the revocation status of a certificate. The OCSP reply is signed by the OCSP server. To verify that it was signed by the correct OCSP server, the OCSP server certificate needs to be added to this trust store.

SSLVPN Trust Store: This trust store is used by the traditional SSL VPN server that is configured as a protected resource of the Access Gateway. The trust store contains the trusted root certificate of the Identity Server that the Access Gateway has been configured to trust.

This trust store does not use the default location; it is located in the `/etc/opt/novell/sslvpn/certs` directory.

ESP Trust Store (SSL VPN): This trust store is used by an SSL VPN server that is ESP-enabled. It contains the trusted root certificate of the Identity Server that it has been configured to trust. It usually contains one certificate unless you have modified the SSL VPN server to trust one Identity Server, and then modify the SSL VPN server to trust a different Identity Server. If you are using certificates generated by the Access Manager CA, the root certificate of this CA is automatically added to this trust store. If the Identity Server is using a certificate generated by an external CA, you need to add the trusted root certificate of that CA to this trust store.

ESP Trust Store (Access Gateway): The Access Gateway EPS trust store contains the trusted root certificate of the Identity Server that it has been configured to trust. It usually contains one certificate unless you configure the Access Gateway to trust one Identity Server, and then modify the Access Gateway to trust a different Identity Server. If you are using certificates generated by the Access Manager CA, the root certificate of this CA is automatically added to this trust store. If the Identity Server is using a certificate generated by an external CA, you need to add the trusted root certificate of that CA to this trust store.

Proxy Trust Store: When SSL is set up between the Access Gateway and its Web servers, the Access Gateway uses this trust store for the trusted root certificates of the Web servers.

This trust store does not use the default location:

- ◆ **Access Gateway Appliance:** `/opt/novell/conf/keys`

- ♦ **Linux Access Gateway Service:** /opt/novell/apache2/cacerts
- ♦ **Windows Access Gateway Service:** \Program Files\Novell\apache\cacerts

ESP Trust Store (J2EE Agent): The agent ESP trust store contains the trusted root certificate of the Identity Server that it has been configured to trust. It usually contains one certificate unless you configure the agent to trust one Identity Server, and then modify the agent to trust a different Identity Server. If you are using certificates generated by the Access Manager CA, the root certificate of this CA is automatically added to this trust store. If the Identity Server is using a certificate generated by an external CA, you need to add the trusted root certificate of that CA to this trust store.

4 Click *Cancel* twice.

3.1.3 Access Manager Keystores

A keystore is a location, such as a file, containing keys and certificates. Access Manager components and agents can access the keystore to retrieve certificates and keys as needed. Keystores for Access Manager are already defined for the components.

The Administration Console creates a keystore in the file system of the device that is assigned to the keystore. The operating system of the device determines the location:

- ♦ **Linux:** /opt/novell/devman/jcc/certs/<device>
- ♦ **Windows Server 2003:** \Program Files\novell\devman\jcc\certs\<device>
- ♦ **Windows Server 2008 Identity Server:** \Program Files (x86)\novell\devman\jcc\certs\ <device>
- ♦ **Windows Server 2008 Access Gateway Service:** \Program Files\novell\devman\jcc\certs\<device>

The <device> can be idp (for the Identity Server), esp (for the Embedded Service Providers, including Access Gateways, J2EE agents, and SSL VPN servers), or sslvpn (for the SSL VPN server).

To view the keystores:

- 1 In the Administration Console, click *Security > Certificates*.
- 2 Click the name of a certificate, then click *Add Certificate to Keystores*.
- 3 Click the *Select Keystore* icon.

Access Manager creates keystores for the following devices:

- ♦ [“Identity Server Keystores” on page 47](#)
- ♦ [“Access Gateway Keystores” on page 48](#)
- ♦ [“J2EE Agent Keystores” on page 48](#)
- ♦ [“SSL VPN Keystores” on page 49](#)
- ♦ [“Keystores When Multiple Devices Are Installed on the Administration Console” on page 49](#)

4 Click *Cancel* twice.

Identity Server Keystores

Access Manager creates the following keystores for each Identity Server cluster configuration:

Signing: Contains the certificate that is used for signing the assertion or specific parts of the assertion.

Encryption: Contains the certificate that is used to encrypt specific fields or data in assertions.

SSL Connector: Contains the certificate that the Identity Server uses for SSL connections. If multiple devices are installed on the same machine, the Identity Server uses the COMMON_TOMCAT_CLUSTER keystore.

Provider Introductions SSL Connector: Contains the certificate that you configure when you set up the Identity Server to provide introductions to service providers that are trusted members of a service domain. The subject name of this certificate needs to match the DNS name of the service domain.

Consumer Introductions SSL Connector: Contains the certificate that you configure when you set up the Identity Server to consume authentications provided by other identity providers that are trusted members of a service domain. The subject name of this certificate needs to match the DNS name of the service domain.

Access Gateway Keystores

Access Manager creates the following keystores for each Access Gateway or cluster:

Signing: Contains the certificate that is used for signing the assertion or specific parts of the assertion.

Encryption: Contains the certificate that is used to encrypt specific fields or data in assertions.

ESP Mutual SSL: Contains the certificate that is used for SSL when you have established SSL communication between the Access Gateway and the Identity Server. The public key (trusted root) of the certificate authority that created the certificate needs to be in the Identity Server's trust store.

Proxy Key Store: Contains the certificate that is used for SSL when you have enabled SSL between a reverse proxy and the browsers. The public key (trusted root) of the certificate authority that created the certificate needs to be in browser's trust store for the SSL connection to work without warnings. If you create multiple reverse proxies and enable them for SSL, each reverse proxy needs a certificate, and the subject name of the certificate needs to match the DNS name of the reverse proxy.

This keystore does not use the default location:

- ◆ **Access Gateway Appliance:** /opt/novell/conf/keys
- ◆ **Linux Access Gateway Service:** /opt/novell/apache2/certs
- ◆ **Windows Access Gateway Service:** \Program Files\Novell\apache\certs

J2EE Agent Keystores

Access Manager creates the following keystores for each J2EE Agent:

Signing: Contains the certificate that is used for signing the assertion or specific parts of the assertion.

Encryption: Contains the certificate that is used to encrypt specific fields or data in assertions.

ESP Mutual SSL: Contains the certificate that is used for SSL, when you have established SSL communication between the J2EE agent and the Identity Server. The public key (trusted root) of the certificate authority that created the certificate needs to be in the Identity Server's trust store.

SSL VPN Keystores

Access Manager creates the following keystores for each SSL VPN server or cluster:

Signing: Contains the certificate that is used for signing the assertion or specific parts of the assertion.

Encryption: Contains the certificate that is used to encrypt specific fields or data in assertions.

ESP Mutual SSL: Contains the certificate that is used for SSL when you have established SSL communication between the ESP-enabled SSL VPN server and the Identity Server. The public key (trusted root) of the certificate authority that created the certificate needs to be in the Identity Server's trust store.

SSLVPN Secure Tunnel: Contains the certificate that encrypts the data exchanged between SSL VPN client and the SSL VPN server, after the SSL VPN connection is made.

This keystore does not use the default location; it is located in the `/etc/opt/novell/sslvpn/certs` directory.

SSL Connector: Contains the certificate that encrypts authentication information between the SSL VPN client browser and the SSL VPN server.

Keystores When Multiple Devices Are Installed on the Administration Console

Access Manager creates the following keystore when the Identity Server and the SSL VPN server are installed on the Administration Console.

COMMON_TOMCAT_CLUSTER: Contains the certificate that is used for SSL connections.

The location of this keystore depends upon which device was installed last: the Identity Server or the SSL VPN server. If the Identity Server was installed last, the keystore is in the `idp` directory. If the SSL VPN server was installed last, the keystore is in the `sslvpn` directory.

3.2 Creating Certificates

Access Manager comes with certificates for testing purposes. The test certificates are called test-signing, test-encryption, test-provider, test-consumer, and test-connector. At a minimum, you must create two SSL certificates: one for Identity Server test-connector and one for the Access Gateway reverse proxy. Then you replace the predefined certificates with the new ones.

If you install a secondary Administration Console, the certificate authority (CA) is installed with the first instance of eDirectory, and the secondary consoles have eDirectory replicas and therefore no CA software. All certificate management must be done from the primary Administration Console. Certificate management commands issued from a secondary Administration Console can work only if the primary console is also running properly. Other commands can work independently of the primary console.

IMPORTANT: Before generating any certificates with the Administration Console CA, make sure time is synchronized within one minute among all of your Access Manager devices. If the time of the Administration Console is ahead of the device for which you are creating the certificate, the device rejects the certificate.

1 In the Administration Console, click *Security > Certificates*.

<input type="checkbox"/>	Name	Subject	Devices	Starting Date	Ending Date	State
<input type="checkbox"/>	test-connector	O=novell, OU=accessManager, CN=test-connector	1 Devices ▼	January 27, 2010	January 27, 2012	
<input type="checkbox"/>	test-consumer	O=novell, OU=accessManager, CN=test-consumer		January 27, 2010	January 27, 2012	
<input type="checkbox"/>	test-encryption	O=novell, OU=accessManager, CN=test-encryption	1 Devices ▼	January 27, 2010	January 27, 2012	
<input type="checkbox"/>	test-provider	O=novell, OU=accessManager, CN=test-provider		January 27, 2010	January 27, 2012	
<input type="checkbox"/>	test-signing	O=novell, OU=accessManager, CN=test-signing	1 Devices ▼	January 27, 2010	January 27, 2012	
<input type="checkbox"/>	test-stunnel	O=novell, OU=accessManager, CN=test-stunnel		January 27, 2010	January 27, 2012	

2 Select from the following actions:

New: To create a new certificate, click *New*. For information about the fields you need to fill in, see [Section 3.2.1, “Creating a Locally Signed Certificate,” on page 51](#) and [Section 3.2.4, “Generating a Certificate Signing Request,” on page 57](#).

Delete: To delete a certificate, select the certificate, then click *Delete*. If the certificate is assigned to a keystore, a warning message appears. You must remove a certificate from all keystores before it can be deleted.

Import Private/Public Keypair: To import a key pair, click *Actions > Import Private/Public Keypair*. For more information, see [Section 3.3.6, “Importing a Private/Public Key Pair,” on page 65](#).

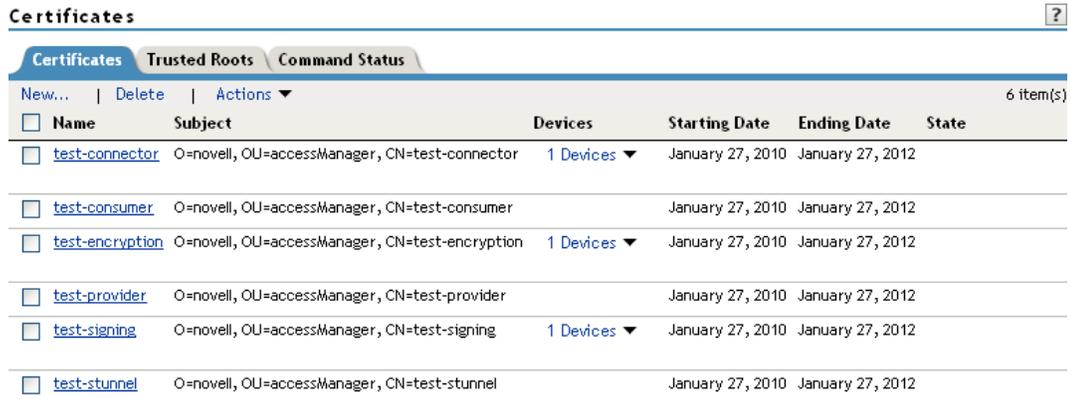
Add Certificate to Keystores: To add a certificate to a keystore, click *Actions > Add Certificate to Keystore*. For more information, see [Section 3.3.2, “Adding a Certificate to a Keystore,” on page 61](#).

View Certificate Details: To view certificate details, renew a certificate, or export keys, click the name of the certificate. For more information, see [Section 3.3.1, “Viewing Certificate Details,” on page 59](#).

3.2.1 Creating a Locally Signed Certificate

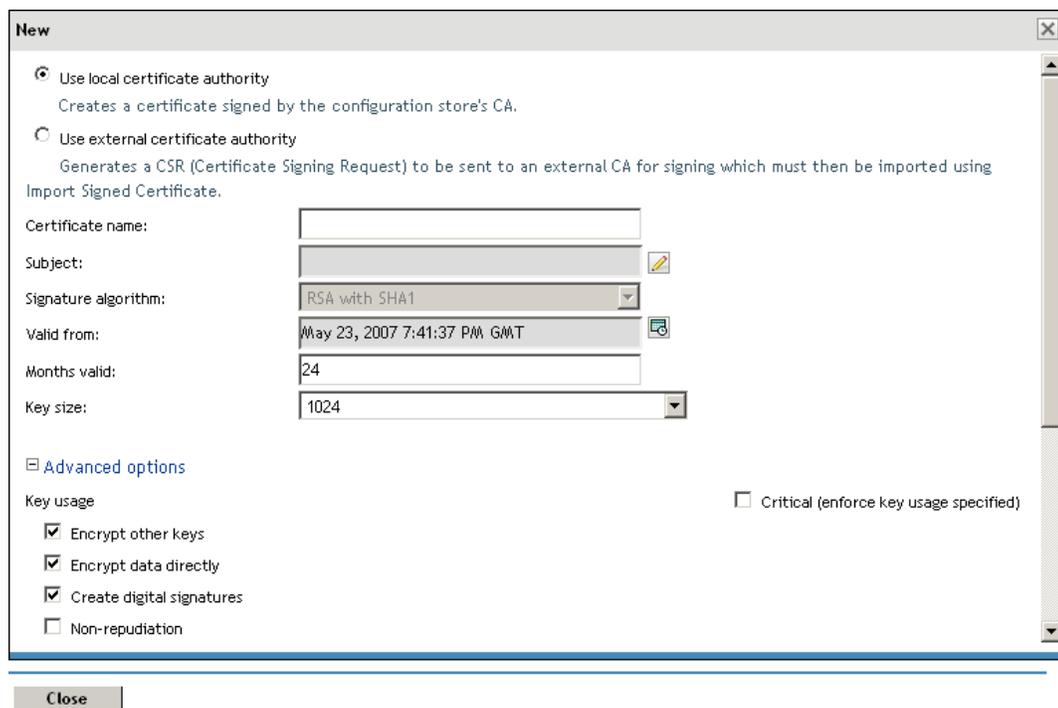
By default, the Access Manager installation process creates the local CA that can issue and sign certificates and installs a certificate server that generates certificates, keys, and CSRs (certificate signing requests) and imports certificates and keys.

- 1 In the Administration Console, click *Security > Certificates*.



Name	Subject	Devices	Starting Date	Ending Date	State
<input type="checkbox"/> test-connector	O=novell, OU=accessManager, CN=test-connector	1 Devices ▼	January 27, 2010	January 27, 2012	
<input type="checkbox"/> test-consumer	O=novell, OU=accessManager, CN=test-consumer		January 27, 2010	January 27, 2012	
<input type="checkbox"/> test-encryption	O=novell, OU=accessManager, CN=test-encryption	1 Devices ▼	January 27, 2010	January 27, 2012	
<input type="checkbox"/> test-provider	O=novell, OU=accessManager, CN=test-provider		January 27, 2010	January 27, 2012	
<input type="checkbox"/> test-signing	O=novell, OU=accessManager, CN=test-signing	1 Devices ▼	January 27, 2010	January 27, 2012	
<input type="checkbox"/> test-stunnel	O=novell, OU=accessManager, CN=test-stunnel		January 27, 2010	January 27, 2012	

- 2 Click *New*.



New

Use local certificate authority
Creates a certificate signed by the configuration store's CA.

Use external certificate authority
Generates a CSR (Certificate Signing Request) to be sent to an external CA for signing which must then be imported using Import Signed Certificate.

Certificate name:

Subject:

Signature algorithm:

Valid from:

Months valid:

Key size:

Advanced options

Key usage Critical (enforce key usage specified)

Encrypt other keys

Encrypt data directly

Create digital signatures

Non-repudiation

- 3 Select the following option:

Use local certificate authority: Creates a certificate signed by the local CA (or Organizational CA), and creates the private key. For information about creating a CSR, see [“Generating a Certificate Signing Request” on page 57](#).

- 4 Provide a certificate name:

Certificate name: The name of the certificate. Pick a unique, system-wide name for the certificate that you can easily associate with the certificate's purpose. The name must contain only alphanumeric characters and no spaces.

- 5 For *Subject*, click the *Edit* button to display a dialog box that lets you add the appropriate attributes for the subject name.

Edit Subject ?

Commonly used attributes

Common name:	<input type="text"/>
Organizational unit:	<input type="text"/>
Organization:	<input type="text"/>
City or town:	<input type="text"/>
State or province:	<input type="text"/>
Country:	<input type="text"/>

Additional attributes

----- Select one ----- ▾	:	<input type="text"/>
----- Select one ----- ▾	:	<input type="text"/>

The subject is an X.500 formatted distinguished name that identifies the entity that is bound to the public key in an X.509 certificate. Choose the subject name that the browser expects to find in the certificate. The name you enter must be fully distinguished. Completing all the fields creates a fully distinguished name that includes the appropriate types (such as C for country, ST for state, L for location, O for organization, OU for organizational unit, and CN for common name). For example, cn=AcmeWebServer.ou=Sales.o=Acme.c=US.

Common name: If you are creating a certificate for an Identity Server, specify the DNS name of the Identity Server. If you are creating a certificate for an Access Gateway, specify the published DNS name of the proxy service. Specifying values for the other attributes is optional. For more information about the other attributes, see [Section 3.2.2, "Editing the Subject Name," on page 54](#).

- 6 Click *OK*, then fill in the following fields:

Signature algorithm: The algorithm you want to use (SHA-1, MD-2, or MD-5). SHA-1 is currently recommended.

Valid from: The date from which the certificate is valid. For externally signed certificates, the external certificate authority sets the validity period.

Months valid: The number of months that the certificate is valid.

Key size: The size of the key. Select 512, 1024, 2048, or 4096.

7 (Optional) To configure advanced options, click *Advanced Options*.

8 Configure the following options as necessary for your organization:

Critical: Specifies that an application should reject the certificate if the application does not understand the key usage extensions.

Encrypt other keys: Specifies that the certificate is used to encrypt keys.

Encrypt data directly: Encrypts data for private transmission to the key pair owner. Only the intended receiver can read the data.

Create digital signatures: Specifies that the certificate is used to create digital signatures.

Non-repudiation: Links a digital signature to the signer and the data. This prevents others from duplicating the signature because no one else has the signer's private key. Additionally, the signer cannot deny having signed the data.

9 (Conditional) If you are creating a key for a certificate authority, configure the following options:

This key is for a Certificate Authority: Specifies that this certificate is for the local configuration (eDirectory) certificate authority.

If you create a new CA, all the keys signed by the CA being replaced no longer have a trusted CA. You might also need to reassign the new CA to all the trust stores that contained the old CA.

Critical: Enforces the basic constraints you specify. Select one of the following:

- ♦ **Unlimited:** Specifies no restriction on the number of subordinate certificates that the CA can verify.
- ♦ **Do not allow intermediate signing certificates in certificate chain:** Prevents the CA from creating other CAs, but it can create server or user certificates.
- ♦ **Number of allowable intermediate signing certificates in signing chain:** Specifies how many subordinate certificates are allowed in the certificate chain. Values must be 1 or more. Entering 0 creates only entity objects.

- 10 (Optional) To create subject alternative names used by the certificate, click the *Edit Subject Alternate Names* button, then click *New*.

The screenshot shows a dialog box titled "Subject Alternative Names" with a help icon. Below the title bar are "New..." and "Delete" buttons. A sub-dialog titled "New" is open, containing a "Name type:" dropdown menu with "Directory name" selected. Below this is a note: "*Note that not all of these Name Types will be supported by all CAs." Underneath is a "Name:" text input field. At the bottom of the "New" dialog are "OK" and "Cancel" buttons.

Alternate names can represent the entity identified by the certificate. The certificate can identify the subject CN=www.OU=novell.O=com, but the subject can also be known by an IP address, such as 222.111.100.101, or a URI, such as www.novell.com, for example. For more information, see [Section 3.2.3, “Assigning Alternate Subject Names,” on page 56](#).

- 11 Click *OK*.
- 12 (Conditional) If you assigned alternate names, determine how you want applications to handle the alternate names. Select *Critical* if you want an application that does not understand the alternate name extensions to reject the certificate.
- 13 Click *OK*.

3.2.2 Editing the Subject Name

- 1 Fill in one or more of the following attributes.

The following attributes are the most common ones used in certificate subjects:

Common name: The DNS name of the server.

Specify the value, for example AcmeWebServer.provo.com. Do not include the type (cn=). The UI adds that for you.

For the Identity Server, this is the domain name of the base URL of the Identity Server configuration. This value cannot be an IP address or begin with a number, in order to ensure that trust does not fail between providers.

For the Access Gateway, this is the published DNS name of the proxy service.

Organizational unit: Describes departments or divisions.

Organization: Differentiates between organizational divisions.

City or town: Commonly referred to as the Locality.

State or province: Commonly referred to as the State. Do not abbreviate the name.

Country: The country, such as US.

- 2 Use the drop-down menus to add additional attributes.

These values allow you to specify additional fields that are supported by eDirectory, and you can include them as part of the subject to further identify the entity represented by the certificate.

CN: The *Common name* attribute in the list of *Commonly used attributes* (OID: 2.5.4.3)

C: The *Country* attribute in the list of *Commonly used attributes* (OID: 2.5.4.6)

SN: The surname attribute (OID: 2.5.4.4)

L: The locality attribute, which is the *City or town* attribute in the list of *Commonly used attributes* (OID: 2.5.4.7)

ST: The *State or province* attribute in the list of *Commonly used attributes* (OID: 2.5.4.8)

S: The *State or province* attribute in the list of *Commonly used attributes* (OID: 2.5.4.8)

O: The Organization attribute in the list of Commonly used attributes (OID: 2.5.4.10)

OU: The Organizational unit attribute in the list of Commonly used attributes (OID: 2.5.4.11)

street: Describes the street address (OID: 2.5.4.9)

serialNumber: Specifies the serial number of a device (OID: 2.5.4.5)

title: Describes the position or function of an object (OID: 2.5.4.12)

description: Describes the associated object (OID: 2.5.4.13)

searchGuide: Specifies a search filter (OID: 2.5.4.14)

businessCategory: Describes the kind of business performed by an organization (OID: 2.5.4.15)

postalAddress: Specifies address information required for the physical delivery of postal messages (OID: 2.5.4.16)

postalCode: Specifies the postal code of an object (OID: 2.5.4.17)

postOfficeBox: Specifies the post office box for the physical delivery of mail (OID: 2.5.4.18)

physicalDeliveryOfficeName: Specifies the name of the city or place where a physical delivery office is located (OID: 2.5.4.19)

telephoneNumber: Specifies a telephone number (OID: 2.5.4.20)

telexNumber: Specifies a telex number (OID: 2.5.4.21)

teletexTerminalIdentifier: Specifies an identifier for a telex terminal (OID: 2.5.4.22)

facsimileTelephoneNumber: Specifies the telephone number for a facsimile terminal (OID: 2.5.4.23)

x121Address: Specifies the address used in electronic data exchange (OID: 2.5.4.24)

internationalISDNNumber: Specifies an international ISDN number used in voice, video, and data transmission (OID: 2.5.4.25)

registeredAddress: Specifies the postal address for the delivery of telegrams or expedited documents (OID: 2.5.4.26)

destinationIndicator: Specifies an attribute used in telegram services (OID: 2.5.4.27)

preferredDeliveryMethod: Specifies the preferred delivery method for a message (OID: 2.5.4.28)

presentationAddress: Specifies an OSI presentation layer address (OID: 2.5.4.29)

supportedApplicationContext: Specifies the identifiers for the OSI application contexts in the application layer (OID: 2.5.4.30)

member: Specifies the distinguished name of an object associated with a group or a list (OID: 2.5.4.31)

owner: Specifies the name of an object that has responsibility for another object (OID: 2.5.4.32)

roleOccupant: Specifies the distinguished name of an object that fulfills an organizational role (OID: 2.5.4.33)

seeAlso: Specifies the distinguished name of an object that contains additional information about the same real-world object (OID: 2.5.4.34)

userPassword: Specifies the object's password (OID: 2.5.4.35)

name: Specifies a name that is in the UTF-8 form of the ISO 10646 character set (OID: 2.5.4.41)

givenName: Specifies the given or first name of an object (OID: 2.5.4.42)

initials: Specifies the initials of an object (OID: 2.5.4.43)

generationQualifier: Specifies the generation of an object, which is usually a suffix (OID: 2.5.4.44)

x500UniqueIdentifier: Specifies an identifier that distinguishes between objects when a DN has been reused (OID: 2.5.4.45)

dnQualifier: Specifies information that makes an object unique when information is being merged from multiple sources and objects could have the same RDNs (OID: 2.5.4.46)

enhancedSearchGuide: Specifies a search filter used by X.500 users (OID: 2.5.4.47)

protocolInformation: Specifies information that is used with the presentationAddress attribute (OID: 2.5.4.48)

distinguishedName: Specifies the distinguished name of an object (OID: 2.5.4.49)

uniqueMember: Specifies the distinguished name of an object associated with a group or a list (OID: 2.5.4.50)

houseIdentifier: Identifies a building within a location (OID: 2.5.4.51)

dmdName: Specifies a directory management domain (OID: 2.5.4.54)

E: Specifies an e-mail address.

EM: Specifies an e-mail address.

DC: Specifies the domain name for an object (OID: 0.9.2342.19200300.100.1.25)

uniqueID: Contains an RDN-type name that can be used to create a unique name in the tree (OID: 0.9.2342.19200300.100.1.1)

T: Specifies the name of the tree root object (OID: 2.16.840.1.113719.1.1.4.1.181)

OID: Specifies an object identifier in dot notation.

- 3 To create a certificate, continue with [Step 6 on page 52](#), or to create a signing request, continue with [Step 5 on page 57](#).

3.2.3 Assigning Alternate Subject Names

- 1 Fill in the following fields:

Name Type: Names as specified by RFC 2459. Use the drop-down list to specify a name type, such as:

- ♦ **Directory name:** An X.500 directory name. The required format for the name is `.<attribute name>=<attribute value>`. For example:

`.O=novell.C=US`

Access Manager supports the following attributes:

Country (C)
Organization (O)
Organizational Unit (OU)
State or Province (S or ST)
Locality (L)
Common Name (CN)

- ♦ **IP Address:** An IP address such as 222.123.123.123
- ♦ **URI:** A URI such as www.novell.com.
- ♦ **Registered ID:** An ASN.1 object identifier.
- ♦ **DNS Name:** A domain name such as novell.com.
- ♦ **Email Address (RFC 822 name):** An e-mail address such as ca@novell.com.
- ♦ **X400 Name:** The messaging and e-mail standard specified by the ITU-TS (International Telecommunications Union - Telecommunication Standard Sector). It is an alternative to the more prevalent Simple Mail Transfer Protocol (SMTP) e-mail protocol. X.400 is common in Europe and Canada.
- ♦ **EDI Party:** EDI (Electronic Data Interchange) is a standard format for exchanging business data.
- ♦ **Other:** A user-defined name.

Name: The display alternative name.

2 Continue with [Step 11 on page 54](#).

3.2.4 Generating a Certificate Signing Request

1 In the Administration Console, click *Security > Certificates*, then click *New*.

2 To create a certificate signing request (CSR), select *Use external certificate authority*.

This option generates a CSR for you to send to the CA for signing. A third-party CA is managed by a third party outside of the eDirectory tree. An example of a third party CA is VeriSign. After the signed certificate is received, you need to import the certificate.

3 Specify a Certificate name.

Pick a unique, system-wide name for the certificate that you can easily associate with the certificate's purpose. The name must contain only alphanumeric characters and no spaces.

4 Click the *Edit* button to display a dialog box that lets you add appropriate locality information types for the subject name.

For more information, see [Section 3.2.2, "Editing the Subject Name," on page 54](#).

5 Click *OK*, then fill in the following fields:

Signature algorithm: The algorithm you want to use (SHA-1, MD-2, or MD-5). SHA-1 is currently recommended.

Valid from: The date from which the certificate is valid. For externally signed certificates, the external certificate authority sets the validity period.

Months valid: The number of months that the certificate is valid.

Key size: The size of the key. Select 512, 1024, 2048, or 4096.

- 6 (Conditional) If you are creating a key for a certificate authority, click *Advanced Options*, then configure the following:

This key is for a Certificate Authority: Select this option.

Critical: Enforces the basic constraints you specify. Select one of the following:

- ♦ **Unlimited:** Specifies no restriction on the number of subordinate certificates that the CA can verify.
- ♦ **Do not allow intermediate signing certificates in certificate chain:** Prevents the CA from creating other CAs, but it can create server or user certificates.
- ♦ **Number of allowable intermediate signing certificates in signing chain:** Specifies how many subordinate certificates are allowed in the certificate chain. Values must be 1 or more. Entering 0 creates only entity objects.

- 7 Click *OK*.

- 8 Click the name of the certificate, copy the CSR data and send the information to the external CA.

The certificate status is CSR Pending until you import the signed certificate.

- 9 Click *Close*.

- 10 When you receive the signed certificate and the trusted root (CA chain), continue with [“Importing a Signed Certificate” on page 58](#).

3.2.5 Importing a Signed Certificate

After you receive the signed certificate and the CA chain, you must import it. There are several ways in which the CA can return the certificate. Typically, the CA either returns one or more files each containing one certificate, or returns a file with multiple certificates in it.

- 1 In the Administration Console, click *Security > Certificates*, then click the name of a certificate that is in a CSR Pending state.
- 2 Click *Import Signed Certificate*.
- 3 In the Import Signed Certificate dialog box, browse to locate the certificate data file, or paste the certificate data text into the *Certificate data text* field.
- 4 To import the CA chain, click *Add trusted root*, then locate the certificate data.
- 5 Click *Add intermediate certificate* if you need to continue adding certificates to the chain.
- 6 Click *OK*, then click *Close* on the Certificate Details page.

The certificate is now available for use by Access Manager devices.

If you receive an error when attempting to import the certificate, see [Chapter 7, “Troubleshooting Certificate Issues,” on page 127](#).

3.3 Managing Certificates and Keystores

You can import certificates created by an external certificate authority. These certificates then need to be assigned to a device by adding the certificate to the device’s keystore. The subject name of the certificate needs to match the DNS name of the device, or if you are using wildcard certificates, the main domain name needs to match. You can perform the following certificate tasks:

- ♦ [Section 3.3.1, “Viewing Certificate Details,” on page 59](#)

- ◆ Section 3.3.2, “Adding a Certificate to a Keystore,” on page 61
- ◆ Section 3.3.3, “Renewing a Certificate,” on page 61
- ◆ Section 3.3.4, “Exporting a Private/Public Key Pair,” on page 63
- ◆ Section 3.3.5, “Exporting a Public Certificate,” on page 64
- ◆ Section 3.3.6, “Importing a Private/Public Key Pair,” on page 65
- ◆ Section 3.3.7, “Reviewing the Command Status for Certificates,” on page 65
- ◆ Section 3.3.8, “Keystore Details,” on page 67

3.3.1 Viewing Certificate Details

The Certificate Details page lists the properties of a certificate, such as certificate type, name, subject, and assigned keystores. The fields are not editable.

- 1 In the Administration Console, click *Security > Certificates*.
- 2 Select one of the following:
 - ◆ Click the name of a certificate that is not in a CSR Pending state. The Certificate Details page contains the following information about the certificate:

Field	Description
<i>Issuer</i>	The name of the CA that created the certificate.
<i>Serial number</i>	The serial number of the certificate.
<i>Subject</i>	The subject name of the certificate.
<i>Valid from</i>	The first date and time that the certificate is valid.
<i>Valid to</i>	The date and time that the certificate expires.
<i>Devices</i>	The devices that are configured to hold this certificate on their file system and the keystore that holds them.
<i>Key size</i>	The key size that was used to create the certificate.
<i>Signature algorithm</i>	The signature algorithm that was used to create the certificate.
<i>Finger print (MD5)</i>	The certificate's message digest that was calculated with the MD5 algorithm. It is embedded into the certificate at creation time. It can be used to uniquely identify a certificate. For example, users can verify that a certificate is the one they think it is by matching this published MD5 fingerprint with the MD5 fingerprint on the local certificate.
<i>Finger print (SHA1)</i>	The certificate's message digest that was calculated with the SHA1 algorithm. It is embedded into the certificate at creation time. It can be used to uniquely identify a certificate. For example, users can verify that a certificate is the one they think it is by matching a published SHA1 fingerprint with the SHA1 fingerprint on the local certificate.
<i>Subject Alternate Names: Critical</i>	Indicates whether an application should reject the certificate if the application does not understand the alternate name extensions. Any configured alternate names are displayed in the list.

Field	Description
<i>Key Usage: Critical</i>	Indicates whether an application should reject the certificate if the application does not understand the key usage extensions.
<i>Sign CRLs</i>	Indicates whether the certificate is used to sign CRLs (Certificate Revocation Lists).
<i>Sign certificates</i>	Indicates whether the certificate is used to sign other certificates.
<i>Encrypt other keys</i>	Indicates whether the certificate is used to encrypt keys.
<i>Encrypt data directly</i>	Indicates whether the certificate can encrypted data for private transmission to the key pair owner. Only the intended receiver can read the data.
<i>Create digital signatures</i>	Indicates whether the certificate can create digital signatures.
<i>Non-repudiation</i>	Indicates whether the certificate links a digital signature to the signer and the data. This prevents others from duplicating the signature because no one else has the signer's private key. Additionally, the signer cannot deny having signed the data.
<i>CRL Distribution Points</i>	A list of Certificate Revocation List (CRL) distribution points that are embedded into the certificate as an extension at certificate creation time. Implementations search the CRL from each distribution point (the distribution point is usually a URI that points to a store of revoked certificates) to see whether a certificate has been revoked.
<i>Authority Info Access (OCSP)</i>	A list of Online Certificate Status Protocol (OCSP) responders that are embedded into the certificate as an extension at certificate creation time. Implementations query the OCSP responder to see whether a certificate has been revoked.

- ◆ Click the name of a certification in a CSR Pending state. The following information is displayed:

<i>Subject</i>	The subject name of the certificate.
<i>Valid from</i>	The date and time that the request was generated.
<i>Valid to</i>	The date and time that the request expires.
<i>Devices</i>	No entries. A CSR cannot be assigned to a device.
<i>Key size</i>	The key size that was used to create the request.
<i>Signature algorithm</i>	The signature algorithm that was used to create the request.
<i>State</i>	Displays <code>CSR Pending</code> , indicating that the request has been generated.
<i>CSR data</i>	The certificate signing request data. You can either export this data or copy and paste it into CA's request tool.

- 3 (Conditional) For a certificate not in a CSR Pending state, select one of the following actions:

Renew: Allows you to renew the certificate. For more information, see [Section 3.3.3, "Renewing a Certificate,"](#) on page 61.

Export Private/Public Keypair: Allows you to export private certificates to obtain a backup copy of the key, to move the key to a different server, or to share the key between servers. For more information, see [Section 3.3.4, “Exporting a Private/Public Key Pair,” on page 63](#)

Export Public Certificate: Allows you to export a public key certificate to a file. For more information, see [Section 3.3.5, “Exporting a Public Certificate,” on page 64](#).

Add Certificate to Keystores: Allows you to assign the certificate to keystore so it can be used by Access Manager. For more information, see [Section 3.3.2, “Adding a Certificate to a Keystore,” on page 61](#).

4 (Conditional) For a certificate in a CSR Pending state, select one of the following actions:

Import Signed Certificate: Allows you to import the certificate that was generated for this request. For more information, see [Section 3.2.5, “Importing a Signed Certificate,” on page 58](#).

Export CSR: Allows you to export the CSR to a CSR file.

3.3.2 Adding a Certificate to a Keystore

After importing a certificate, you need to assign the certificate to keystore before it is used by Access Manager.

- 1** In the Administration Console, click *Security > Certificates*.
- 2** Select a certificate.
- 3** Click *Actions > Add Certificate to Keystores*.
- 4** Specify the keystore to which you are adding the certificate. To locate a keystore:
 - 4a** Click the *Select Keystore* button.

For a description of the Access Manager keystores, see [Section 3.1.3, “Access Manager Keystores,” on page 47](#).
 - 4b** On the Keystore Details page, select the keystore, then click *OK*.
- 5** Fill in the following fields:

Alias: Specify the certificate alias.

Overwrite keys with same alias: Select whether to overwrite certificates with the same alias, if the alias you specify is already in use in that keystore.
- 6** Click *OK*.
- 7** Update the device or devices that are using this keystore.

NOTE: For problems related to failures in adding certificates to a keystore and for validating the cross device-existence of the Trusted Root of the certificates present in a particular keystore, see [Section 7.3, “Troubleshooting Options for Certificate Problems,” on page 129](#).

3.3.3 Renewing a Certificate

The Certificate Details page lists the properties of a certificate, such as certificate type, name, subject, and assigned keystores. This page also includes the original CSR when the certificate is still in a pending state (for example, you have generated the CSR, but you have not yet received and imported the signed certificate). If the certificate is expiring, you can cut and paste its text to send it to the CA to get a renewed certificate, then import the newly signed certificate.

For the certificates that Access Manager uses internally, a certificate process is started with Tomcat. This process runs once every 24 hours. It checks all the internal certificates and determines if they are going to expire within 30 days. If they are due to expire, the process automatically regenerates the certificate or trusted root. When a certificate is regenerated, the following message appears:

```
One or more automatically created certificates were regenerated. Reboot the
entire administration console as soon as possible to avoid interruption of
service.
```

This message appears when the administrator logs into the Administration Console, or if the administrator is already logged in, when the administrator switches from one page to another.

This event is also auditing. Another audit event is also generated which tells the administrator to restart any effected services. When the Administration Console certificate and the eDirectory certificates are expiring, a log entry is written to the app_sc log file. The log entry contains the “Recreating auto-generated certificates” string as well as a couple success or failure messages per key re-generated.

Certificates and trusted roots that are manually created with the Access Manager CA or are imported into Administration Console use a different process. The administrator is warned that these certificates are expiring when the administrator logs in to the Administration Console. The following message is displayed:

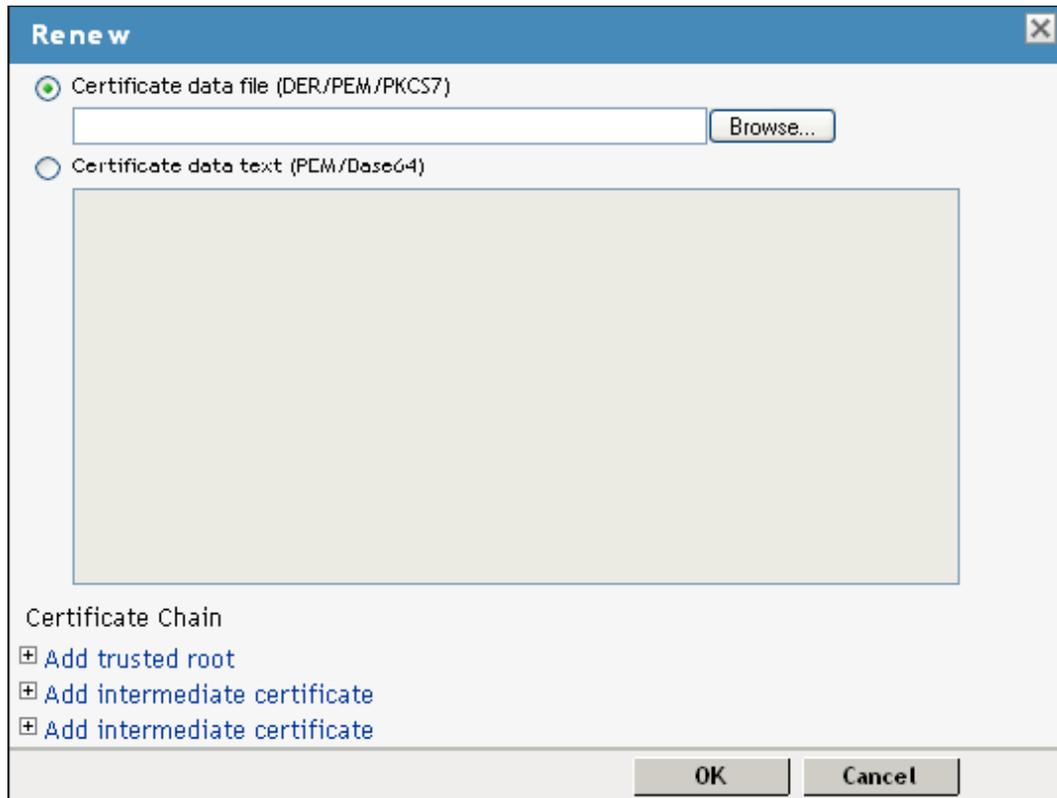
```
Warning: the following certificates are expired or will expire within X days:
<certA>, <certB>.
```

This message is displayed each time the administrator logs into the Administration Console. Events for the expiration of these certificates are not audited and are not logged.

To renew a certificate:

- 1** In the Administration Console, click *Security > Certificates*.
- 2** Click the certificate name.

- 3 Click *Renew*.



- 4 On the Renew page, either browse to locate and select the certificate or select the *Certificate data text (PCM/Base64)* option and paste the certificate data into the text box.
- 5 Click *OK*.
- 6 Update the device using the certificate.

3.3.4 Exporting a Private/Public Key Pair

When you create a certificate, you can specify whether it is exportable. If a key is exportable, it can be extracted and put in a file along with the associated certificate. The file is written in an industry standard format, PKCS#12, which allows it to be transported to other platforms. It is encrypted with a user-specified password to protect the private key. You can export private certificates to obtain a backup copy of the key, to move the key to a different server, or to share the key between servers.

You cannot export a certificate if you enabled the *Do not allow private key to be exportable option* while creating the certificate.

- 1 In the Administration Console, click *Security > Certificates*.
- 2 On the Certificates page, click the certificate.

3 On the Certificate Details page, click *Export Private/Public Keypair*.

Certificate: jwilson1_provo_novell_com

Renew... | Export Private/Public Keypair... | Export Public Certificate ▼ | Add Certificate to Keystores...

Issuer: O=jwilson_tree, OU=Organizational CA
Serial number: 02:1C:11:FF:A4:FF:85:59:5E:48:FF:DB:C3:A3:A5:AC:4F:0D:90:DC:C4:CD:1F:95:21:C3:92:D4:7A:22:02:02:55:FE:5C
Subject: CN=jwilson1.provo.novell.com
Valid from: Monday, April 20, 2009 3:43:06 AM GMT
Valid to: Wednesday, April 20, 2011 3:43:06 AM GMT
Devices: Multiple devices in NIDP Configuration: IDP-as-PR
[SSL Connector](#)
ag18 [Access Gateway]
[Proxy Key Store](#)
137.65.159.18 [Access Gateway]
[ESP Mutual SSL](#)

Key size: 1024
Signature algorithm: RSA with SHA1
Finger print (MD5): 0B:E0:D3:CB:6D:6F:38:25:1E:29:1B:76:CC:AB:A2:5E
Finger print (SHA1): 11:7A:64:D1:5A:84:61:AB:E5:9F:31:F5:64:BD:E8:DF:41:D2:8A:F4



4 Select the format for the key:

PFX/PKCS12: Public Key Cryptography Standards #12 (PKCS#12) format, which is also called PFX format. This format can be used to create JKS or PEM files.

JKS: Java keystore format.

5 Specify the password in the *Encryption/decryption* password field, then click OK.

IMPORTANT: Remember this password because you need it to re-import the key.

6 Click *OK*.

3.3.5 Exporting a Public Certificate

You can export a trusted root or a public key certificate to a file so that a client can use it to verify the certificate chain sent by a cryptography-enabled application, or to have a backup copy of the file.

You can export the certificate in the following formats:

- ♦ DER-encoded (.der) to a file.
- ♦ PEM-encoded to a file. This is a Base64-encoded DER certificate that is enclosed between the BEGIN CERTIFICATE and END CERTIFICATE tags.
- ♦ PEM CUT/Paste Buffer. This displays the certificate data so you can copy it to the system Clipboard. You can then pasted it directly into a cryptography-enabled application.

To export the public certificate:

- 1 In the Administration Console, click *Security > Certificates*.
- 2 Click the certificate name.
- 3 On the Certificate Details page, click *Export Public Certificate*, then click the file type.
- 4 Save the output file to the location of your choosing.

3.3.6 Importing a Private/Public Key Pair

If you created a key pair that was exported from another certificate management system, you can import the key pair and then assign it to an Access Manager device. The file needs to be in PFX/PKCS12 (*.pfx or *.p12) format.

- 1 In the Administration Console, click *Security > Certificates*.
- 2 Choose *Actions > Import Private/Public Keypair*.
- 3 Fill in the following fields:

Certificate name: The name of the certificate. This is a system-wide, unique name used by Access Manager. The name must contain only alphanumeric characters and no spaces. If the name starts with a number, an underline (_) prefix is added to the name so that the name conforms to XML requirements. If the name contains invalid characters, it is automatically renamed.

Keystore password: Type the encryption/decryption password established when exporting the certificate.

Certificate data file (PFX/PKCS12): The certificate file to import. You can browse to locate the *.pfx or *.p12 file.

Certificate data file (JKS): To locate a JKS file, select this option, then click the *Browse* button.

- 4 Click *OK*.

If you receive an error when importing the certificate, the error comes from either NCI or PKI. For a description of these error codes, see [Novell Certificate Server Error Codes and Novell International Cryptographic Infrastructure \(http://www.novell.com/documentation/nwec/index.html\)](#). For general certificate import issues, see [Section 7.1.1, “Importing an External Certificate Key Pair,” on page 127](#).

- 5 Continue with [“Adding a Certificate to a Keystore” on page 61](#).

3.3.7 Reviewing the Command Status for Certificates

You can view the status of the commands that have been sent to the certificate server for execution.

- 1 In the Administration Console, click *Security > Certificates*, then click *Command Status*.
- 2 Use the following options to review or change a server’s certificate command status:
 - ♦ **Delete:** To delete a command, select the check box for the command, then click *Delete*. The selected command is cleared.
 - ♦ **Refresh:** Click *Refresh* to update the current cache of recently executed commands.
 - ♦ **Name:** Click this box to select all the commands in the list, then click *Refresh* or *Delete*.

The following table describes the features on this page:

Column Name	Description
<i>Name</i>	Contains the display name of the command. Click the link to view additional details about the command.
<i>Status</i>	Specifies the status of the command. Some of the possible states of the command include Pending, Incomplete, Executing, and Succeeded.

Column Name	Description
<i>Type</i>	Specifies the type of server, such as Identity Server or Access Gateway.
<i>Commands</i>	Specifies the command given, such as <code>Import certificate</code> , or <code>Import trusted root</code> .
<i>Admin</i>	Specifies if the system or a user issued the command. If a user issued the command, the DN of the user is displayed.
<i>Date & Time</i>	Specifies the local date and time the command was issued.

- 3 To review command information, click a link under the *Name* column.

Server Details Edit: Server Scheduled Command

Note: Date and time entries are specified in local time.

Command Information

Refresh | Delete

Name:	Import trusted root with name (configCA) to trust store (Proxy Trust Store) on (151.155.1
Type:	Import trusted root
Admin:	cn=admin,o=novell
Status:	Succeeded
Last Executed On:	Jun 4, 2007 8:22 AM

Command Execution Details

Command	Command Result
CertTRImport	Success

Close

This page displays status information about the command and allows you to perform the following tasks:

Refresh: Select this option to refresh the data for this command.

Delete: Select this option to clear this command.

The following command information is listed:

Name: Specifies the display name that has been given to the command.

Type: Specifies the type of command.

Admin: Specifies whether the system or a user issued the command. If a user issued the command, the field contains the DN of the user.

Status: Specifies the status of the command, and includes such states as *Pending*, *Incomplete*, *Executing*, and *Succeeded*.

Last Executed On: Specifies when the command was issued. The date and time are displayed in local time. If the command failed, additional information is available.

For a command that the Administration Console can successfully process, the page displays a *Command Execution Details* section with the name of the command and the command results.

- 4 Click *Close*.

3.3.8 Keystore Details

The Keystore Details page allows you to view associated cluster member keystores and to replace certificates associated with the keystore.

Not all keystores are associated with a cluster configuration. Those that are (for example, the Signing and Encryption keystores) display the following information:

Column	Description
Keystore Name	The name of the keystore.
Type	The type of keystore, such as Java or PKCS12.
Device or Cluster Name	The name of the device or of the cluster that is using the keystore.

Some keystores require a single certificate, so you can only replace the certificate. Other keystores can contain multiple certificates. In this type of keystore, you can add and remove certificates.

To view a keystore:

- 1 In the Administration Console, click *Security > Certificates*.
- 2 Click the down-arrow in the *Devices* column, then select a keystore.
- 3 To remove a certificate, select the certificate, then click *Remove*.
This option is not available for all keystores.
- 4 To add or replace a certificate:
 - 4a Click either *Add* or *Replace*.
 - 4b Fill in the following fields:
 - Certificate:** Specifies the certificate you want to add. You can browse to locate the certificate. When you browse, the system displays the Select Certificate page. Select the certificate, then click *OK*.
 - Alias(es):** Specifies the certificate alias. This name is displayed among the list of certificates assigned to the keystore.
 - Overwrite keys with the same alias:** (If available) Select if you want only one certificate with the specified alias in the keystore.
 - 4c Click *OK*.
- 5 Click *Close*.

3.4 Managing Trusted Roots and Trust Stores

A certificate from a certificate authority (CA) is commonly referred to as trusted root. A trusted root is a trusted certificate, or the certificate of a known CA. These certificates are self-signed and are recognized as representing a CA that is trusted. In order to validate a digital signature, you must trust at least one of the certificates in the user or server's certificate chain. You can directly trust the certificate of the user or server, or you can choose to trust any other certificate in the chain. Typically, the certificate that is trusted is the root CA's certificate.

When an external certificate authority creates certificates, you need to import the trusted root of the certificate authority and assign the trusted root to the trust store of the device that needs to trust the certificate.

1 In the Administration Console, click *Security > Trusted Roots*.

2 Select from the following actions:

Import: Allows you to import trusted roots so that Access Manager devices can trust the certificate sent by other computers at runtime. For more information, see [Section 3.4.1, “Importing Public Key Certificates \(Trusted Roots\),” on page 68](#).

Delete: To delete a trusted root, select the trusted root, then click *Delete*.

Add Trusted Roots to Trust Stores: Allows you to assign a trusted root to a device so it can be used by that device. For more information, see [Section 3.4.2, “Adding Trusted Roots to Trust Stores,” on page 68](#).

Auto Import From Server: To import a trusted root from another server, click *Auto Import From Server*. For more information, see [Section 3.4.3, “Auto-Importing Certificates from Servers,” on page 69](#).

View Trusted Root Details: To view information about a trusted root, click the name of a trusted root. For more information, see [Section 3.4.6, “Viewing Trusted Root Details,” on page 70](#).

3.4.1 Importing Public Key Certificates (Trusted Roots)

You import trusted roots so that the specific device can trust the certificate sent by other computers at runtime. After you import a trusted root, you can assign it to the proper trust store associated with a device, which allows the device to trust certificates signed by the trusted root.

1 In the Administration Console, click *Security > Trusted Roots*.

2 Click *Import*, then specify a name for the certificate.

This is a system-wide, unique name used by Access Manager.

3 Select one of the following methods for importing the public key:

- **Certificate data file (DER/PEM/PKCS7):** Select this method to browse to a file. Click *Browse* to locate the file on your file system.
- **Certificate data text (PEM/Base64):** Select this method to paste Base64-encoded certificate data text.

4 Click *OK*.

5 Continue with [“Adding Trusted Roots to Trust Stores” on page 68](#)

3.4.2 Adding Trusted Roots to Trust Stores

After importing a trusted root, you need to assign it to a device before it is used by Access Manager.

To add a trusted root to an existing trust store:

1 In the Administration Console, click *Security > Trusted Roots*.

2 Select the trusted root, then click *Add Trusted Roots to Trust Stores*.

3 Fill in the following fields:

Trusted roots: Select the trusted root store. To locate the trusted root store, click the *Select Keystore* icon. When you browse, the system displays the Select Trusted Roots page. Select the trusted root store, then click *OK*.

Alias(es): Specify an alias for the trusted root.

- 4 Click *OK*.
- 5 Update the device that is using this trust store.

3.4.3 Auto-Importing Certificates from Servers

You can import certificates from other servers (such as an LDAP server, an identity provider, or service provider) and make them available for use in Access Manager. You must provide the IP address, port, and certificate name.

- 1 In the Administration Console, click *Security > Trusted Roots > Auto-Import from Server*.
- 2 Fill in the following fields:
 - Server IP Address:** Specify the server IP address. You can use a DNS name.
 - Server Port:** Specify the server port.
 - Certificate Name:** Specify a unique name of the certificate to store in Access Manager.
- 3 Click *OK*.

3.4.4 Exporting the Public Certificate of a Trusted Root

You can export a trusted root or a public key certificate to a file so that a client can use it to verify the certificate chain sent by a cryptography-enabled application, or to have a backup copy of the file.

You can export the certificate in the following formats:

- ♦ DER-encoded (.der) to a file.
- ♦ PEM-encoded to a file. This is a Base64-encoded DER certificate that is enclosed between BEGIN CERTIFICATE and END CERTIFICATE tags.
- ♦ PEM CUT/Paste Buffer. This displays the certificate data so you can copy it to the system Clipboard. You can then pasted it directly into a cryptography-enabled application.

To export the public certificate:

- 1 In the Administration Console, click *Security > Trusted Roots*.
- 2 Click the name of the trusted root.
- 3 On the Certificate Details page, click *Export Public Certificate*, then click the file type.
- 4 Save the output file to the location of your choosing.

3.4.5 Viewing Trust Store Details

- 1 In the Administration Console, click *Security > Trusted Roots*.
- 2 Under the *Devices* column, click the name of a trust store.

3 View the following information:

Field	Description
Trust store name	The name of the selected trust store.
Trust store type	The type of trust store, such as Java, PEM, or DER.
Cluster of Device name	The name of the cluster using this trust store or the single device that is using the trust store.
Cluster Members' Trust Stores	The trust stores assigned to a cluster. If a device does not belong to a cluster, this section does not appear.
Trusted Roots	The trusted roots that are stored in the trust store.

4 Click *Close*.

3.4.6 Viewing Trusted Root Details

1 In the Administration Console, click *Security > Trusted Roots*.

2 Click the name of a trusted root.

3 View the following information:

Field	Description
<i>Issuer</i>	The name of the CA that created the certificate.
<i>Serial number</i>	The serial number of the certificate.
<i>Subject</i>	The subject name of the certificate.
<i>Valid from</i>	The first date and time that the certificate is valid.
<i>Valid to</i>	The date and time that the certificate expires.
<i>Devices</i>	The devices that are configured to hold this certificate on their file system.
<i>Key size</i>	The key size that was used to create the certificate.
<i>Signature algorithm</i>	The signature algorithm that was used to create the certificate.
<i>Finger print (MD5)</i>	The certificate's message digest that was calculated with the MD5 algorithm. It is embedded into the certificate at creation time. It can be used to uniquely identify a certificate. For example, users can verify that a certificate is the one they think it is by matching this published MD5 fingerprint with the MD5 fingerprint on the local certificate.
<i>Finger print (SHA1)</i>	The certificate's message digest that was calculated with the SHA1 algorithm. It is embedded into the certificate at creation time. It can be used to uniquely identify a certificate. For example, users can verify that a certificate is the one they think it is by matching a published SHA1 fingerprint with the SHA1 fingerprint on the local certificate.

The *Subject Alternate Names* section indicates whether an application should reject the certificate if the application does not understand the alternate name extensions. Any configured alternate names are displayed in the list.

The *Key Usage* section indicates whether an application should reject the certificate if the application does not understand the key usage extensions. The following are possible:

Sign CRLs: Indicates whether the certificate is used to sign CRLs (Certificate Revocation Lists).

Sign certificates: Indicates that the certificate is used to sign other certificates.

Encrypt other keys: Indicates that the certificate is used to encrypt keys.

Encrypt data directly: Indicates that the certificate encrypts data for private transmission to the key pair owner. Only the intended receiver can read the data.

Create digital signatures: Indicates that the certificate is used to create digital signatures.

Non-repudiation: Indicates that the certificate links a digital signature to the signer and the data. This prevents others from duplicating the signature because no one else has the signer's private key. Additionally, the signer cannot deny having signed the data.

CRL Distribution Points: Displays a list of Certificate Revocation List (CRL) distribution points that are embedded into the certificate as an extension at certificate creation time. Implementations search the CRL from each distribution point (the distribution point is usually a URI that points to a store of revoked certificates) to see whether a certificate has been revoked.

Authority Info Access (OCSP): Displays a list of Online Certificate Status Protocol (OCSP) responders that are embedded into the certificate as an extension at certificate creation time. Implementations query the OCSP responder to see whether a certificate has been revoked.

4 Select from the following actions:

Export Public Certificate: Allows you to export a trusted root to a file so that a client can use it to verify the certificate chain sent by a cryptography-enabled application. For more information, see [Section 3.3.5, "Exporting a Public Certificate,"](#) on page 64.

Add Trusted Root to Trust Stores: Allows you to assign a trusted root to a device so it can be used by that device. For more information, see [Section 3.4.2, "Adding Trusted Roots to Trust Stores,"](#) on page 68

5 Click *Close*.

3.5 Security Considerations for Certificates

Your security deployment plan should contain policies for the following:

- ♦ **Key size for certificates:** The Access Manager product ships with a CA that can create certificates with a key size of 512, 1024, 2048 or 4096. Select the maximum size supported by the applications that you are protecting with Access Manager.
- ♦ **Certificate renewal dates:** We recommend that certificates should be renewed every two years. Your security needs might allow for a longer or shorter period.
- ♦ **Trusted certificate authorities:** The Access Manager ships with a CA, and during installation of the various components, it creates and distributes certificates. For added security, you might want to replace these certificates with certificates from a well-known CA.

3.6 Assigning Certificates to Access Manager Devices

After you assign certificates to devices, the certificates are placed in keystores. Ensure that you update the device so that the certificates are pushed into active use.

This section discusses how you update, renew, and assign certificates to Access Manager devices.

- ◆ [Section 3.6.1, “Importing a Trusted Root to the LDAP User Store,” on page 72](#)
- ◆ [Section 3.6.2, “Managing Identity Server Certificates,” on page 73](#)
- ◆ [Section 3.6.3, “Assigning Certificates to an Access Gateway,” on page 75](#)
- ◆ [Section 3.6.4, “Assigning Certificates to J2EE Agents,” on page 75](#)
- ◆ [Section 3.6.5, “Configuring SSL for Authentication between the Identity Server and Access Manager Components,” on page 76](#)
- ◆ [Section 3.6.6, “Changing a Non-Secure \(HTTP\) Environment to a Secure \(HTTPS\) Environment,” on page 76](#)
- ◆ [Section 3.6.7, “Creating Keystores and Trust Stores,” on page 77](#)

3.6.1 Importing a Trusted Root to the LDAP User Store

When you specify the settings of a user store for an Identity Server configuration, or add a user store, you can import the trusted root certificate to the LDAP user store device.

- 1 In the Administration Console, click *Devices > Identity Servers > Edit > Local > [User Store]*.
- 2 Under *Server Replicas*, click the name of the server replica.

Identity Servers ▶ IDP_A ▶

Installed User Store

Name:	<input type="text" value="Installed User Store"/>
Admin name:	<input type="text" value="cn=admin,o=novell"/> (Ex: cn=admin,o=novell)
Admin password:	<input type="password" value="•••••"/>
Confirm password:	<input type="password" value="•••••"/>
Directory type:	

Server replicas

New | Delete

- Name
- [Installed User Store Replica](#)

Search Contexts

Specify server replica information [X]

Name:	<input type="text" value="Installed User Store Replica"/>
IP Address:	<input type="text" value="151.155.167.52"/> : <input type="text" value="389"/>
	<input type="checkbox"/> Use secure LDAP connections
	Auto import trusted root
Connection limit:	<input type="text" value="20"/> [▲] [▼]
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

- 3 Enable the *Use secure LDAP connections* option.

This option allows SSL communication to occur between the Identity Server and the user store.

- 4 Click *Auto import trusted root*.
- 5 Click *OK* to confirm the import.

Ensure that you have pop-ups enabled, or the browser cannot display the Confirm dialog box.

Select Certificate to Trust

Alias:

Server Certificate

Subject: CN=jwilson.provo.novell.com, O=JWILSON_TREE
 Issuer: O=jwilson_tree, OU=Organizational CA
 Valid starting date: Apr 6, 2009
 Valid ending date: Apr 6, 2011
 Signature algorithm: SHA1withRSA
 Finger print (MD5): 37:0E:0F:89:48:42:CE:D9:50:19:E3:15:4B:BB:64:F5
 Finger print (SHA1): 29:D3:B3:27:D9:4E:D8:CF:47:77:7F:07:36:3A:6E:19:12:2C:51:6D

Root CA Certificate

Subject: O=jwilson_tree, OU=Organizational CA
 Issuer: O=jwilson_tree, OU=Organizational CA
 Valid starting date: Apr 6, 2009
 Valid ending date: Feb 3, 2036
 Signature algorithm: SHA1withRSA
 Finger print (MD5): 65:97:32:84:D0:04:4B:AE:2C:C1:79:01:47:63:42:03
 Finger print (SHA1): F4:26:E5:62:A0:14:FD:29:5E:A5:3D:40:25:C3:1F:2E:7F:AD:70:5B

- 6 Select one of the certificates in the list.
 You are prompted to choose either a server certificate or a root CA certificate. To trust one certificate, choose *Server Certificate*. Choose *Root CA Certificate* to trust any certificate signed by that certificate authority.
- 7 Specify an alias, then click *OK*.
 You use the alias to identify the certificate in Access Manager.
- 8 On the User Store page, click *OK*.
- 9 Restart the Identity Server.

3.6.2 Managing Identity Server Certificates

The Identity Server stores certificates in keystores and trust stores. Keystores can hold only one certificate; trust stores can hold multiple trusted roots. After you install the Identity Server, you should replace the default certificates in the keystore. You should create an SSL certificate for the Identity Server and use it to replace the predefined test-connector certificate that comes with Access Manager. You can also replace the test-provider and test-consumer certificates in the *Provider Introductions SSL Connector* and *Consumer Introductions SSL Connector* keystores. The steps for replacing the signing, encryption, provider, and consumer certificates are similar.

You can also add trusted roots to the Trust Store used by the Identity Server, delete imported trusted roots, or auto-import them from a server. The Trust Store is the certificate container for CA certificates that the Identity Server has been configured to trust. It needs to contain the trusted root for the identity providers, service providers, and embedded service providers that it has been configured to trust.

You can also access the OCSP trust store to add OCSP server certificates. Online Certificate Status Protocol is a method used for checking the revocation status of a certificate. For this feature, you must set up an OCSP server. The Identity Server sends an OCSP request to the OCSP server to determine if a certain certificate has been revoked. The OCSP server replies with the revocation status. If this revocation checking protocol is used, the Identity Server does not cache or store the information in the reply, but sends a request every time it needs to check the revocation status of a certificate. The OCSP reply is signed by the OCSP server. To verify that it was signed by the correct OCSP server, the OCSP server certificate needs to be added to this trust store. The OCSP server certificate itself is added to the trust store, not the CA certificate

1 In the Administration Console, click *Devices > Identity Servers > Edit > Security*.

2 To replace a certificate in a keystore:

2a Click the keystore link that contains the certificate you want to replace:

Encryption: Displays the encryption certificate keystore. The encryption certificate is used to encrypt specific fields or data in the assertions.

Signing: Displays the signing certificate keystore. Click this option to access the keystore and replace the signing certificate as necessary. The signing certificate is used to sign the assertion or specific parts of the assertion.

SSL: Displays the SSL connector keystore. Click this option to access the keystore and replace the SSL certificate as necessary. This certificate is used for SSL connections.

Provider: Displays the identity provider keystore. Click this option to access the keystore and replace the identity provider certificate.

Consumer: Displays the identity consumer keystore. Click this option to access the keystore and replace the identity consumer certificate as necessary.

2b Click *Replace*.

A keystore stores only one certificate at a time. When you replace a certificate, you overwrite the existing one.

2c In the Replace dialog box, click the *Select Certificate* icon and browse to select the certificate you created in [Section 3.2, “Creating Certificates,” on page 49](#).

2d Click *OK*.

2e Click *OK* in the Replace dialog box.

2f Restart Tomcat, as prompted by the system.

The system restarts Tomcat for you if you click *Restart Now* at the prompt. If you want to restart at your convenience, select *Restart Later* and then manually restart Tomcat.

Linux: Enter the following command:

```
/etc/init.d/novell-tomcat5 restart
```

Windows: Enter the following commands:

```
net stop Tomcat5
```

```
net start Tomcat5
```

- 3 To modify the trusted roots in the Trust Store:
 - 3a Click the name of the trust store that you want to manage.

NIDP Trust Store: Contains the trusted root certificates of all the providers that the Identity Server trusts.

OCSP Trust Store: Contains the certificates of the OCSP servers that the Identity Server trusts.
 - 3b To add a trusted root that you have saved in a file, click *Add*.
 - 3c To remove a trusted root, select the trusted root, then click *Delete*.
 - 3d To download the trusted root from the server, click *Auto-Import From Server*, specify the DNS or IP address of the server, enter the port, then click *OK*.
 - 3e Select the certificate to add, specify an alias, then click *OK*.
 - 3f Update the Identity Server configuration on the Servers page, as prompted.

3.6.3 Assigning Certificates to an Access Gateway

The Access Gateway can be configured to use certificates for SSL communication with three types of entities:

- ♦ **Identity Server:** The Access Gateway uses the Embedded Service Provider to communicate with the Identity Server. The Access Manager CA automatically generates the required certificates for secure communication when you set up a trusted relationship with the Identity Server. To manage these certificates in the Administration Console, click *Access Gateways > [Configuration Link] > Service Provider Certificates*. For more information, see “[Managing Embedded Service Provider Certificates](#)” in the *Novell Access Manager 3.1 SP2 Access Gateway Guide*.
- ♦ **Client browsers:** You can enable SSL communication between the client browsers and the Access Gateway. When setting up this feature, you can either have the Access Manager CA automatically generate a certificate key or you can select a certificate key you have already imported (or created) for the reverse proxy. To manage this certificate in the administration console, click *Access Gateways > [Configuration Link] > [Name of Reverse Proxy]*. For more information, see “[Managing Reverse Proxies and Authentication](#)” in the *Novell Access Manager 3.1 SP2 Access Gateway Guide*.
- ♦ **Protected Web servers:** You can enable SSL communication between the Access Gateway and the Web servers it is protecting. This option is only available if you have enabled SSL communication between the browsers and the Access Gateway. You can enable SSL or mutual SSL. To manage these certificates in the Administration Console, click *Access Gateways > [Configuration Link] > [Name of Reverse Proxy] > [Name of Proxy Service] > Web Servers*. For more information, see “[Configuring the Web Servers of a Proxy Service](#)” in the *Novell Access Manager 3.1 SP2 Access Gateway Guide*.

3.6.4 Assigning Certificates to J2EE Agents

To enable the J2EE agent for SSL, you must set up the following trust relationships:

- ♦ The J2EE server with the Identity Server
- ♦ The J2EE agent with the Identity Server

For instructions on setting up these certificates, see “[Configuring SSL Certificate Trust](#)” in the *Novell Access Manager 3.1 SP2 J2EE Agent Guide*.

3.6.5 Configuring SSL for Authentication between the Identity Server and Access Manager Components

By default, all Access Manager components (Identity Server, Access Gateway, SSL VPN, and J2EE agents) trust the certificates signed by the local CA. However, if the Identity Server is configured to use an SSL certificate signed externally, the trusted store of the service provider for each component must be configured to trust this new CA. Import the public certificate of the CA into the following trust stores:

- ♦ For an Access Gateway, click *Devices > Access Gateways > Edit > Service Provider Certificates > Trusted Roots*.
- ♦ For a J2EE agent, click *Devices > J2EE Agents > Edit > Trusted Roots*.
- ♦ For an SSL VPN server, click *Devices > SSL VPNs > Edit > SSL VPN Certificates > Trusted Root*.

If an Access Gateway, a J2EE agent, or an SSL VPN server is configured to use an SSL certificate signed externally, the trusted store of the Identity Server must be configured to trust this new CA. Import the public certificate of the CA into the Identity Server configuration that the component is using for authentication.

In the Administration Console, click *Devices > Identity Servers > Edit > Security > NIDP Trust Store* and add the certificate to the Trusted Roots list.

NOTE: Whenever you replace certificates on a device, you must update the Identity Server configuration (by clicking *Update Servers* on the Servers page), or restart the Embedded Service Provider.

3.6.6 Changing a Non-Secure (HTTP) Environment to a Secure (HTTPS) Environment

If you are running in a non-secure staging environment, and you’re ready to move to production, you must perform the following steps to enable security.

- 1 Change the Identity Server configuration protocol to HTTPS. (See “[Configuring Secure Communication on the Identity Server](#)” in the *Novell Access Manager 3.1 SP2 Setup Guide*.)
- 2 Replace the test certificates with your own. (See “[Using Access Manager Certificates](#)” or “[Using Externally Signed Certificates](#)” in the *Novell Access Manager 3.1 SP2 Setup Guide*.)
- 3 Update all devices that are trusting this Identity Server configuration.
This causes the Embedded Service Provider to reimport the metadata of the Identity Server.
- 4 (Conditional) If you have set up federation, reimport metadata for trusted service and identity providers. (See “[Managing Metadata](#)” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.)
- 5 Change the Access Gateway configuration to HTTPS. (See “[Configuring the Access Gateway for SSL](#)” in the *Novell Access Manager 3.1 SP2 Setup Guide*.)

3.6.7 Creating Keystores and Trust Stores

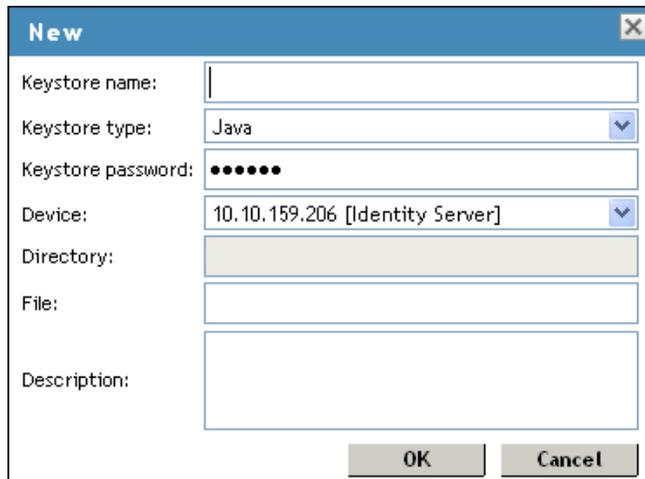
A keystore is storage file containing keys, certificates, and trusted roots. Access Manager agents can access them to retrieve certificates, keys, and trusted roots as needed. A trust store is a keystore containing only trusted roots. Intermediate CAs and end entity public certificates can be part of a trust store.

Access Manager comes with predefined stores for certificate management. However, in certain situations you might need to create a keystore or trust store. For example, if you are using JBoss keystore certificates that you need to import into Access Manager, you must create a keystore and assign it to the JBoss agent. It is probable that the keystore already exists on the JBoss file system, as created and configured by JBoss. Creating it again through Access Manager does not delete the existing keystore. This does allow Access Manager to recognize the existing keystore and add or remove the certificates. Access Manager cannot manage certificates that were created before the keystore is created in Access Manager.

The easiest way to create a keystore is to do so when you are adding the certificate to the keystore. If you want to create a trust store, the steps are identical, except you select trusted roots from the Trusted Roots page, rather than the certificates from the Certificates page.

A keystore stores only one certificate at a time. When you replace a certificate, you overwrite the existing one.

- 1 In the Administration Console, click *Security > Certificates*.
- 2 Import the certificate, if you have not done so already. See [“Importing a Private/Public Key Pair” on page 65](#).
- 3 Click the certificate name.
- 4 In the Certificate Details page, click *Add Certificate to Keystores*.
- 5 On the Add Certificate to Keystores dialog box, click the *Select Keystore* button to browse for key stores.
- 6 On the Keystore page, click *New*.



- 7 Fill in the following fields:

Keystore name: Specifies the name of the keystore. This maps to a name that the server communication recognizes to identify the keystore on the device.

Keystore type: Specifies whether to use Java, PEM, or PKCS12.

Keystore password: Specifies the password to revise the keystore settings.

Device: Specifies the device (by IP) to which you assign the keystore. The device can be an Identity Server or SSL VPN. You cannot assign one keystore to multiple devices.

Directory: Specifies the directory where PKCS12 or PEM files are stored.

For example, `/var/opt/novell/keystores/`.

File: Specifies the path and filename of the Java keystore (JKS).

For example, `/var/opt/novell/keystores/myKeystore.keystore`.

Description: Describes the keystore.

8 Click *OK*.

This creates the keystore.

9 (Optional) On the Keystore page, assign a certificate to the new keystore by selecting the store's check box.

10 Click *OK* in the *Add Certificate to Keystores* dialog box.

Access Manager Logging

4

- ♦ [Section 4.1, “Understanding the Types of Logging,” on page 79](#)
- ♦ [Section 4.2, “Downloading the Log Files,” on page 80](#)
- ♦ [Section 4.3, “Using the Log Files for Troubleshooting,” on page 86](#)

4.1 Understanding the Types of Logging

Access Manager supports two types of logging:

- ♦ [Section 4.1.1, “Component Logging for Troubleshooting Configuration or Network Problems,” on page 79](#)
- ♦ [Section 4.1.2, “HTTP Transaction Logging for Proxy Services,” on page 80](#)

4.1.1 Component Logging for Troubleshooting Configuration or Network Problems

Each Access Manager component maintains log files that contain entries documenting the operation of the component. Component file logging records the processing and interactions between the Access Manager components that occur while satisfying user and administrative requests and during general system processing. By enabling the correct levels of logging for the various Access Manager components, an administrator can monitor how the Access Manager processes user and administrative requests. Transaction flows have been defined to help the administrator identify the processing steps that occur during the execution of specific types of user or administrative requests. All component file logs include tags and values that allow the administrator to identify and correlate which component file log entries pertain to a given transaction and user.

Component file logs are not primarily intended for debugging the software itself, although they can be used to detect software that is not behaving properly. Rather, the intent of component file logging is to document the operational processing of the Access Manager components so that system administrators and support personnel can identify and isolate problems caused by configuration errors, invalid user data, or network problems such as broken connections. However, component file logging is typically the first step in identifying software bugs.

Component file logging is more verbose than audit logging. It increases processing load, and on a day-to-day basis, it should be enabled only to log error conditions and system warnings. If a specific problem occurs, component file logging can be set to *info* or *config* to gather the information needed to isolate and repair the detected problem. When the problem is resolved, component file logging should be reconfigured to log only error conditions and system warnings.

Log files can be configured to include entries for the following events:

- ♦ Initialization and shutdown
- ♦ Configuration
- ♦ Events processed by the component, such as authentication, role assignment, resource access, and policy evaluation
- ♦ Error conditions

See “[Configuring Component Logging](#)” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.

4.1.2 HTTP Transaction Logging for Proxy Services

The Access Gateway allows you to log HTTP transactions. You can log what happens with an HTTP request and response during certain times:

- ◆ Between the browser and the Access Gateway
- ◆ Between the Access Gateway and the back-end Web server

You select fields from the HTTP header of a request and these fields are logged. You can then use these logged transactions to bill customers for Web services or to troubleshoot whether a request is refused because the browser didn’t send the required information or because the Access Gateway didn’t send the Web server the required information.

This type of logging conforms to the W3C specification for proxy server logging in the common and extended log formats. This type of logging provides no information about the exchanges between the Access Gateway and the Identity Server. If you need to discover whether the Access Gateway is obtaining the correct information from the Identity Server for an Identity Injection or Form Fill policy, you need to turn on Component logging. See “[Configuring Component Logging](#)” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.

For HTTP transaction logging, see “[Configuring Logging for a Proxy Service](#)” in the *Novell Access Manager 3.1 SP2 Access Gateway Guide*.

4.2 Downloading the Log Files

The *General Logging* page displays the location of the files that the Access Manager components use for logging system messages. There are some exceptions:

- ◆ **J2EE Agent:** The J2EE Agent uses the J2EE global logger, and the location of this file is customizable. For information about J2EE agent log files, see “[Viewing Log Files](#)” in the *Novell Access Manager 3.1 SP2 J2EE Agent Guide*.
- ◆ **Default Auditing File:** If you have configured Novell Audit to send events to the default audit file (on Linux, this is `/var/opt/novell/naudit/logs/auditlog`), this file does not appear in the list. (On a Windows machine that has different security restraints, the file appears in the list.)

If you want this file to appear in this list on a Linux machine, you must make this file readable by the `novlwww` user. It is a breach of Novell Audit security for Access Manager code to change the permissions on this file. You must decide whether changing its permissions and displaying the file in this list compromises your security.

To have it appear in the list of files for the Administration Console, configure the following:

- ◆ Use commands similar to the following to grant the `novlwww` user executable permissions to the `naudit` directories:

```
chmod o+x /var/opt/novell/naudit
chmod o+x /var/opt/novell/naudit/logs
```

- ◆ Use a command similar to the following to grant the `novlwww` user read access to the `auditlog` file:

```
chmod o+r /var/opt/novell/naudit/logs/auditlog
```

- ◆ **Proxy Service Log Files:** If you enable proxy service logging, these files are not available for downloading from this page because there could be potentially hundreds of these files. If this type of logging has been enabled, the directory where they are located is displayed. For more information about this type of logging, see “[Configuring Logging for a Proxy Service](#)” in the *Novell Access Manager 3.1 SP2 Access Gateway Guide*.

To view or download the log file:

- 1 In the Administration Console, click *Auditing > General Logging*.
- 2 Select one or more log files, click *Download*, then open it or save it to disk.

You can use any text editor to view the file.

Each Access Manager component generates multiple log files. The following tables lists these files and the types of messages they contain.

- ◆ [Section 4.2.1, “Linux Administration Console Logs,” on page 81](#)
- ◆ [Section 4.2.2, “Windows Server 2003 Administration Console Logs,” on page 82](#)
- ◆ [Section 4.2.3, “Windows Server 2008 Administration Console Logs,” on page 82](#)
- ◆ [Section 4.2.4, “Linux Identity Server Logs,” on page 83](#)
- ◆ [Section 4.2.5, “Windows Server 2003 Identity Server Logs,” on page 83](#)
- ◆ [Section 4.2.6, “Windows Server 2008 Identity Server Logs,” on page 83](#)
- ◆ [Section 4.2.7, “Linux Access Gateway Appliance Logs,” on page 84](#)
- ◆ [Section 4.2.8, “Linux Access Gateway Service Logs,” on page 85](#)
- ◆ [Section 4.2.9, “Windows Access Gateway Service Logs,” on page 85](#)
- ◆ [Section 4.2.10, “SSL VPN Server Logs,” on page 86](#)

4.2.1 Linux Administration Console Logs

Filename	Description
/var/opt/novell/tomcat5/logs/catalina.out	Contains Tomcat errors.
/opt/novell/devman/share/logs/app_sc.0.log	Contains events related to importing devices, device configuration changes, health status changes, statistics reporting, and communication problems.
/opt/novell/devman/share/logs/app_cc.0.log	Contains events related to policy configuration.
/opt/novell/devman/share/logs/platform.0.log	Contains XML events for configuration changes. This log file contains very little useful information for system administrators.

4.2.2 Windows Server 2003 Administration Console Logs

Filename	Description
\Program Files\Novell\Tomcat\logs\stderr.log	Contains Tomcat error messages directed to stderr. This file is reset whenever Tomcat is restarted.
\Program Files\Novell\Tomcat\logs\stdout.log	Contains Tomcat error messages directed to stdout. This file is reset whenever Tomcat is restarted.
\Program Files\Novell\logs\app_sc.0.log	Contains events related to importing devices, device configuration changes, health status changes, statistics reporting, and communication problems.
\Program Files\Novell\logs\app_cc.0.log	Contains events related to policy configuration.
\Program Files\Novell\logs\platform.0.log	Contains XML events for configuration changes. This log file contains very little useful information for system administrators.
\Program Files\Novell\Nsure Audit\logs\auditlog	Contains the log entries for Novell auditing.

4.2.3 Windows Server 2008 Administration Console Logs

Filename	Description
\Program Files (x86)\Novell\Tomcat\logs\stderr.log	Contains Tomcat error messages directed to stderr. This file is reset whenever Tomcat is restarted.
\Program Files (x86)\Novell\Tomcat\logs\stdout.log	Contains Tomcat error messages directed to stdout. This file is reset whenever Tomcat is restarted.
\Program Files (x86)\Novell\logs\app_sc.0.log	Contains events related to importing devices, device configuration changes, health status changes, statistics reporting, and communication problems.
\Program Files (x86)\Novell\logs\app_cc.0.log	Contains events related to policy configuration.
\Program Files (x86)\Novell\logs\platform.0.log	Contains XML events for configuration changes. This log file contains very little useful information for system administrators.
\Program Files (x86)\Novell\Nsure Audit\logs\auditlog	Contains the log entries for Novell auditing.

4.2.4 Linux Identity Server Logs

Filename	Description
<code>/var/opt/novell/tomcat5/logs/catalina.out</code>	Logging to this file occurs only if you have selected the <i>Echo to Console</i> option from the <i>Identity Servers > Servers > Edit > Logging</i> page. When component logging has been set to info for Applications, it contains entries tracing user authentication and role assignments.
<code>/opt/novell/devman/jcc/logs/jcc-0.log.0</code>	Contains the log entries for the server communications module related to interaction of the Identity Server with the Administration Console, such as imports, certificates, health checks, and configuration.

4.2.5 Windows Server 2003 Identity Server Logs

Filename	Description
<code>\Program Files\Novell\Tomcat\logs\stderr.log</code>	Contains Tomcat error messages directed to stderr. This file is reset whenever Tomcat is restarted.
<code>\Program Files\Novell\Tomcat\logs\stdout.log</code>	Logging to this file occurs only if you have selected the <i>Echo to Console</i> option from the <i>Identity Servers > Servers > Edit > Logging</i> page. When component logging has been set to info for Applications, it contains entries tracing user authentication and role assignments. This file is reset whenever Tomcat is restarted.
<code>\Program Files\Novell\devman\jcc\logs\jcc-0.log.0</code>	Contains the log entries for the server communications module related to interaction of the Identity Server with the Administration Console, such as imports, certificates, health checks, and configuration.

4.2.6 Windows Server 2008 Identity Server Logs

Filename	Description
<code>\Program Files (x86)\Novell\Tomcat\logs\stderr.log</code>	Contains Tomcat error messages directed to stderr. This file is reset whenever Tomcat is restarted.

Filename	Description
<code>\Program Files (x86)\Novell\Tomcat\logs\stdout.log</code>	<p>Logging to this file only occurs if you have selected the <i>Echo to Console</i> option from the <i>Identity Servers > Servers > Edit > Logging</i> page.</p> <p>When component logging has been set to info for Applications, it contains entries tracing user authentication and role assignments.</p> <p>This file is reset whenever Tomcat is restarted.</p>
<code>\Program Files (x86)\Novell\devman\jcc\logs\jcc-0.log.0</code>	<p>Contains the log entries for the server communications module related to interaction of the Identity Server with the Administration Console, such as imports, certificates, health checks, and configuration.</p>

4.2.7 Linux Access Gateway Appliance Logs

Filename	Description
<code>/var/opt/novell/tomcat5/logs/catalina.out</code>	<p>Logging to this file only occurs if you have selected the <i>Echo to Console</i> option from the <i>Identity Servers > Servers > Edit > Logging</i> page.</p> <p>Check this file for entries tracing the evaluation of authorization, identity injection, and form fill policies.</p>
<code>/var/log/novell/reverse/<name></code>	<p>If logging is enabled on one or more reverse proxies, this directory contains the log files. To enable this type of logging, see “Configuring Logging for a Proxy Service” in the <i>Novell Access Manager 3.1 SP2 Access Gateway Guide</i>.</p> <p>A directory is listed for each reverse proxy on which you have enabled logging.</p>
<code>/var/log/ics_dyn.log</code>	<p>Contains all log entries generated by the Linux Access Gateway Appliance. Use syslog to control file rolling and log file distribution.</p>
<code>/opt/novell/devman/jcc/logs/jcc-0.log.0</code>	<p>Contains the log entries for the server communications module related to interaction of the Access Gateway with the Administration Console, such as imports, certificates, health checks, and configuration.</p>
<code>/var/log/lagsoapmessages</code>	<p>Logs all the SOAP messages between the Linux Access Gateway and the Embedded Service Provider.</p>

Filename	Description
/var/log/laghttpheaders	Contains a log of the HTTP headers to and from the Linux Access Gateway.

4.2.8 Linux Access Gateway Service Logs

Filename	Description
/var/log/novell-apache2	<p>If logging is enabled on one or more reverse proxies, this directory contains the log files. To enable this type of logging, see “Configuring Logging for a Proxy Service” in the <i>Novell Access Manager 3.1 SP2 Access Gateway Guide</i>.</p> <p>This directory also contains the Apache generated log files such as the <code>error_log</code> file.</p>
/var/opt/novell/amlogging/logs	<p>If you have enabled log profiles, this directory contains these log files. To enable this type of logging, see “Access Gateway Service Logs” in the <i>Novell Access Manager 3.1 SP2 Access Gateway Guide</i>.</p>
/var/opt/novell/amlogging/logs/ags_error.log	<p>Contains the messages generated for configuration, device imports, health, and statistics. It also contains entries for the policy evaluation processes done by the Gateway Service Manager module.</p>
/var/opt/novell/tomcat5/logs/catalina.out	<p>Contains the log messages generated by the Embedded Service Provider. Logging to this file occurs only if you have selected the <i>Echo to Console</i> option from the <i>Identity Servers > Servers > Edit > Logging</i> page.</p> <p>Check this file for entries tracing the evaluation of authorization, identity injection, and form fill policies.</p>

4.2.9 Windows Access Gateway Service Logs

Filename	Description
\\Program Files\\Novell\\amlogging\\logs\\ags_error.log	<p>Contains the messages generated for configuration, device imports, health, and statistics. It also contains entries for the policy evaluation processes done by the Gateway Service Manager module.</p>

Filename	Description
\Program Files\Novell\amlogging\logs\	If you have enabled log profiles, this directory contains these log files. To enable this type of logging, see “Access Gateway Service Logs” in the <i>Novell Access Manager 3.1 SP2 Access Gateway Guide</i> .
\Program Files\Novell\Apache\logs\ <name>	If logging is enabled on one or more reverse proxies, this directory contains the log files. To enable this type of logging, see “Configuring Logging for a Proxy Service” in the <i>Novell Access Manager 3.1 SP2 Access Gateway Guide</i> . This directory also contains the Apache generated log files such as the <code>error_log</code> file.
\Program Files\Novell\Tomcat\logs\ stdout.log	Contains the log messages generated by the Embedded Service Provider. Logging to this file only occurs if you have selected the <i>Echo to Console</i> option from the <i>Identity Servers > Servers > Edit > Logging</i> page. Check this file for entries tracing the evaluation of authorization, identity injection, and form fill policies. This file is reset whenever Tomcat is restarted.

4.2.10 SSL VPN Server Logs

Filename	Description
/var/opt/novell/tomcat5/logs/ catalina.out	Logging to this file occurs only if you have selected the <i>Echo to Console</i> option from the <i>Identity Servers > Servers > Edit > Logging</i> page.
/opt/novell/devman/jcc/logs/ jcc0.log.0	Contains the log entries for the server communications module related to interaction of the SSL VPN with the Administration Console, such as imports, certificates, and configuration.
/var/log/messages	Contains the log entries for the Connection Manager and SOCKS servers.
/var/log.novell-openvpn.log	Contains log entries for the OpenVPN server or the Enterprise mode server.
/var/log/stunnel.log	Contains log entries for Stunnel or the Kiosk mode server.

4.3 Using the Log Files for Troubleshooting

The following sections describe the logging features available in Access Manager and provide information on how you can use them for troubleshooting problems:

- ◆ [Section 4.3.1, “Enabling Logging,” on page 87](#)

- ♦ [Section 4.3.2, “Understanding the Log Format,” on page 87](#)
- ♦ [Section 4.3.3, “Sample Authentication Traces,” on page 90](#)

For information about policy tracing, see “[Understanding Policy Evaluation Traces](#)” in the *Novell Access Manager 3.1 SP2 Policy Guide*.

4.3.1 Enabling Logging

Each Access Manager device has configuration options for logging:

Identity Server: Logging is turned off and must be enabled. When you enable Identity Server logging, you also enable logging for the Embedded Service Providers that are configured to use the Identity Server for authentication. For configuration information, see “[Configuring Component Logging](#)” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.

Embedded Service Providers: Each Access Manager device has an Embedded Service Provider that communicates with the Identity Server. Its log level is controlled by configuring Identity Server logging.

Access Gateway Appliance: A log notice level of logging is enabled by default. You can change the level from the command line interface. For information, see “[Access Gateway Appliance Logs](#)” in the *Novell Access Manager 3.1 SP2 Access Gateway Guide*.

Access Gateway Service: The Gateway Service logs contain the messages sent between the Gateway Service and the Embedded Service Provider and between the Gateway Service and the Web server. This type of logging is turned off and must be enabled. For information, see “[Access Gateway Service Logs](#)” in the *Novell Access Manager 3.1 SP2 Access Gateway Guide*

4.3.2 Understanding the Log Format

Access Manager does not have a fixed format for file log entries. However, to facilitate the use of non-interactive stream-oriented editors such as `sgrep`, `sed`, `awk`, and `grep` and to improve log entry readability, the log entries in the `catalina.out` files use some standard elements. These entries use the beginning and ending log entry tags and the log entry correlation tags. The data portion of log entries is the most flexible part. A log entry has the following fields:

```
<amLogEntry> [\n]
  time-date-stamp
  [log preamble]:
  AM#event-code:
  AMDEVICE#device-id:
  AMAUTHID#auth-id:
  AMEVENTID#event-id:
  [...additional correlating information][\n]
  [supplementary log entry data and text ... \n]
</amLogEntry> [\n]
```

Most log entries do not use the optional line breaks (`[\n]`). Notice that the time-date-stamp, the log preamble, the correlation tags, and optional additional correlating information are on the same line so that stream-oriented editors that use only one line (such as `grep`) can be used to locate log entries that are related. The following entry is a typical entry that is logged when a user has initiated a login sequence.

```
<amLogEntry> 2009-06-08T21:06:25Z INFO NIDS Application: AM#500105014:
AMDEVICEID#9921459858EAAC29: AMAUTHID#BB11C254B7521B5E836D8703826287 AF:
Attempting to authenticate user cn=jwilson,o=novell with provided credentials.
</amLogEntry>
```

Table 4-1 *Fields in a Log Entry*

Field	Description
Beginning, ending tags	The <code><amLogEntry></code> and <code></amLogEntry></code> tags mark the beginning and the end of a log entry. These tags allow stream-oriented editors to extract log entries for processing.
Time-date-stamp tag	The date and time is specified in the W3C profile format of ISO 8061. It has the following fields: year-month-day-T-hour-minutes-seconds-time zone. The Z value for the time zone indicates that the time is specified in UTC.
Log preamble	This information is optional, and usually consists of a string indicating the logging level (such as warning, informational, or debug) and a string identifying the type of module making the entry. In the example log entry, the preamble has a log level and a module identifier and contains the following strings: INFO NIDS Application:
Correlation tags	The correlation tags uniquely identify the event, the device that produced the event, and the user who requested the action. The example log entry contains the following correlation tags: AM#500105014: AMDEVICEID#9921459858EAAC29: AMAUTHID#BB11C254B7521B5E836D8703826287AF: For more information, see “Understanding the Correlation Tags in the Log Files” on page 89 .
Additional correlation information	This information is optional, and contains correlation tags and data unique to a specific type of trace. For example, a policy evaluation trace created by the Embedded Service Provider contains the following additional tags: <ul style="list-style-type: none"> ◆ NXPESID#value ◆ POLICYID#value The example log entry does not contain any additional correlation information. For a log entry that does, see “Identity Injection Traces” in the Novell Access Manager 3.1 SP2 Policy Guide .
Supplementary information	This information is optional, and contains information that is specific to the log entry. It can be as simple as an informational string, such as the string in the example log entry: Attempting to authenticate user cn=jwilson,o=novell with provided credentials. The supplementary information can have a very specific format. For an example and explanation of the policy trace information, see “Understanding Policy Evaluation Traces” in the Novell Access Manager 3.1 SP2 Policy Guide .

Understanding the Correlation Tags in the Log Files

There is no fixed field format for log file entries. However, because most requests handled by Access Manager are processed by multiple Access Manager components, there is a mechanism that facilitates the correlation of log entries for a single Access Manager request in the various component log files. A correlation tag has the following general format:

```
<tag name>#<tag value>:
```

The <tag name> is a fixed value, defined in the Format column of [Table 4-2](#). It is always terminated by the # character. The <tag value> immediately follows the # character and is always terminated by the : character. The <tag value> is not a fixed value, but a uniquely assigned value to identify an event, a user, or a transaction. [Table 4-2](#) lists the defined correlation tags:

Table 4-2 Correlation Tags

Type	Format	Description
Event code	AM#<Event-Code>:	An event number defined in Novell Access Manager 3.1 SP2 Event Codes . This tag is included in all log entries that record an event and in all events that are presented to the user as an informational or error page.
User ID	AMAUTHID#<ID>:	<p>An authentication identifier that the Identity Server or the Embedded Service Provider assigns to each authenticated user. This tag is included in all entries that pertain to a request made by an authenticated user.</p> <p>Currently the Identity Server and the Embedded Service Provider (ESP) assign different authentication IDs. When correlating the flow of events between the Identity Server and the ESP for an authentication sequence, you can use the event code of the authentication events and find the artifact that the ESP and the Identity Server exchange.</p> <p>In the <code>catalina.out</code> file of the Identity Server, search for <code>AM#500105018</code> events. This is the event that sends the artifact to the ESP. Search for a corresponding artifact in the Access Gateway log. Events <code>AM#500105020</code> and <code>AM#500105021</code> contain the artifact value.</p>
Device ID	AMDEVICE#<ID>	<p>An identifier that uniquely identifies the Access Manager device that is generating the log entry.</p> <p>You can view the identifier that is assigned to each device on the General Logging page in the Administration Console (click <i>Auditing > General Logging</i>). The ID begins with a prefix that identifies the type of device such as <code>idp</code> for Identity Server, <code>ag</code> for an Access Gateway, and <code>idp-esp</code> for the Embedded Service Provider of the device. The prefix is followed by a 16-digit hexadecimal number.</p> <p>In log entries, the <code>idp</code> prefix is not recorded. For example, the General Logging page displays <code>idp-AA257DA77ED48DB0</code> for the ID of the Identity Server, but in the <code>catalina.out</code> file, the value is <code>AMDEVICE#AA257DA77ED48DB0</code>.</p>

Type	Format	Description
Transaction ID	AMEVENTID#<ID>:	<p>An identifier assigned to each Access Manager or system administration transaction. Access Manager transactions are actions such as authenticating a user, processing a request for access to a resource, and federating an identity.</p> <p>If a user requests access to multiple resources, each request is given a separate transaction ID. When the Access Gateway evaluates a policy for a protected resource page and the page contains links, the policy is evaluated for each link, and each of these evaluations generates a new transaction ID.</p> <p>System administration transactions are actions such as importing a device, deleting a device, stopping or starting a device, and configuring or modifying the configuration of a device.</p>

Sample Scenario

The following scenario illustrates how these tags can be used. A user receives an error page indicating that he or she has been refused access to a protected resource. The error page contains an event code. The user contacts the system administrator and reports the event code contained in the message. The code displayed to the user includes both an event number and an identifier indicating the device detecting the error, for example, 300101023-92E1B234. The 300101023 value is the event number and 92E1B234 is the device identifier. The device identifier is the number assigned to the Access Manager device reporting the error. You can make a textual search of log entries using the tags and values `AM#300101023:` and `AMDEVICEID#92E1B234:` to locate candidate log entries of the target Access Manager transaction flow. When the desired log entry is found, the `AMEVENTID#` tag and value and the `AMAUTHID#` tag (assuming the user has been authenticated) from the log entry can be used to locate all other log entries pertaining to the user in the context of the transaction.

4.3.3 Sample Authentication Traces

An authentication trace is logged to the `catalina.out` file of the Identity Server that authenticates the user. If the Access Gateway initiates the authentication because of a user request to a protected resource, the Embedded Service Provider log file of the Access Gateway also contains entries for the authentication sequence. Identity Server logging must be enabled to produce authentication traces (see “[Configuring Component Logging](#)” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*).

This section describes the following types of authentication traces:

- ◆ “[Direct Authentication Request to the Identity Server](#)” on page 91
- ◆ “[Protected Resource Authentication Trace](#)” on page 93

Direct Authentication Request to the Identity Server

The following trace is an example of a user logging directly into the Identity Server to access the End User Portal. The log entries are numbered, so that they can be described.

```
1. <amLogEntry> 2009-06-14T17:14:30Z INFO NIDS Application: AM#500105015:
AMDEVICEID#9921459858EAAC29: AMAUTHID#F35A3C7AD7F2EEDEB3D17F99EC3F39D1:
Processing login request with TARGET = http://10.10.15.19:8080/nidp/app, saved
TARGET = . </amLogEntry>

2. <amLogEntry> 2009-06-14T17:14:30Z INFO NIDS Application: AM#500105009:
AMDEVICEID#9921459858EAAC29: AMAUTHID#F35A3C7AD7F2EEDEB3D17F99EC3F39D1:
Executing contract Name/Password - Form. </amLogEntry>

3. <amLogEntry> 2009-06-14T17:14:30Z INFO NIDS Application: AM#500105010:
AMDEVICEID#9921459858EAAC29: AMAUTHID#F35A3C7AD7F2EEDEB3D17F99EC3F39D1:
Contract Name/Password - Form requires additional interaction. </amLogEntry>

4. <amLogEntry> 2009-06-14T17:14:39Z INFO NIDS Application: AM#500105015:
AMDEVICEID#9921459858EAAC29: AMAUTHID#F35A3C7AD7F2EEDEB3D17F99EC3F39D1:
Processing login request with TARGET = http://10.10.15.19:8080/nidp/app, saved
TARGET = http://10.10.15.19:8080/nidp/app. </amLogEntry>

5. <amLogEntry> 2009-06-14T17:14:39Z INFO NIDS Application: AM#500105009:
AMDEVICEID#9921459858EAAC29: AMAUTHID#F35A3C7AD7F2EEDEB3D17F99EC3F39D1:
Executing contract Name/Password - Form. </amLogEntry>

6. <amLogEntry> 2009-06-14T17:14:39Z INFO NIDS Application: AM#500105014:
AMDEVICEID#9921459858EAAC29: AMAUTHID#F35A3C7AD7F2EEDEB3D17F99EC3F39D1:
Attempting to authenticate user cn=bcf,o=novell with provided credentials. </
amLogEntry>

7. <amLogEntry> 2009-06-14T17:14:39Z WARNING NIDS Application: Event Id:
3014666, Target: cn=bcf,o=novell, Sub-Target:
F35A3C7AD7F2EEDEB3D17F99EC3F39D1, Note 1: Local, Note 2: This Identity
Provider, Note 3: name/password/uri, Numeric 1: 0 </amLogEntry>

8. <amLogEntry> 2009-06-14T17:14:39Z WARNING NIDS Application: Event Id:
3015456, Note 1: F35A3C7AD7F2EEDEB3D17F99EC3F39D1, Note 2: Manager, Note 3:
Document=(ou=xpemplPEP,ou=mastercdn,ou=Content
PublisherContainer,ou=Partition,ou=PartitionsContainer,ou=VCDN_Root,ou=access
ManagerContainer,o=novell:romaContentCollectionXMLDoc),Policy=(Manager),Rule=
(1::RuleID_1181251958207),Action=(AddRole::ActionID_1181252224665), Numeric
1: 0 </amLogEntry>

9. <amLogEntry> 2009-06-14T17:14:39Z WARNING NIDS Application: Event Id:
3015456, Note 1: F35A3C7AD7F2EEDEB3D17F99EC3F39D1, Note 2: authenticated, Note
3: system-generated-action, Numeric 1: 0 </amLogEntry>

10. <amLogEntry> 2009-06-14T17:14:39Z INFO NIDS Application: AM#500199050:
AMDEVICEID#9921459858EAAC29: AMAUTHID#F35A3C7AD7F2EEDEB3D17F99EC3F39D1: IDP
RolesPep.evaluate(), policy trace:
  ~RL~1~~~~Rule Count: 1~~Success(67)
  ~RU~RuleID_1181251958207~Manager~DNF~~1:1~~Success(67)
  ~CS~1~~ANDs~~1~~True(69)
  ~CO~1~LdapGroup(6645):no-param:hidden-value:~ldap-group-is-member-
of~SelectedLdapGroup(66455):hidden-param:hidden-value:~~~True(69)
  ~PA~ActionID_1181252224665~~AddRole~Manager~~~Success(0)
  ~PC~ActionID_1181252224665~~Document=(ou=xpemplPEP,ou=mastercdn,
```

```
ou=ContentPublisherContainer,ou=Partition,ou=PartitionsContainer,ou=VCDN_Root
,ou=accessManagerContainer,o=novell:romaContentCollectionXMLDoc),Policy=(Manager),Rule=(1::RuleID_1181251958207),Action=(AddRole::ActionID_1181252224665)~
AdditionalRole(6601):unknown():Manager:~~~Success(0)
</amLogEntry>
```

```
11. <amLogEntry> 2009-06-14T17:14:39Z INFO NIDS Application: AM#500105013:
AMDEVICEID#9921459858EAAC29: AMAUTHID#F35A3C7AD7F2EEDEB3D17F99EC3F39D1:
Authenticated user cn=bcf,o=novell in User Store Local Directory with roles
Manager,authenticated. </amLogEntry>
```

```
12. <amLogEntry> 2009-06-14T17:14:39Z INFO NIDS Application: AM#500105017:
AMDEVICEID#9921459858EAAC29: AMAUTHID#F35A3C7AD7F2EEDEB3D17F99EC3F39D1:
nLogin succeeded, redirecting to http://10.10.15.19:8080/nidp/app. </
amLogEntry>
```

Table 4-3 Log Entry Descriptions for an Authentication Trace from an Identity Server

Entry	Description
1	Indicates that a login request is in process. This is the first entry for a login request. The requester, even though login has not been successful, is assigned an authentication ID. You can use this ID to find the log entries related to this user. The entry also specifies the URL of the requested resource, in this case the /nidp/app resource called the End User Portal. The saved TARGET message does not contain a value, so this step will be repeated.
2	Specifies the contract that is being used to perform the login.
3	Indicates that the contract requires interaction with the user.
4	Indicates that the a login request is in process. The saved TARGET message contains a value, so the required information has been gathered to start the authentication request. The AM# correlation tag is AM#500105015, which is the same value as the first log entry.
5	Indicates that an exchange is occurring between the client and the Identity Server to obtain the required credentials. Each contract requires a different exchange. The AM# correlation tag is AM#500105009, which is the same value as the second log entry.
6	Provides the DN of the user attempting to log in and indicates that the user's credentials are being sent to the LDAP server for verification.
7	Provides information about an auditing event. If you have not enabled auditing or you have not selected the login events, this entry does not appear in your log file. This event contains information about who is logging in and the contract that is being used.
8	Provides information about an auditing event. If you have not enabled auditing or you have not selected the login events, this entry does not appear in your log file. This event contains information about the Manager policy that is evaluated during login.
9	Provides information about an auditing event. If you have not enabled auditing or you have not selected the login events, this entry does not appear in your log file.
10	Contains the entry for processing a Role policy. When a user logs in, all Role policies are evaluated and the user is assigned any roles that the user has the qualifications for. For more information, see " Understanding Policy Evaluation Traces " in the <i>Novell Access Manager 3.1 SP2 Policy Guide</i> .
11	Contains a summary of who logged in from which user store and the names of the Role policies that successfully assigned roles to the user.

Entry	Description
12	Contains the final results of the login, with the URL that the request is redirected to.

Protected Resource Authentication Trace

When a protected resource is configured to require authentication, both the Identity Server and the Embedded Service Provider of the Access Gateway (or J2EE Agent) generate log entries for the process. The following sections explain how to correlate the entries from the logs.

- ◆ [“Entries from an Identity Server Log” on page 93](#)
- ◆ [“Entries from an Access Gateway Log” on page 94](#)
- ◆ [“Correlating the Log Entries between the Identity Server and the Access Gateway” on page 94](#)

Entries from an Identity Server Log

```
<amLogEntry> 2009-07-31T17:36:39Z INFO NIDS Application: AM#500105016:
AMDEVICEID#AA257DA77ED48DB0: AMAUTHID#83778AE09DCA5A35B57842D754A60D67:
Processing login resulting from Service Provider authentication request. </
amLogEntry>
```

```
<amLogEntry> 2009-07-31T17:36:39Z INFO NIDS Application: AM#500105009:
AMDEVICEID#AA257DA77ED48DB0: AMAUTHID#83778AE09DCA5A35B57842D754A60D67:
Executing contract Name/Password - Form. </amLogEntry>
```

```
<amLogEntry> 2009-07-31T17:36:39Z INFO NIDS Application: AM#500105010:
AMDEVICEID#AA257DA77ED48DB0: AMAUTHID#83778AE09DCA5A35B57842D754A60D67:
Contract Name/Password - Form requires additional interaction. </amLogEntry>
```

```
<amLogEntry> 2009-07-31T17:36:49Z INFO NIDS Application: AM#500105016:
AMDEVICEID#AA257DA77ED48DB0: AMAUTHID#83778AE09DCA5A35B57842D754A60D67:
Processing login resulting from Service Provider authentication request. </
amLogEntry>
```

```
<amLogEntry> 2009-07-31T17:36:49Z INFO NIDS Application: AM#500105009:
AMDEVICEID#AA257DA77ED48DB0: AMAUTHID#83778AE09DCA5A35B57842D754A60D67:
Executing contract Name/Password - Form. </amLogEntry>
```

```
<amLogEntry> 2009-07-31T17:36:49Z INFO NIDS Application: AM#500105014:
AMDEVICEID#AA257DA77ED48DB0: AMAUTHID#83778AE09DCA5A35B57842D754A60D67:
Attempting to authenticate user cn=admin,o=novell with provided credentials.
</amLogEntry>
```

```
<amLogEntry> 2009-07-31T17:36:49Z INFO NIDS Application: AM#500105012:
AMDEVICEID#AA257DA77ED48DB0: AMAUTHID#83778AE09DCA5A35B57842D754A60D67:
Authenticated user cn=admin,o=novell in User Store Internal with no roles. </
amLogEntry>
```

```
<amLogEntry> 2009-07-31T17:36:49Z INFO NIDS Application: AM#500105018:
AMDEVICEID#AA257DA77ED48DB0: AMAUTHID#83778AE09DCA5A35B57842D754A60D67:
Responding to AuthnRequest with artifact AAMoz+rm2jQjDSHjea8U9zm3Td/U2ax0YZCo/
qBNool8WkZiTct7N7Jx </amLogEntry>
```

```
<amLogEntry> 2009-07-31T17:36:49Z INFO NIDS Application: AM#500105019:
AMDEVICEID#AA257DA77ED48DB0: AMAUTHID#C2D8D52704918AF2D5D62F6EDC2FFAC6:
Sending AuthnResponse in response to artifact AAMoz+rm2jQjDSHjea8U9zm3Td/
U2ax0YZCo/qBNool8WkZiTct7N7Jx </amLogEntry>
```

Entries from an Access Gateway Log

```
<amLogEntry> 2009-07-31T17:35:05Z INFO NIDS Application: AM#500105005:  
AMDEVICEID#esp-2FA73CE1A376FD91: AMAUTHID#C6D119FD93EEBBEBEC50BEB27B9E2832:  
Processing proxy request for login using contract name/password/uri and return  
url http://jwilson.provo.novell.com/ </amLogEntry>
```

```
<amLogEntry> 2009-07-31T17:35:05Z INFO NIDS Application: AM#500105015:  
AMDEVICEID#esp-2FA73CE1A376FD91: AMAUTHID#C6D119FD93EEBBEBEC50BEB27B9E2832:  
Processing login request with TARGET = http://jwilson.provo.novell.com/, saved  
TARGET = . </amLogEntry>
```

```
<amLogEntry> 2009-07-31T17:35:05Z INFO NIDS Application: AM#500105009:  
AMDEVICEID#esp-2FA73CE1A376FD91: AMAUTHID#C6D119FD93EEBBEBEC50BEB27B9E2832:  
Executing contract IDP Select. </amLogEntry>
```

```
<amLogEntry> 2009-07-31T17:35:05Z INFO NIDS Application: AM#500105010:  
AMDEVICEID#esp-2FA73CE1A376FD91: AMAUTHID#C6D119FD93EEBBEBEC50BEB27B9E2832:  
Contract IDP Select requires additional interaction. </amLogEntry>
```

```
<amLogEntry> 2009-07-31T17:35:15Z INFO NIDS Application: AM#500105020:  
AMDEVICEID#esp-2FA73CE1A376FD91: AMAUTHID#C6D119FD93EEBBEBEC50BEB27B9E2832:  
Received and processing artifact from IDP - AAMoz+rm2jQjDSHjea8U9zm3Td/  
U2ax0YZCo/qBNool8WkZiTct7N7Jx </amLogEntry>
```

```
<amLogEntry> 2009-07-31T17:35:15Z INFO NIDS Application: AM#500105021:  
AMDEVICEID#esp-2FA73CE1A376FD91: AMAUTHID#C6D119FD93EEBBEBEC50BEB27B9E2832:  
Sending artifact AAMoz+rm2jQjDSHjea8U9zm3Td/U2ax0YZCo/qBNool8WkZiTct7N7Jx to  
URL http://jwilson1.provo.novell.com:8080/nidp/idff/soap at IDP </amLogEntry>
```

Correlating the Log Entries between the Identity Server and the Access Gateway

You can see that these two trace sequences are for the same authentication request because the artifact (**AAMoz+rm2jQjDSHjea8U9zm3Td/U2ax0YZCo/qBNool8WkZiTct7N7Jx**) that is exchanged is the same. You can use the **AMAUTHID** in each file to search for other requests that this user has made.

To associate a distinguished name with the **AMAUTHID**, use the `catalina.out` file of the Identity Server. Event `AM#500105014` contains the DN of the user.

Changing the IP Address of Access Manager Devices

5

The following sections explain how to change the IP address on the following devices:

- ♦ [Section 5.1, “Changing the IP Address of the Administration Console,” on page 95](#)
- ♦ [Section 5.2, “Changing the IP Address of an Identity Server,” on page 95](#)
- ♦ [Section 5.3, “Changing the IP Address of the Access Gateway Appliance,” on page 97](#)
- ♦ [Section 5.4, “Changing the IP Address of the Access Gateway Service,” on page 98](#)
- ♦ [Section 5.5, “Changing the IP Address of the Audit Server,” on page 99](#)

NOTE: Changing the IP address of an SSL VPN component is not recommended.

5.1 Changing the IP Address of the Administration Console

We recommend that you install the Administration Console with the IP address that it will always use because all of the devices that import into the Administration Console use this address to establish secure communication with the Administration Console.

The only tested method of changing the IP address so that all other devices trust the Administration Console is to install a secondary console with the new IP address and then promote the secondary console to be the primary console.

See the following sections:

- ♦ [“Installing Secondary Versions of the Administration Console” in the *Novell Access Manager 3.1 SP2 Setup Guide*](#)
- ♦ [“Converting a Secondary Console into a Primary Console” on page 110](#)

Converting a secondary console into a primary console is not a simple task. The task was designed as a disaster recovery solution when the primary console is no longer available.

5.2 Changing the IP Address of an Identity Server

These instructions assume that your Identity Server and Administration Console are not on the same machine. If they are on the same machine, see [Section 5.1, “Changing the IP Address of the Administration Console,” on page 95](#).

To move a machine or change the IP address for the Identity Server:

- 1 In the Administration Console, click *Devices > Identity Servers*.
- 2 Click the server name.
- 3 On the General page, click *Edit*.

- 4 Specify the new IP address in the *Management IP Address* field and, if necessary, a port.
- 5 Click *OK*, then click *Close*.
- 6 On the Identity Server, stop the server communication service by using the following command:
Linux: `/etc/init.d/novell-jcc stop`
Windows: `net stop jccserver`
- 7 Change the IP address by using an operating system utility:
Linux: Click *YaST > Network Devices > Network Card*, select a method, select the card, then click *Edit*.
Windows: Click *Control Panel > Network Connections > Local Area Connection > Properties > Internet Protocol (TCP/IP) > Properties*.
- 8 Change to the `jcc` directory:
Linux: `/opt/novell/devman/jcc`
Windows Server 2003: `\Program Files\Novell\devman\jcc`
Windows Server 2008: `\Program Files (x86)\Novell\devman\jcc`
- 9 Run the configure command:
Linux: `conf/Configure.sh`
Windows: `conf\configure.cmd`
The command must be run from the `jcc` directory because it needs access to files that are available from this directory.
- 10 When you are prompted for the local listener IP address, enter the new IP.
- 11 When you are prompted for the administration server IP, enter the IP address of the Administration Console.
- 12 Follow the prompts and accept the defaults for ports and admin user.
- 13 Replace all references to the old IP address in the `server.xml` file with the new IP address:
 - 13a Change to the Tomcat configuration directory:
Linux: `/var/opt/novell/tomcat5/conf`
Windows Server 2003: `\Program Files\Novell\Tomcat\conf`
Windows Server 2008: `\Program Files (x86)\Novell\Tomcat\conf`
 - 13b In a text editor, open the `server.xml` file.
 - 13c Search for the old IP address and replace it with the new IP address.
 - 13d Save your changes.
- 14 Start the server communication service by using the following command:
Linux: `/etc/init.d/novell-jcc start`
Windows: `net start jccserver`
- 15 Restart Tomcat:
Linux: Enter the following command:
`/etc/init.d/novell-tomcat5 restart`
Windows: Enter the following commands:

```
net stop Tomcat5
net start Tomcat5
```

For information about deleting an Identity Server, see “[Maintaining an Identity Server](#)” in the *Novell Access Manager 3.1 SP2 Identity Server Guide*.

5.3 Changing the IP Address of the Access Gateway Appliance

If you need to change the IP address of the Access Gateway machine, you need to configure the Access Gateway for this change. This is especially significant when the Access Gateway Appliance has only one IP address.

IMPORTANT: The new IP address must be configured in the Administration Console before you change it on the Access Gateway. If you change the address on the Access Gateway first, the Administration Console does not trust the Access Gateway and cannot establish communication.

- 1 In the Administration Console, click *Devices > Access Gateways > Edit > Adapter List*.
- 2 (Conditional) If the machine belongs to a cluster, select the Access Gateway from the *Cluster Member* list.
- 3 From the Adapter List, select the subnet mask that contains the IP address you want to change. When you select the subnet mask, the Adapter page appears.

Adapter eth0

Subnet: 10.10.159.206

Subnet Mask: *

IP Address List *

[New...](#) | [Delete](#) | [Change IP Address...](#)

<input type="checkbox"/> IP Addresses
<input type="checkbox"/> 10.10.159.206

Changes made on this panel must be applied or scheduled from the [Configuration](#) Panel.

- 4 Select the old IP address, click *Change IP Address*, specify the new IP address, then click *OK*. This option changes all configuration instances of the old IP address to the new IP address. For example, any reverse proxies that have been assigned the old IP address as a listening address are modified to use the new IP address as the listening address.
- 5 To save your changes to browser cache, click *OK*.
- 6 To apply your changes, click the *Access Gateways* link, then click *Update > OK*.
- 7 If you are physically moving the machine, move it before completing the rest of these steps.

- 8 Check the IP address that the Administration Console uses for managing the Access Gateway. Click *Access Gateways* > [*Name of Access Gateway*] > *Edit*.
- 9 If the old IP address is listed as the *Management IP Address*, select the new IP address. If your Access Gateway has multiple IP addresses, select the one that you want the Administration Console to use for communication with the Access Gateway.

The port should only be modified if there is another device on the Access Gateway that is using the default port of 1443.
- 10 If the name of the Access Gateway is the old IP address, modify the *Name* option.
- 11 Click *OK*.

The Administration Console uses the configured IP address to find the Access Gateway.

- 12 On the Identity Server, restart Tomcat:

Linux: Enter the following command:

```
/etc/init.d/novell-tomcat5 restart
```

Windows: Enter the following commands:

```
net stop Tomcat5
```

```
net start Tomcat5
```

If your Access Gateway stops reporting to the Administration Console after completing these steps, you need to trigger an auto-import. See “[Triggering an Import Retry](#)” in the *Novell Access Manager 3.1 SP2 Installation Guide*.

5.4 Changing the IP Address of the Access Gateway Service

- 1 On the Access Gateway Service, use a system utility to add the new IP address.

Do not delete the old IP address at this time.

Linux: Start YaST, click *Network Devices* > *Network Card*, then select the *Traditional Method*.
Windows: Access the Control Panel, click *Network Connections* > *Local Area Connection* > *Properties*, then select *Internet Protocol (TCP/IP)*. Click *Properties* > *Advanced*.
- 2 In the Administration Console, import the new IP address:
 - 2a Click *Access Gateways* > [*Name of Access Gateway*] > *New IP*.
 - 2b Click *OK*.

Wait for the command to complete.
- 3 Change the management IP address:
 - 3a On the Server Details page, click *Edit*.
 - 3b If the old IP address is listed as the *Management IP Address*, select the new IP address.

If your Access Gateway has multiple IP addresses, select the one that you want the Administration Console to use for communication with the Access Gateway.
 - 3c (Conditional) Modify the port if there is another device on the Access Gateway that is using the default port of 1443.
 - 3d If the name of the Access Gateway is the old IP address, modify the *Name* option.

3e Click *OK*.

The Administration Console uses the configured IP address to find the Access Gateway.

- 4** To verify that the new IP address is being used, check the health of the Access Gateway.
- 5** Edit the Access Gateway configuration so that the reverse proxies use the new IP address:
You need to complete these steps for each reverse proxy.
 - 5a** In the Administration Console, click *Access Gateways > Edit > [Name of Reverse Proxy]*.
 - 5b** (Conditional) If a member of a cluster, select the cluster member that has a new IP address.
 - 5c** For the listening address, deselect the old IP address and select the new IP address.
 - 5d** Apply the settings and update the Access Gateway.
 - 5e** Verify that everything is working correctly by accessing a resource protected by this reverse proxy.
- 6** On the Access Gateway Service machine, use a system utility to remove the old IP address.
- 7** Remove the old IP address from the Administration Console:
 - 7a** Click *Access Gateways > [Name of Access Gateway] > New IP*.
 - 7b** Click *OK*.
Wait for the command to complete.
 - 7c** To verify that the old address has been removed, click *Edit* and verify that the old address is not an option for the *Management IP Address*.

5.5 Changing the IP Address of the Audit Server

To move a machine or change the IP address for the audit server:

- 1** In the Administration Console, click *Auditing > Novell Auditing*.
- 2** On the Novell Auditing page, change the IP address for the server and, if necessary, the port.
- 3** Click *OK*.
- 4** Update all Access Gateways.
- 5** Reboot all servers, including the Access Gateways, to use the new configuration.

Troubleshooting the Administration Console

6

This section discusses general troubleshooting issues found in the Administration Console:

- ◆ [Section 6.1, “Global Troubleshooting Options,” on page 101](#)
- ◆ [Section 6.2, “Stopping Tomcat on Windows,” on page 108](#)
- ◆ [Section 6.3, “Logging,” on page 108](#)
- ◆ [Section 6.4, “Event Codes,” on page 109](#)
- ◆ [Section 6.5, “Restoring a Failed Secondary Console,” on page 109](#)
- ◆ [Section 6.6, “Moving the Primary Administration Console to New Hardware,” on page 109](#)
- ◆ [Section 6.7, “Converting a Secondary Console into a Primary Console,” on page 110](#)
- ◆ [Section 6.8, “Orphaned Objects in the Trust/Configuration Store,” on page 122](#)
- ◆ [Section 6.9, “Repairing the Configuration Datastore,” on page 123](#)
- ◆ [Section 6.10, “Session Conflicts,” on page 123](#)
- ◆ [Section 6.11, “Unable to Log In to the Administration Console,” on page 123](#)
- ◆ [Section 6.12, “\(Linux\) Exception Processing IdentityService_ServerPage.JSP,” on page 124](#)
- ◆ [Section 6.13, “Backup/Restore Failure Because of Special Characters in Passwords,” on page 124](#)
- ◆ [Section 6.14, “Unable to Install NMAS SAML Method,” on page 125](#)

6.1 Global Troubleshooting Options

The following options allow you to view the status of multiple devices and identify the devices that are not healthy.

- ◆ [Section 6.1.1, “Checking for Potential Configuration Problems,” on page 101](#)
- ◆ [Section 6.1.2, “Checking for Version Conflicts,” on page 103](#)
- ◆ [Section 6.1.3, “Checking for Invalid Policies,” on page 104](#)
- ◆ [Section 6.1.4, “Viewing Device Health,” on page 104](#)
- ◆ [Section 6.1.5, “Viewing Health by Using the Hardware IP Address,” on page 104](#)
- ◆ [Section 6.1.6, “Using the Dashboard,” on page 105](#)
- ◆ [Section 6.1.7, “Viewing System Alerts,” on page 108](#)

6.1.1 Checking for Potential Configuration Problems

If your Access Manager components are not behaving in the way you have configured them to run, you might want to check the system to see if any of the components have configuration or network problems.

- 1 In the Administration Console, click *Auditing > Troubleshooting > Configuration*.

- 2 All of the options should be empty, except the *Cached Access Gateway Configurations* option (see [Step 4](#)) and the *Current Access Gateway Configurations* option (see [Step 5](#)). If an option contains an entry, you need to clear it. Select the appropriate action from the following table:

Option	Description and Action
<i>Device Pending with No Commands</i>	If you have a device that remains in the pending state, even when all commands have successfully executed, that device appears in this list. Before deleting the device from this list, check its Command Status. If the device has any commands listed, select the commands, then delete them. Wait a few minutes. If the device remains in a pending state, return to this troubleshooting page. Find the device in the list, then click <i>Remove</i> . The Administration Console clears the pending state.
<i>Other Known Device Manager Servers</i>	If a secondary Administration Console is in a non-reporting state, perhaps caused by hardware failure, its configuration needs to be removed from the primary Administration Console. As long as it is part of the configuration, other Access Manager devices try to contact it. If you cannot remove it by running the uninstall script on the secondary Administration Console, you can remove it by using this troubleshooting page. Click the <i>Remove</i> button next to the console that is in the non-reporting state. All references to the secondary Administration Console are removed from the configuration database.
<i>Access Gateways with Corrupt Protected Resource Data</i>	If you modify the configuration for a protected resource, update the Access Gateway with the changes, then review the configuration for the protected resource and the changes have not been applied, the configuration for the protected resource is corrupted. Click the <i>Repair</i> button next to the protected resource that has a corrupted configuration. You should then be able to modify its configuration, and when you update the Access Gateway, the changes should be applied and saved.
<i>Access Gateways with Duplicate Protected Resource Data</i>	After an upgrade, if you get errors related to invalid content for policy enforcement lists, you need to correct them. The invalid elements that do not have an associated resource data element are listed in this section. Click the <i>Repair</i> button to remove them.
<i>Access Gateways with Protected Resources Referencing Nonexistent Policies</i>	Protected resources have problems when policies are deleted before their references to the protected resources are removed. If you have protected resources in this condition, they are listed in this section. Click the <i>Repair</i> button to remove these references. Then verify that your protected resources have the correct policies enabled. Click <i>Access Gateways > Edit > [Name of Reverse Proxy] > [Name of Proxy Service] > Protected Resources</i> , then change to the <i>Policy View</i> .

Option	Description and Action
<i>Access Gateways with Invalid Alert Profile References</i>	You can create XML validation errors on your Access Gateway Appliance if you start to create an alert profile (click <i>Access Gateways > Edit > Alerts > New</i>), but you do not finish the process. The incomplete alert profile does not appear in the configuration for the Access Gateway, so you cannot delete it. If such a profile exists, it appears in the <i>Access Gateways with Invalid Alert Profile References</i> list. Click the <i>Remove</i> button by the invalid profile. You should then be able to modify its configuration, and when you update the Access Gateway, the changes should be applied and saved.
<i>Devices with Corrupt Data Store Entries</i>	If an empty value is written to an XML attribute, the device with this invalid configuration appears in this list. Click the <i>Repair</i> button to rewrite the invalid attribute values.

- 3** When you have finished repairing or deleting invalid Access Gateway configurations, click the *Access Gateways* link, then click *Update > OK*.
- 4** (Optional) Verify that all members of an Access Gateway cluster have the same configuration in cache:
 - 4a** Click *Auditing > Troubleshooting > Configuration*.
 - 4b** Scroll to the *Cached Access Gateway Configuration* option.
 - 4c** Click *View* next to the cluster configuration or next to an individual Access Gateway.
 This option allows you to view the Access Gateway configuration that is currently residing in browser cache. If the Access Gateway belongs to a cluster, you can view the cached configuration for the cluster as well as the cached configuration for each member. The + and - buttons allow you to expand and collapse individual configurations. The configuration is displayed in XML format
 To search for particular configuration parameters, you need to copy and paste the text into a text editor.
- 5** (Conditional) After viewing the Access Gateway configuration (see [Step 4](#)) and discovering that an Access Gateway does not have the current configuration, select the Access Gateway in the *Current Access Gateway Configurations* section, then click *Re-push Current Configuration*.

6.1.2 Checking for Version Conflicts

The Version page displays all the installed components along with their currently running version. Use this page to verify that you have updated all components to the latest compatible versions. There are two steps to ensuring that your Access Manager components are running compatible versions:

- ♦ All components of the same type should be running the same version. If you have components that display multiple versions, identify the components that need to be upgraded and upgrade them to the newer version.
- ♦ All components need to be running versions that are compatible with each other. For the latest release, view the list in the “Novell Access Manager Readme” (http://www.novell.com/documentation/novellaccessmanager31/readme/accessmanager_readme.html).

To view the current version of all Access Manager devices:

- 1 In the Administration Console, click *Auditing > Troubleshooting*.
- 2 Click *Versions*.

A list of the devices with their version information is displayed. If a device also has an embedded service provider, the version of the Embedded Service Provider is also displayed.

6.1.3 Checking for Invalid Policies

The Policies page displays the policies that are in an unusable state because of configuration errors.

- 1 In the Administration Console, click *Auditing > Troubleshooting > Policies*.
If you have configured a policy without defining a valid rule for it, the policy appears in this list.
- 2 Select the policy, then click *Remove*.

6.1.4 Viewing Device Health

You can monitor all of the devices hosted by a server and quickly isolate and correct server issues. The system displays a status (green, yellow, white, or red) for the server.

- 1 In the Administration Console, click *Auditing > Device Health*.
The Device Health page shows the health status by IP address of the server and lists all the devices installed on the server. The health of the least healthy device is used for the status of the server.
- 2 To view more information about the health of each device, click the IP address of the machine.

Health information can also be viewed at the following locations:

- ♦ *Access Manager > Dashboard*
The Dashboard page shows the health status at the device level. The status displayed is the status of the least healthy device.
- ♦ *Devices > [Component] > Servers*
The Servers page for each component provides a health status for each device.

6.1.5 Viewing Health by Using the Hardware IP Address

The Hardware IP Address page allows you to view the devices and agents managed through the selected IP address. You can monitor all of the devices hosted by a server and quickly isolate and correct server issues. The system displays statuses (green, yellow, white, or red) for the Access Manager devices.

- 1 In the Administration Console, click *Access Manager > Auditing > Device Health*.
- 2 To view information about the health of each installed device, click an IP address.
- 3 Select one of the following actions:
 - ♦ To return to the Device Health page, click *Close*.
 - ♦ To edit the details of a device, click the server name.

- ◆ To view health details, click the *Health* icon.
- ◆ To view the alerts, click the alerts link.
- ◆ To view device statistics, click the statistics link.
- ◆ To view or configure audit events for the device, click the *Edit Events* link.

6.1.6 Using the Dashboard

The Dashboard page is the starting point and central place to monitor and manage all product components and policies. The status of each device is available, with colored warnings or alert conditions.

- 1 In the Administration Console, click *Access Manager > Dashboard*.
- 2 Click a box to view a component or click the link to view the alerts:
 - ◆ [Identity Servers](#)
 - ◆ [Access Gateways](#)
 - ◆ [SSL VPNs](#)
 - ◆ [J2EE Agents](#)
 - ◆ [Policies](#)
 - ◆ [Alerts](#)

For conventions that apply to all pages in the interface, see [Section 1.2.4, “Understanding Administration Console Conventions,”](#) on page 20.

Identity Servers

The Identity Server is the central authentication and identity access point for all Access Manager devices. The Identity Server is responsible for authenticating users and distributing role information to facilitate authorization decisions. It also provides the Liberty Alliance Web Service Framework to distribute identity information.

An Identity Server always operates as an identity provider and can optionally be configured to run as an identity consumer (also known as a service provider), using either Liberty, SAML 1.1, or SAML 2.0 protocols. As an identity provider, the Identity Server is the central store for a user’s identity information and is the heart of the user’s identity federations or account linkage information. As an authentication authority, the identity provider is viewed by internal and external service providers as a trusted identity store.

In an Access Manager configuration, the Identity Server is responsible for managing the following:

- ◆ **Authentication:** Verifies user identities through various forms of authentication, both local (user supplied) and indirect (supplied by external providers). The identity information can be some characteristic attribute of the user, such as a role, e-mail address, name, or job description.
- ◆ **Identity Stores:** Stores user identities in eDirectory, Microsoft Active Directory, and Sun ONE Directory Server.
- ◆ **Identity Federation:** Enables user identity federation and provides access to Liberty-enabled services.

- ◆ **Account Provisioning:** Enables service provider account provisioning when federating, which automatically creates user accounts.
- ◆ **Custom Attribute Mapping:** Allows you to define custom attributes by mapping Liberty Alliance keywords to LDAP-accessible data, in addition to the available Liberty Alliance Employee and Person profiles.
- ◆ **SAML Assertions:** Processes and generates SAML assertions. Using SAML assertions in each Access Manager component protects confidential information by removing the need to pass user credentials between the components to handle session management.
- ◆ **Single Sign-on and Log-out:** Enables users to log in only once to gain access to multiple applications and platforms. Single sign-on and single log-out are primary features of Access Manager and are achieved after the federation and trust model is configured among trusted providers and the components of Access Manager.
- ◆ **Embedded Service Providers:** Provides authentication and identity services for the other Access Manager components. The Access Gateways, J2EE Agents, and the SSL VPN server include an Embedded Service Provider that sets up a trusted relationship with the Identity Server.
- ◆ **Roles:** Provides RBAC (role-based access control) management. RBAC is used to provide a convenient way to assign a user to a particular job function or set of permissions within an enterprise, in order to control access. The Identity Server establishes the active set of roles for a user session each time the user is authenticated. Roles can be assigned to subsets of users based on constraints outlined in a role policy. The established role can then be used in authorization policies and J2EE permissions, to form the basis for granting and restricting access to particular Web resources.
- ◆ **Clustering:** Adds capacity and failover management. An Identity Server can be a member of a cluster of Identity Servers that is configured to act as a single server.

Access Gateways

An Access Gateway provides secure access to HTTP-based Web servers by hiding the IP addresses and DNS names of the Web servers. It provides the typical security services (authorization, single sign-on, and data encryption) previously provided by Novell iChain, and is integrated with the new identity and policy services of Access Manager.

An Access Gateway works with the Identity Server to enable existing Web services for the Liberty and SAML protocols. It provides single sign-on to Web servers through Identity Injection policies that supply required user information and Form Fill policies that automatically fill in requested form information. If your Web servers have not been configured to enforce authentication and authorization, you can configure an Access Gateway to provide these services. Authentication contracts and authorization policies can be assigned so that they protect the entire Web server, a single page, or somewhere in between.

An Access Gateway can also be configured so that it caches requested pages. When the user meets the authentication and authorization requirements, the user is sent the page from cache rather than requesting it from the Web server.

An Access Gateway can be installed as a soft appliance (includes both the operating system and the Access Gateway software) and as a service (includes just the Access Gateway software).

SSL VPNs

You install and configure the SSL VPN components when you need to protect non-HTTP and Java applications. The SSL VPN component provides secure access to such applications as an e-mail server, an FTP client, or Telnet service. SSL VPN is a Linux-based service, which can be installed in one of two ways:

- ♦ As a protected resource of an Access Gateway, which allows it to share session information with the Access Gateway.
- ♦ With an Embedded Service Provider, which allows it to set up a trusted relationship with the Identity Server.

The requests are delivered in the form of a servlet. An ActiveX plug-in or Java applet is delivered to the client on successful authentication. Roles and policies determine authorization decisions for back-end applications. Client integrity checking is available to ensure the existence of approved firewall and virus scanning software, before the SSL VPN session is established.

J2EE Agents

You install and configure the J2EE Agent components when you need to protect applications running on J2EE servers. Access Manager provides JBoss, WebLogic, and IBM WebSphere server agents for Java 2 Enterprise Edition (J2EE) application servers. These agents allow J2EE applications to leverage the product's authentication and authorization functionality without any code changes, as long as the applications rely on the J2EE application servers for authentication and authorization.

These agents leverage the Java Authentication and Authorization Service (JAAS) and Java Authorization Contract for Containers (JACC) standards for Access Manager-controlled authentication and authorization to Java Web applications and Enterprise JavaBeans. For more information about these Java authentication and authorization standards, see the [JAAS Authentication Tutorial \(http://java.sun.com/j2se/1.4.2/docs/guide/security/jaas/tutorials/GeneralAcnOnly.html\)](http://java.sun.com/j2se/1.4.2/docs/guide/security/jaas/tutorials/GeneralAcnOnly.html) and the [Java Authorization Contract for Containers \(http://java.sun.com/j2ee/javaacc/index.html\)](http://java.sun.com/j2ee/javaacc/index.html).

Like the Access Gateway, J2EE Agents are enabled for the Liberty Alliance and therefore operate as service provider agents. As such, they redirect all authentication requests to the Identity Server, which returns a SAML assertion to the component. This process has the added security benefit of removing the need to pass user credentials between the components to handle session management.

Policies

Policies provide the authorization component of Access Manager. The administrator of the Identity Server uses policies to define how properties of a user's authenticated identity map to the set of active roles for the user. This role definition serves as the starting point for role-based authorization policies of the Access Gateway and J2EE components. Additionally, authorization policies can be defined for the Access Gateway and J2EE components that control access to protected resources based on user and system attributes other than assigned roles.

The flexibility built into the policy component is nearly unlimited. You can, for example:

- ♦ Set up a URL-based policy that permits or denies users access to a protected Web site, depending on their roles, such as employee or manager.
- ♦ Specify whether an administrator has access to the policy management component of the Access Manager administration console. The administrator could create, edit, and manage policies that are assigned to specific components.

Each Access Gateway and J2EE component includes an Embedded Service Provider agent that interacts with the Identity Server to provide authentication, policy decision, and enforcement. For the Java application servers, the agent also provides role pass-through to allow integration with the Java Application server's authorization processes.

6.1.7 Viewing System Alerts

The System Alerts page displays how many unacknowledged alerts have been generated for all the devices imported into this Administration Console.

- 1 In the Administration Console, click *Access Manager > Dashboard > Alerts*.
- 2 To acknowledge and clear the alerts for a device, select the name of the server, then click *Acknowledge Alerts*.

The following columns display information about the alerts for each server.

Column	Description
<i>Server Name</i>	Specifies the name of the server receiving alerts. Click the server name to view more information about an alert before acknowledging it.
<i>Severe</i>	Indicates how many severe alerts have been sent to the server.
<i>Warning</i>	Indicates how many warning alerts have been sent to the server.
<i>Informational</i>	Indicates how many informational alerts have been sent to the server.

6.2 Stopping Tomcat on Windows

If you use the *Administrative Tools > Services* option on Windows to stop Tomcat, Tomcat does not shut down cleanly and displays an error. To continue, click *OK*.

The following command sometimes produces an error:

```
net stop Tomcat5
```

Whichever method you use to stop Tomcat, Tomcat is stopped. Wait a minute before restarting Tomcat.

6.3 Logging

You can troubleshoot by configuring component logging. In the Administration Console, click *Devices > Identity Server > Edit > Logging*.

For more information, see [Section 4.3, "Using the Log Files for Troubleshooting,"](#) on page 86.

6.4 Event Codes

A description of the Access Manager event codes is available in [Novell Access Manager 3.1 SP2 Event Codes](#).

6.5 Restoring a Failed Secondary Console

If a secondary console fails, you need to remove its configuration from the primary console before installing a new secondary console. As long as the failed console is part of the configuration, other Access Manager devices try to contact it.

- 1 On the primary console, click *Auditing > Troubleshooting*.
- 2 In the *Other Known Device Manager Servers* section, click the *Remove* button next to the secondary console that has failed.
- 3 Remove traces of the secondary console from the configuration datastore:
 - 3a In the iManager menu bar, select *View Objects*.



- 3b In the Tree view, select *novell*, and view the objects.
- 3c Delete all objects that reference the failed secondary console.

You should find the following types of objects:

- ♦ SAS Service object with the hostname of the secondary console
 - ♦ An object that starts with the last octet of the IP address of the secondary console
 - ♦ DNS AG object with the hostname of the secondary console
 - ♦ DNS IP object with the hostname of the secondary console
 - ♦ SSL CertificateDNS with the hostname of the secondary console
 - ♦ SSL CertificateIP with the hostname of the secondary console
- 4 Install a new secondary console. For installation instructions, see “[Installing Secondary Versions of the Administration Console](#)” in the *Novell Access Manager 3.1 SP2 Setup Guide*.

6.6 Moving the Primary Administration Console to New Hardware

If you do not have any secondary consoles:

- 1 Perform a backup. For instructions, see [Section 2.2, “Backing Up the Access Manager Configuration,”](#) on page 34.
- 2 Install the Administration Console on the new hardware, using the same DNS name and IP address.
- 3 Restore the configuration. For instructions, see [Section 2.3, “Restoring an Administration Console Configuration,”](#) on page 35.

If you have secondary consoles, you need to re-create the replica ring. When you install secondary consoles, they are added to the replica ring of the configuration datastore. The Access Manager backup script does not back up the replica ring information. It backs up only the Access Manager configuration information. The following instructions explain how you can re-create the replica ring when you install the primary Administration Console on new hardware.

- 1 Perform a backup. For instructions, see [Section 2.2, “Backing Up the Access Manager Configuration,” on page 34](#).
- 2 Down any secondary consoles.
- 3 Install the Administration Console on the new hardware, using the same DNS name and IP address.
- 4 Restore the configuration. For instructions, see [Section 2.3, “Restoring an Administration Console Configuration,” on page 35](#).
- 5 Remove any secondary consoles from the configuration:
 - 5a In the Administration Console, click *Auditing > Troubleshooting*.
 - 5b In the *Other Known Device Manager Servers* section, use the *Remove* button to remove any secondary consoles.
- 6 Uninstall the secondary consoles. For instructions, see “[Uninstalling the Administration Console](#)” in the *Novell Access Manager 3.1 SP2 Installation Guide*.
- 7 Reinstall the secondary consoles as secondary consoles to the new primary console.

6.7 Converting a Secondary Console into a Primary Console

In order for a secondary Administration Console to be converted into a primary Administration Console, a recent backup of the Administration Console must be available. For information on how to perform a backup, see [Section 2.2, “Backing Up the Access Manager Configuration,” on page 34](#). A backup is necessary in order to restore the certificate authority (CA).

If the failed server holds a master replica of any partition, you must use `ndsrepair` to designate a new master replica on a different server in the replica list.

WARNING: Perform these steps only if the primary Administration Console cannot be restored. If you have a recent backup, you can restore the primary Administration Console to new hardware. This is an easier configuration task than converting a secondary console into a primary console. See [Section 6.6, “Moving the Primary Administration Console to New Hardware,” on page 109](#)

This conversion includes the following tasks:

- ◆ [Section 6.7.1, “Shutting Down the Administration Console,” on page 111](#)
- ◆ [Section 6.7.2, “Changing the Master Replica,” on page 111](#)
- ◆ [Section 6.7.3, “Restoring CA Certificates,” on page 112](#)
- ◆ [Section 6.7.4, “Editing the vcdn.conf File,” on page 112](#)
- ◆ [Section 6.7.5, “Deleting Objects from the eDirectory Configuration Store,” on page 113](#)
- ◆ [Section 6.7.6, “Performing Component-Specific Procedures,” on page 113](#)
- ◆ [Section 6.7.7, “Enabling Backup on the New Primary Administration Console,” on page 121](#)

6.7.1 Shutting Down the Administration Console

If your primary Administration Console is running, you must log in as the administrator and shut down the service.

- ♦ **Linux:** Start YaST, click *System > System Services (Runlevel)*, then select to stop the `nds` service.
- ♦ **Windows:** Open the Control Panel, click *Administrative Tools > Services*, then select to stop the NDS Server.

6.7.2 Changing the Master Replica

Changing the master replica to reside on the new primary Administration Console makes this Administration Console into the certificate authority for Access Manager. You need to first designate the replica on the new primary Administration Console as the master replica. Then you need to remove the old primary Administration Console from the replica ring.

- ♦ [“Linux Secondary Administration Console” on page 111](#)
- ♦ [“Windows Secondary Administration Console” on page 111](#)

Linux Secondary Administration Console

- 1 At the secondary Administration Console, log in as `root`.
- 2 Change to the `/opt/novell/eDirectory/bin` directory.
- 3 Run `DSRepair` with the following options:

```
./ndsrepair -P -Ad
```
- 4 Select the one available replica.
- 5 Select *Designate this server as the new master replica*.
- 6 Run `ndsrepair -P -Ad` again.
- 7 Select the one available replica.
- 8 Select *View replica ring*.
- 9 Select the name of the failed primary server.
- 10 Select *Remove this server from replica ring*.
- 11 Enter the DN of the admin user in leading dot notation. For example:

```
.admin.novell
```
- 12 Continue with [Section 6.7.3, “Restoring CA Certificates,” on page 112](#).

Windows Secondary Administration Console

- 1 At the secondary Administration Console, log in as the administrator.
- 2 Change to the `C:\Novell\NDS` directory.
- 3 Start the `NDSCons.exe` program.
- 4 Select `dsrepair.dlm`.
- 5 In the *Parameters* box, specify `-A`, then click *Start*.
- 6 Click *Partitions > Root > Designate This Server As The New Master Replica*.

- 7 Open *Partitions > Root*, select the server, and verify that the replica is the master replica.
- 8 Run `ndsrepair` again with `-A` in the *Parameters* box.
- 9 Click *Partitions > Root*, then select the name of the failed primary server.
- 10 From the menu, click *Partitions > Replica Rings > Remove Server From Ring*.
- 11 Enter the DN of the admin user in leading dot notation. For example:
 `.admin.novell`
- 12 Continue with [Section 6.7.3, “Restoring CA Certificates,”](#) on page 112.

6.7.3 Restoring CA Certificates

The following steps are performed on the machine that you are promoting to be a primary console.

- 1 Copy your most recent Administration Console backup files to your new primary Administration Console.
- 2 Change to the backup `bin` directory:
 - Linux:** `/opt/novell/devman/bin`
 - Windows Server 2003:** `\Program Files\Novell\bin`
 - Windows Server 2008:** `\Program Files (x86)\Novell\bin`
- 3 Verify the IP address in the backup file.
 - 3a Open the backup file:
 - Linux:** `defbkparm.sh`
 - Windows:** `defbkparm.properties`
 - 3b Verify that the value in the `IP_Address` parameter is the IP address of your new primary console.
 - 3c Save the file
- 4 Run the certificate restore script:
 - Linux:** `sh aminst-certs.sh`
 - Windows:** `aminst-certs.bat`
- 5 When prompted, enter the location of the backup files.
- 6 Continue with [Section 6.7.4, “Editing the `vcdn.conf` File,”](#) on page 112.

6.7.4 Editing the `vcdn.conf` File

The `vcdn.conf` file contains the IP address of the failed primary Administration Console.

- 1 Change to the Administration Console configuration directory:
 - Linux:** `opt/novell/devman/share/conf`
 - Windows Server 2003:** `\Program Files\Novell\Tomcat\webapps\roma\WEB-INF\conf`
 - Windows Server 2008:** `\Program Files (x86)\Novell\Tomcat\webapps\roma\WEB-INF\conf`
- 2 Open the `vcdn.conf` file.

- 3 Search for all occurrences of the old IP address and replace them with the IP address of your new primary console.
- 4 Save the file.
- 5 Restart the Administration Console by entering the following command from the command line interface:


```
Linux: /etc/init.d/novell-tomcat5 restart
```

```
Windows: net stop Tomcat5
```

```
net start Tomcat5
```
- 6 Continue with [Section 6.7.5, “Deleting Objects from the eDirectory Configuration Store,”](#) on page 113.

6.7.5 Deleting Objects from the eDirectory Configuration Store

Several objects representing the failed primary Administration Console in the configuration store must be deleted.

- 1 Log in to the new Administration Console, then click *Auditing > Troubleshooting*.
- 2 In the *Other Known Device Manager Servers* section, select the old primary Administration Console, then click *Remove*.
- 3 Remove traces of the failed primary console from the configuration datastore:
 - 3a In the iManager menu bar, select *View Objects*.



- 3b In the Tree view, select *novell*, and view the objects.
- 3c Delete all objects that reference the failed primary console.

You should find the following types of objects:

 - ♦ SAS Service object with the hostname of the failed primary console
 - ♦ An object that starts with the last octet of the IP address of the failed primary console
 - ♦ DNS AG object with the hostname of the failed primary console
 - ♦ DNS IP object with the hostname of the failed primary console
 - ♦ SSL CertificateDNS with the hostname of the failed primary console
 - ♦ SSL CertificateIP with the hostname of the failed primary console
- 4 Continue with [Section 6.7.6, “Performing Component-Specific Procedures,”](#) on page 113.

6.7.6 Performing Component-Specific Procedures

If you have installed the following components, perform the cleanup steps for the component:

- ♦ [“Identity Server Installed with the Failed Primary Administration Console”](#) on page 114
- ♦ [“Third Administration Console”](#) on page 114
- ♦ [“Linux Access Gateway Appliances”](#) on page 115

- ♦ “Access Gateway Services” on page 117
- ♦ “Linux Identity Server” on page 118
- ♦ “Windows Identity Server” on page 118
- ♦ “Linux J2EE Agents” on page 119
- ♦ “Windows J2EE Agents” on page 119
- ♦ “SSL VPN” on page 120
- ♦ “Old Primary Administration Console” on page 121

Identity Server Installed with the Failed Primary Administration Console

If you had an Identity Server installed with your failed primary Administration Console, you need to clean up the configuration database to remove references to this Identity Server.

- 1 Log in to the Administration Console.
- 2 Remove the Identity Server:
 - 2a Click *Devices > Identity Servers*.
 - 2b Select the Identity Server that was installed with the primary Administration Console.
 - 2c Remove it from the cluster, then delete it.
- 3 Remove traces of the failed Identity Server from the configuration datastore:
 - 3a In the iManager menu bar, select *View Objects*.



- 3b In the Tree view, select *novell*, and view the objects.
- 3c Delete all objects that reference the failed Identity Server.

You should find the following types of objects:

- ♦ SAS Service object with the hostname of the failed Identity Server
- ♦ An object that starts with the last octet of the IP address of the failed Identity Server
- ♦ DNS AG object with the hostname of the failed Identity Server
- ♦ DNS IP object with the hostname of the failed Identity Server
- ♦ SSL CertificateDNS with the hostname of the failed Identity Server
- ♦ SSL CertificateIP with the hostname of the failed Identity Server

Third Administration Console

If you installed a third Administration Console used for failover, you must manually perform the following steps on that server:

- 1 Open the `vcdn.conf` file.
 - Linux:** `/opt/novell/devman/share/conf`
 - Windows Server 2003:** `\Program Files\Novell\Tomcat\webapps\roma\WEB-INF\conf`

Windows Server 2008: \Program Files (x86)\Novell\Tomcat\webapps\roma\WEB-INF\conf

- 2 In the file, look for the line that is similar to the following:

```
<vcdnPrimaryAddress>10.1.1.1</vcdnPrimaryAddress>
```

In this line, 10.1.1.1 represents the failed primary Administration Console IP address.

- 3 Change this IP address to the IP address of the new primary Administration Console.
- 4 Restart the Administration Console by entering the following command from the command line interface:

Linux: /etc/init.d/novell-tomcat5 restart

Windows: Use the following commands:

```
net stop Tomcat5
```

```
net start Tomcat5
```

Linux Access Gateway Appliances

For each Access Gateway Appliance imported into the Administration Console, you must edit the `config.xml` file and the `settings.properties` file on the Access Gateway and edit the current config and working config XML documents in the configuration store on the new primary Administration Console.

- 1 At the Access Gateway Appliance, log in as the `root` user.
- 2 Open a terminal window and shut down all services by entering the following commands:

```
/etc/init.d/novell-jcc stop  
/etc/init.d/novell-tomcat5 stop  
/etc/init.d/novell-vmc stop
```

- 3 If you are running SSL VPN, enter the following command to stop SSL VPN:

```
/etc/init.d/novell-sslvpn stop
```

- 4 Edit the `config.xml` file:

- 4a Enter:

```
vi /var/novell/cfgdb/.current/config.xml
```

- 4b Enter `/Remote`, then press Enter.

In the `IPv4Address` field, change the IP address from the failed Administration Console to the new primary Administration Console address.

- 4c (Conditional) If your audit server was on the primary Administration Console, enter

```
/NsureAuditSetting, then press Enter.
```

In the `IPv4Address` field, change the IP address from the failed Administration Console to the new primary Administration Console address.

- 4d Enter `:wq!` to save and exit.

- 5 Edit the `settings.properties` file:

- 5a Enter:

```
vi /opt/novell/devman/jcc/conf/settings.properties
```

- 5b** Change the IP address in the `remotemgmtip` list from the IP address of the failed Administration Console to the address of the new primary Administration Console.
 - 5c** Enter `:wq!` to save and exit.
- 6** At the new primary Administration Console, open an LDAP browser and edit the CurrentConfig object of the Access Gateway Appliance.

IMPORTANT: You should use an LDAP browser for the following steps, rather than iManager. Because iManager is slow at saving large files, your iManager connection might time out before your modifications are saved.

- 6a** Browse to the following container: `novell > accessManagerContainer > VCDN_Root > PartitionsContainer > Partition > AppliancesContainer`.
A list of devices appears. Access Gateways have an `ag` prefix.
 - 6b** Expand an Access Gateway container, then select the CurrentConfig object.
 - 6c** Select the `romaAGConfigurationXMLDoc` attribute and open it so you can view its value.
The value is a large XML file.
 - 6d** Copy the contents of the attribute to a text editor.
 - 6e** (Conditional) To verify which Access Gateway Appliance you are changing, search for the `<Local>` element.
The IP address should match the IP address of the Access Gateway Appliance that you are configuring for the new primary Administration Console.
 - 6f** Search for the `<Remote>` element.
 - 6g** Change the IP address of the `<Remote>` element so that it matches the IP address of the new primary Administration Console.
 - 6h** (Conditional) If your audit server was on the primary Administration Console, search for the `<NsureAuditSetting>` element.
Change the IP address of the `<NsureAuditSetting>` element so that it matches the IP address of the new primary Administration Console.
 - 6i** Copy the modified document in the text editor to the value field of the `romaAGConfigurationXMLDoc` attribute.
 - 6j** Save your changes.
- 7** At the new primary Administration Console, edit the WorkingConfig object of the Access Gateway Appliance:
Use an LDAP browser for these steps.
- 7a** Browse to the following container: `novell > accessManagerContainer > VCDN_Root > PartitionsContainer > Partition > AppliancesContainer`.
A list of devices appears. Expand the Access Gateway container.
 - 7b** Select the WorkingConfig object.
 - 7c** Select the `romaAGConfigurationXMLDoc` attribute and open it so you can view its value.
 - 7d** Copy the contents of the attribute to a text editor.
 - 7e** Search for the `<Remote>` element.
 - 7f** Change the IP address of the `<Remote>` element so that it matches the IP address of the new primary Administration Console.

- 7g** (Conditional) If your audit server was on the primary Administration Console, search for the `<NsureAuditSetting>` element.
- Change the IP address of the `<NsureAuditSetting>` element so that it matches the IP address of the new primary Administration Console.
- 7h** Copy the modified document in the text editor to the value field of the `romaAGConfigurationXMLDoc` attribute.
- 7i** Save your changes.
- 8** At the Access Gateway Appliance, start all services by entering the following commands:
- ```
/etc/init.d/novell-jcc start
/etc/init.d/novell-tomcat5 start
/etc/init.d/novell-vmc start
/etc/init.d/novell-sslvpn start
```
- 9** (Conditional) Repeat this process for each Linux Access Gateway that has been imported into the Administration Console.

## Access Gateway Services

For each Access Gateway Service imported into the Administration Console, you must edit the `config.xml` file and the `settings.properties` file on the Access Gateway.

- 1** At the Access Gateway Service, log in as the `root` or the Administrator user.
- 2** Shut down all Access Gateway services.
- Linux:** Enter the following commands:
- ```
/etc/init.d/novell-jcc stop
/etc/init.d/novell-tomcat5 stop
/etc/init.d/novell-apache2 stop
```
- Windows:** Click *Control Panel > Administrative Tools > Services*, then stop the following services:
- ```
Apache Tomcat
JCCServer
```
- Stopping Apache Tomcat causes Apache 2.2 to also stop.
- 3** (Conditional) If your audit server was on the primary Administration Console, edit the `config.xml` file:
- 3a** Change to the directory and open the file.
- Linux:** `/var/opt/novell/tomcat5/webapps/agm/WEB-INF/config/current`
- Windows:** `\Program Files\Novell\Tomcat\webapps\agm\ WEB-INF\config\current`
- 3b** Find the `NsureAuditSetting` entry.
- In the `IPv4Address` field, change the IP address from the failed Administration Console to the new primary Administration Console address.
- 3c** Save and exit.
- 4** Edit the `settings.properties` file:
- 4a** Change to the directory and open the file.
- Linux:** `/opt/novell/devman/jcc/conf`
- Windows:** `\Program Files\Novell\devman\jcc\conf`

- 4b** Change the IP address in the `remotemgmtip` list from the IP address of the failed Administration Console to the address of the new primary Administration Console.
  - 4c** Save and exit.
  - 5** At the Access Gateway Service, start all services by entering the following commands:
    - Linux:** Enter the following commands:
 

```
/etc/init.d/novell-jcc start
/etc/init.d/novell-tomcat5 start
/etc/init.d/novell-apache2 start
```
    - Windows:** Click *Control Panel > Administrative Tools > Services*, then start the following services:
 

```
Apache Tomcat
JCCServer
```
- Starting Apache Tomcat causes Apache 2.2 to also start.
- 6** (Conditional) Repeat this process for each Access Gateway Service that has been imported into the Administration Console.

## Linux Identity Server

For each Linux Identity Server imported into the Administration Console, perform the following steps:

- 1** Log in as the `root` user.
- 2** Open a terminal window and shut down all services by entering the following commands:
 

```
/etc/init.d/novell-jcc stop
/etc/init.d/novell-tomcat5 stop
```
- 3** Edit the `settings.properties` file:
  - 3a** Enter:
 

```
vi /opt/novell/devman/jcc/conf/settings.properties
```
  - 3b** Change the IP address in the `remotemgmtip` list from the IP address of the failed Administration Console to the address of the new primary Administration Console.
  - 3c** Enter `:wq!` to save and exit.
- 4** Start the services by entering the following commands:
 

```
/etc/init.d/novell-jcc start
/etc/init.d/novell-tomcat5 start
```

## Windows Identity Server

For each Windows Identity Server imported into the Administration Console, perform the following steps:

- 1** Open a terminal window and shut down all services by entering the following commands:
 

```
net stop JCCServer
net stop Tomcat5
```
- 2** Edit the `settings.properties` file:
  - 2a** Change to the following directory:
 

```
Windows Server 2003: \Program Files\Novell\devman\jcc\conf
```

**Windows Server 2008:** \Program Files (x86)\Novell\devman\jcc\conf

- 2b** Open the `settings.properties` file.
- 2c** Change the IP address in the `remotemgmtip` list from the IP address of the failed Administration Console to the address of the new primary Administration Console.
- 2d** Save your changes.
- 3** Start the services by entering the following commands:

```
net start JCCServer
net start Tomcat5
```

## Linux J2EE Agents

For each Linux J2EE agent imported into the Administration Console, perform the following steps:

- 1** Log in as the `root` user.
- 2** Open a terminal window and shut down all services by entering

```
/etc/init.d/novell-jcc stop
```
- 3** Edit the `settings.properties` file:
  - 3a** Enter:

```
vi /opt/novell/devman/jcc/conf/settings.properties
```
  - 3b** Change the IP address in the `remotemgmtip` list from the IP address of the failed Administration Console to the address of the new primary Administration Console.
  - 3c** Enter `:wq!` to save and exit.
- 4** Start the services by entering

```
/etc/init.d/novell-jcc start
```

## Windows J2EE Agents

For each Windows J2EE agent imported into the Administration Console, you must perform the following steps:

- 1** Log in as a user with administration rights.
- 2** In the Control Panel, click *Administrative Tools > Services*.
- 3** Select the JCCServer, then click *Stop*.
- 4** In a text editor, open the `settings.properties` file in the JCC configuration directory:

**Windows Server 2003:** \Program Files\Novell\devman\jcc\conf  
**Windows Server 2008:** \Program Files (x86)\Novell\devman\jcc\conf
- 5** Change the IP address in the `remotemgmtip` list from the IP address of the failed Administration Console to the address of the new primary Administration Console.
- 6** Save your changes and exit.
- 7** In the Control Panel, click *Administrative Tools > Services*.
- 8** Select the JCCServer, then click *Start*.

## SSL VPN

For each SSL VPN component imported into the Administration Console, you must edit the `config.xml` file and the `settings.properties` file on the SSL VPN server and edit the current config and working config XML documents in the configuration store on the new primary Administration Console.

- 1 At the SSL VPN machine, log in as the `root` user.
- 2 Open a terminal window and shut down all services by entering the following commands:

```
/etc/init.d/novell-jcc stop
/etc/init.d/novell-tomcat5 stop
/etc/init.d/novell-sslvpn stop
```

- 3 Edit the `config.xml` file:

**3a** Enter:

```
vi /etc/opt/novell/sslvpn/config.xml
```

**3b** Enter `/DeviceManagerAddress`, then press `Enter`.

**3c** Change the IP address to that of the new primary Administration Console.

**3d** Enter `:wq!` to save and exit.

- 4 Edit the `settings.properties` file:

**4a** Enter:

```
vi /opt/novell/devman/jcc/conf/settings.properties
```

**4b** Change the IP address in the `remotemgmtip` list from the IP address of the failed Administration Console to the address of the new primary Administration Console.

**4c** Enter `:wq!` to save and exit.

- 5 At the new primary Administration Console, open an LDAP browser and edit the `CurrentConfig` object of the SSL VPN.

---

**IMPORTANT:** You should use an LDAP browser for the following steps, rather than `iManager`. `iManager` is slow at saving large files, and your `iManager` connection might time out before your modifications are saved.

---

**5a** Browse to the following container: `novell > accessManagerContainer > VCDN_Root > PartitionsContainer > Partition > AppliancesContainer`.

A list of devices appears. SSL VPN devices have an `sslvpn` prefix.

**5b** Expand an SSL VPN container, then select the `CurrentConfig` object.

**5c** Select the `romaSSLVPNConfigurationXMLDoc` attribute and open it.

**5d** Copy the contents of the attribute to a text editor.

**5e** Search for the `<DeviceManagerAddress>` element.

**5f** Change the IP address of the `<DeviceManagerAddress>` element so that it matches the IP address of the new primary Administration Console.

**5g** Copy the modified document in the text editor to the value field of the `romaSSLVPNConfigurationXMLDoc` attribute.

**5h** Save your changes.

- 6 At the new primary Administration Console, edit the `WorkingConfig` object of the SSL VPN container:

Use an LDAP browser for these steps.

- 6a** Browse to the SSL VPN object by expanding the following containers: novell > accessManagerContainer > VCDN\_Root > PartitionsContainer > Partition > AppliancesContainer.  
A list of devices appears.
  - 6b** Expand the SSL VPN container, then select the WorkingConfig object.
  - 6c** Select the romaSSLVPNConfigurationXMLDoc attribute and open it.
  - 6d** Copy the contents of the attribute to a text editor.
  - 6e** Search for the <DeviceManagerAddress> element.
  - 6f** Change the IP address of the <DeviceManagerAddress> element so that it matches the IP address of the new primary Administration Console.
  - 6g** Copy the modified document in the text editor to the value field of the romaSSLVPNConfigurationXMLDoc attribute.
  - 6h** Save your changes.
- 7** At the SSL VPN machine, start all services by entering the following commands:
- ```
/etc/init.d/novell-jcc start
/etc/init.d/novell-tomcat5 start
/etc/init.d/novell-sslvpn start
```
- 8** (Conditional) If the SSL VPN server is still not functioning, restart the Linux server by entering `reboot`.
 - 9** (Conditional) Repeat this process for each SSL VPN server that has been imported into the Administration Console.

Old Primary Administration Console

After the secondary console has been promoted to be the primary console, uninstall the Administration Console software of the old primary Administration Console. Before uninstalling, make sure the machine is disconnected from the network. For instructions, see “[Uninstalling the Administration Console](#)” in the *Novell Access Manager 3.1 SP2 Installation Guide*.

If you want to use the old primary console as a secondary console, you need to first uninstall the Administration Console software. Connect the machine to the network, then reinstall the software, designating this console as a secondary console.

6.7.7 Enabling Backup on the New Primary Administration Console

If you installed your Administration Consoles using the 3.1 version of Access Manager, the backup utility is properly configured.

If you have upgraded the Linux Administration Consoles from 3.0 SP4 to 3.1, you need to modify the `defbkparm.sh` file before performing a backup.

- 1** On the new primary Administration Console, change to the `/opt/novell/devman/bin` directory.
- 2** Open the `defbkparm.sh` file and find the following lines:

```
EDIR TREE=<tree_name>
EDIR CA=<CA name>
```

These lines contain values using the hostname of the Administration Console you are on.

- 3 Modify these lines to use the hostname of the failed Administration Console.

When you install the primary Administration Console, the EDIR TREE parameter is set to the hostname of the server with `_tree` appended to it. The EDIR CA parameter is set to the hostname of the server with `_tree CA` appended to it.

If the failed Administration Console had `amlab` as its hostname, you would change these lines to have the following values:

```
EDIR TREE="amlab_tree"
EDIR CA="amlab_tree CA"
```

- 4 Save your changes.
- 5 Make a backup from your new primary Administration Console.

WARNING: After configuring the secondary console to be the new primary console and performing all the cleanup steps, you cannot restore an old backup from the primary console.

Make a new backup as soon as your new primary console is functional.

6.8 Orphaned Objects in the Trust/Configuration Store

If you delete a User object in LDAP, the objects in the trust/configuration datastore related to that user can become orphaned. The system uses these objects for federated identity and user profiles. Currently, there are no known issues related to orphaned identity objects, but they might affect system performance. Orphaned user profile objects might also affect user lookup operations, and therefore you should remove them.

To do so, you first delete the user's profile before you delete a User object, as described in the following steps:

- 1 In iManager or an LDAP browser, edit the attributes of the User object that you are going to delete.
- 2 Note the value of the User object's `GUID` attribute (for eDirectory), `objectGUID` attribute (for Active Directory), or the `nsuniqueid` attribute (for Sun One).
- 3 In the Access Manager trust/configuration datastore, locate any containers that use the following naming patterns:

```
cn=LUP*,cn=SCC*,cn=cluster,cn=nids,ou=accessManagerContainer,o=novell,cn=LibertyUserProfiles*,cn=SCC*,cn=cluster,cn=nids,ou=accessManagerContainer,o=novell.
```
- 4 Look for a child profile object inside of these containers that is named by using the GUID noted in [Step 2](#). There should only be one profile object for each GUID.
- 5 Delete that child profile object.
- 6 Repeat these steps for each User object that you want to delete.
- 7 Delete the User objects.

6.9 Repairing the Configuration Datastore

The configuration datastore is an embedded version of eDirectory 8.8. If it becomes corrupted, you can run DSRepair to fix it. Or, you can restore a recent backup. To restore a backup, see [Section 2.3, “Restoring an Administration Console Configuration,”](#) on page 35.

To run DSRepair:

- 1 In a browser, enter the following URL.

```
http://<ip_address>:8028/nds
```

Replace *<ip_address>* with the IP address of your Administration Console.

- 2 At the login prompt, enter the username and password of the admin user for the Administration Console.

The NDS iMonitor application is launched. For more information, see [Accessing iMonitor \(http://www.novell.com/documentation/edir88/edir88/data/a6160f7.html\)](#).

- 3 In the *View* bar, select the *Repair* icon.

For more information about DSRepair, see the following:

- Click the *Help* icon.
- [Using NdsRepair \(http://www.novell.com/documentation/edir88/edir88tshoot/data/bq0gv5l.html\)](#)

6.10 Session Conflicts

Do not use two instances of the same browser to simultaneously access the same Administration Console. Browser sessions share settings, which can result in problems when you apply changes to configuration settings. However, you can use two different brands of browsers simultaneously, such as Internet Explorer and Firefox, which makes it possible to avoid the session conflicts.

6.11 Unable to Log In to the Administration Console

If you experience problems logging in to the Administration Console, you might need to restart Tomcat.

- 1 In a terminal window on the console machine, restart Tomcat:

Linux: Enter the following command:

```
/etc/init.d/novell-tomcat5 restart
```

Windows: Enter the following commands:

```
net stop Tomcat5
```

```
net start Tomcat5
```

- 2 If this does not solve the problem, check the log file:

Linux: `/var/opt/novell/tomcat5/logs/catalina.out`

Windows Server 2003: `\Program Files\Novell\Tomcat\logs\stdout.log`

Windows Server 2008: `\Program Files (x86)\Novell\Tomcat\logs\stdout.log`

- 3 Check for the following error:

Error Starting up core services.
Application manager is Shutting down the Device Manager suite.
Shutting down Device Manager suite.

4 (Linux) If you see this error, check the status of eDirectory:

4a Enter the following command:

```
/etc/init.d/ndsd status
```

If the status command returns nothing, you need to manually start eDirectory

4b Enter the following command:

```
/etc/init.d/ndsd start
```

4c Restart Tomcat.

5 (Windows) If you see this error, check the status of eDirectory:

5a Enter the following command:

```
net start "nds server0"
```

If the service has been started, this command returns a message that the service has been started. If the service has been stopped, it starts eDirectory.

5b Verify that the agent is running. Click *Control Panel > Novell eDirectory Services*, then verify that the *Server* box does not contain an agent closed message.

5c If the agent is closed, run `dsrepair`.

5d Restart Tomcat.

6.12 (Linux) Exception Processing IdentityService_ServerPage.JSP

If you see the message `Exception processing IdentityService_ServerPage.jsp` on a Linux Administration Console, it is an indication that the system has run out of available file handles. You need to use the command line to increase the `ulimit` value (`ulimit -n [new limit]`), which sets the number of open file descriptors allowed.

To set this value permanently, you can create the `/etc/profile.local` file with the `ulimit` value, such as:

```
ulimit -n 4096
```

You can make changes to `/etc/security/limits.conf` file with a line just to change the limit for a specific user, in this case the `novlwww`. You do this by adding the following line:

```
novlwww soft nofile [new limit]
```

6.13 Backup/Restore Failure Because of Special Characters in Passwords

Administration passwords with special characters such as dollar signs might cause the `ambkup` utility to fail. The `ambkup` utility creates a command line for the `ICE` utility, and the special characters might be interpreted by it. If you must use special characters, and this issue arises, modify the `defbkparm` file so that the special characters are escaped.

For example, if the administrator's password is mi\$\$le, then the field DS_ADMIN_PWD should be mi\\$\\$le.

This file is located in the following directory:

Linux: /opt/novell/devman/bin/defbkparm.sh

Windows Server 2003: \Program Files\Novell\bin\defbkparm.properties

Windows Server 2008: \Program Files (x86)\Novell\bin\defbkparm.properties

6.14 Unable to Install NMAS SAML Method

When you try to create an Identity Server cluster configuration with an eDirectory user store and with the *Install NMAS SAML method* option enabled and you have not installed the dependent packages, the following error message is displayed:

```
Warning: Failed to create SAML Affiliate Object  
com.novell.security.japi.nmas.LoginMethodModel.getLsmWINNNTStatus() I
```

One of the installation requirements for the Linux Administration Console is to install the compat and the libstdc++ packages. On SLES 10, these are separate packages. On SLES 11, the compat package contains the libstdc++ library. The Identity Server also requires the compat package. For more information on installing these packages, see [TID 7004701: iManager: Certificate Server Plugin Errors](http://www.novell.com/support/php/search.do?cmd=displayKC&docType=kc&externalId=7004701&sliceId=1&docTypeID=DT_TID_1_1&dialogID=68926420&stateId=0%20%20130264119) (http://www.novell.com/support/php/search.do?cmd=displayKC&docType=kc&externalId=7004701&sliceId=1&docTypeID=DT_TID_1_1&dialogID=68926420&stateId=0%20%20130264119).

Troubleshooting Certificate Issues

7

- ♦ [Section 7.1, “Resolving Certificate Import Issues,” on page 127](#)
- ♦ [Section 7.2, “Mutual SSL with X.509 Produces Untrusted Chain Messages,” on page 129](#)
- ♦ [Section 7.3, “Troubleshooting Options for Certificate Problems,” on page 129](#)
- ♦ [Section 7.4, “Can’t Log In with Certificate Error Messages,” on page 131](#)
- ♦ [Section 7.5, “When a User Accesses a Resource, the Browser Displays Certificate Errors,” on page 131](#)
- ♦ [Section 7.6, “Access Gateway Canceled Certificate Modifications,” on page 131](#)

7.1 Resolving Certificate Import Issues

Use the following sections to resolve issues created when a full certificate chain is not imported into Access Manager:

- ♦ [Section 7.1.1, “Importing an External Certificate Key Pair,” on page 127](#)
- ♦ [Section 7.1.2, “Resolving a -1226 PKI Error,” on page 128](#)
- ♦ [Section 7.1.3, “When the Full Certificate Chain Is Not Returned During an Automatic Import of the Trusted Root,” on page 128](#)
- ♦ [Section 7.1.4, “Using Internet Explorer to Add a Trusted Root Chain,” on page 129](#)

7.1.1 Importing an External Certificate Key Pair

The Access Manager Certificate Authority requires that all certificate key pairs in `.pfx` format contain the complete certificate chain. If a key pair was created with multiple CAs and the exported certificate does not contain the complete certificate chain, the file cannot be imported into Access Manager. When you try to import such a certificate, the following error message is displayed:

```
"Error importing certificate key pair: Error: Error: -1403
```

When exporting the certificate key pair, make sure you include all the certificates in the certification path.

To ensure that your certificate contains all the intermediate certificates and contains them in the right order, import the certificate into Internet Explorer or Firefox.

- ♦ For Internet Explorer, click *Tools > Internet Options > Content > Certificates > Personal > Import*.
- ♦ For Firefox, click *Tools > Options > Advanced > Encryption > View Certificates > Your Certificates > Import*.

Make sure the browser contains the public key for all the intermediate CAs. Then select the certificate and export the certificate in `.pfx` format. In Internet Explorer, you must select to include all the certificates in the chain. In Firefox, all the certificates in the chain are automatically included.

If you receive an error when importing the certificate, the error comes from either NCI or PKI. For a description of these error codes, see [Novell Certificate Server Error Codes and Novell International Cryptographic Infrastructure \(http://www.novell.com/documentation/nwec/index.html\)](http://www.novell.com/documentation/nwec/index.html).

7.1.2 Resolving a -1226 PKI Error

When you create a certificate signing request, send it to a third-party issuer to be signed, and receive the server certificate from the third-party issuer, you sometimes receive a -1226 error when you try to import the signed certificate. You receive this error when the issuer does not send back the trusted roots required to validate the issuer of the server certificate.

Use one of the following options to resolve this issue:

- ◆ If the issuer included the trusted root and any intermediate certificates in a separate file or files, specify these files during the import by clicking the + character that allows you to add a trusted root or an intermediate certificate.
- ◆ If the issuer did not send you any additional files, you can go to the issuer's Web site, download them, then specify these files during the import by clicking the + character that allows you to add a trusted root or an intermediate certificate.
- ◆ You can try importing the certificate into Internet Explorer, which has the trusted roots from all major CAs, then export the certificate with the required chain of trusted roots. See [Section 7.1.4, "Using Internet Explorer to Add a Trusted Root Chain," on page 129](#).

7.1.3 When the Full Certificate Chain Is Not Returned During an Automatic Import of the Trusted Root

Access Manager allows you to automatically import the trusted root under the following conditions:

- ◆ When enabling SSL communication between the Access Gateway and the Web server, you can automatically import the root CA from the Web server.
- ◆ When setting up the user stores for the Identity Server and adding the server replicas, you can automatically import the root CA of the LDAP server.

If there are multiple certificates in the chain, sometimes the server does not send all the certificates in the chain. When this happens, the following message is displayed:

```
The root CA certificate was not returned by the server. It might be necessary to manually import the root CA certificate and possible intermediate CA certificates in order to complete the chain.
```

To correct this problem, you need to manually import the missing entries. The easiest method to obtain all the certificates in the chain, including the root CA, is to import the server certificate into Internet Explorer, then export the chain and import it into Access Manager. If Access Manager already has some of the certificates, it skips their import and imports only the missing certificates.

For instructions on this process, see [Section 7.1.4, "Using Internet Explorer to Add a Trusted Root Chain," on page 129](#).

7.1.4 Using Internet Explorer to Add a Trusted Root Chain

The following procedure works only when Internet Explorer contains the trusted root certificate of the issuer of your certificate.

- 1 In Internet Explorer, click *Tools > Internet Options > Content > Certificates*.
- 2 Click *Import* and import your server certificate into the *Other People* tab.
- 3 Click *Other People*, then double-click your certificate.
- 4 Click *Certification Path*.
 - ♦ If the *Certification Path* shows that the certificate is OK, you now have the full certificate chain available for export. Click *OK*, then continue with [Step 5](#).
 - ♦ If the *Certification Path* is not OK, you cannot use this method. Click *OK*, then contact your issuer for the certificate chain.
- 5 Select the certificate, then click *Export > Next*.
- 6 Select *Cryptographic Message Syntax Standard - PKCS #7 Certificates (.P7B)* as the format and select *Include all certificates in the certification path if possible* to include the certificate chain.
- 7 Click *Next*, then specify a filename and path for the file.
- 8 Click *Next > Finish*.
- 9 Use this P7B file to import your server certificate into Access Manager.

7.2 Mutual SSL with X.509 Produces Untrusted Chain Messages

When you set up an X.509 contract for mutual SSL authentication, you must ensure that the Identity Server trust store (NIDP-truststore) contains the trusted root from each CA that has signed the client certificates. If a client has a certificate signed by a CA that is not in the Identity Server Trust Store, authentication fails.

To add a certificate to the Identity Server Trust Store:

- 1 In the Administration Console, click *Devices > Identity Servers > Edit > Security > NIDP Trust Store*.
- 2 Click either *Add* or *Auto-Import From Server* and follow the prompts.

7.3 Troubleshooting Options for Certificate Problems

If certificate problems are reported after upgrading or replacing certificates or after restoring a device, use the following options to solve the problem:

- ♦ [“Verifying Certificates after Upgrading or Restoring” on page 130](#)
- ♦ [“Validating Trusted Root Certificates after Replacing Certificates” on page 130](#)

Verifying Certificates after Upgrading or Restoring

Certificate commands are generated when you upgrade the Administration Console and when you restore a device. You should ensure that they have completed successfully.

- 1** In the Administration Console, click *Security > Command Status* to determine whether a certificate command has failed.
- 2** Note the destination trust store or keystore of any failed command.
- 3** Click *Auditing > Troubleshooting > Certificates*.
The Certificates page displays all the keystores and trust stores configured for Access Manager.
- 4** Select the store, then click *Re-push certificates*.
This pushes all assigned certificates to the store. You can re-push certificates multiple times without causing any problems.

Validating Trusted Root Certificates after Replacing Certificates

When you replace certificates, you should validate that the Identity Server configuration is storing a valid trusted root for the Access Gateways or SSL VPN servers that are using the Identity Server for authentication. You should also validate that the Access Gateway cluster and SSL VPN cluster are storing a valid trusted root for the Identity Server.

You cannot use the following process to validate that the Identity Server and J2EE agents are storing valid trusted roots for each other.

To validated the availability of required trusted root certificates:

- 1** In the Administration Console, click *Auditing > Troubleshooting > Certificates*.
The Certificates page displays all the keystores and trust stores configured for Access Manager.
- 2** Validate the trusted root certificates of the Identity Server configuration:
 - 2a** Select one of the following keystores that belong to the NIDP Configuration device:
 - ◆ Signing
 - ◆ Encryption
 - ◆ SSL Connector
 - ◆ Provider Introductions SSL Connector
 - ◆ Consumer Introductions SSL Connector
 - 2b** Click *Validate trusted root*.
 - 2c** If an error is reported, add the missing trusted root to a trust store.
To identify the trust store, check the ESP Trust Store of the devices that are using the Identity Server for authentication. For instructions, see the following sections:
[Section 3.4.5, “Viewing Trust Store Details,” on page 69](#)
[Section 3.4.2, “Adding Trusted Roots to Trust Stores,” on page 68](#)
 - 2d** Repeat [Step 2a](#) and [Step 2c](#) for each keystore that you want to validate.
- 3** Validate the trusted root certificates of the Access Gateway cluster or the SSL VPN cluster:
 - 3a** Select one of the following keystores that belong to the cluster:
 - ◆ Signing

- ◆ Encryption
- ◆ ESP Mutual SSL

3b Click *Validate trusted root*.

3c If an error is reported, add the missing trusted root to the Trust Store of the Identity Server. For instructions, see [Section 3.4.2, “Adding Trusted Roots to Trust Stores,” on page 68](#).

3d Repeat [Step 3a](#) and [Step 3c](#) for each keystore that you want to validate.

7.4 Can’t Log In with Certificate Error Messages

After an upgrade if your users can’t log in to access protected resources, and the failure messages contain certificate error messages, you might need to manually push the certificates from the Administration Console to the Access Gateway.

To re-push a certificate:

- ◆ For a reverse proxy certificate, go to the Reverse Proxy page, select a different certificate, click *OK*, return to the Reverse Proxy page, select the correct certificate, then click *OK*.
- ◆ For a Web server certificate, go to the Web Server page, select a different SSL mutual certificate, click *OK*, return to the Web Server page, select the correct certificate, click *OK*, then apply the changes.

7.5 When a User Accesses a Resource, the Browser Displays Certificate Errors

When you configure the Identity Server to use SSL (the HTTPS protocol), the browser must be configured to trust the CA that created the certificate for the Identity Server. If you use a well-known CA, the browser is usually already configured to trust certificates from the CA. If you use a less-known CA or the Access Manager CA to create the certificate, you need to import the public key of the trusted root certificate into the browsers to establish the trust. For the Access Manager CA, this certificate is called configCA.

For instructions on how to export the public key of a trusted root certificate, see [“Viewing Trusted Root Details” on page 70](#).

To import a public key into the browser, access the certificate options, then follow the prompts:

- ◆ For Internet Explorer, click *Tools > Internet Options > Content > Certificates > Trusted Root Certification Authorities > Import*.
- ◆ For Firefox, click *Tools > Options > Advanced > Encryption > View Certificates > Authorities > Import*.

7.6 Access Gateway Canceled Certificate Modifications

An Access Gateway has the following issue when canceling changes to certificate modifications:

If you make certificate changes on the Reverse Proxy or the Web Servers page, click the *Configuration Panel* link, and then cancel the changes on the Configuration page, the Reverse Proxy is configured with an invalid certificate.

To correct the problem, return to the page and select the old certificate. As soon as you exit the page, the certificate is pushed to the device. Because you did not change the certificate, you do not need to restart the Embedded Service Provider.

Certificates Terminology

A

Access Manager uses certificates to provide secure communication between devices, encrypt sensitive information, facilitate single sign-on, and to verify that the user sending the message is who he or she claims to be. The following is a list of certificate terminology used in Access Manager:

certificate authority (CA): An entity that issues digital certificates attesting to the authenticity of the information in the certificate.

certificate: Information attached to an electronic message. It is used to verify that the sender is who he or she claims to be. A certificate is signed. The signer of the certificate (a CA), if trusted, verifies the accuracy of the information in the certificate.

certificate chain: In addition to identifying a user, server, or computer, certificates can validate the identity and trustworthiness of other certificates. A certificate that asserts an identity is signed by a certificate that trusts the contents of the certificate it is signing. The signing certificate in turn can be signed by another certificate, which can be signed by another certificate, and so forth, thus forming a certificate chain. The last certificate in the certificate chain is referred to as the root certificate and is a self-signed certificate.

When a certificate or certificate chain is sent from one computer to another, the receiving computer examines the certificate chain to determine if it can be trusted. To verify certificate trust in a chain, the receiving computer examines its own configuration store to see if it contains a CA certificate that matches the root certificate of the certificate chain. If so, the receiver compares its copy of the certificate with the chain's root certificate to verify its authenticity.

certificate signing request (CSR): Requesting a signed certificate is accomplished by sending a CSR to the CA. A CSR is created with information about the person or organization that desires the signed certificate. A public key is also generated and included in the CSR. A private key is also generated, but not included in the CSR.

When the CA receives the CSR, the CA uses it in combination with the CA's guidelines and practices to establish that the person or organization represented by the CSR is properly identified and authorized as the owner of the information in CSR. The CA creates and signs a certificate that the requesting person or organization can use. The signature of the CA in the certificate identifies that the entity is who it claims to be. The signed certificate is delivered to its owner, who adds it to the keystore (usually the same keystore where the private key created with the original CSR resides).

issuer: The CA that issues a certificate.

intermediate certificate: A subordinate certificate issued by the trusted root specifically for end-entity server certificates. The result is a certificate chain that begins at the trusted root CA, proceeds through the intermediate certificate, and ends with the SSL certificate issued to you. Using intermediate certificates adds more levels of security, but does not cause performance, installation, or compatibility issues.

key: A string or variable value used for encrypting and decrypting information.

key pair: Public and private keys generated by a cryptography system and used in combination with each other.

keystore: A storage file containing keys, certificates, and trusted roots. Access Manager agents can access keystores to retrieve certificates, keys, and trusted roots as needed.

local CA: The CA of the administration console's instance of eDirectory. Also known as the Organizational CA.

private key: The unpublished key in a security system that uses two keys. It is used for authentication, data encryption/decryption, digital signing, and secure e-mail. One of the most common uses is sending and receiving digitally signed and encrypted e-mail by using the S/MIME standard.

The public and private keys have the following relationships:

- ◆ Data encrypted with the public key can be decrypted with the private key only.
- ◆ Data signed with the private key can be verified with the public key only.
- ◆ Exposing a public key does not expose the corresponding private key.

public key: The publicly distributed key in a security system that uses two keys.

root CA: The issuing authority for the root certificate.

root certificate: The last certificate in a certificate chain.

self-signed certificate: A certificate whose issuer is itself.

SSL connections: When two computers connect and need to establish trust and a secure connection, certificates are exchanged and an encryption algorithm is established. Public keys shared in the exchanged certificates, as well as the associated private keys (which are not exchanged) are used as part of the encryption algorithm. After security is established, a secure SSL session is established and the two computers are able to communicate securely.

trusted certificate: The certificate of a known CA. These certificates are self-signed and are recognized as representing a CA that is trusted.

trusted root: The same as a trusted certificate. A trusted root provides the basis for trust in public key cryptography. Trusted roots enable security for SSL, secure e-mail, and certificate-based authentication. These certificates are for root CAs, so they are called "trusted roots."

trust store: A keystore containing only trusted roots. Intermediate CAs and end entity public certificates can be part of a trust store.

Troubleshooting XML Validation Errors on the Access Gateway Appliance

An XML validation error is often ignored because the returning message does not appear to be serious. However, closer inspection of the Access Gateway Appliance shows that none of the changes have been applied. When a change is applied by using the UI, the system writes the configuration to the configuration store on the Administration Console, as well as to the `/var/novell/cfgdb/vcdn/config.xml` file on the Access Gateway Appliance. If this file passes the schema checks on the Access Gateway Appliance, the `/var/novell/cfgdb/.current/config.xml` file is updated with the configuration.

This is the file that the Access Gateway Appliance reads when it loads or refreshes. If the `config.xml` file from `/var/novell/cfgdb/vcdn/` and `/var/novell/cfgdb/.current/` are not in sync, then all changes you defined have not been applied to the Access Gateway Appliance.

You need to pay attention to XML validation errors and identify the key steps required to solve such problems. There are two main scenarios that are discussed in this section:

- [Section B.1, “Modifying a Configuration That References a Removed Object,” on page 135](#)
- [Section B.2, “Configuration UI Writes Incorrect Information to the Local Configuration Store,” on page 137](#)

B.1 Modifying a Configuration That References a Removed Object

One scenario that causes XML validation errors occurs when a configuration references an object that has been removed. For example, a custom authentication contract was created and assigned to a protected resource. The contract was manually deleted from the Identity Server configuration, but the Access Gateway protected resource still references it, even though it is not displayed in the user interface. After you identify the missing link, you can use the Access Manager interface to work around the problem.

To troubleshoot this scenario:

- 1 Search the `/opt/novell/devman/share/logs/app_sc.0.log` file on the Administration Console server for #200904025: Error - XML VALIDATION FAILED.

After you find the entry, work backwards to identify the start of the Java exception. Locate the problem strings or entry from the configuration, such as the following string `authprocedure_NEIL__Name_Password__Form` found in the following entry:

```
871 (D) Wed May 23 15:45:06 BST
2007 (L) webui.sc (T) 26 (C) com.volera.vcdn.webui.sc.dispatcher.ConfigWorkDispatcher (M) A (E) org.jdom.input.JDOMParseException: Error on
line 1120: cvc-id.1: There is no ID/IDREF binding for IDREF
'authprocedure_NEIL__Name_Password__Form'.
at org.jdom.input.SAXBuilder.build(SAXBuilder.java:468)
```

```

at org.jdom.input.SAXBuilder.build(SAXBuilder.java:770)
at com.volera.vcdn.platform.util.XmlUtil.validateXML(y:3304)
at com.volera.vcdn.webui.sc.dispatcher.ConfigWorkDispatcher.A(y:793)
at com.volera.vcdn.webui.sc.dispatcher.ConfigWorkDispatcher.do_deviceCon
fig(y:648)
:
:
:
at org.apache.coyote.http11.Http11Processor.process(Http11Processor.java
:799)
at org.apache.coyote.http11.Http11Protocol$Http11ConnectionHandler.proce
ssConnection(Http11Protocol.java:705)
at org.apache.tomcat.util.net.TcpWorkerThread.runIt(PoolTcpEndpoint.java
:577)
at
org.apache.tomcat.util.threads.ThreadPool$ControlRunnable.run(ThreadPool.
java:683)
at java.lang.Thread.run(Thread.java:534)
(Msg) <amLogEntry> 2007-05-23T15:45:06Z ERROR DeviceManager: AM#200904025:
Error
- XML VALIDATION FAILED. PLEASE CHECK APP_SC LOG </amLogEntry>

```

- 2** On the Access Gateway Appliance, change to the `/var/novell/cfgdb/vcdn` directory and open the `config.xml` file. Search for the problem string and the corresponding protected resource.

The example below shows that the problem string is tied to the ProtectedResourceID_svhttp_mylag_iMon_root resource. This maps to the HTTP reverse proxy called mylag, the service called iMon, and the protected resource called root.

```

----- snippet from problem area of config.xml -----
<ProtectedResource Name="root" Enable="1" Description=""
LastModified="116973455
5995" LastModifiedBy="cn=admin,o=novell"
UserInterfaceID="ProtectedResourceID_sv
http_mylag_iMon_root"
ProtectedResourceID="ProtectedResourceID_svhttp_mylag_iMon
_root">

    <URLPathList LastModified="4294967295" LastModifiedBy="String">

        <URLPath URLPath="/*" UserInterfaceID="/*"/>

    </URLPathList>

    <PolicyEnforcementList LastModified="1168947602067"
schemaVersion="1.34"
    LastModifiedBy="cn=admin,o=novell"
RuleCombiningAlgorithm="DenyOverridesWithPri
ority">

        <PolicyRef ElementRefType="ExternalWithIDRef"
ExternalDocRef="ou=xpemplPEP,ou=mastercdn,ou=ContentPublisherContainer,ou=
Partition,ou=PartitionsContainer,ou=VCDN_Root,ou=accessManagerContainer,o
=novell:romaContentCollectionXMLDoc"

```

```

UserInterfaceID="PolicyID_xpemlPEP_AGFormFill_1168947167634"
ExternalElementRef="PolicyID_xpemlPEP_AGFormFill_1168947167634"/>

    </PolicyEnforcementList>

    <AuthenticationProcedureRef
AuthProcedureIDRef="authprocedure_NEIL___Name_Password___Form"/>

</ProtectedResource>

----- end of snippet from problem area of config.xml -----

```

- 3 Look at the `AuthenticationProcedureRef` variable, which points to the contract assigned to the protected resource. You can see that the `authprocedure_NEIL___Name_Password___Form` contract is assigned to it.

However, when you look at the Access Gateway Appliance configuration in the Administration Console, you can see that the assigned contract is *[None]*, which is not the contract shown in the example. Change it to another contract name, apply the change, then set the contract back to *[None]* to clear the problem entry. The setup now operates with no XML validation errors.

B.2 Configuration UI Writes Incorrect Information to the Local Configuration Store

In this scenario, you apply the same change twice in quick succession, and the information written to the configuration store is invalid. Subsequent schema checks detect this invalid configuration and return an XML validation error. This scenario is more complex because it involves changing the configuration store on the Administration Console.

Troubleshooting Steps

- 1 On the Administration Console, search the `/opt/novell/devman/share/logs/app_sc.0.log` file for `#200904025: Error - XML VALIDATION FAILED`.

After you find the entry, work backwards to identify the start of the Java exception. From this, locate the problem strings or entry from the configuration, such as

`ProtectedResourceID_svhttp_sjh_portal_sjh_portal_1179933619340`. This message also indicates that a defined protected resource might not be unique. The configuration shows that before the Java exception, there is not enough information to narrow down the problem, so more troubleshooting is required.

The following is a snippet from the problem area of `app_sc.0.log` file that indicates that there are multiple occurrences of a protected resource.

```

Caused by: org.xml.sax.SAXParseException: cvc-id.2: There are multiple
occurrences of ID value
'ProtectedResourceID_svhttp_sjh_portal_sjh_portal_1179933619340'.
at
org.apache.xerces.util.ErrorHandlerWrapper.createSAXParseException(Unknown
Source)
at org.apache.xerces.util.ErrorHandlerWrapper.error(Unknown Source)
at org.apache.xerces.parsers.XML11Configuration.parse(Unknown Source)
at org.apache.xerces.parsers.XMLParser.parse(Unknown Source)
at org.apache.xerces.parsers.AbstractSAXParser.parse(Unknown Source)
at org.jdom.input.SAXBuilder.build(SAXBuilder.java:453)
at org.jdom.input.SAXBuilder.build(SAXBuilder.java:770)
at com.volera.vcdn.platform.util.XmlUtil.validateXML(y:3304)

```

```

at com.volera.vcdn.webui.sc.dispatcher.ConfigWorkDispatcher.A(y:793)
at
com.volera.vcdn.webui.sc.dispatcher.ConfigWorkDispatcher.do_deviceconfig(
y:648)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java
:39)
at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorI
mpl.java:25)
at java.lang.reflect.Method.invoke(Method.java:324)
at com.volera.vcdn.webui.sc.dispatcher.DefaultDispatcher.invoke(y:469)
at
com.volera.vcdn.webui.sc.dispatcher.DefaultDispatcher.processRequest(y:17
32)
at com.volera.roma.app.handler.DispatcherHandler.processRequest(y:3168)
at com.volera.roma.servlet.GenericController.doPost(y:53)
at javax.servlet.http.HttpServlet.service(HttpServlet.java:716)
at javax.servlet.http.HttpServlet.service(HttpServlet.java:809)
at
org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(Applicat
ionFilterChain.java:200)
at
org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterC
hain.java:146)
at
org.apache.catalina.core.StandardPipeline$StandardPipelineValveContext.in
vokeNext(StandardPipeline.java:594)
at com.novell.accessmanager.tomcat.SynchronizationValve.invoke(y:297)
at
org.apache.catalina.core.StandardPipeline.invoke(StandardPipeline.java:43
3)
at org.apache.catalina.core.ContainerBase.invoke(ContainerBase.java:948)
at
org.apache.coyote.tomcat5.CoyoteAdapter.service(CoyoteAdapter.java:152)
at
org.apache.coyote.http11.Http11Protocol$Http11ConnectionHandler.processCo
nnection(Http11Protocol.java:705)
at
org.apache.tomcat.util.threads.ThreadPool$ControlRunnable.run(ThreadPool.
java:683)
at java.lang.Thread.run(Thread.java:534)
(Msg) <amLogEntry> 2007-05-23T13:22:15Z ERROR DeviceManager: AM#200904025:
Error - XML VALIDATION FAILED. PLEASE CHECK APP_SC LOG </amLogEntry>

```

2 Confirm that the change has not been applied at the Access Gateway Appliance:

2a Enable the most verbose level of logging in the `/etc/laglogs.conf` file:

`log_level=LOG_DEBUG`. See “[Configuring Log Levels](#)” in the *Novell Access Manager 3.1 SP2 Access Gateway Guide*.

2b Restart the vmc services by using the following command:

```
/etc/init.d/novell-vmc restart
```

2c Search for in-memory errors in the `ics_dyn` log file. When these errors are displayed, the working Access Gateway Appliance configuration has not been updated with the latest changes.

2d Identify the protected resource with these issues. In the following case, the protected resource is the same, so you must look at the `config.xml` file and search for this specific protected resource. For example:

```
May 23 13:22:14 chw-amtlag1-176 : 404502 0: 7168: 0: 0:
VcpConfiguration::reconfigure starting AafLog
May 23 13:22:14 chw-amtlag1-176 : 404502 0: 7168: 0: 0:
VcpConfiguration::reconfigure finished
Error at file "in-memory", line 328, column 306
    Message: Datatype error: Type:InvalidDatatypeValueException,
Message:ID
'ProtectedResourceID_svhttp_sjh_portal_sjh_portal_1179933619340' is
not unique.
ERROR: Error retrieving config.xml: No data available
```

3 Search for the preceding string in the `/var/novell/cfgdb/vcdn/config.xml` file. You should see the following type of information:

```
<ProtectedResourceList>
<ProtectedResource Name="sjh_redirect" Enable="1"
  Description="" LastModified="1179934022767"

  LastModifiedBy="cn=admin,o=novell"UserInterfaceID="ProtectedResourceID_sv
http_sjh_portal_sjh_portal_1179933619340"
  ProtectedResourceID="ProtectedResourceID_svhttp_sjh_portal_sjh_portal_117
9933619340">
  <URLPathList LastModified="4294967295" LastModifiedBy="String">
<URLPath URLPath="/*" UserInterfaceID="/*"/>
  </URLPathList>
  <PolicyEnforcementList LastModified="1179934011081" schemaVersion="0.1"
LastModifiedBy="cn=admin,o=novell"
RuleCombiningAlgorithm="DenyOverridesWithPriority"
IncludedPolicyCategories=""/>
  <AuthenticationProcedureRef
AuthProcedureIDRef="authprocedure_Name_Password__Form"/>
  </ProtectedResource>
</ProtectedResourceList>
```

You should also see the following information:

```
<ProtectedResourceList LastModified="1179949051828"
LastModifiedBy="cn=admin,o=novell">
  <ProtectedResource Name="sjh_redirect" Enable="1" Description=""
LastModified="1179949051828" LastModifiedBy="cn=admin,o=novell"
UserInterfaceID="ProtectedResourceID_svhttp_sjh_portal_sjh_portal_1179933
619340"
ProtectedResourceID="ProtectedResourceID_svhttp_sjh_portal_sjh_portal_117
9933619340">
  <URLPathList LastModified="4294967295" LastModifiedBy="String">
  <URLPath URLPath="/*" UserInterfaceID="/*"/>
  </URLPathList>
  <PolicyEnforcementList LastModified="1179949047445"
schemaVersion="0.1" LastModifiedBy="cn=admin,o=novell"
RuleCombiningAlgorithm="DenyOverridesWithPriority"
IncludedPolicyCategories="">
  <PolicyRef ElementRefType="ExternalWithIDRef"
ExternalDocRef="ou=xpemplPEP,ou=mastercdn,ou=ContentPublisherContainer,ou=
Partition,ou=PartitionsContainer,ou=VCDN_Root,ou=accessManagerContainer,o
=novell:romaContentCollectionXMLDoc"
```

```

UserInterfaceID="PolicyID_xpemlPEP_Authorization_1176770874051"
ExternalElementRef="PolicyID_xpemlPEP_Authorization_1176770874051"/>
  </PolicyEnforcementList>
  <AuthenticationProcedureRef
AuthProcedureIDRef="authprocedure_Name_Password_Form"/>
  </ProtectedResource>
</ProtectedResourceList>

```

This is the duplicate entry that is causing the problem. You need to clear one of the entries from the configuration. If you clear it from the `/var/novell/cfgdb/vcdn/config.xml` file, then any change applied in the UI rewrites the information to the `config.xml` file.

- 4** Remove the duplicate entry from the Administration Console server's configuration store. To do this, you need an LDAP browser.

You can download a free Java-based tool from the Internet.

- 4a** Start the LDAP browser, then locate the `ag-xxxx` that matches the Access Gateway Appliance you are having problems with.

The easiest way is to go to the *Auditing > General Logging* tab of the Access Manager Administration Console and identify your Access Gateway Appliance ID. This ID corresponds to the first four digits of the `ag-xxxx` in the LDAP browser.

- 4b** Click the `ag-xxxx` container. You should see *CurrentConfig* and *WorkingConfig* containers within this Access Gateway container.

- 4c** Select the *CurrentConfig*, then the `RomaAGConfigurationXMLDoc` attribute. Copy and paste the attribute value into any editor. This is the configuration from the LAG.

- 4d** Search for the `RomaAGConfigurationXMLDoc` attribute string and remove the entire section on one of the hits starting with `<ProtectedResourceList>` and ending with `</ProtectedResourceList>`.

- 4e** Select and save the modified text.

- 4f** Paste the saved text into the `RomaAGConfigurationXMLDoc` attribute value.

- 4g** Repeat these steps for the `RomaAGConfigurationXMLDoc` attribute in *WorkingConfig*, and remove the duplicate entry that is causing the XML validation errors.

- 5** Restart Tomcat on the Administration Console machine.

- 6** Log in to the Administration Console again. Make a small change to the setup and apply that change, and verify that the XML validation error has disappeared.

Access Manager Audit Events and Data



The sections contains all the Novell audit events logged by Access Manager. Each event has the EventID, Description, Originator Title, Target Title, Subtarget Title, Text1 Title, Text2 Title, Text3 Title, Value1 Title, Value1 Type, Group Title, Data Length, and Data Type values stored. Each field contains a single character token (such as B, U, Y, and so on) that represent the data fields of the audit event, with each letter representing a different data field. The mapping of the character tokens to data fields is found in the `nids_en.lsc` and `sslvpn_en.lsc` files.

Novell Access Manager is listed among the log applications on the *General* tab on the Logging Server Options page (*Auditing and Logging > Logging Server Options*). You can view events on the Event list page in *Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*.

When you run an SQL query (*Auditing and Logging > Queries > [Name] > Run*), the system displays the results on the Query Results page. The *EventID* column displays the description of the event. Below, the event ID is listed with the description, to help you quickly locate the data for each audit event.

This section discusses the following audit events:

- ◆ [Section C.1, “NIDS: Sent a Federate Request \(002e0001\),” on page 143](#)
- ◆ [Section C.2, “NIDS: Received a Federate Request \(002e0002\),” on page 144](#)
- ◆ [Section C.3, “NIDS: Sent a Defederate Request \(002e0003\),” on page 144](#)
- ◆ [Section C.4, “NIDS: Received a Defederate Request \(002e0004\),” on page 145](#)
- ◆ [Section C.5, “NIDS: Sent a Register Name Request \(002e0005\),” on page 145](#)
- ◆ [Section C.6, “NIDS: Received a Register Name Request \(002e0006\),” on page 146](#)
- ◆ [Section C.7, “NIDS: Logged Out an Authentication that Was Provided to a Remote Consumer \(002e0007\),” on page 146](#)
- ◆ [Section C.8, “NIDS: Logged out a Local Authentication \(002e0008\),” on page 147](#)
- ◆ [Section C.9, “NIDS: Provided an Authentication to a Remote Consumer \(002e0009\),” on page 147](#)
- ◆ [Section C.10, “NIDS: User Session Was Authenticated \(002e000a\),” on page 148](#)
- ◆ [Section C.11, “NIDS: Failed to Provide an Authentication to a Remote Consumer \(002e000b\),” on page 149](#)
- ◆ [Section C.12, “NIDS: User Session Authentication Failed \(002e000c\),” on page 149](#)
- ◆ [Section C.13, “NIDS: Received an Attribute Query Request \(002e000d\),” on page 150](#)
- ◆ [Section C.14, “NIDS: User Account Provisioned \(002e000e\),” on page 150](#)
- ◆ [Section C.15, “NIDS: Failed to Provision a User Account \(002e000f\),” on page 151](#)
- ◆ [Section C.16, “NIDS: Web Service Query \(002e0010\),” on page 152](#)
- ◆ [Section C.17, “NIDS: Web Service Modify \(002e0011\),” on page 152](#)
- ◆ [Section C.18, “NIDS: Connection to User Store Replica Lost \(002e0012\),” on page 153](#)

- ◆ Section C.19, “NIDS: Connection to User Store Replica Reestablished (002e0013),” on page 154
- ◆ Section C.20, “NIDS: Server Started (002e0014),” on page 154
- ◆ Section C.21, “NIDS: Server Stopped (002e0015),” on page 155
- ◆ Section C.22, “NIDS: Server Refreshed (002e0016),” on page 155
- ◆ Section C.23, “NIDS: Intruder Lockout (002e0017),” on page 156
- ◆ Section C.24, “NIDS: Severe Component Log Entry (002e0018),” on page 156
- ◆ Section C.25, “NIDS: Warning Component Log Entry (002e0019),” on page 157
- ◆ Section C.26, “NIDS: Roles PEP Configured (002e0300),” on page 157
- ◆ Section C.27, “Access Gateway: PEP Configured (002e0301),” on page 158
- ◆ Section C.28, “J2EE Agent: Web Service Authorization PEP Configured (002e0305),” on page 158
- ◆ Section C.29, “J2EE Agent: JACC Authorization PEP Configured (002e0306),” on page 159
- ◆ Section C.30, “Roles Assignment Policy Evaluation (002e0320),” on page 160
- ◆ Section C.31, “Access Gateway: Authorization Policy Evaluation (002e0321),” on page 160
- ◆ Section C.32, “Access Gateway: Form Fill Policy Evaluation (002e0322),” on page 161
- ◆ Section C.33, “Access Gateway: Identity Injection Policy Evaluation (002e0323),” on page 161
- ◆ Section C.34, “J2EE Agent: Web Service Authorization Policy Evaluation (002e0324),” on page 162
- ◆ Section C.35, “J2EE Agent: Web Service SSL Required Policy Evaluation (002e0325),” on page 162
- ◆ Section C.36, “J2EE Agent: Startup (002e0401),” on page 163
- ◆ Section C.37, “J2EE Agent: Shutdown (002e0402),” on page 163
- ◆ Section C.38, “J2EE Agent: Reconfigure (002e0403),” on page 164
- ◆ Section C.39, “J2EE Agent: Authentication Successful (002e0404),” on page 164
- ◆ Section C.40, “J2EE Agent: Authentication Failed (002e0405),” on page 165
- ◆ Section C.41, “J2EE Agent: Web Resource Access Allowed (002e0406),” on page 166
- ◆ Section C.42, “J2EE Agent: Clear Text Access Allowed (002e0407),” on page 166
- ◆ Section C.43, “J2EE Agent: Clear Text Access Denied (002e0408),” on page 167
- ◆ Section C.44, “J2EE Agent: Web Resource Access Denied (002e0409),” on page 167
- ◆ Section C.45, “J2EE Agent: EJB Access Allowed (002e040a),” on page 168
- ◆ Section C.46, “J2EE Agent: EJB Access Denied (002e040b),” on page 169
- ◆ Section C.47, “Access Gateway: Access Denied (0x002e0505),” on page 169
- ◆ Section C.48, “Access Gateway: URL Not Found (0x002e0508),” on page 170
- ◆ Section C.49, “Access Gateway: System Started (0x002e0509),” on page 171
- ◆ Section C.50, “Access Gateway: System Shutdown (0x002e050a),” on page 171
- ◆ Section C.51, “Access Gateway: Identity Injection Parameters (0x002e050c),” on page 172
- ◆ Section C.52, “Access Gateway: Identity Injection Failed (0x002e050d),” on page 173

- ◆ Section C.53, “Access Gateway: Form Fill Authentication (0x002e050e),” on page 173
- ◆ Section C.54, “Access Gateway: Form Fill Authentication Failed (0x002e050f),” on page 174
- ◆ Section C.55, “Access Gateway: URL Accessed (0x002e0512),” on page 175
- ◆ Section C.56, “Access Gateway: IP Access Attempted (0x002e0513),” on page 176
- ◆ Section C.57, “Access Gateway: Webserver Down (0x002e0515),” on page 176
- ◆ Section C.58, “Access Gateway: All WebServers for a Service is Down (0x002e0516),” on page 177
- ◆ Section C.59, “Management Communication Channel: Health Change (0x002e0601),” on page 178
- ◆ Section C.60, “Management Communication Channel: Device Imported (0x002e0602),” on page 178
- ◆ Section C.61, “Management Communication Channel: Device Deleted (0x002e0603),” on page 179
- ◆ Section C.62, “Management Communication Channel: Device Configuration Changed (0x002e0604),” on page 180
- ◆ Section C.63, “Management Communication Channel: Device Alert (0x002e0605),” on page 180

C.1 NIDS: Sent a Federate Request (002e0001)

This event is generated when you select the *Federation Request Sent* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Sent a federate request.

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier

Data Description: LDAP Auth: User DN

Other Auth: User GUID

SubTarget (Y): null

Text1 (S): null

Text2 (T): null

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.2 NIDS: Received a Federate Request (002e0002)

This event is generated when you select the *Federation Request Handled* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Received a federate request.

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier

Data Description: LDAP Auth: User DN

Other Auth: User GUID

SubTarget (Y): null

Text1 (S): Schema Title: Provider Identifier; Data Description: Service Provider ID

Text2 (T): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.3 NIDS: Sent a Defederate Request (002e0003)

This event is generated when you select the *Defederation Request Sent* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Sent a defederate request.

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier

Data Description: LDAP Auth: User DN

Other Auth: User GUID

SubTarget (Y): null

Text1 (S): Schema Title: Provider Identifier; Data Description: Service Provider ID

Text2 (T): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.4 NIDS: Received a Defederate Request (002e0004)

This event is generated when you select the *Defederation Request Handled* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Received a defederate request

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier

Data Description: LDAP Auth: User DN

Other Auth: User GUID

SubTarget (Y): null

Text1 (S): Schema Title: Provider Identifier

Data Description: Service Provider ID

Text2 (T): null

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.5 NIDS: Sent a Register Name Request (002e0005)

Description: NIDS: Sent a register name request

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): null

Text2 (T): null

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.6 NIDS: Received a Register Name Request (002e0006)

This event is generated when you select the *Register Name Request Handled* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Received a register name request

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): null

Text2 (T): null

Text3 (F): null

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.7 NIDS: Logged Out an Authentication that Was Provided to a Remote Consumer (002e0007)

This event is generated when you select the *Logout Provided* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Logged out an authentication that was provided to a remote consumer

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier

Data Description: LDAP Auth: User DN

Other Auth: User GUID

SubTarget (Y): null

Text1 (S): Schema Title: Authentication Identifier

Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): null

Text3 (F): null

Value1 (I): Schema Title: Timed Out

Data Description: 0 = other reason

1 = timed out

Group (G): 0

Data Length (X): 0

Data (D): null

C.8 NIDS: Logged out a Local Authentication (002e0008)

This event is generated when you select the *Logout Local* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Logged out a local authentication

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier

Data Description: LDAP Auth: User DN

Other Auth: User GUID

SubTarget (Y): null

Text1 (S): Schema Title: Authentication Identifier

Data Description: IDP Session ID (AMAUTHID#auth_id:

Text2 (T): null

Text3 (F): null

Value1 (1): Schema Title: Timed Out

Data Description: 0 = other reason

1 = timed out

Group (G): 0

Data Length (X): 0

Data (D): null

C.9 NIDS: Provided an Authentication to a Remote Consumer (002e0009)

This event is generated when you select the *Login Consumed* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Provided an authentication to a remote consumer

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier

Data Description: User DN

SubTarget (Y): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text1 (S): Schema Title: Authentication Type
Data Description: Authentication Profile

Text2 (T): Schema Title: Authentication Entity Name
Data Description: Authentication Source

Text3 (F): Schema Title: Contract Class or Method Name
Data Description: Authentication Contract URI

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.10 NIDS: User Session Was Authenticated (002e000a)

This event is generated when you select the *Login Provided* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: User session was authenticated

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier
Data Description: User DN

SubTarget (Y): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text1 (S): Schema Title: Authentication Type
Data Description: Authentication Profile

Text2 (T): Schema Title: Authentication Entity Name
Data Description: Authentication Source

Text3 (F): Schema Title: Contract Class or Method Name
Data Description: Authentication Contract URI

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.11 NIDS: Failed to Provide an Authentication to a Remote Consumer (002e000b)

This event is generated when you select the *Login Consumed Failure* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Failed to provide an authentication to a remote consumer

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier

Data Description: User DN

SubTarget (Y): null

Text1 (S): Schema Title: Authentication Identifier

Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Provider Identifier

Data Description: Service Provider ID

Text3 (F): Schema Title: Reason

Data Description: Reason Message

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.12 NIDS: User Session Authentication Failed (002e000c)

This event is generated when you select the *Login Provided Failure* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration. Use the *Description* field and the *Text3 (F)* field to determine whether the failure came from a contract, SAML 1.1, SAML 2.0, or Liberty.

Description: NIDS: User session authentication failed. This string plus one of the following phrases: for a contract failure, *Contract Execution*; for a SAML 1.1 failure, *SAML Assertion*; for a SAML 2.0 failure, *SAML2 SSO*; for a Liberty failure, *Liberty SSO*.

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: Authentication Contract Name

Data Description: Contract URI

SubTarget (Y): null

Text1 (S): Schema Title: Authentication Identifier

Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Reason
Data Description: Reason Message

Text3 (F): Schema Title: Authentication Source
Data Description: For a contract, contains the authentication method name; for Liberty, contains the service provider IP; for SAML 1.1, contains the SAML assertion issuer; for SAML 2.0, contains the service provider IP.

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.13 NIDS: Received an Attribute Query Request (002e000d)

This event is generated when you select the *Attribute Query Request Handled* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Received an attribute query request

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier
Data Description: LDAP Auth: User DN
Other Auth: User GUID

SubTarget (Y): null

Text1 (S): Schema Title: Provider Identifier
Data Description: Service Provider ID

Text2 (T): Schema Title: Attribute Names
Data Description: Requested Attributes

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.14 NIDS: User Account Provisioned (002e000e)

This event is generated when you select the *User Account Provisioned* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: User account provisioned

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Store Identifier

Data Description: Displayable user name

SubTarget (Y): null

Text1 (S): Schema Title: User Identifier

Data Description: Authentication User Name

Text2 (T): Schema Title: Authentication Identifier

Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.15 NIDS: Failed to Provision a User Account (002e000f)

This event is generated when you select the *User Account Provisioned Failure* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Failed to provision a user account

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Store Identifier

Data Description: Displayable User Name

SubTarget (Y): null

Text1 (S): Schema Title: User Identifier

Data Description: Authentication User Name

Text2 (T): Schema Title: Reason

Data Description: Reason Message

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.16 NIDS: Web Service Query (002e0010)

This event is generated when you select the *Web Service Query Handled* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration. The Identity Server uses this event for two types of Web service queries:

- ♦ **Discovery:** This is a query to discover a service. For this type of query, the *Group (G)* field is not used. For a remote query, the *Data Description* of the *Value1* field is set to 0. For a local query, the *Data Description* of the *Value1* field is set to 1.
- ♦ **Profile:** This is a query to get attributes for a user from a profile (personal, credential, etc.). For this type of query, the *Group (G)* field contains a GroupingID for all attributes selected in the request. A separate event is generated for each attribute select list in the request. For a remote query, the *Data Description* of the *Value1* field is set to 0. For a local query, the *Data Description* of the *Value1* field is set to 1.

Description: NIDS: Web Service query

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier

Data Description: User DN

SubTarget (Y): null

Text1 (S): Schema Title: Provider Identifier

Data Description: Requesting Provider ID

Text2 (T): Schema Title: Select String

Data Description: Requested attributes; select string

Text3 (F): Schema Title: Service Identifier

Data Description: Web Service URI

Value1 (I): Schema Title: Local

Data Description: 0 – Remote

1 – Local

Group (G): Schema Title: Query Group

Data Description: If this is a profile query, it contains the grouping ID for all attributes selected in this request. Otherwise, this field is not used in the event.

Data Length (X): 0

Data (D): null

C.17 NIDS: Web Service Modify (002e0011)

This event is generated when you select the *Web Service Modify Handled* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration. The Identity Server uses this event for two types of Web service modify requests:

- ♦ **Discovery:** This is a request to discover a service to modify. For this type of request, the *Group (G)* field is not used. For a remote request, the *Data Description* of the *Value1* field is set to 0. For a local request, the *Data Description* of the *Value1* field is set to 1.

- ♦ **Profile:** This is a request to modify the attributes of a user in a profile (personal, credential, etc.). For this type of request, the *Group (G)* field contains a GroupingID for all attributes selected in the request. A separate event is generated for each attribute select list in the modify request. For a remote request, the *Data Description* of the *Value1* field is set to 0. For a local request, the *Data Description* of the *Value1* field is set to 1.

Description: NIDS: Web Service modify

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier

Data Description: User DN

SubTarget (Y): null

Text1 (S): Schema Title: Provider Identifier

Data Description: Requesting Provider ID

Text2 (T): Schema Title: Select String

Data Description: Modified attributes select string

Text3 (F): Schema Title: Service Identifier

Data Description: Web Service URI

Value1 (I): Schema Title: Local

Data Description: 0 – Remote; 1 – Local

Group (G): Schema Title: Modify Group

Data Description: If this is a profile modify, it contains the grouping ID for each attribute select list in the request. Otherwise, this field is not used in the event.

Data Length (X): 0

Data (D): null

C.18 NIDS: Connection to User Store Replica Lost (002e0012)

This event is generated when you select the *LDAP Connection Lost* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Connection to user store replica lost

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Store Replica Name

Data Description: Replica name

SubTarget (Y): null

Text1 (S): Schema Title: User Store Replica Host

Data Description: IP Address of User Store replica server

Text2 (T): null

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.19 NIDS: Connection to User Store Replica Reestablished (002e0013)

This event is generated when you select the *LDAP Connection Reestablished* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Connection to user store replica reestablished

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Store Replica Name

Data Description: Replica name

SubTarget (Y): null

Text1 (S): Schema Title: User Store Replica Host

Data Description: IP Address of User Store replica server

Text2 (T): null

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.20 NIDS: Server Started (002e0014)

This event is generated when you select the *Server Started* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Server started

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: Configuration Identifier

Data Description: Configuration Object DN

SubTarget (Y): null

Text1 (S): Schema Title: Server Identifier
Data Description: Unique server ID also used to create Liberty and SAML artifacts

Text2 (T): null

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.21 NIDS: Server Stopped (002e0015)

This event is generated when you select the *Server Stopped* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Server stopped

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: Configuration Identifier
Data Description: Configuration object DN

SubTarget (Y): null

Text1 (S): Schema Title: Server Identifier
Data Description: Unique server ID also used to create Liberty and SAML artifacts

Text2 (T): null

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.22 NIDS: Server Refreshed (002e0016)

This event is generated when you select the *Server Refreshed* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Server Refreshed

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: Configuration Identifier
Data Description: Configuration Object DN

SubTarget (Y): null

Text1 (S): Schema Title: Server Identifier

Data Description: Unique server ID also used to create Liberty and SAML artifacts

Text2 (T): null

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.23 NIDS: Intruder Lockout (002e0017)

This event is generated when you select the *Intruder Lockout Detected* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Intruder Lockout

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier

Data Description: User DN

SubTarget (Y): null

Text1 (S): Schema Title: Server Identifier

Data Description: IP address of the User Store replica server

Text2 (T): null

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.24 NIDS: Severe Component Log Entry (002e0018)

This event is generated when you select the *Component Log Severe Messages* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Severe Component Log Entry

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Component Log Text
Data Description: Server Error Text

Text2 (T): null

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.25 NIDS: Warning Component Log Entry (002e0019)

This event is generated when you select the *Component Log Warning Messages* option under *Novell Audit Logging* on the Logging page of an Identity Server configuration.

Description: NIDS: Warning Component Log Entry

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Component Log Text
Data Description: Warning Error Text

Text2 (T): null

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.26 NIDS: Roles PEP Configured (002e0300)

This event is generated for Identity Server roles.

Description: NIDS: Roles PEP Configured

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): null

Text2 (T): null

Text3 (F): null

Value1 (I): 0

Group (G): 0

Data Length (X): Schema Title: Policy Enforcement List Length
Data Description: Byte length of PEL

Data (D): Schema Title: Policy Enforcement List
Data Description: Policy Enforcement List (PEL) data

C.27 Access Gateway: PEP Configured (002e0301)

This event is generated when you enable auditing.

Description: Access Gateway: policy enforcement point (PEP) configured

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Text2 (T): null

Text3 (F): null

Value1 (I): Schema Title: Audit Enabled
Data Description: 0 = No; 1 = Yes

Group (G): 0

Data Length (X): Schema Title: Policy Enforcement List Length
Data Description: Byte length of PEL

Data (D): Schema Title: Policy Enforcement List
Data Description: Policy Enforcement List (PEL) data

C.28 J2EE Agent: Web Service Authorization PEP Configured (002e0305)

This event is generated when you enable auditing.

Description: J2EE Agent: Web Service Authorization policy enforcement point (PEP) Configured

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Event Identifier

Data Description: Event Tracking Identifier

Text2 (T): null

Text3 (F): null

Value1 (I): Schema Title: Audit Enabled

Data Description: 0 = Yes; 1 = No

Group (G):

Data Length (X): Schema Title: Protected Resource List Length

Data Description: Byte length of PWRL

Data (D): Schema Title: Protected Resource List

Data Description: Protected Web Resource List (PWRL)

C.29 J2EE Agent: JACC Authorization PEP Configured (002e0306)

This event is generated when you enable auditing.

Description: J2EE Agent: JACC Authorization policy enforcement point (PEP) configured

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Event Identifier

Data Description: Event Tracking Identifier

Text2 (T): null

Text3 (F): null

Value1 (I): Schema Title: audit enabled

Data Description: 0 = No; 1 = Yes

Group (G):

Data Length (X): Schema Title: Protected Resource List Length

Data Description: Byte length of PWML

Data (D): Schema Title: Protected Resource List
Data Description: Protected Web Module List (PWML)

C.30 Roles Assignment Policy Evaluation (002e0320)

This event is generated when you enable auditing.

Description: Roles assignment policy evaluation

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Assigned Roles
Data Description: Assigned Role or error message

Text3 (F): Schema Title: Policy Action
Data Description: Policy Action FDN

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.31 Access Gateway: Authorization Policy Evaluation (002e0321)

This event is generated when you enable auditing.

Description: Access Gateway: Authorization policy evaluation

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Text3 (F): Schema Title: Policy Action
Data Description: Policy Action FDN

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.32 Access Gateway: Form Fill Policy Evaluation (002e0322)

This event is generated when you enable auditing.

Description: Access Gateway: Form Fill policy evaluation

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Authentication Identifier

Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Event Identifier

Data Description: Event Tracking Identifier

Text3 (F): Schema Title: Policy Action

Data Description: Policy Action FDN

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.33 Access Gateway: Identity Injection Policy Evaluation (002e0323)

This event is generated when you enable auditing.

Description: Access Gateway: Identity Injection policy evaluation

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Authentication Identifier

Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Text3 (F): Schema Title: Policy Action
Data Description: Policy Action FDN

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.34 J2EE Agent: Web Service Authorization Policy Evaluation (002e0324)

This event is generated when you enable auditing.

Description: J2EE Agent: Web Service Authorization policy evaluation

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: Protected Resource URL
Data Description: Protected resource URL

SubTarget (Y): null

Text1 (S): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Text3 (F): Schema Title: Policy Action
Data Description: Policy Action FDN

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.35 J2EE Agent: Web Service SSL Required Policy Evaluation (002e0325)

This event is generated when you enable auditing.

Description: J2EE Agent: Web Service SSL Required policy evaluation

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: Protected Resource URL
Data Description: Protected Resource URL

SubTarget (Y): null

Text1 (S): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Text3 (F): null

Value1 (1): Schema Title: SSL Required
Data Description: 0 = No; 1 = Yes

Group (G): 0

Data Length (X): 0

Data (D): null

C.36 J2EE Agent: Startup (002e0401)

This event is generated when you select the *Startup, shutdown, and reconfigure* option in the *Audit Configuration* section of the Server Configuration page for the J2EE Agents.

Description: J2EE Agent: Startup

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): null

Text2 (T): null

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.37 J2EE Agent: Shutdown (002e0402)

This event is generated when you select the *Startup, shutdown, and reconfigure* option in the *Audit Configuration* section of the Server Configuration page for the J2EE Agents.

Description: J2EE Agent: Shutdown

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): null

Text2 (T): null

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.38 J2EE Agent: Reconfigure (002e0403)

This event is generated when you select the *Startup, shutdown, and reconfigure* option in the *Audit Configuration* section of the Server Configuration page for the J2EE Agents.

Description: J2EE Agent: Reconfigure

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Text2 (T): null

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.39 J2EE Agent: Authentication Successful (002e0404)

This event is generated when you select the *Successful authentications* option in the *Audit Configuration* section of the Server Configuration page for the J2EE Agents.

Description: J2EE Agent: Authentication successful

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier
Data Description: User DN

SubTarget (Y): null

Text1 (S): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.40 J2EE Agent: Authentication Failed (002e0405)

This event is generated when you select the *Unsuccessful authentications* option in the *Audit Configuration* section of the Server Configuration page for the J2EE Agents.

Description: J2EE Agent: Authentication failed

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier
Data Description: User DN

SubTarget (Y): null

Text1 (S): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Text3 (F): null

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.41 J2EE Agent: Web Resource Access Allowed (002e0406)

This event is generated when you select the *Allowed web resource access* option in the *Audit Configuration* section of the Server Configuration page for the J2EE Agents.

Description: J2EE Agent: Web Resource access allowed

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier
Data Description: User DN

SubTarget (Y): Schema Title: Source IP Address
Data Description: User IP Address

Text1 (S): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Permission Requested
Data Description: Web resource permission

Text3 (F): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.42 J2EE Agent: Clear Text Access Allowed (002e0407)

This event is generated when you select the *Allowed clear text access* option in the *Audit Configuration* section of the Server Configuration page for the J2EE Agents.

Description: J2EE Agent: Clear text access allowed

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier
Data Description: User DN

SubTarget (Y): Schema Title: Source IP Address
Data Description: User IP Address

Text1 (S): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Permission Requested
Data Description: Web User Data Permission

Text3 (F): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.43 J2EE Agent: Clear Text Access Denied (002e0408)

This event is generated when you select the *Denied clear text access* option in the *Audit Configuration* section of the Server Configuration page for the J2EE Agents.

Description: J2EE Agent: Clear text access denied

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier
Data Description: User DN

SubTarget (Y): Schema Title: Source IP Address
Data Description: User IP Address

Text1 (S): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Permission Requested
Data Description: Web User Data Permission

Text3 (F): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.44 J2EE Agent: Web Resource Access Denied (002e0409)

This event is generated when you select the *Denied web resource access* option in the *Audit Configuration* section of the Server Configuration page for the J2EE Agents.

Description: J2EE Agent: Web resource access denied

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier
Data Description: User DN

SubTarget (Y): Schema Title: Source IP Address
Data Description: User IP Address

Text1 (S): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Permission Requested
Data Description: Web User Data Permission

Text3 (F): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.45 J2EE Agent: EJB Access Allowed (002e040a)

This event is generated when you select the *Allowed EJB access* option in the *Audit Configuration* section of the Server Configuration page for the J2EE Agents.

Description: J2EE Agent: EJB access allowed

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier
Data Description: User DN

SubTarget (Y): null

Text1 (S): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Permission Requested
Data Description: EJB Method Permission

Text3 (F): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.46 J2EE Agent: EJB Access Denied (002e040b)

This event is generated when you select the *Denied EJB access* option in the *Audit Configuration* section of the Server Configuration page for the J2EE Agents.

Description: J2EE Agent: EJB access denied

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: User Identifier

Data Description: User DN

SubTarget (Y): null

Text1 (S): Schema Title: Authentication Identifier

Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text2 (T): Schema Title: Permission Requested

Data Description: EJB Method Permission

Text3 (F): Schema Title: Event Identifier

Data Description: Event Tracking Identifier

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.47 Access Gateway: Access Denied (0x002e0505)

This event is generated when you select the *Access Denied* option on the Novell Audit page of an Access Gateway.

Description: Access Gateway: Access Denied

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e0505

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: Protected Resource Name

Data Description: Configured Name of Protected Resource

SubTarget (Y): Schema Title: Protected Resource URL

Data Description: Protected Resource URL

Text1 (S): Schema Title: User Identifier
Data Description: User DN

Text2 (T): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text3 (F): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Value1 (I): Schema Title: Source IP Address
Data Description: User IP address (numeric format – host order)

Group (G): 0

Data Length (X): 0

Data (D): null

C.48 Access Gateway: URL Not Found (0x002e0508)

This event is generated when you select the *URL Not Found* option on the Novell Audit page of an Access Gateway.

Description: Access Gateway: URL Not Found

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e0508

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): Schema Title: Protected Resource URL
Data Description: Protected Resource URL

Text1 (S): Schema Title: User Identifier
Data Description: User DN

Text2 (T): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text3 (F): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Value1 (I): Schema Title: Source IP Address
Data Description: User IP address (numeric format – host order)

Group (G): 0

Data Length (X): 0

Data (D): null

C.49 Access Gateway: System Started (0x002e0509)

This event is generated when you select the *System Started* option on the Novell Audit page of an Access Gateway.

Description: Access Gateway: System Started

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e0509

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): null

Text2 (T): null

Text3 (F): null

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.50 Access Gateway: System Shutdown (0x002e050a)

This event is generated when you select the *System Shutdown* option on the Novell Audit page of an Access Gateway.

Description: Access Gateway: System Shutdown

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e050a

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): null

Text2 (T): null

Text3 (F): null

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.51 Access Gateway: Identity Injection Parameters (0x002e050c)

This event is generated when you select the *Identity Injection Parameters* option on the Novell Audit page of an Access Gateway.

Description: Access Gateway: Identity Injection Parameters

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e050c

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): Schema Title: Protected Resource URL

Data Description: Protected Resource URL

Text1 (S): Schema Title: User Identifier

Data Description: User DN

Text2 (T): Schema Title: Authentication Identifier

Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text3 (F): Schema Title: Event Identifier

Data Description: Event Tracking Identifier

Value1 (I): Schema Title: Injection Location

Data Description: 2710 – Auth Header 2720 – Custom Header
2730 – Query Parameters

Group (G): 0

Data Length (X): 0

Data (D): null

C.52 Access Gateway: Identity Injection Failed (0x002e050d)

This event is generated when you select the *Identity Injection Failed* option on the Novell Audit page of an Access Gateway.

Description: Access Gateway: Identity Injection Failed

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e050d

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): Schema Title: Protected Resource URL

Data Description: Protected Resource URL

Text1 (S): Schema Title: User Identifier

Data Description: User DN

Text2 (T): Schema Title: Authentication Identifier

Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text3 (F): Schema Title: Event Identifier

Data Description: Event Tracking Identifier

Value1 (I): Schema Title: Injection Location

Data Description: 2710 – Auth Header 2720 – Custom Header
2730 – Query Parameters

Group (G): 0

Data Length (X): 0

Data (D): null

C.53 Access Gateway: Form Fill Authentication (0x002e050e)

This event is generated when you select the *Form Fill Success* option on the Novell Audit page of an Access Gateway.

Description: Access Gateway: Form Fill Authentication

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e050e

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: Protected Resource Name
Data Description: Configured name of protected resource

SubTarget (Y): Schema Title: Protected Resource URL
Data Description: Protected Resource URL

Text1 (S): Schema Title: User Identifier
Data Description: User DN

Text2 (T): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text3 (F): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.54 Access Gateway: Form Fill Authentication Failed (0x002e050f)

This event is generated when you select the *Form Fill Failed* option on the Novell Audit page of an Access Gateway.

Description: Access Gateway: Form Fill Authentication Failed

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e050f

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): Schema Title: Protected Resource Name
Data Description: Configured name of protected resource

SubTarget (Y): Schema Title: Protected Resource URL
Data Description: Protected Resource URL

Text1 (S): Schema Title: User Identifier
Data Description: User DN

Text2 (T): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text3 (F): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.55 Access Gateway: URL Accessed (0x002e0512)

This event is generated when you select the *URL Accessed* option on the Novell Audit page of an Access Gateway.

Description: Access Gateway: URL Accessed

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e0512

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): Schema Title: Protected Resource URL
Data Description: Protected Resource URL

Text1 (S): Schema Title: User Identifier
Data Description: User DN

Text2 (T): Schema Title: Authentication Identifier
Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text3 (F): Schema Title: Event Identifier
Data Description: Event Tracking Identifier

Value1 (I): Schema Title: Source IP Address
Data Description: User IP address (numeric format – host order)

Group (G): 0

Data Length (X): 0

Data (D): null

C.56 Access Gateway: IP Access Attempted (0x002e0513)

This event is generated when you select the *IP Access Attempted* option on the Novell Audit page of an Access Gateway.

Description: Access Gateway: IP Access Attempted

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e0513

Originator (B): Schema Title: Originator

Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): Schema Title: Protected Resource URL

Data Description: Protected Resource URL

Text1 (S): Schema Title: User Identifier

Data Description: User DN

Text2 (T): Schema Title: Authentication Identifier

Data Description: IDP Session ID (AMAUTHID#auth_id:)

Text3 (F): Schema Title: Event Identifier

Data Description: Event Tracking Identifier

Value1 (I): Schema Title: Source IP Address

Data Description: User IP address (numeric format – host order)

Group (G): 0

Data Length (X): 0

Data (D): null

C.57 Access Gateway: Webserver Down (0x002e0515)

This event is generated when you select the *IP Access Attempted* option on the Novell Audit page of an Access Gateway.

Description: Access Gateway: One of the Web servers is not reachable

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e0515

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): WebServer hostname

Text2 (T): null

Text3 (F): null

Value1 (1): WebServer IP Address

Group (G): 0

Data Length (X): 0

Data (D): null

C.58 Access Gateway: All WebServers for a Service is Down (0x002e0516)

This event is generated when you select the IP Access Attempted option on the Novell Audit page of an Access Gateway.

Description: Access Gateway: All Web servers for a service are down

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e0516

Originator (B): Schema Title: Originator
Data Description: JCC Device ID (AMDEVICEID#device_id:)

Target (U): null

SubTarget (Y): null

Text1 (S): WebServer Hostname

Text2 (T): null

Text3 (F): null

Value1 (1): WebServer IP address

Group (G): 0

Data Length (X): 0

Data (D): null

C.59 Management Communication Channel: Health Change (0x002e0601)

This event is generated when you select the *Health Changes* option on the Access Manager Auditing page.

Description: Management Communication Channel: Health Change

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e0601

Originator (B): Schema Title: Originator
Data Description: “devmanagement” (AMDEVICEID#devmanagement:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Changed Device
Data Description: IP address and device type of the changed device

Text2 (T): Schema Title: Old State
Data Description: Old State

Text3 (F): Schema Title: New State
Data Description: New State

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.60 Management Communication Channel: Device Imported (0x002e0602)

This event is generated when you select the *Server Imports* option on the Access Manager Auditing page.

Description: Management Communication Channel: Device Imported

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e0602

Originator (B): Schema Title: Originator
Data Description: “devmanagement” (AMDEVICEID#devmanagement:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Device

Data Description: IP address and device type of the changed device

Text2 (T): blank string

Text3 (F): blank string

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.61 Management Communication Channel: Device Deleted (0x002e0603)

This event is generated when you select the *Server Deletes* option on the Access Manager Auditing page.

Description: Management Communication Channel: Device Deleted

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e0603

Originator (B): Schema Title: Originator

Data Description: "devmanagement" (AMDEVICEID#devmanagement:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Device

Data Description: IP address and device type of the changed device

Text2 (T): Schema Title: Administrator

Data Description: DN of the administrator deleting the device

Text3 (F): blank string

Value1 (I): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.62 Management Communication Channel: Device Configuration Changed (0x002e0604)

This event is generated when you select the *Configuration Changes* option on the Access Manager Auditing page.

Description: Management Communication Channel: Device Configuration Changed

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e0604

Originator (B): Schema Title: Originator

Data Description: “devmanagement” (AMDEVICEID#devmanagement:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Device

Data Description: IP address and device type of the changed device

Text2 (T): Schema Title: Administrator

Data Description: DN of the administrator invoking the configuration change

Text3 (F): blank string

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

C.63 Management Communication Channel: Device Alert (0x002e0605)

This event is generated when you enable auditing.

Description: Management Communication Channel: Device Alert

In the Event list (*Auditing and Logging > Logging Server Options > [Name of Novell Audit Secure Logging Server] > Novell Access Manager > Events*), this column is called *Event Name*.

In a query, this column is called *EventID*.

Event ID: 0x002e0605

Originator (B): Schema Title: Originator

Data Description: “devmanagement” (AMDEVICEID#devmanagement:)

Target (U): null

SubTarget (Y): null

Text1 (S): Schema Title: Device

Data Description: IP address of the device generating the alert

Text2 (T): Schema Title: Alert Message

Data Description: alert message string

Text3 (F): blank string

Value1 (1): 0

Group (G): 0

Data Length (X): 0

Data (D): null

