

Novell Nsure™ Identity Manager

2

www.novell.com

ADMINISTRATION GUIDE

April 1, 2004



Novell®

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.

You may not export or re-export this product in violation of any applicable laws or regulations including, without limitation, U.S. export regulations or the laws of the country in which you reside.

Copyright © 2004 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.

U.S. Patent Nos. 5,349,642; 5,608,903; 5,671,414; 5,677,851; 5,758,344; 5,784,560; 5,818,936; 5,828,882; 5,832,275; 5,832,483; 5,832,487; 5,870,561; 5,870,739; 5,873,079; 5,878,415; 5,884,304; 5,919,257; 5,933,503; 5,933,826; 5,946,467; 5,956,718; 6,016,499; 6,065,017; 6,105,062; 6,105,132; 6,108,649; 6,167,393; 6,286,010; 6,308,181; 6,345,266; 6,424,976; 6,516,325; 6,519,610; 6,539,381; 6,578,035; 6,615,350; 6,629,132. Patents Pending.

Novell, Inc.
1800 South Novell Place
Provo, UT 84606
U.S.A.

www.novell.com

Novell Nsure Identity Manager 2 Administration Guide

April 1, 2004

Online Documentation: To access the online documentation for this and other Novell products, and to get updates, see www.novell.com/documentation.

Novell Trademarks

Novell is a registered trademark of Novell, Inc. in the United States and other countries.

Nsure is a trademark of Novell, Inc.

Third-Party Trademarks

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

Contents

	About This Guide	11
1	Overview	13
	What's New in Identity Manager 2?	15
	Policy Builder Interface and DirXML Script for Creating Policies	15
	Password Management	15
	Role-Based Entitlements	16
	Reporting and Notification Using Novell Nsure Audit Lite	16
	Global Configuration Values	17
	Driver Heartbeat	17
	Understanding the Identity Manager Architecture	17
	DirXML Engine	18
	DirXML Driver Shim	19
	Driver Configuration Files	19
	Identity Manager Event Cache	19
	Identity Manager Components	19
	Driver Set	19
	Driver Object	20
	Driver Shim	21
	Publisher and Subscriber Channels	22
	Events and Commands	22
	Policies and Filters	22
	Associations	22
2	Planning	25
	Planning the Project Management Aspects of Identity Manager Implementation	25
	Novell Identity Manager Deployment	25
	Planning the Technical Aspects of Identity Manager Implementation	31
	Replicating the Objects that Identity Manager Needs on the Server	31
	Managing Users on Different Servers Using Scope Filtering	32
3	Upgrading	37
	Upgrading Password Synchronization	37
	Upgrading from RNS to Nsure Audit	37
	Upgrading Driver Configurations	37
4	Installation	39
	Before You Install	39
	Installing Identity Manager Components	40
	Identity Manager System Requirements	41
	Feature Support for eDirectory 8.6.2 and eDirectory 8.7.3	43
	Installing Identity Manager on NetWare	44
	Installing Identity Manager on Windows	45
	Installing Identity Manager on Linux or Solaris	46
	Post-Installation Tasks	47
	Using the Remote Loader Service	47

Using the Remote Loader Console on Windows	48
Configuring SSL between the DirXML Engine and the Remote Loader	49
Configuring a DirXML Driver with the Remote Loader	50
Configuring the Remote Loader on Solaris and Linux	51
Configuring the Remote Loader for Remote Drivers	51
Configuring the Remote Loader with Open SSL on UNIX	51
Running the Remote Loader Configuration Wizard	52
Setting Command Line Options and Parameters	53
Running the Remote Loader	58
Setting Up Multiple Instances of the Remote Loader	58
Activating Identity Manager Products	58
Installing a Custom Driver	59
5 Managing DirXML Drivers	61
Creating and Configuring a Driver	61
Creating a Driver Object	61
Creating Multiple Drivers	62
Managing DirXML 1.x Drivers in an Identity Manager Environment	62
Upgrading a Driver Configuration from DirXML 1.x to Identity Manager Format	63
Starting, Stopping, or Restarting a Driver	63
Using Global Configuration Values	63
Using the DirXML Command Line Utility	64
Using Named Passwords	64
Creating a Named Password in the DirXML Command Line Utility	64
Using Named Passwords in Driver Policies	66
Removing a Named Password in the DirXML Command Line Utility	66
Reassociating a Driver Object with a Server	67
Adding Driver Heartbeat	68
6 Creating Policies	71
7 Managing Passwords Using Password Policies	73
Overview of Password Policy Features	73
Set Advanced Password Rules	74
Enable Universal Password	75
Provide Users with Forgotten Password Self-Service	76
Provide Users with Reset Password Self-Service	76
Assign Policies to eDirectory Users	76
Enforce Policies in eDirectory	77
Enforce Policies on Connected Systems	79
View Which Password Policy Is in Effect for a User	79
Set Universal Password for a User	79
Planning for Password Policies	80
Planning How to Assign Password Policies in the Tree	80
Planning the Rules for Your Password Policies	80
Planning How to Manage Login Methods for your Users	80
Prerequisites for Using Password Policies	81
Deploying Password Policies without Universal Password	82
(NetWare 6.5 only) Re-Creating Universal Password Assignments	82
Creating Password Policies	84
Assigning Password Policies to Users	84
Making Sure Password Policies Are Correct for Identity Manager	85
Finding Out Which Policy a User Has	86
Setting A User's Password	86
Creating or Editing Challenge Sets	86
Configuring Notification for Password Features	87

Troubleshooting Password Policies	87
8 Password Self-Service	89
Overview of Self-Service Features	89
Provide Users with Forgotten Password Self-Service	89
Provide Users with Reset Password Self-Service	90
Prerequisites for Using Password Self-Service.	91
Planning for Login Methods for Password Self-Service	92
Providing End Users with Forgotten Password Self-Service	92
Challenge Sets	92
Forgotten Password Actions	94
Password Hints	94
Prompting End Users to Set Up Forgotten Password.	94
How the End User Sets Up Forgotten Password Self-Service	95
What End Users See When They Forget Passwords.	104
Turning Off the Forgotten Password Link.	108
Providing End Users with Password Reset Self-Service	109
Creating or Editing Challenge Sets.	110
Configuring Notification for Password Self-Service.	111
Troubleshooting Password Self-Service	111
9 Password Synchronization across Connected Systems	113
Overview	113
What is Bidirectional Password Synchronization?	114
Features of Identity Manager Password Synchronization.	114
Diagrams of Password Synchronization Flow.	118
Connected System Support for Password Synchronization	119
Pieces Required for Password Synchronization to Work.	120
Password Synchronization Capabilities Declared in the Driver Manifest	121
Password Synchronization Settings You Create Using Global Configuration Values	121
Policies Required in the Driver Configuration	125
Filters You Install on the Connected System to Capture Passwords	126
Password Policies You Create for Your Users	127
NMAS Login Methods.	127
Preparing to Use Identity Manager Password Synchronization and Universal Password	127
Switching Users from NDS Password to Universal Password	127
Changing Passwords Using iManager Self-Service Console.	128
Preparing to Use Universal Password	128
Replica Planning and Password Policies	130
Setting Up E-Mail Notification.	130
New Driver Configuration and Identity Manager Password Synchronization.	131
Upgrading Password Synchronization 1.0 to Identity Manager Password Synchronization	132
Upgrading Existing Driver Configurations to Support Identity Manager Password Synchronization	132
Implementing Password Synchronization	138
Overview of Identity Manager Relationship to NMAS.	139
Scenario 1: eDirectory to eDirectory Password Synchronization Using NDS Password	140
Scenario 2: Synchronizing Universal Password	142
Scenario 3: Synchronizing eDirectory and Connected Systems with Identity Manager Updating the Distribution Password	152
Scenario 4: Tunneling — Synchronizing Connected Systems but not eDirectory, with Identity Manager Updating the Distribution Password	160
Troubleshooting Scenario 4.	164
Setting Up Password Filters	165
Setting Up Password Synchronization Filters for Active Directory and NT Domain	166
Setting Up Password Synchronization Filters for NIS.	166
Managing Password Synchronization	166

Setting the Flow of Passwords Across Systems	166
Enforcing Password Policies on Connected Systems	169
Keeping the eDirectory Password Separate from the Synchronized Password	170
Checking the Password Synchronization Status for a User	170
Configuring E-Mail Notification	170
Prerequisites	171
Setting Up the SMTP Server To Send E-Mail Notification	171
Setting Up E-Mail Templates for Notification	172
Adding Your Own Replacement Tags to E-Mail Notification Templates	173
Sending E-Mail Notifications to the Administrator	180
Localizing E-Mail Notification Templates	180
Troubleshooting Password Synchronization	180
10 Using Role-Based Entitlements	183
Overview	183
How it Works	185
Prerequisites	185
Getting Started	186
Creating a Driver Object for the Entitlements Driver	186
Configuring Drivers to Use Entitlement Policies	186
Creating Entitlement Policies	187
Defining Membership for an Entitlement Policy	188
Choosing Entitlements for an Entitlement Policy	189
Keeping Accounts Safe	194
Controlling Adding or Removing Entitlements	195
Conflict Resolution between Entitlement Policies	196
Overview	196
Changing the Conflict Resolution Method for an Individual Entitlement	197
Prioritizing Entitlement Policies	198
Password Synchronization and Role-Based Entitlements	199
Troubleshooting Role-Based Entitlements	199
11 Managing Engine Services	201
Entitlements Service Driver	201
Loopback Service Driver: Facilitating Moving Objects Using the Move Proxy Service	201
Understanding the Move Proxy Service	202
Setting Up the Move Proxy Service	202
Configuring Other Drivers To Delegate Moves to the Move Proxy Service	203
Manual Task Service Driver (Workflow Service Request Driver)	204
12 Logging Events Using Nsure Audit	205
Overview	205
Setting Up Nsure Audit	206
Setting Up the Platform Agent	206
Setting Up the Secure Logging Server	207
Logging Configuration	207
Selecting Events to Log	207
User-Defined Events	209
eDirectory Objects	211
Viewing Events in Nsure Audit Report	212
Importing the Identity Manager Application Schema	212
Viewing Identity Manager Events	212
Sending Notifications Based on Events	212
Using Status Logs	213
Setting the Maximum Log Size	213
Viewing Status Logs	213

A	Activating Novell Identity Manager Products	215
	Generating a Product Activation Request	215
	Submitting a Product Activation Request.	217
	Installing a Product Activation Credential	218
	Viewing Product Activations for Identity Manager and DirXML Drivers.	219
B	Updates	221
	March 2004	221
	April 1, 2004	221

About This Guide

Novell® Nsure™ Identity Manager 2, formerly DirXML™, is a data sharing and synchronization service that enables applications, directories, and databases to share information. It links together scattered information and enables you to establish policies that govern automatic updates to designated systems when identity changes occur.

Identity Manager provides the foundation for account provisioning, security, single sign-on, user self-service, authentication, authorization, automated workflow and Web services. It allows you to integrate, manage and control your distributed identity information so you can securely deliver the right resources to the right people.

This guide provides an overview of the Identity Manager technologies, and also describes installation, administration, and configuration functions.

Additional Documentation

For documentation on using the DirXML drivers, see the [Identity Manager Documentation Web site \(http://www.novell.com/documentation/lg/dirxmldrivers/index.html\)](http://www.novell.com/documentation/lg/dirxmldrivers/index.html).

Documentation Updates

For the most recent version of this document, see the [Identity Manager Documentation Web site \(http://www.novell.com/documentation/lg/dirxml20/index.html\)](http://www.novell.com/documentation/lg/dirxml20/index.html)

Documentation Conventions

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

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

User Comments

We want to hear your comments and suggestions about this manual and the other documentation included with this product. To contact us, send e-mail to proddoc@novell.com.

1

Overview

Novell® Nsure™ Identity Manager 2, formerly DirXML®, is an award-winning data-sharing and synchronization solution that revolutionizes how you manage data. This service leverages your identity vault to synchronize, transform, and distribute information across applications, databases, and directories.

When data from one system changes, the DirXML engine detects and propagates these changes to other connected entities based on the business rules you define. This solution enables you to enforce authoritative data sources for any particular piece of data (for example, an HR application owns a user's ID, while a messaging system might own a user's e-mail account information).

Identity Manager lets an application (such as SAP*, PeopleSoft*, Lotus Notes*, Microsoft* Exchange, Active Directory*, and others) do the following:

- ♦ Share data with the identity vault (Novell eDirectory™.)
- ♦ Synchronize and transform shared data with the identity vault when it is modified in the application database.
- ♦ Synchronize and transform shared data with the application database when the data is modified in the identity vault.

Identity Manager does this by providing a bidirectional framework that allows administrators to specify which data will flow from the identity vault to the application and from the application to the identity vault. The framework uses XML to provide data and event translation capabilities that convert identity vault data and events into the specified application-specific format. It also converts application-specific formats into a format that can be understood by the identity vault. All interactions with the application take place using the application's native API.

Identity Manager lets you select only the eDirectory attributes and classes that correspond to relevant application-specific records and fields. For example, an eDirectory database can choose to share User-type objects with a Human Resources database but not share network resource objects such as Servers, Printers, and Volumes. The Human Resources database can in turn share users' given names, surnames, initials, telephone numbers, and work locations with eDirectory but not share the users' family information and employment history.

If eDirectory doesn't have classes or attributes for data you want to share with other applications, you can extend the eDirectory schema to include them. In this case, eDirectory becomes a repository of information that eDirectory does not need, but which other applications can use. The application-specific database maintains the repository for the information that is required only by the application.

Identity Manager accomplishes the following tasks:

- ♦ Uses events to capture changes in the identity vault.
- ♦ Centralizes or distributes data management by acting as a hub to pull all the data together.

- ◆ Exposes directory data in XML format, allowing it to be used and shared by XML applications or applications integrated through Identity Manager.
- ◆ Controls the flow of data using specific filters that govern data elements defined in the system.
- ◆ Enforces authoritative data sources by using permissions and filters.
- ◆ Applies rules to directory data that is in an XML format. These rules govern the interpretation and transformation of the data as changes flow through Identity Manager.
- ◆ Transforms the data from XML into virtually any data format. This provides Identity Manager the ability to share data with any application.
- ◆ Carefully maintains associations between identity vault objects and objects within all other integrated systems, in order to ensure that data changes are appropriately reflected across all integrated systems.

With Identity Manager, your business can simplify HR processes, reduce data management costs, build customer relationships through highly customized service, and remove interoperability barriers that inhibit success. Below are several examples activities that Identity Manager enables:

Activity	Identity Manager Solution
Manage User Accounts	<p>With a single operation:</p> <p>Identity Manager grants or removes access for an employee to resources almost immediately.</p> <p>Identity Manager provides automated employee provisioning capability where a new employee has access to network, e-mail, applications, resources, and so forth.</p> <p>Identity Manager can also restrict or disable access upon termination or leave.</p>
Track and Integrate Asset Inventory	Identity Manager can add profiles for all asset inventory items (computers, monitors, phones, library resources, chairs, desks, etc.) to eDirectory and integrate them with user profiles such as individuals, departments, or organizations.
Automate White/Yellow Page Directories	Identity Manager can create unified directories with varying levels of information for internal and external use. External directories might contain only e-mail addresses; internal directories might include location, phone, fax, cell, home address, etc.
Enhance User Profiles	Identity Manager augments user profiles by adding or synchronizing information such as e-mail address, phone number, home address, preferences, reporting relationships, hardware assets, phone, keys, inventory, and more.
Unify Communications Access	Identity Manager simplifies network, phone, pagers, Web, wireless access, etc. for individual users or groups by synchronizing directories for each to a common management interface.
Strengthen Partner Relationships	Identity Manager strengthens partnerships by creating profiles (employee, customer, etc.) in partner systems outside the firewall to enable partners to provide immediate service as needed.

Activity	Identity Manager Solution
Improve Supply Chain	Identity Manager improves customer services by recognizing and consolidating instances of multiple accounts per customer.
Build Customer Loyalty	Identity Manager offers new services in response to recognizing customer needs as a result of viewing data together that was previously isolated in silos.
Customize Service	<p>Identity Manager provides users (employees, customers, partners, etc.) with profiles complete with synchronized information, including relationships, status, and service records.</p> <p>These profiles can be used to provide varying levels of access to services and information, and offer real-time, customized services based on a customer's standing.</p>

What's New in Identity Manager 2?

In this section:

- ♦ [“Policy Builder Interface and DirXML Script for Creating Policies” on page 15](#)
- ♦ [“Password Management” on page 15](#)
- ♦ [“Role-Based Entitlements” on page 16](#)
- ♦ [“Reporting and Notification Using Novell Nsure Audit Lite” on page 16](#)
- ♦ [“Global Configuration Values” on page 17](#)
- ♦ [“Driver Heartbeat” on page 17](#)

Policy Builder Interface and DirXML Script for Creating Policies

In previous releases of DirXML, the policies used in a driver configuration were called Rule objects and Stylesheet objects. In Identity Manager 2, each part of the driver configuration is called a Policy object, and these policies contain individual rules.

For common tasks, you can now use the new Policy Builder interface to create policies for your drivers without having to write XSLT code. The Policy Builder helps you set up twenty-five of the most common rules using the new DirXML Script.

For more information, see [“Creating Policies” on page 71](#).

Password Management

Identity Manager 2 includes new and enhanced password management features:

- ♦ New Password Policies let you create rules for passwords and assign them to users, containers, or the whole eDirectory tree. You can enable Universal Password, which lets you enforce detailed criteria for passwords and allows for special characters.
- ♦ Identity Manager Password Synchronization is now cross-platform, and it lets you enforce your Password Policies across connected systems. New notification templates let you automatically send messages to users about their password synchronization status.

- ◆ Using Password Policies, you can also provide Forgotten Password Self-Service and Reset Password Self-Service to your users. These new features can help you reduce help desk calls. Notification templates are also included for automatically sending forgotten password and password hint messages to users.

For more information, see [Chapter 7, “Managing Passwords Using Password Policies,” on page 73](#) and [Chapter 9, “Password Synchronization across Connected Systems,” on page 113](#).

Role-Based Entitlements

Role-Based Entitlements let you grant entitlements on connected systems to a group of Novell eDirectory users. Using Entitlement Policies, you can streamline management of business policies and reduce the need to configure your DirXML drivers.

Role-Based Entitlements is an alternative way to administer Identity Manager. You might choose to use it if you prefer a centralized model of Identity Manager administration.

An Entitlement Policy is an eDirectory dynamic group object with additional features added for connected systems. When you create an Entitlement Policy, you define the membership for the policy and the entitlements that should be granted to the members of the Entitlement Policy.

Role-Based Entitlements let you grant entitlements on connected systems and rights in eDirectory. Entitlements on connected systems can be any of the following:

- ◆ Accounts
- ◆ Membership in e-mail distribution lists
- ◆ Group membership
- ◆ Attributes for the corresponding objects in connected systems, populated with values you specify
- ◆ Other entitlements on connected systems that you customize

Because Role-Based Entitlements functionality is based on Identity Manager, you must have DirXML drivers installed and configured properly in order to be able to administer connected systems. In addition, to avoid possible conflicts between Entitlement Policy assignments and DirXML driver configurations, you should be aware of your business policies and how they are administered through Identity Manager.

For more information, see [Chapter 10, “Using Role-Based Entitlements,” on page 183](#).

Reporting and Notification Using Novell Nsure Audit Lite

With Identity Manager 2, you can now use Novell Nsure Audit for reporting and notification services. Novell Nsure Audit is a centralized, cross-platform auditing service. It collects event data from multiple applications across multiple platforms and writes the data to a single, non-repudiable data store. Nsure Audit is also capable of creating filtered data stores. Based on criteria you define, Nsure Audit captures specific types of events and writes those events to secondary data stores.

For more information, see the [NSure Audit 1.0 Administration Guide \(http://www.novell.com/documentation/lg/nsureaudit/index.html\)](http://www.novell.com/documentation/lg/nsureaudit/index.html).

The Identity Manager 2 engine is backward compatible with RNS. The engine continues to process RNS functions if you are currently using RNS. However, we recommend that you move to Nsure Audit because it provides all the functionality of RNS with the addition of many new features. For RNS documentation, see the [DirXML 1.1a Administration Guide \(http://www.novell.com/documentation/lg/dirxml11a/dirxml/data/afae8bz.html\)](http://www.novell.com/documentation/lg/dirxml11a/dirxml/data/afae8bz.html).

Global Configuration Values

Global configuration values (GCVs) are new settings that are similar to driver parameters. Global configuration values can be specified for a driver set as well as an individual driver. If a driver does not have a value for a particular GCV, the driver inherits the value for that GCV from the driver set.

GCVs allow you to specify settings for new features such as Password Synchronization, as well as settings that are specific to the function of an individual driver configuration. Some GCVs are provided with the drivers, but you can also add your own. You can refer to these values in a policy to help you customize your driver configuration.

For more information, see [“Using Global Configuration Values” on page 63](#).

Driver Heartbeat

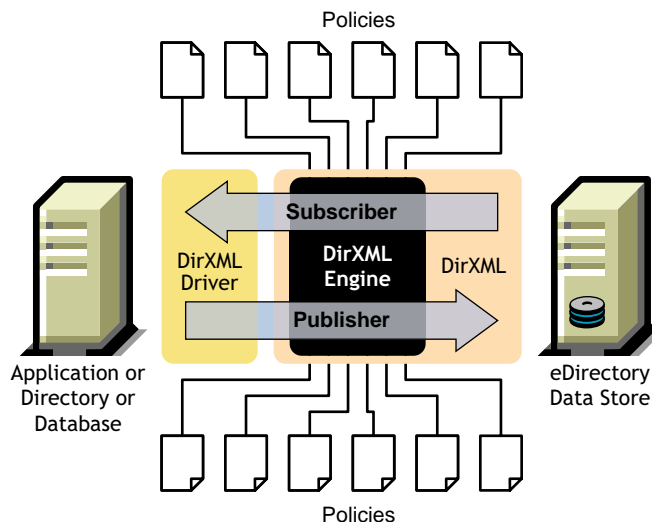
The DirXML engine now accepts driver heartbeat documents from drivers, and drivers can be configured to send them.

For more information, see [“Adding Driver Heartbeat” on page 68](#).

Understanding the Identity Manager Architecture

The Identity Manager technology is composed of several different components. Its main purpose is to provide for the clean movement of data between the identity vault and any application, directory, or database. To accomplish this, Identity Manager has a well-defined interface that translates directory data and events into XML format. This interface allows the data to flow in and out of eDirectory in a bidirectional manner.

The following illustration shows the basic Identity Manager components and their relationships.



The DirXML engine is the key module in the Identity Manager architecture. It provides the interface that allows DirXML drivers to synchronize information with eDirectory, allowing even disparate data systems to connect and share data.

The DirXML engine exposes eDirectory data and eDirectory events using an XML format. The DirXML engine employs a rules processor and a data transformation engine to manipulate the data as it flows between two systems.

When eDirectory initializes, it does the following:

1. Reads the filter for all DirXML drivers.
2. Registers the drivers for the appropriate eDirectory events.
3. Filters the data according to each driver's specifications.
4. Sets up a cache for the eDirectory events passing through to each driver.

After an event is cached, the driver that owns the cache will read the event.

The driver receives eDirectory data in eDirectory native format, translates it into XDS format (the XML vocabulary used by Identity Manager that can be transformed by a policy), and sends the event to the DirXML engine. The engine reads all the policies you have set up for your application driver (Mapping, Matching, Placement, Create, Transformation, and Style Sheets) and creates XML-formatted data according to those policies, then sends the data to your application driver. It then sends the data to the application and monitors the update until it receives a successful completion code.

The Publisher portion of the driver performs the gathering and sending of updates from the external application database to the identity vault. When the application driver is informed of changes to the information the two databases are sharing, the application driver gathers the information, ensures that it has been filtered to the correct set of data, converts the data to DirXML format, and sends the data to the engine.

DirXML Engine

The DirXML engine, sometimes referred to as the Join engine, can be broken down into two components: the NDS[®] interface and the Join engine.

NDS Interface

The NDS Interface built into the DirXML engine is used to detect events that take place in eDirectory. This interface guarantees the delivery of events to Identity Manager by using the event cache. The NDS interface supports multiple driver loading, which means that only one instance of Identity Manager is running, but it can communicate with multiple applications. Loopback detection is built into this interface to prevent event loops from occurring between eDirectory and the application. Although the interface contains loopback protection, developers are encouraged to build loopback detection into the individual application drivers.

Join Engine

The Join engine applies the Identity Manager XML-based rules (XDS) to each event presented to it. Identity Manager rules can also be in the form of Extensible Stylesheet Language Transformation (XSLT), which is a more powerful XML vocabulary defined for operating on and then transforming XML documents

The Join engine applies each type of rule to the source document. The ability to complete these transformations is one of the most powerful capabilities of Identity Manager. Data is transformed in real-time as it is shared between eDirectory and individual applications.

DirXML Driver Shim

The DirXML driver shim, commonly referred to as the driver, is the conduit through which information is transferred between eDirectory and the application, directory, or database. Communication between the DirXML engine and the driver is handled through XML documents that describe events, queries, and results.

The shim is written in either Java* or C++.

Driver Configuration Files

Driver configurations are preconfigured XML files that are included with Identity Manager. You can import these configuration files through the wizards in iManager.

These driver configurations contain sample policies. They are not intended for use in a production environment, but rather as a template for you to modify.

Identity Manager Event Cache

All of the events generated through eDirectory are stored in an event cache until they are successfully processed. This guarantees that no data will be lost because of a bad connection, loss of system resources, unavailability of a driver, or any other network failure.

Identity Manager Components

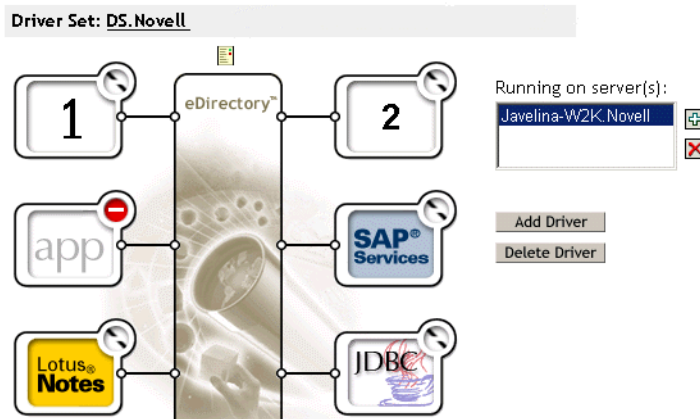
This section introduces the concepts and the components that comprise Identity Manager.

Driver Set

A driver set is a container that holds DirXML drivers. Only one driver set can be active on a server at a time. As a result, all active drivers must be grouped into the same driver set. It is not necessary to activate all of the drivers in a driver set on every server that is using the driver set.

The driver set object must exist in a full read/write replica on any server that is using it, so we recommend partitioning the driver set. This is recommended so that if replicas of users are moved to another server, the driver objects are not.

The following image displays the driver set in iManager.



From the Overview page in iManager (shown above), you can:

- ◆ View and modify the driver set and its properties
- ◆ View the drivers within the driver set
- ◆ Change the status of a driver
- ◆ Associate a driver set with a server
- ◆ Add or remove drivers
- ◆ View activation information for the driver set
- ◆ View the status log for the driver set

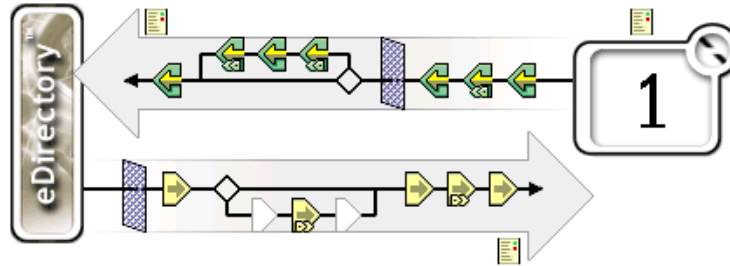
Driver Object

A driver object represents a driver that connects to an application that integrates with eDirectory. The following components comprise the driver object and its configuration parameters:

- ◆ A driver object in the eDirectory tree contained by a driver set object.
- ◆ A Subscriber channel object contained by the Driver object.
- ◆ A Publisher object contained by the Driver object.
- ◆ Several policy objects that are referenced by the Driver, Subscriber, and Publisher objects.
- ◆ An executable driver shim that is referenced by the Driver object.
- ◆ Shim-specific parameters that are configured by the administrator.
- ◆ An eDirectory password for the Driver object, which can be used by the shim to authenticate a remote part of the shim.
- ◆ Authentication parameters used to connect to and authenticate to the supported directory or application.
- ◆ A startup option for the driver that includes the following:
 - ◆ Disabled: The driver will not run.
 - ◆ Manual: The driver must be started manually through iManager.
 - ◆ Auto start: The driver will start automatically when eDirectory starts.
- ◆ A reference to a Schema Mapping policy.
- ◆ An XML representation of the supported application or directory's schema. This is typically obtained automatically from the application or directory through the shim.

In iManager, you can view the DirXML Driver Overview and modify existing driver parameters, rules, and style sheets. The DirXML Driver Overview is shown below.

Driver: 1.DS.Novell



In addition, the driver object is used for eDirectory rights checking. The driver object must be granted sufficient eDirectory rights to any object it reads or writes. You can do this by making the driver object a Trustee of the eDirectory objects the driver will synchronize with, or by granting Security Equivalences to the driver object.

See "eDirectory Rights" (<http://www.novell.com/documentation/lg/edir87/edir87/data/fbachifb.html>) in the *Novell eDirectory Administration Guide* for more information on rights assignments.

Driver Shim

The driver shim serves as a conduit for information between the application, directory, or database and eDirectory. The shim is written in Java, C, or C++.

The communication between the DirXML engine and the driver shim is in the form of XML documents that describe events, queries, and results.

The following object events are supported by the shim:

- ◆ Add (creation)
- ◆ Modify
- ◆ Delete
- ◆ Rename
- ◆ Move

In addition, the shim must support a defined query capability so that Identity Manager can query the synchronized application, directory, or database.

When an event occurs in eDirectory that will cause an action in the synchronized application or directory, Identity Manager creates an XML document that describes the eDirectory event and submits it through the Subscriber channel to the driver shim.

When an event occurs in the synchronized application, directory, or database, the driver shim generates an XML document that describes the application event. The driver shim then submits the XML document to Identity Manager through the Publisher channel. After processing the event through any application rules, Identity Manager causes eDirectory to take the appropriate action.

Publisher and Subscriber Channels

DirXML drivers contain two channels for processing data: the Publisher channel and Subscriber channel. Each channel contains its own policies that define how to process and transform data.

Events and Commands

The distinction between events and commands in Identity Manager is important. If an element is being sent to a driver, the element is a command. If the element is being sent to Identity Manager, the element is an event notification. When the driver sends an event notification to Identity Manager, the driver is informing Identity Manager of a change that occurred in the application. Based on configurable rules, Identity Manager will then determine what commands, if any, must be sent to eDirectory.

When Identity Manager sends a command to the driver, Identity Manager has already taken an eDirectory event as input, applied the appropriate policies, and determined that the change in the application represented by the command is necessary.

Policies and Filters

Policies and filters give you the ability to control how data flows from one system to another. For detailed information on policies and filters, refer to the *Policy Builder and Driver Customization Guide* (<http://www.novell.com/documentation/lg/dirxml20/policies/data/boswupw.html>).

Associations

Most other identity management products require the connected application to store an identifier of some sort to map objects from an application to the directory. With Identity Manager, no changes are required of the application. Each object in eDirectory contains an association table that maps the eDirectory object with a unique identifier in the connected directories and applications. The table is reverse-indexed so that the connected application does not need to supply an eDirectory identifier (such as a distinguished name) to the integration driver when updating eDirectory.

The creation of an association between two objects happens when an event occurs to an object that has not yet been associated with another object in the network. For an association to be created, the minimum set of definable criteria must match between each object. For example, you can create a rule stating that if any two of four attributes match by more than 90% (full name, telephone number, employee ID, and e-mail address) the object will be associated.

Matching policies define the criteria for determining if two objects are the same. If no match is found for the changed object, a new object can be created. For this to occur, all of the minimum creation criteria must be met. These criteria are defined by a Create policy. Finally, the Placement policy defines where, in the naming hierarchy, the new object is created.

Associations can be created in one of two ways:

- ♦ As a match between objects
- ♦ As a new creation of an object in a specific location

After an association between objects is formed, this association remains in effect until the objects are deleted or the association is deleted by an eDirectory administrator.

Association Table

In Identity Manager, associations refer to the matching of objects in eDirectory with objects residing in connected systems. When Identity Manager is initially installed, the eDirectory schema is extended on NetWare[®] and Windows* NT*/2000. If you are using Solaris* or Linux*, the schema is not automatically extended. Part of this extension is a new attribute tied to the base class of all eDirectory objects. This attribute is an association table. Association tables keep track of all the external application objects that an eDirectory object is linked to. This table is built and maintained automatically so there is rarely a reason to manually edit this information, although it is often helpful to view it.

2

Planning

In this section:

- ♦ “Planning the Project Management Aspects of Identity Manager Implementation” on page 25
- ♦ “Planning the Technical Aspects of Identity Manager Implementation” on page 31

Planning the Project Management Aspects of Identity Manager Implementation

This section outlines high-level political and project management aspects of implementing Identity Manager. (For the technical aspects, see “[Planning the Technical Aspects of Identity Manager Implementation](#)” on page 31.)

This planning material provides an overview of the type of activities that would normally be taken from the inception of an Identity Manager project to its full production deployment. Implementing an identity management strategy requires you to discover what the needs are and who the stakeholders are in your environment, design a solution, get buy-in from stakeholders, and test and roll out the solution. This section is intended to provide you with sufficient understanding of the process so that you can maximize the benefit from working with Identity Manager.

We strongly recommend that a trained Identity Manager expert be engaged to assist in each phase of the solution deployment. For more information about partnership options, see the [Novell Nsure Solution Partner Web site \(http://www.novell.com/solutions/nsure/partners\)](http://www.novell.com/solutions/nsure/partners). Novell Education also offers courses that address Identity Manager implementation.

This section is not exhaustive; it is not intended to address all possible configurations, nor is it intended to be rigid in its execution. Each environment is different and will require flexibility in the type of activities to be used.

Novell Identity Manager Deployment

There are several activities suggested as best practices when deploying Identity Manager:

- ♦ “Discovery” on page 26
- ♦ “Requirements and Design Analysis” on page 26
- ♦ “Proof of Concept” on page 29
- ♦ “Data Validation and Preparation” on page 30
- ♦ “Production Pilot” on page 30
- ♦ “Production Rollout Planning” on page 30
- ♦ “Production Deployment” on page 31

Discovery

You might want to begin your Identity Manager implementation with a discovery process that can do the following:

- ◆ Identify the primary objectives in managing identity information
- ◆ Define or clarify the business issues being addressed
- ◆ Determine what initiatives are required to address outstanding issues
- ◆ Determine what it would take to carry out one or more of these initiatives
- ◆ Develop a high-level strategy or "solution roadmap" and an agreed execution path

Discovery provides a common understanding of the issues and solutions for all stakeholders. It provides an excellent primer for the analysis phase that requires stakeholders to have a basic knowledge of directories, Novell eDirectory™, Novell Nsure Identity Manager®, and XML integration in general.

- ◆ It can establish a base level understanding among all stakeholders
- ◆ It can capture key business and systems information from stakeholders
- ◆ It can enable a solution roadmap to be developed

The discovery also identifies immediate next steps, which might include the following:

- ◆ Identifying planning activities in preparation of a requirements and design phase
- ◆ Defining additional education for stakeholders

Key Deliverables

- ◆ Structured interviews with key business and technical stakeholders
- ◆ High-level summary report of the business and technical issues
- ◆ Recommendations for the next steps
- ◆ An executive presentation outlining the outcome of the discovery

Requirements and Design Analysis

This analysis phase captures both technical and business aspects of the project in detail and produces the data model and high-level Identity Manager architecture design. This activity is a crucial first step from which the solution is implemented.

The focus of the design will be specifically on identity management; however, many of the elements traditionally associated with a resource management directory, such as file and print, can also be addressed. Here is a sample of items that you may want to assess:

- ◆ What versions of system software are being used?
- ◆ Is the directory design appropriate?
- ◆ Is the quality of the data in all systems appropriate? (If the data are not of usable quality, business policy may not be implemented as desired.)

After the requirements analysis, you can establish the scope and project plan for the implementation, and can determine if any prerequisite activities need to occur. To avoid costly mistakes, be as complete as possible in gathering information and documenting requirements.

The following tasks might be completed during the requirements assessment:

- ♦ “Define the Business Requirements” on page 27
- ♦ “Analyze Your Business Processes” on page 28
- ♦ “Design an Enterprise Data Model” on page 28

Define the Business Requirements

Gather your organization's business processes and the business requirements that define these business processes.

For example, a business requirement for terminating an employee might be that the employee's network and e-mail account access must be removed the same day the employee is terminated.

The following tasks can guide you in defining the business requirements:

- ♦ Establish the process flows, process triggers, and data mapping relationships.
For example, if something is going to happen in a certain process, what will happen because of that process? What other processes are triggered?
- ♦ Map data flows between applications.
- ♦ Identify data transformations that need to take place from one format to another, such as 2/25/2002 to 25 Feb 2002.
- ♦ Document the data dependencies that exist.
If a certain value is changed, it is important to know if there is a dependency on that value. If a particular process is changed, it is important to know if there is a dependency on that process.
For example, selecting a "temporary" employee status value in a human resources system might mean that the IT department needs to create a user object in eDirectory with restricted rights and access to the network during certain hours.
- ♦ List the priorities.
Not every requirement, wish, or desire of every party can be immediately fulfilled. Priorities for designing and deploying the provisioning system will help plan a roadmap.
It might be advantageous to divide the deployment into phases that will enable implementation of a portion of the deployment earlier and other portions of the deployment later.
- ♦ Define the prerequisites.
The prerequisites required for implementing a particular phase of the deployment should be documented.
- ♦ Identify authoritative data sources.
Learning early on which items of information system administrators and managers feel belong to them can help in obtaining and keeping buy-in from all parties.
For example, the account administrator might want ownership over granting rights to specific files and directories for an employee. This can be accommodated by implementing local trustee assignments in the account system.

Analyze Your Business Processes

The analysis of business processes often commences by interviewing essential individuals such as managers, administrators, and employees who actually use the application or system. Issues to be addressed include:

- ◆ Where does the data originate?
- ◆ Where does the data go?
- ◆ Who is responsible for the data?
- ◆ Who has ownership for the business function to which the data belongs?
- ◆ Who needs to be contacted to change the data?
- ◆ What are all the implications of the data being changed?
- ◆ What work practices exist for data handling (gathering and/or editing)?
- ◆ What types of operations take place?
- ◆ What methods are used to ensure data quality and integrity?
- ◆ Where do the systems live (on what servers, in which departments)?
- ◆ What processes are not suitable for automated handling?

For example, questions that might be posed to an administrator for a PeopleSoft system in Human Resources may include

- ◆ What data are stored in the PeopleSoft database?
- ◆ What appears in the various panels for an employee account?
- ◆ What actions are required to be reflected across the provisioning system (such as add, modify, or delete)?
- ◆ Which of these are required? Which are optional?
- ◆ What actions need to be triggered based on actions taken in PeopleSoft?
- ◆ What operations/events/actions are to be ignored?

Interviewing key people can lead to other areas of the organization that can provide a more clear picture of the entire process.

Design an Enterprise Data Model

After your business processes have been defined, you can begin to design a data model that reflects your current business process.

The model should illustrate where data originates, where they move to, and where they can't move. It should also account for how critical events affect the data flow.

You might also wish to develop a diagram that illustrates the proposed business process and the advantages of implementing automated provisioning in that process.

The development of this model begins by answering questions such as the following:

- ◆ What types of objects (users, groups, etc) are being moved?
- ◆ Which events are of interest?
- ◆ Which attributes need to be synchronized?
- ◆ What data is stored throughout your business for the various types of objects being managed?

- ♦ Is the synchronization one-way or two-way?
- ♦ Which system is the authoritative source for which attributes?

It is also important to consider the interrelationships of different values between systems.

For example, an employee status field in PeopleSoft might have three set values: employee, contractor, and intern. However, the Active Directory system might have only two values: permanent and temporary. In this situation, the relationship between the "contractor" status in PeopleSoft and the "permanent" and "temporary" values in Active Directory needs to be determined.

The focus of this work should be to understand each directory system, how they relate to each other, and what objects and attributes need to be synchronized across the systems.

Key Deliverables

- ♦ Data model showing all systems, authoritative data sources, events, information flow and data format standards
- ♦ Appropriate Identity Manager architecture for the solution
- ♦ Detail for additional system connection requirements
- ♦ Strategies for data validation and record matching
- ♦ Directory design to support the Identity Manager infrastructure

Dependencies

- ♦ Staff familiar with all external systems (such as HR database administrator, network and messaging system administrator)
- ♦ Availability of system schemas and sample data
- ♦ Data model from the analysis and design phase
- ♦ Availability of basic information such as organizational chart, WAN and server infrastructure

Proof of Concept

The outcome of this activity is to have a sample implementation in a lab environment that reflects your company's business policy and data flow. It is based on the design of the data model developed during the requirement analysis and design and is a final step before the production pilot.

NOTE: This step is often beneficial in gaining management support and funding for a final implementation effort.

Key Deliverables

- ♦ A functioning Identity Manager proof of concept with all system connections operational

Dependencies

- ♦ Hardware platform
- ♦ Necessary software
- ♦ Analysis and design phase that identifies the required connections
- ♦ Availability and access to other systems for testing purposes
- ♦ Data model from the analysis and design phase

Data Validation and Preparation

The data in production systems can be of varying quality and consistency and therefore may introduce inconsistencies when synchronizing systems. This phase presents an obvious point of separation between the Nsure Resources implementation team and the business units or groups who "own" or manage the data in the systems to be integrated. At times, the associated risk and cost factors may not belong in a provisioning project.

Key Deliverables

- ◆ Production data sets appropriate for loading into eDirectory (as identified in the analysis and design activities). This includes the likely method of loading (either bulk load or via connectors). The requirement for data that are validated, or otherwise formatted is also identified.

Dependencies

- ◆ Data model from analysis and design phase (proposed record matching and data format strategy)
- ◆ Access to production data sets

Production Pilot

The purpose of this activity is to begin the migration into a production environment. During this phase, there may be additional customization that occurs. In this limited introduction, desired outcomes of the preceding activities can be confirmed and agreement obtained for production rollout.

NOTE: This phase might provide the acceptance criteria for the solution and/or the necessary milestone en route to full production.

Key Deliverables

- ◆ Pilot solution providing live proof of concept and validation for data model and desired process outcomes

Dependencies

- ◆ All previous activities (analysis and design, Identity Manager technology platform).

Production Rollout Planning

This phase is where the production deployment is planned. The plan should:

- ◆ Confirm server platforms, software revisions, and service packs
- ◆ Confirm the general environment
- ◆ Confirm introduction of eDirectory and mixed tree coexistence
- ◆ Confirm partitioning and replication strategies
- ◆ Confirm Identity Manager implementation
- ◆ Plan the legacy process cutover
- ◆ Plan a rollback contingency strategy

Key Deliverables

- ♦ Production rollout plan
- ♦ Legacy process cutover plan
- ♦ Rollback contingency plan

Dependencies

- ♦ All previous activities

Production Deployment

This phase is where the pilot solution is expanded to affect all live data in the production environment. It typically follows agreement that the production pilot meets all the technical and business requirements.

Key Deliverables

- ♦ Production solution ready for transition

Dependencies

- ♦ All previous activities

Planning the Technical Aspects of Identity Manager Implementation

Replicating the Objects that Identity Manager Needs on the Server

As part of your planning, you need to make sure that certain eDirectory objects are replicated on servers where you want to run DirXML drivers.

You can use filtered replicas, as long as all of the objects and attributes that the driver needs to read or synchronize are included in the filtered replica.

Keep in mind that you must give the DirXML Driver object sufficient eDirectory rights to any objects it is to synchronize, either by explicitly granting it rights or by making the Driver object security equivalent to an object that has the desired rights.

An eDirectory server that is running a DirXML driver (or that the driver refers to, if you are using Remote Loader) must hold a master or read-write replica of the following:

- ♦ The Driver Set object for that server.

You should have one Driver Set object for each server that is running Identity Manager. Unless you have specific needs, don't associate more than one server with the same Driver Set object.

NOTE: When creating a Driver Set object, the default setting is to create a separate partition, but this is not required.

- ♦ The Server object for that server.

The Server object is necessary because it allows the driver to generate key pairs for objects. It also is important for remote loader authentication.

- ♦ The objects that you want this instance of the driver to synchronize.

The driver can't synchronize objects unless a replica of those objects is on the same server as the driver. In fact, a DirXML driver will synchronize the objects in *all* the containers that are replicated on the server unless you create rules to specify otherwise (rules for "scope filtering").

If you want a driver to synchronize all user objects, for example, the simplest way is to use one instance of the driver on a server that holds a master or read/write replica of all your users.

However, many environments don't have a single server that contains a replica of all the users. Instead, the complete set of users is spread across multiple servers. In this case, you have two choices:

- ♦ **Aggregate users onto a single server.** You can create a single server that holds all users by adding replicas to an existing server. Filtered replicas can be used to reduce the size of the eDirectory database if desired, as long as the necessary user objects and attributes are part of the filtered replica.
- ♦ **Use multiple instances of the driver on multiple servers, with scope filtering.** If you *don't* want to aggregate users onto a single server, you will need to determine which set of servers holds all the users, and set up one instance of the DirXML driver on each of those servers.

To prevent separate instances of a driver from trying to synchronize the same users, you will need to use "scope filtering" to define which users each instance of the driver should synchronize. Scope filtering means that you add rules to each driver to limit the scope of the driver's management to specific containers. See [“Managing Users on Different Servers Using Scope Filtering” on page 32](#).

- ♦ The Template objects you want the driver to use when creating users, if you choose to use templates.

DirXML drivers do not require you to specify eDirectory Template objects for creating users. But if you specify that a driver should use a template when creating users in eDirectory, the Template object must be replicated on the server where the driver is running.

- ♦ Any containers you want the DirXML driver to use for managing users.

For example, if you have created a container named Inactive Users to hold user accounts that have been disabled, you must have a master or read/write replica (preferably master replica) of that container on the server where the driver is running.

- ♦ Any other objects that the driver needs to refer to (for example, work order objects for the Avaya PBX driver).

If the other objects are only to be read by the driver, not changed, the replica for those objects on the server can be a read-only replica.

Managing Users on Different Servers Using Scope Filtering

Scope filtering means adding rules to each driver to limit the scope of the driver's actions to specific containers. The following are two situations in which you would need to use scope filtering:

- ♦ You want the driver to synchronize only users that are in a particular container.

A DirXML driver by default will synchronize objects in all the containers that are replicated on the server where it is running. To narrow that scope, you must create scope filtering rules.

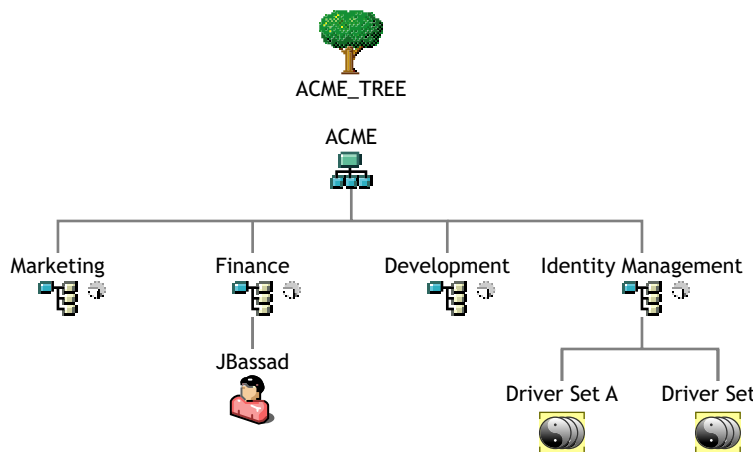
- ♦ You want a DirXML driver to synchronize all users, but you don't want all users to be replicated on the same server.

To synchronize all users without having them replicated on one single server, you will need to determine which set of servers holds all the users, and then create an instance of the DirXML driver on each of those servers. To prevent two instances of the driver from trying to synchronize the same users, you will need to use scope filtering to define which users each instance of the driver should synchronize.

NOTE: You should use scope filtering even if your server's replicas don't overlap currently. In the future, replicas could be added to your servers and an overlap could be created unintentionally. If you have scope filtering in place, your DirXML drivers will not try to synchronize the same users, even if replicas are added to your servers in the future.

Here's an example of how scope filtering is used.

The following illustration shows a tree with three containers that contain users: Marketing, Finance, and Development. Each is a separate partition.



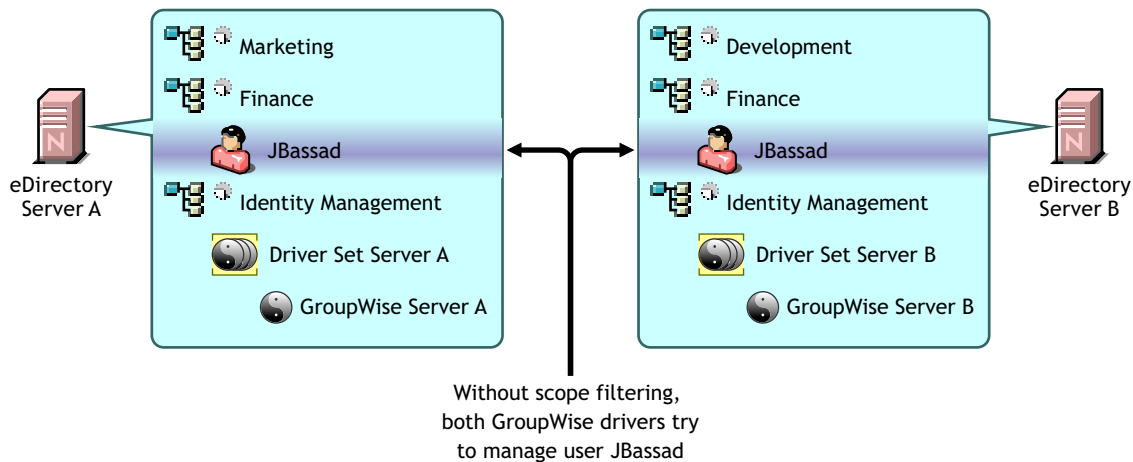
In this example, the eDirectory administrator has two eDirectory servers, Server A and Server B, shown in the next illustration. Neither server contains a copy of all the users. Each server contains two of the three partitions, so the scope of what the servers hold is overlapping.

The administrator wants all the users in the tree to be synchronized by the GroupWise driver, but does not want to aggregate replicas of the users onto a single server. He chooses instead to use two instances of the GroupWise driver, one on each server. He installs Identity Manager and sets up the GroupWise driver on each eDirectory server.

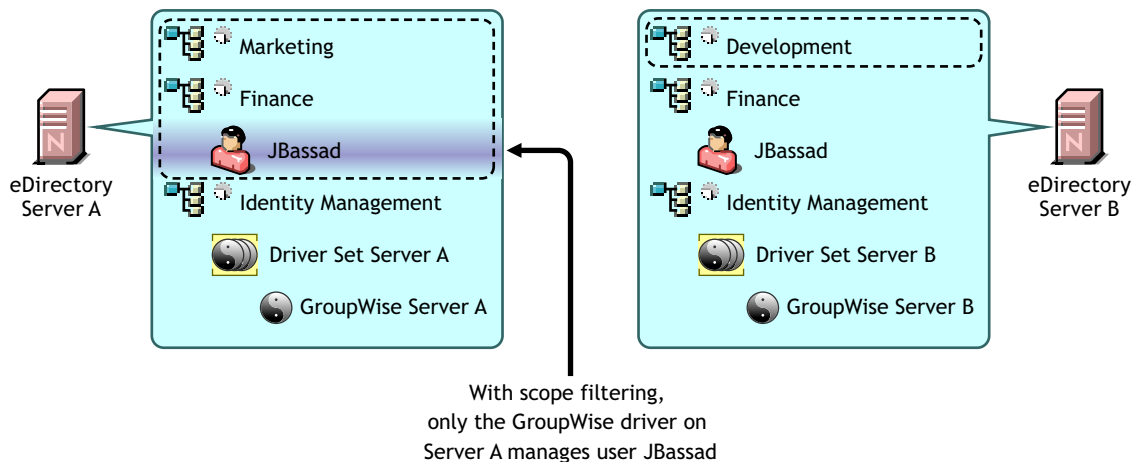
Server A holds replicas of the Marketing and Finance containers. Also on the server is a replica of the Identity Management container, which holds the Driver Set for Server A and the GroupWise Driver object for Server A.

Server B holds replicas of the Development and Finance containers, and the Identity Management container holding the Driver Set for Server B and the GroupWise Driver object for Server B.

Because Server A and Server B both hold a replica of the Finance container, both servers hold the user JBassad, who is in the Finance container. Without scope filtering, both GroupWise Driver A and GroupWise Driver B would synchronize JBassad.



The next illustration shows that scope filtering prevents the two instances of the driver from managing the same user, because it defines which drivers synchronize each container.



Here is a sample of how you would create a rule for scope filtering. You would place the rule in the Subscriber Event Transformation style sheet.

```
<xsl:transform version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:jstring="http://www.novell.com/nxsl/java/java.lang.String"
  exclude-result-prefixes="jstring">
```

```
<!--
```

```
To select different containers for scoping, add/delete/modify the <value>
elements in the body of the variable "in-scope-containers-rtf"
```

```
Note that if the container is not in the root of the tree, then the DN
(minus the tree name) of the container must be specified, e.g.,
Corporate\Executives
```

```
Note: THESE MUST BE ENTERED IN THE TABLE AS ALL UPPERCASE
-->
```

```
<xsl:variable name="in-scope-containers-rtf">
  <value>CORPORATE\USERS\ACTIVE</value>
  <value>CORPORATE\USERS\INACTIVE</value>
</xsl:variable>
```

```

    <xsl:variable name="in-scope-containers" select="document('')/xsl:transform/
xsl:variable[@name='in-scope-containers-rtf']/value"/>

<!--
"identity" transformation - copies unchanged everything not explicitly
matched by other templates
-->

<xsl:template match="node()|@"*>
    <xsl:copy>
        <xsl:apply-templates select="@*|node()"/>
    </xsl:copy>
</xsl:template>

<!-- throw away events that are out of scope -->

<xsl:template match="input/*[@src-dn]">
    <xsl:variable name="in-scope">
        <xsl:call-template name="in-scope"/>
    </xsl:variable>
    <xsl:choose>
        <xsl:when test="$in-scope = '1'">
            <xsl:copy>
                <xsl:apply-templates select="@*|node()"/>
            </xsl:copy>
        </xsl:when>
        <xsl:otherwise>
            <xsl:message>
                <status level="warning">Operation vetoed by Event Transformation
                Rule - out of scope</status>
            </xsl:message>
        </xsl:otherwise>
    </xsl:choose>
</xsl:template>

<!--
check to see if an object is in the scope defined by the variable
"in-scope-containers"
-->

<xsl:template name="in-scope">
    <!-- validate that the container is in scope -->
    <xsl:variable name="src-dn" select="substring-after(substring-after(@src-dn,'\'),'\\')"/>
    <xsl:variable name="src-dn-i" select="jstring:indexOf($src-dn,'\')"/>
    <xsl:if test="$src-dn-i != -1">
        <xsl:variable name="src-dn-container" select="jstring:substring($src-dn, 0, $src-dn-
i)"/>

        <!--
        the following test takes advantage of the XPath existential
        quantification semantics:
        basically, if one node in the node-set has a string value that matches
        the string, then the statement is true
        -->

        <xsl:if test="jstring:toUpperCase($src-dn-container) = $in-scope-containers">
            <xsl:value-of select="'1'"/>
        </xsl:if>
    </xsl:if>
</xsl:template>
</xsl:transform>

```


3

Upgrading

This section explains how to upgrade Identity Manager:

- ♦ “Upgrading Password Synchronization” on page 37
- ♦ “Upgrading from RNS to Nsure Audit” on page 37
- ♦ “Upgrading Driver Configurations” on page 37

Upgrading Password Synchronization

See “Upgrading Password Synchronization 1.0 to Identity Manager Password Synchronization” on page 132.

Upgrading from RNS to Nsure Audit

The DirXML engine that ships with Identity Manager 2 is backward compatible with RNS. The engine will continue to process RNS functions if you are currently using it. However, we recommend that you move to Nsure Audit because it provides all the functionality of RNS with the addition of many new features.

For more information, see the *NSure Audit 1.0 Administration Guide* (<http://www.novell.com/documentation/lg/nsureaudit/index.html>).

Upgrading Driver Configurations

Upgrading driver configurations has two aspects:

- ♦ Converting the rules to Identity Manager policies. This is done by a conversion tool, and it does not enhance the functionality of the driver. Legacy drivers run without this conversion, but doing the conversion allows you to view the existing driver configuration in the DirXML iManager plugins.
- ♦ Upgrading the driver policies to add new functionality. This is best handled by an Identity Manager expert.

See “Upgrading a Driver Configuration from DirXML 1.x to Identity Manager Format” on page 63, and “Managing DirXML 1.x Drivers in an Identity Manager Environment” on page 62.

Another alternative is to begin with the Identity Manager driver configurations and customize them to do the same things your DirXML 1.x configuration does.

4

Installation

This section contains requirements and instructions for installing Identity Manager and DirXML[®] drivers.

- ♦ “Before You Install” on page 39
- ♦ “Installing Identity Manager Components” on page 40
- ♦ “Identity Manager System Requirements” on page 41
- ♦ “Feature Support for eDirectory 8.6.2 and eDirectory 8.7.3” on page 43
- ♦ “Installing Identity Manager on NetWare” on page 44
- ♦ “Installing Identity Manager on Windows” on page 45
- ♦ “Installing Identity Manager on Linux or Solaris” on page 46
- ♦ “Post-Installation Tasks” on page 47
- ♦ “Using the Remote Loader Service” on page 47
- ♦ “Activating Identity Manager Products” on page 58
- ♦ “Installing a Custom Driver” on page 59

Before You Install

Before you install Identity Manager, review the following information:

- ♦ Review the planning information found in “Planning the Project Management Aspects of Identity Manager Implementation” on page 25 and “Planning the Technical Aspects of Identity Manager Implementation” on page 31.
- ♦ Ensure that you meet all “Identity Manager System Requirements” on page 41.
- ♦ You should back up your Novell[®] eDirectory[™] server. Refer to the Novell documentation for [Backing Up and Restoring eDirectory](http://www.novell.com/documentation/lg/edir871/edir871/data/a2n4mb6.html) (<http://www.novell.com/documentation/lg/edir871/edir871/data/a2n4mb6.html>).
- ♦ A Identity Manager installation on a server will synchronize only the information that is contained physically in the host eDirectory server’s partition replicas. This may necessitate aggregating several partitions onto a single server if a tree-wide view of eDirectory data is desired for a particular synchronization application. For more information, see “Replicating the Objects that Identity Manager Needs on the Server” on page 31.
- ♦ The Driverset object must exist in a full read/write replica on the server that is hosting the drivers.
- ♦ The Driver object must be granted sufficient eDirectory rights to any objects it is to synchronize, or the Driver object must be made security equivalent to an object that has the desired rights.

Installing Identity Manager Components

Nsure™ Identity Manager contains components that can be installed within your environment on multiple systems and platforms. Depending on your system configuration, you might need to run the Identity Manager installation program several times to install Identity Manager components on the appropriate systems.

The following images illustrate the various system types and the components you can install per system.

A DirXML Server

The DirXML Engine and core components run on this server.



DirXML Components Include:

- The DirXML engine
- Nsure Audit agent
- DirXML Service drivers
- DirXML Driver for eDirectory
- Novell Modular Authentication Services (NMAS) components

Prerequisites:

A supported Operating System:

- NetWare 6, or 6.5
- Windows NT, 2000, XP or 2003
- Linux Red Hat or SuSE
- Solaris 8 or 9
- eDirectory 8.6.2 or 8.7.3

NOTE: eDirectory 8.7.3 is required to use full password functionality

DirXML Connected System Server

This server establishes the link between the connected system and the DirXML Server. A connected system server need not be an eDirectory server.



DirXML Components Include:

- The DirXML Remote Loader
- Remote Loader configuration tool
- Nsure Audit agent
- Driver Shim for the connected system
- Tools for this connected system

Prerequisites:

A supported Operating System

Web-based Administration Server

This server contains the administration and management software you can use with DirXML.



DirXML Components Include:

- DirXML & Password Mgt. plug-ins
- Driver configurations
- End user password self-service
- eGuide

Prerequisites:

A supported OS:

- NetWare 6 or 6.5
- Windows 2000
- Linux Red Hat or SuSE
- Solaris 8 or 9

A supported Web Server and Servlet Engine

Novell iManager 2.0.x or later

DirXML Utilities

These additional utilities can be installed on the administration server, connected system, or other computers you specify.



DirXML Components Include:

- DirXML License Auditing Tool
- Sample PeopleSoft application
- Nsure Audit Setup tool

Prerequisites:

A supported OS

Identity Manager System Requirements

The following table lists the minimum software requirements for each system.

System Component	System Requirements	Notes
DirXML Server	<p>One of the following operating systems.</p> <ul style="list-style-type: none">♦ NetWare® 6 or 6.5 with the latest Support Pack♦ Windows NT or 2000 with the latest Service Pack♦ Linux Red Hat AS 2.1 or ES 2.1♦ SuSE Linux Enterprise Server 8♦ Solaris 8 or 9 <p>One of the following versions of eDirectory.</p> <ul style="list-style-type: none">♦ eDirectory 8.6.2 with the latest Support Pack♦ eDirectory 8.7.3 with the latest Support Pack	<ul style="list-style-type: none">♦ Some features are not supported on eDirectory 8.6.2. See versions of eDirectory, see "Feature Support for eDirectory 8.6.2 and eDirectory 8.7.3" on page 43.♦ To use the full password capabilities provided with Identity Manager, eDirectory 8.7.3 must be installed on your identity vault server.♦ The DirXML server is not yet supported on Windows 2003.

System Component	System Requirements	Notes
Connected System Server	Refer to the Identity Manager Driver documentation (http://www.novell.com/documentation/lg/dirxml/drivers) for operating system and connected system requirements specific to each system.	<ul style="list-style-type: none"> Identity Manager supports using Microsoft Windows Server 2003 as the platform for connected systems that are "remote loaded." In other words, you can install a driver shim and remote loader service on the Windows 2003 computer, and it will communicate correctly with the DirXML engine on a separate box.
Web-based Administration Server	<ul style="list-style-type: none"> NetWare 6 or 6.5 with the latest Support Pack Windows 2000, XP, or 2003 with the latest Service Pack Linux Red Hat AS 2.1 or ES 2.1 (Glibc version 2.1.1 or later and Kernel version 2.2.xx or later.) Solaris 8 or 9 Novell iManager 2.0.2. Apache 2.0.44 or later and Tomcat 4.1.18 or later 	<ul style="list-style-type: none"> Browser support is determined by iManager 2.0.2. For details and known issues, refer to the Novell iManager Administration guide (http://www.novell.com/documentation/lg/imanager20/imanager20/data/bobxl9n.html). You must go through the iManager Configuration Wizard to install portal content into eDirectory. This is required to use Identity Manager tasks out of the box, such as Password Self-Service and Forgotten Password features. This can be done during installation of iManager, which is the default. If you choose not to do it during installation, you must do it post-installation. If you are installing eGuide, you must have a Web server and an Application server installed prior to running the installation program. If you install iManager 2.0.2 on the same server where eDirectory is installed, the version of eDirectory must be 8.7.3. (Netware) In order to install Novell eGuide 2.1.2, a valid JVM must be installed on the server. (Windows) The Novell Client™ 4.83 is available from Novell Software Downloads (http://www.novell.com/download/index.html). When logging into other trees with iManager to manage remote Identity Manager servers, you might encounter errors if you use the server name instead of the IP address for the remote server. Additionally, the LDAP server group object in NDS must be configured to require TLS on simple binds, and the trusted root certificate of the remote tree must be imported as a trusted certificate onto the web server.
DirXML Utilities	Refer to the Identity Manager Driver documentation (http://www.novell.com/documentation/lg/dirxml/drivers) for requirements specific to each system.	

Feature Support for eDirectory 8.6.2 and eDirectory 8.7.3

The following table lists the features that are not supported when running eDirectory 8.6.2, and also points out a few considerations for eDirectory 8.7.3.

NOTE: Versions earlier than eDirectory 8.6.2 are referred to as legacy NDS®. The DirXML engine that ships with Identity Manager cannot run on legacy NDS.

Identity Manager has not been tested with eDirectory 8.7, so it is not supported. However, eDirectory 8.7.3 is supported, and it is a free upgrade for eDirectory 8.7.

Feature	eDirectory 8.6.2	eDirectory 8.7.3
Policy Builder	Supported.	Supported.
DirXML Script	Supported.	Supported.
Password Policies: Advanced Password Rules	<p>Not supported by Identity Manager on eDirectory 8.6.2, because the password rules require Universal Password.</p> <p>However, in a mixed environment, Advanced Password Rules could be enforced for an 8.6.2 tree if you are synchronizing between two trees and one is eDirectory 8.7.3. For example, if you have an identity vault running 8.7.3, and you only allow users to change passwords in that tree, you could turn on Universal Password in the identity vault and then synchronize one-way to the eDirectory 8.6.2 tree. You could synchronize the Universal Password to the NDS password, and enforce the password rules.</p> <p>If eDirectory 8.6.2 is being used, the password restrictions you can use are the ones available for the NDS password.</p>	<p>Supported, if Universal Password is enabled in a Password Policy.</p> <p>You can also enforce Password Policies on connected systems.</p>
Password Policies: Forgotten Password Self-Service	<p>All the features are supported except the following:</p> <ul style="list-style-type: none">♦ Allow user to reset password on page♦ E-mail current password to user <p>This feature requires a reversible password. Because eDirectory 8.6.2 does not support Universal Password, this feature is not available.</p>	<p>All features supported if Universal Password is enabled for the Password Policy.</p> <p>If Universal Password is disabled for a Password Policy, the administrator cannot provide the following options for users of that policy:</p> <ul style="list-style-type: none">♦ Allow user to reset password on page♦ E-mail current password to user <p>These features require a reversible password, so they can't be used if Universal Password is not enabled.</p>
Password Policies: Challenge Sets	Supported.	Supported.

Feature	eDirectory 8.6.2	eDirectory 8.7.3
Password Policies: Reset Password Self-Service	Supported. The Reset Password gadget changes the NDS password if the Universal Password is not available, so it can be used on eDirectory 8.6.2.	Supported. The Reset Password gadget changes the NDS password if the Universal Password is not available, so it can be used even if the Universal Password is not enabled in a user's Password Policy.
Password Policies: Set Universal Password task	Not supported for changing the NDS password; instead, use the Modify Object task or other help desk task to change the user's NDS password.	Supported, if Universal Password is enabled. Unlike the Reset Password gadget, the Set Universal Password task works only if Universal Password is enabled in the user's Password Policy.
Password Synchronization	Only password publishing to Identity Manager is supported. Using 8.6.2, you can configure your drivers to mimic the same functionality that is provided with Password Synchronization 1.0, with the addition of support for new platforms. Identity Manager can accept passwords from connected systems to update the NDS password. But without Universal Password, Identity Manager can't distribute passwords to connected systems unless the system is another eDirectory tree.	Supported. However, if Universal Password is not enabled in a Password Policy, passwords can't be distributed to connected systems, and password policies can be enforced on incoming passwords but can't be enforced on the connected systems.
Role-Based Entitlements	Not supported. Entitlement Policies are dynamic groups, and some features of dynamic groups were not supported in eDirectory 8.6.2.	Supported.
Reporting and Notification	Supports Novell Nsure™ Audit. For upgrade customers only. It also supports RNS, which is the legacy reporting and notification service. The RNS plugins are included with Identity Manager; the RNS components for the DirXML engine are not included.	Supports Novell Nsure Audit. For upgrade customers only. It also supports RNS, which is the legacy reporting and notification service. The RNS plug-ins are included with Identity Manager; the RNS components for the DirXML engine are not included.
eGuide	Supported.	Supported.

Installing Identity Manager on NetWare

Before you begin, make sure your system meets the requirements listed in [“Identity Manager System Requirements” on page 41](#).

- 1 At the server console, enter **nwconfig.nlm**.
- 2 Select Product Options > Install a Product Not Listed.
- 3 Press F3 (F4 if you're using RCONSOLE), then specify the path to the Identity Manager NetWare installation files.

The graphical installation utility will start after a few moments.

4 Click Next.

After the files have finished copying, the DirXML Welcome Screen appears. Click Next to begin the installation.

5 Review the Overview pages describing the system types, then click Next to continue.

6 Read the license agreement, then click I Accept.

7 Select the components you want to install. The following options are available:

- ♦ **DirXML Server:** Installs the DirXML engine and service drivers; DirXML Drivers for eDirectory, LDAP, JDBC, GroupWise®, Delimited Text, and SIF; NMASTM components; Nsure Audit agent; and extends the eDirectory schema.

Novell eDirectory must be installed before you can install this option.

- ♦ **Connected System:** Installs the Remote Loader Service, and the following drivers: LDAP, JDBC, eDirectory, SAP, Delimited Text, and Lotus Notes.
- ♦ **DirXML Web Components:** Installs the DirXML plug-ins, DirXML driver configurations, and Novell eGuide.

Novell iManager must be installed before you can install this option.

- ♦ **Utilities:** Installs additional scripts for the JDBC driver.

8 Click Next.

9 In the Schema Extension page, specify the following:

- ♦ **User Name:** Specify the username (in LDAP format) of a user who has rights to extend the schema.
- ♦ **User Password:** Specify the user's password.

10 Click Next.

11 Read and verify your selections on the Summary page, then click Finish.

12 After the installation completes and displays the Installation Complete dialog box, click Close.

Installing Identity Manager on Windows

Before you begin, make sure your system meets the requirements listed in **“Identity Manager System Requirements” on page 41**.

1 Download and extract the Identity Manager installation file.

2 Run install.exe from the NT directory.

3 Read the Welcome information, then click Next.

4 Read the License Agreement, then click I Accept.

5 Review the Overview pages about the various systems and components, then click Next to begin the installation.

6 Select the components you want to install

- ♦ **DirXML Server:** Installs the DirXML engine and service drivers, DirXML drivers, NMASTM components, Nsure Audit agent, and extends the eDirectory schema.

Novell eDirectory must be installed before you can install this option.

- ♦ **Connected System:** Installs the Remote Loader Service and the DirXML drivers you select.
- ♦ **DirXML Web Components:** Installs the DirXML plug-ins, DirXML driver configurations, and Novell eGuide.

Novell iManager must be installed before you can install this option.

- ♦ **Utilities:** Installs the Application utilities you select.

7 In the Schema Extension page, specify the following:

- ♦ **User Name:** Specify the username (in LDAP format) of a user who has rights to extend the schema.
- ♦ **User Password:** Specify the user's password.

8 Select the Web components you want to install, then click Next.

9 Select the utilities you want to install, then click Next. The installation program then display the install path. If you would like to change the default location, type (or browse to) the desired location, then click Next.

10 Select the system components you want to install (JDBC, PeopleSoft, DirXML License Auditing Utility, Active Directory Discovery Tool, Lotus Notes Discovery Tool), then click Next.

11 Review the items listed in the Summary page. If you approve, click Finish to install the components.

12 Click Close to exit the installation program.

Installing Identity Manager on Linux or Solaris

Before you begin, make sure your system meets the requirements listed in **“Identity Manager System Requirements” on page 41.**

- 1** Download and extract the tar file to a location of your choice.
- 2** On the Solaris or Linux host, log in as root.
- 3** From the directory where you extracted the tar file, enter one of the following commands to run the installation program.

On Linux: `/unix/Linux/setup/dirxml_linux.bin`

On Solaris: `/unix/Solaris/setup/dirxml_solaris.bin`

- 4** Review the Welcome information, then press Enter to continue the installation.
- 5** Read the license agreement and enter **Y** if you agree to the usage terms. If not, enter **N** to exit the installation program.
- 6** Specify the appropriate number (1-4) for the install set you want to install. The install sets contain the following components:

- ♦ **DirXML Server:** Installs the DirXML engine and service drivers, DirXML drivers, NMAS components, Nsure Audit agent, and extends the eDirectory schema.

Novell eDirectory must be installed before you can install this option.

- ♦ **Connected System Server:** Installs the Remote Loader Service and the following drivers: LDAP, JDBC, eDirectory, SAP, Delimited Text, and Lotus Notes.

- ♦ **Web-Based Administration Server:** Installs the DirXML plug-ins, DirXML driver configurations, and Novell eGuide.

Novell iManager must be installed before you can install this option.

- ♦ **Customize:** Installs the specific components you select from a list of all components.

NOTE: Enter **prev** to return to previous menus and modify your installation options.

- 7 (Optional) Depending on the options you entered, you might be prompted to specify the LDAP user name and password, or the Web Server Secure port.
- 8 Verify that the information contained in the summary is correct because eDirectory temporarily shuts down (when installing the DirXML Engine and schema files.) If the Install Summary is correct, press Enter to start installing the packages.
- 9 When finished, enter **OK** to close the installation program.

Post-Installation Tasks

If eDirectory is running, it automatically launches the Identity Manager module. You do not need to manually load or unload Identity Manager. After Identity Manager is installed, you should configure it (and the drivers you installed) to meet the policies and requirements defined by your business processes. Post-installation tasks typically include the following items:

- ♦ Configuring an Application System (refer to the [Identity Manager Driver Documentation \(http://www.novell.com/documentation/lg/dirxml/drivers\)](http://www.novell.com/documentation/lg/dirxml/drivers) for specific driver configuration instructions.)
- ♦ “Creating and Configuring a Driver” on page 61
- ♦ “Creating Policies” on page 71
- ♦ “Starting, Stopping, or Restarting a Driver” on page 63
- ♦ “Activating Identity Manager Products” on page 58

Using the Remote Loader Service

The Remote Loader Service allows DirXML drivers running on one platform to communicate with the DirXML engine running on a different platform.

For example, the DirXML Driver for PeopleSoft must operate on the Windows 2000 platform. The Remote Loader lets you run the PeopleSoft driver on a Windows platform, and enables you to synchronize data with the DirXML engine running on a Solaris, Linux, or NetWare platform.

A new feature in Identity Manager 2 is the Remote Loader Console. The console runs on Windows and allows you to easily manage all of your remote loader instances from a single location. For information on using the console, refer to “[Using the Remote Loader Console on Windows](#)” on page 48.

This section contains information on the following:

- ♦ “Using the Remote Loader Console on Windows” on page 48
- ♦ “Configuring a DirXML Driver with the Remote Loader” on page 50

Using the Remote Loader Console on Windows

The Remote Loader Console allows you to easily manage remote instances of your DirXML drivers. Use the Console to add and configure new remote loader instances on the local computer, to edit configuration settings, and to start and stop Remote Loader instances.

NOTE: If you are upgrading to Identity Manager 2, the Console detects and imports existing instances of the Remote Loader (configurations must be stored in the /remoteloader directory to be imported automatically.) You can then use the Console to manage these remote drivers. Do not continue to use the Remote Loader Configuration Wizard that shipped prior to Identity Manager 2. You might encounter errors when using the Wizard and Console together.

When you install the Remote Loader service, an icon is installed on your desktop. Double-click the icon to run the Remote Loader Console.

Entering Information for Remote Loader Instances

When configuring a new remote loader instance, you will be prompted for the following information:

Remote Driver Configuration

- ◆ Description: Specify a description to identify the remote loader instance.
- ◆ Driver: Browse to and select the appropriate shim for your driver.
- ◆ Config File: Specify a name for the configuration file. The Remote Loader Console places configuration parameters into this text file.

Communication

- ◆ IP Address: Specify the IP address on which the Remote Loader service listens.
- ◆ Connection Port - DirXML Server: Specify the connection parameters to the DirXML server running the remote driver. The default TCP/IP port for this connection is 8090. With each new instance you create, the port number automatically increases by one.
- ◆ Command Port - Local Host Communication Only: Specify the TCP port number on which a Remote Loader service listens for commands such as Stop, Change Trace Level, and so forth. Each instance of the Remote Loader that is running on a particular computer must have a different command port number. The default command port is 8000. With each new instance you create, the port number automatically increases by one.

NOTE: You can run multiple instances of the Remote Loader (on the same server hosting different driver instances) by specifying different connection ports and command ports.

Remote Loader Password

- ◆ Password: This password is used to control access to a Remote Loader instance for a driver. It must be the same password you specify when configuring the driver to connect remotely (via the Driver Parameters page in Novell iManager.)
- ◆ Confirm: Re-enter the password to confirm.

Driver Object Password

- ◆ Password: This password is used by the Remote Loader to authenticate itself to the DirXML server. It must be the same password you specify when configuring the driver to connect remotely (via the Driver Parameters page in Novell iManager.)
- ◆ Confirm: Re-enter the password to confirm.

Secure Socket Link (SSL)

- ♦ Use an SSL Connection: Check this option to specify an SSL connection.
- ♦ Trusted Root File: Browse to the Java* keystore file containing the appropriate trusted root certification. This is the exported self-signed certificate from the eDirectory tree's Organization Certificate Authority. The certificate must be exported in Base64 format. For example, MY-TREE CA.b64.

Trace File

- ♦ Trace Level: Set a trace level greater than zero for the Remote Loader instance to display a trace window that contains informational messages from both the loader and the driver.
- ♦ Trace File: Specify a trace filename where trace messages will be written. Each Remote Loader instance running a particular machine must use a different trace file. Trace messages are written to the trace file only if the trace level is greater than zero.
- ♦ Maximum Disk Space Allowed for all Trace Logs (MB): Specify the approximate maximum size that trace file data can occupy on disk. If Unlimited is not selected, the default value is set to 4096 MB or 4 gigabytes.

Establish a Remote Loader Service for this Driver Instance

- ♦ Check this option to configure the Remote Loader instance as a service. This automatically starts the Remote Loader when the computer boots.

Configuring SSL between the DirXML Engine and the Remote Loader

The following instructions provide information regarding how to create and export certificates, and how to configure your SSL connection between the DirXML Engine and the Remote Loader for secure data transfers.

- ♦ [“Creating and Exporting a Certificate” on page 49](#)
- ♦ [“Configuring the Driver to use a SSL Connection” on page 50](#)

Creating and Exporting a Certificate

- 1** In Novell iManager, click Novell Certificate Server > Create Server Certificate.
- 2** Select the server that will own the certificate and give it a certificate nickname. (For example, remotecert.)
IMPORTANT: Write down the certificate nickname, because you will use this for the KMO name in the driver's remote connection parameters. (For example, remotecert)
- 3** Leave the Creation method set to Standard, then click Next.
- 4** Review the Summary, then click Finish.
You've just created a server certificate. You now need to export the Trusted Root certificate.
- 5** Click eDirectory Administration > Modify Object.
- 6** Browse to the Certificate Authority in the Security container, then click OK.
The Certificate Authority (CA) is named after the tree name (Treename-CA.Security).
- 7** Click the Certificates tab, then click Export.
- 8** In the Export Certificate Wizard, select No, then click Next. You do not want to export the private key with the certificate.

- 9** Choose to export the file in Base64 format, then click Next.
- 10** Click the link to Save the Exported Certificate to a File, specify a location, then click Save.
- 11** Copy this file to a secure location that can be accessed by Remote Loader service.

You need to configure the Remote Loader parameters in the driver to use SSL. Refer to [“Configuring the Driver to use a SSL Connection” on page 50](#) for more information.

Configuring the Driver to use a SSL Connection

Before configuring your SSL connection, make sure that you have exported the Trusted Root certificate and that the Remote Loader has access to the exported file. Refer to [“Creating and Exporting a Certificate” on page 49](#) for detailed instructions.

You now need to modify the driver parameters to use this certificate.

- 1** In Novell iManager, click DirXML Management > Overview.
- 2** Browse to the driver object for which you want to configure an SSL connection.
- 3** Append `kmo=kmo object nickname` to the Remote Loader Connection parameters. For example, `hostname=198.162.0.1 port=8090 kmo=remotecert`

If you used spaces in the certificate name, you need to enclose the KMO object nickname in quotation marks.

TIP: The KMO object name is the nickname value you specified in Step 2 of [“Creating and Exporting a Certificate” on page 49](#).

Configuring a DirXML Driver with the Remote Loader

Complete these tasks to configure a driver you want to use with the Remote Loader:

- ♦ [“Installing the Remote Loader Service and Drivers” on page 50](#)
- ♦ [“Configuring the Driver Object Properties for the Remote Loader” on page 50](#)

Installing the Remote Loader Service and Drivers

For information on installing the Remote Loader Service and remote DirXML drivers, refer to [“Installing Identity Manager Components” on page 40](#).

Configuring the Driver Object Properties for the Remote Loader

After you have installed the Remote Loader and DirXML Driver, you must also specify parameters for the Remote Loader on the driver object.

- 1** In Novell iManager, click DirXML Management > Overview.
- 2** Browse to the driver object you want to configure remotely.
- 3** Click the driver status icon, then click Edit Properties.
- 4** The Driver Configuration parameters will display. Enter the following Authentication information to configure the Remote Loader settings:
 - ♦ Remote Loader Connection parameters: Enter the communications parameters for the Remote Loader. You enter three parameters as a series of “name=value” pairs for these connection parameters. This is an example of the full communications parameters:
`hostname=198.162.0.1 port=8090 kmo=remotecert`

- ♦ **hostname:** Specifies the address or name of the machine on which the Remote Loader will run. For example, hostname=198.162.0.1. (If you don't enter this communication parameter, this value defaults to "localhost.")
- ♦ **port:** Specifies the port on which the Remote Loader will accept connections from the remote interface shim. For example, port=8090. (If you don't enter this communication parameter, this value defaults to "8090.")
- ♦ **kmo:** Specifies the key name of the Key Material Object containing the keys and certificate used for SSL. For example, kmo=remotedrivercert. (If you don't enter this communication parameter, no value will be stored. This means, there will be no SSL.)
- ♦ **Application Password:** Enter the password of the application user ID. This is used to pass eDirectory subscription information to the application.
- ♦ **Remote Loader Password:** Enter the password for the Remote Loader. This password is used by the Remote Interface shim to authenticate itself to the Remote Loader.

NOTE: When entering passwords for the application and the Remote Loader, you must set or reset both of these passwords at the same time.

5 Click OK to apply your changes.

Configuring the Remote Loader on Solaris and Linux

You can configure the Remote Loader to work in either a Java or a native environment. In a native environment, the binary component, rdxml, loads the JVM and the Java Remote Loader to support Java drivers using a native interface. It also loads native drivers.

Setting Environment Variables on Solaris and Linux

After installing the Remote Loader, you can set the environmental variable RDXML_BASE_PATH to point to your preferred directory. This directory will then be taken as the base path for files that are subsequently created. To set the value of the RDXML_BASE_PATH variable, enter the following commands:

- ♦ set RDXML_BASE_PATH=<path>
- ♦ export RDXML_BASE_PATH

Configuring the Remote Loader for Remote Drivers

To configure the Remote Loader for hybrid drivers, enter the following command:

```
rdxml option_name
```

For more information on the options associated with the rdxml component, refer to [“Setting Command Line Options and Parameters” on page 53](#).

Configuring the Remote Loader with Open SSL on UNIX

To configure the Remote Loader with jsse:

- 1** Edit the java.security file in the /jre/lib/security/ directory to add the following line:
security.provider.n=com.sun.net.ssl.internal.ssl.Provider

Specify the value of the variable *n* depending upon the number of security providers you already have. For example, if this is your second security provider, add the line:

```
security.provider.2=com.sun.net.ssl.internal.ssl.Provider
```

- 2** Set the CLASSPATH variable to the path of the jsse.jar file in the /usr/lib/dirxml/classes/ directory.
- 3** Export the Trusted Root Certificate of a KMO from the eDirectory tree to the machine hosting the Remote Loader. In case of an FTP transfer, export the file in binary mode.
- 4** To run the keytool, enter the following command:

```
keytool -import -alias TrustedRoot -file <pathname_with_filename>
```
- 5** If you are unable to execute the command in Step 4, you might have multiple java.security files. In that case, enter the following command:

```
java sun.security.tools.KeyTool -import -alias TrustedRoot -file <pathname_with_filename>
```

Running the Remote Loader Configuration Wizard

You run the Remote Loader Configuration Wizard to set the communication and configuration parameters to run the Remote Loader. You can only run this Wizard on NetWare and Windows NT/2000. You run the Remote Loader Configuration Wizard on the application server where you installed the Remote Loader.

To run the Remote Loader Configuration Wizard:

- 1** Run the dirxml_remote.exe found in c:\novell\remoteloader with no command line parameters to launch the wizard.
- 2** Use the information in the following table to set parameters for the Remote Loader:

Parameters	Description
CommandPortNumber	The TCP port number on which a Remote Loader instance listens for commands such as Unload, Change Trace Level, Trace Window On/Off, and so forth. Each instance of the Remote Loader that is running on a particular machine must have a different command port number.
Configuration File Name	The configuration file is a text file into which the Remote Loader configuration parameters are placed. Generally, this is in the same directory as the Remote Loader executable.
Java Class Name	The name of the Java class that will be instantiated for the application driver component of the driver if the driver is a Java driver such as the Delimited or LDAP driver. The Java class can either be in the c:\novell\remoteloader\classes directory in a .class file or in the c:\novell\remoteloader\lib directory in a .jar file.
DLL File Name	Specify the name of the native library that implements the driver if the driver is a native driver such as the Active Directory or PeopleSoft drivers. If a native module is specified, it should be located in the same directory as dirxml_remote.exe, or the full path should be provided.
Port Number	The connection port number is the TCP port on which the Remote Loader will listen for connections from the DirXML server.
Address	The IP address on which the loader will listen.

Parameters	Description
Use SSL	Mark to specify SSL.
Trusted Root File	<p>If you already have a Java keystore file containing the appropriate trusted root certificate, you can specify this file in the field.</p> <p>This is the exported "Self-Signed Certificate" from the eDirectory tree's Organizational Certificate Authority. The certificate must be exported in Base 64 format.</p>
Keystore Password	This field only appears if the trusted root name has an extension of .keystore. Enter the access password for the keystore file in this field.
Trace Level	Set a trace level greater than zero for the Remote Loader instance to display a trace window that contains informational messages from both the loader and the driver.
Trace File	Specify a trace file for trace messages to be written to a file. Each Remote Loader instance running a particular machine must use a different trace file. Trace messages will only be written to the trace file if the trace level is greater than zero.
Install This Remote Loader Instance as a Service	Check this to set up the Remote Loader as a service to start the Remote Loader automatically when the machine boots.
Remote Loader Password	This is used to control access to a Remote Loader instance that is hosting a driver. It must be the same password that is specified as the Remote Loader password using the Driver Object Properties page. Confirm the password in the Confirm password field.
Driver Object Password	The Driver object password is used by the Remote Loader to authenticate itself to the DirXML server. It must be the same password as the Driver Object Password using the Driver Object Properties page.

Setting Command Line Options and Parameters

You can configure the Remote Loader by using command line options and parameters without using the Remote Loader Configuration Wizard. In general, the command line parameters set the shim module or class, connection parameters between the shim and the DirXML server, and the trace level. Other parameters include opening and closing the trace window, setting passwords, and determining whether the Remote Loader instance is a Windows service. You specify command line options and parameters in a configuration file.

- 1 Open or create a configuration file in a text editor.
- 2 Enter the commands in the following table to configure the Remote Loader:

Option	Secondary Name	Parameter	Description
-class	-cl	Java class name	Specifies the Java class name of the DirXML application shim that is to be hosted. The class option and the module option are mutually exclusive.

Option	Secondary Name	Parameter	Description
-commandport	-cp	port number	<p>Specify the TCP/IP port that the Remote Loader instance uses for control purposes.</p> <p>If the Remote Loader instance is hosting an application shim, the command port is the port on which another Remote Loader instance will communicate with the instance hosting the shim.</p> <p>If the Remote Loader instance is sending a command to an instance that is hosting an application shim, the command port is the port on which the hosting instance is listening.</p> <p>If not specified, the default command port is 8000.</p> <p>Multiple instances of the Remote Loader can be run on the same server hosting different driver instances by specifying different connection ports and command ports.</p>
-config	None	filename	<p>Specifies a configuration file. The configuration file can contain any command line options except config. Options specified on the command line override options specified in the configuration file.</p>
-connection	-conn	connection configuration string	<p>This specifies the parameters for the connection to the DirXML server running the remote interface shim.</p> <p>The default connection method for the Remote Loader is TCP/IP using SSL. The default TCP/IP port for this connection is 8090.</p> <p>Multiple instances of the Remote Loader can be run on the same server hosting different driver instances by specifying different connection ports and command ports.</p>
-java	-j	None	<p>Specifies that the passwords are to be set for a Java shim instance. This option is only useful in conjunction with the setpasswords option.</p> <p>If -class is specified with -setpasswords, this option is not necessary.</p>

Option	Secondary Name	Parameter	Description
-module	-m	modulename	Specifies the module containing the DirXML application shim that is to be hosted. The module option and the class option are mutually exclusive.
-password	-p	password	<p>Specifies the password for command authentication. This password must be the same as the first password specified with setpasswords for the loader instance being commanded.</p> <p>If a command option (unload, tracechange, etc.) is specified and the password option is not specified, the user will be prompted to enter the password for the loader that is the target of the command.</p>
-service	-serv	None, or install/uninstall	<p>To install an instance as a service, use the install argument together with any other arguments necessary to host an application shim. For example, the arguments used must include -module, but any argument can include -connection, -commandport, and so forth.</p> <p>To uninstall an instance running as a service, use the uninstall argument together with any other arguments necessary to host the application shim. Note that this option installs the Win32* service; this option does not start the service.</p> <p>The no-argument version of this option is only used on the command line to an instance being run as a Win32 service. This is automatically set up when installing an instance as a service.</p>

Option	Secondary Name	Parameter	Description
-setpasswords	-sp	password password	<p>Specify the password for the Remote Loader instance and the password of the Driver object of the remote interface shim with which the Remote Loader will communicate.</p> <p>The first password in the argument is the password for the Remote Loader. The second password in the argument is the password for the Driver object associated with the remote interface shim on the DirXML server.</p> <p>Both passwords must be specified.</p> <p>This is a configuration option. Using this option configures the Remote Loader instance with the passwords specified but does not load a DirXML application shim or communicate with another loader instance.</p>
-trace	-t	integer	Specify the trace level. This is only used when hosting an application shim. Trace levels correspond to those used on the DirXML server.
-tracechange	-tc	integer	<p>Command a Remote Loader instance that is hosting an application shim to change its trace level.</p> <p>Trace levels correspond to those used on the DirXML server.</p>
-tracefile	-tf	filename	Specify a file to which to write trace messages. Trace messages will be written to the file if the trace level is greater than zero. Trace messages will be written to the file even if the trace window is not open.
-tracelfilechange	-tfc	None, or filename	<p>Command a Remote Loader instance that is hosting an application shim to start using a trace file, or to close one already in use and use a new one.</p> <p>Using the no-argument version of this option will cause the hosting instance to close any trace file being used.</p>

Option	Secondary Name	Parameter	Description
-unload	-u	None	Unload the Remote Loader instance. If the Remote Loader is running as a Win32 service this will stop the service.
-window	-w	On/Off	Turn the trace window on or off in a Remote Loader instance. This option is not available on Solaris or Linux platforms.
-wizard	-wiz	None	Launches the Configuration Wizard. Note that running dirxml_remote.exe with no command line parameters also launches the wizard. This option is useful if a configuration file is also specified. In this case, the wizard will start with values from the configuration file and the wizard can be used to change the configuration without editing the configuration file directly. This option is not available on Solaris or Linux platforms.

Setting Up Connection Parameters Using the Connection Command Line Option

This section contains the argument names and parameters for the TCP/IP connection method.

- 1 Open or create a configuration file in a text editor.
- 2 Use the following table to set up the TCP/IP connection:

Option	Parameter	Description
address	IP address	Specifies that the Remote Loader will listen on a particular local IP address. This is useful if the server hosting the Remote Loader has multiple IP addresses and the Remote Loader must listen on only one of the addresses. If an address is not specified, the Remote Loader will listen on all local IP addresses.
keypass	keypass	Used only for DirXML application shims contained in .jar files. Specifies the password for the Java keystore specified by the keystore parameter.
keystore	keystore	Used only for DirXML application shims contained in .jar files. Specifies the filename of the Java keystore that contains the trusted root certificate of the issuer of the certificate used by the remote interface shim. This will typically be the Certificate Authority of the eDirectory tree that is hosting the remote interface shim.

Option	Parameter	Description
port	decimal port number	Specifies the TCP/IP port on which the Remote Loader will listen for connections from the remote interface shim.
rootfile	filename	Used only for DirXML application shims contained in .DLL files. Specifies the file containing the trusted root certificate of the issuer of the certificate used by the remote interface shim. This will typically be the Certificate Authority of the eDirectory tree that is hosting the remote interface shim. The certificate file must be in Base 64 format (PEM).

Running the Remote Loader

The following section contains information about running the Remote Loader service on Solaris and Linux.

Solaris and Linux

On Solaris or Linux, the binary component `rdxml` provides the Remote Loader functionality. This component is located in the `/usr/bin/` directory.

To run the Remote Loader:

- 1** Enter a command to start the configuration file, for example:
`rdxml -config <path_to_config_file>`
 The default sample configuration file, `remote.conf`, is present in the `/usr/lib/dirxml/rules/` directory.
- 2** Start the Remote Loader interface shim using `iManager` or `ConsoleOne`.
- 3** Confirm that the Remote Loader is operating properly. You can do this by using either the `ps` or `ndstat` commands, or a trace file to check if the command and connection ports are listening.
- 4** To stop the Remote Loader, enter a command, like this example:
`rdxml -config <path_to_config_file> -U`

Setting Up Multiple Instances of the Remote Loader

You can run multiple instances of the Remote Loader on the same server hosting different driver instances by specifying different connection ports and command ports.

Activating Identity Manager Products

Activation is required for your Identity Manager products. Refer to [Appendix A, “Activating Novell Identity Manager Products,” on page 215](#) for more information.

Installing a Custom Driver

A custom driver is delivered as a single .jar or .zip file, and might consist of the following:

- ♦ A set of .jar or native (.dll or .nlm files
- ♦ XML rules files for configuring the driver
- ♦ Documentation

For more information on creating a custom driver or installing one, see the [Novell Developer Kit \(http://developer.novell.com/ndk/dirxml-index.htm\)](http://developer.novell.com/ndk/dirxml-index.htm).

5

Managing DirXML Drivers

This section contains information that will help you create and manage your DirXML[®] driver. Topics include:

- ♦ [“Creating and Configuring a Driver” on page 61](#)
- ♦ [“Managing DirXML 1.x Drivers in an Identity Manager Environment” on page 62](#)
- ♦ [“Upgrading a Driver Configuration from DirXML 1.x to Identity Manager Format” on page 63](#)
- ♦ [“Starting, Stopping, or Restarting a Driver” on page 63](#)
- ♦ [“Using Global Configuration Values” on page 63](#)
- ♦ [“Using Named Passwords” on page 64](#)
- ♦ [“Reassociating a Driver Object with a Server” on page 67](#)
- ♦ [“Adding Driver Heartbeat” on page 68](#)

Creating and Configuring a Driver

For each DirXML driver you plan to use, you should create a driver object and import a driver configuration. The driver object contains configuration parameters and policies for that driver. As part of creating a driver object, you import a driver-specific configuration file. Driver configurations contain a default set of policies. These policies are intended to give you a good start as you implement your data sharing model. Most of the time, you will set up a driver using the shipping default configuration, and then modify the driver configuration according to the requirements of your environment.

There are two methods you can use for creating driver objects.

- ♦ The Create Driver task lets you create a single driver and import its driver configuration. For more information, refer to [“Creating a Driver Object” on page 61](#).
- ♦ The Import Driver task lets you create multiple drivers at the same time and import their configurations. For more information, refer to [“Creating Multiple Drivers” on page 62](#).

Creating a Driver Object

The driver configuration (XML) file creates and configures the objects needed in order for driver work properly. It also includes basic policies you can modify for your implementation.

- 1** In iManager, select DirXML Utilities > Create Driver.
- 2** Select a driver set where you want to create the driver, then click Next.

If you place this driver in a new driver set, you must specify a driver set name, context, and associated server.

- 3** Mark Import a Driver Configuration from the Server and select the.xml file.

The driver configuration file is installed on the Web server when you set up iManager.

- 4** Follow the prompts to finish importing the driver configuration.

The necessary Identity Manager objects are created. If you didn't define security equivalences or exclude administrative users during the import, you can complete these tasks by modifying the properties of the driver object.

Creating Multiple Drivers

Identity Manager provides the capability to create several drivers at once. The process is similar to creating a single driver in that the driver configuration (XML) files still create and configure the objects needed in order for drivers to work properly.

To import several drivers at the same time:

- 1** In iManager, select DirXML Utilities > Import Drivers.
- 2** Select a driver set where you want to create new drivers, then click Next.

If you place these drivers in a new driver set, you must specify a driver set name, context, and associated server.

- 3** Select the application drivers to add to the driver set, then click Next.
- 4** Follow the prompts and specify the requested data, then click Next.

The necessary Identity Manager objects for each driver are created. If you didn't define security equivalences or exclude administrative users during the import, you can complete these tasks by modifying the properties of the driver object.

Managing DirXML 1.x Drivers in an Identity Manager Environment

Existing drivers that were created for DirXML 1.x will continue to run with Identity Manager.

The DirXML engine that ships with Nsure Identity Manager 2 is backward compatible with older drivers (as long as the older driver shims and configurations have been updated with all the latest product updates and patches). It accomplishes this by converting their driver configurations to Identity Manager format, on the fly. This conversion is only for the use of the engine, and does not make any permanent changes to existing DirXML 1.x driver configurations. Because the engine is backward compatible, you can run DirXML 1.x drivers on Identity Manager servers as long as you want to, without making any changes to them.

However, the iManager plug-ins have only limited backward compatibility. Older drivers can be viewed in the Overview of a driver set, but the driver configuration can't be viewed or edited. When you click a DirXML 1.x driver in the driver set Overview, the DirXML plugins discover that the driver is in 1.x format, and prompt you to convert the driver to 2.0 format using the wizard.

If you don't want to make any changes to an existing driver yet, you can cancel out of the wizard.

To edit a 1.x driver in 1.x format, you must use the DirXML 1.x plug-ins. To do this, you must use a separate iManager web server with the 1.x plug-ins installed on it. You can't use the DirXML plug-ins that ship with Identity Manager to edit a driver configuration without converting the driver to Identity Manager 2 format.

Upgrading a Driver Configuration from DirXML 1.x to Identity Manager Format

The Identity Manager installation installs new driver shims, but it does not change existing driver objects or driver configurations.

Existing driver configurations that were created for DirXML 1.x will continue to run with Identity Manager. However, the iManager DirXML plug-ins for Identity Manager let you edit only drivers that are in Identity Manager format.

IMPORTANT: Running an Identity Manager DirXML driver shim or driver configuration with a DirXML 1.x Engine is not supported.

A wizard is provided to help you convert DirXML 1.x drivers to Identity Manager format.

To begin the wizard:

- 1** In iManager, click DirXML Management > Overview. Select the driver set that contains the driver you want to convert.
- 2** Click the icon for the driver you want to convert.
You are prompted to convert the driver to the new format.
- 3** Follow the steps in the wizard to complete the conversion.

Starting, Stopping, or Restarting a Driver

You can easily start, stop, or restart a driver through iManager.

- 1** In iManager, click DirXML Management > Overview.
- 2** Browse to the driver set where the driver exists.
- 3** Click the driver whose status you want to change, then choose the appropriate option (start, stop, restart.)

Using Global Configuration Values

Global configuration values (GCVs) are new settings that are similar to driver parameters. Global configuration values can be specified for a driver set as well as an individual driver. If a driver does not have a GCV value, the driver inherits the value for that GCV from the driver set.

GCVs allow you to specify settings for new Identity Manager features such as password synchronization and driver heartbeat, as well as settings that are specific to the function of an individual driver configuration. Some GCVs are provided with the drivers, but you can also add your own. You can refer to these values in a policy to help you customize your driver configuration.

IMPORTANT: Password synchronization settings are GCVs, but it's best to edit them in the graphical interface provided on the Server Variables page for the driver, instead of the GCV page. The Server Variables page which shows Password Synchronization settings is accessible as a tab like other driver parameters, or by clicking Password Management > Password Synchronization, searching for the driver, and clicking the driver name. The page contains online help for each Password Synchronization setting.

To add, remove, or edit GCVs that are not related to Identity Manager Password Synchronization:

- 1** In iManager, click DirXML Management > Overview.
- 2** Browse to and click the driver set or driver object, then click the Global Config Values tab.
- 3** Add, remove, or edit the XML, then click OK to apply your changes.

Using the DirXML Command Line Utility

The DirXML Command Line Utility provides access to all DirXML subverbs and allows you to perform common driver management tasks, such as starting and stopping a driver, from the command line. It's installed with Identity Manager, but also works with DirXML 1.x implementations.

While we recommend using iManager to administer Identity Manager, the DirXML Command Line Utility is another option available to perform common operations. You can use the utility in an interactive mode or in a pure command line mode.

The utility and scripts are installed on all platforms during the Identity Manager installation. The utility is installed to the following locations:

Windows: \Novell\Nds\dxcmd.bat

NetWare: SYS:\SYSTEM\DXCMD.NCF

Unix: /usr/bin/dxcmd

Using Named Passwords

DirXML 1.x provided the ability to store a single password securely, so that a driver could use that password without having it hard-coded in clear text in the driver policies.

Identity Manager allows you to store multiple passwords securely for a particular driver. This new functionality is referred to as Named Passwords. Each different password is accessed by a key, or name.

You can create or remove these passwords using the DirXML Command Line Utility.

To use a named password in a driver policy, you refer to it by the name of the password, instead of using the actual password, and the DirXML engine sends the password to the driver. The method described in this section for storing and retrieving Named Passwords can be used with any driver without making changes to the driver shim.

NOTE: The sample configurations provided for the DirXML Driver for Lotus Notes include an example of using Named Passwords in this way. The Notes driver shim has also been customized to support other ways of using Named Passwords, and examples of those methods are also included. For more information, see the section on Named Passwords in the *DirXML Driver for Lotus Notes Implementation Guide* (<http://www.novell.com/documentation/lg/dirxmldrivers>).

In this section:

- ◆ “Creating a Named Password in the DirXML Command Line Utility” on page 64
- ◆ “Using Named Passwords in Driver Policies” on page 66
- ◆ “Removing a Named Password in the DirXML Command Line Utility” on page 66

Creating a Named Password in the DirXML Command Line Utility

- 1** Run the DirXML Command Line Utility.

For information, see “Using the DirXML Command Line Utility” on page 64.

- 2** Enter your user name and password.

The following list of options appears.

DirXML commands

```
1: Start driver
2: Stop driver
3: Driver operations...
4: Driver set operations...
5: Log events operations...
6: Get DirXML version
99: Quit
```

Enter choice:

3 Enter 3 for driver operations.

A numbered list of drivers appears.

4 Enter the number for the driver you want to add a Named Password to.

The following list of options appears.

Select a driver operation for:

driver_name

```
1: Start driver
2: Stop driver
3: Get driver state
4: Get driver start option
5: Set driver start option
6: Resync driver
7: Migrate from application into DirXML
8: Submit XDS command document to driver
9: Check object password
10: Initialize new driver object
11: Passwords operations
12: Cache operations
99: Exit
```

Enter choice:

5 Enter 11 for password operations.

The following list of options appears.

Select a password operation

```
1: Set shim password
2: Reset shim password
3: Set named password
4: Clear named password(s)
5: List named passwords
99: Exit
```

Enter choice:

6 Enter 3 to set a new Named Password.

The following prompt appears:

Enter password name:

7 Enter the name by which you want to refer to the Named Password.

8 Enter the actual password that you want to secure, at the following prompt that appears:

Enter password:

The characters you type for the password are not displayed.

9 Confirm the password by entering it again, at the following prompt that appears:

Confirm password:

- 10** After you enter and confirm the password, you are returned to the password operations menu.

After completing this procedure, you can use the 99 option twice to exit the menu and quit the DXCommand utility.

Using Named Passwords in Driver Policies

The following example shows how a named password can be referenced in a driver policy on the Subscriber channel in XSLT:

```
<xsl:value-of select="query:getNamedPassword($srcQueryProcessor,'mynamedpassword')">  
xmlns:query="http://www.novell.com/java/com.novell.nds.dirxml.driver.XdsQueryProcessor/>
```

Removing a Named Password in the DirXML Command Line Utility

This option is useful if you no longer need Named Passwords you previously created.

- 1** Run the DirXML Command Line Utility.

For information, see [“Using the DirXML Command Line Utility” on page 64](#).

- 2** Enter your user name and password.

The following list of options appears.

DirXML commands

```
1: Start driver  
2: Stop driver  
3: Driver operations...  
4: Driver set operations...  
5: Log events operations...  
6: Get DirXML version  
99: Quit
```

Enter choice:

- 3** Enter 3 for driver operations.

A numbered list of drivers appears.

- 4** Enter the number for the driver you want to remove Named Passwords from.

The following list of options appears.

Select a driver operation for:

driver_name

```
1: Start driver  
2: Stop driver  
3: Get driver state  
4: Get driver start option  
5: Set driver start option  
6: Resync driver  
7: Migrate from application into DirXML  
8: Submit XDS command document to driver  
9: Check object password  
10: Initialize new driver object  
11: Passwords operations  
12: Cache operations  
99: Exit
```

Enter choice:

5 Enter 11 for password operations.

The following list of options appears.

Select a password operation

```
1: Set shim password
2: Reset shim password
3: Set named password
4: Clear named password(s)
5: List named passwords
99: Exit
```

Enter choice:

6 (Optional) Enter 5 to see the list of existing Named Passwords.

The list of existing Named Passwords is displayed.

This step can help you make sure you are removing the correct password.

7 Enter 4 to remove one or more Named Passwords.

8 Enter No to remove a single Name Password, at the following prompt that appears:

Do you want to clear all named passwords? (yes/no):

9 Enter the name of the Named Password you want to remove, at the following prompt that appears:

Enter password name:

After you enter the name of the Named Password you want to remove, you are returned to the password operations menu:

Select a password operation

```
1: Set shim password
2: Reset shim password
3: Set named password
4: Clear named password(s)
5: List named passwords
99: Exit
```

Enter choice:

10 (Optional) Enter 5 to see the list of existing Named Passwords.

The list of existing Named Passwords is displayed.

This step lets you verify that you have removed the correct password.

After completing this procedure, you can use the 99 option twice to exit the menu and quit the DXCommand utility.

Reassociating a Driver Object with a Server

A driver object is associated with a server.

If the association becomes invalid for some reason, it is indicated by one of the following:

- ♦ When upgrading eDirectory on your DirXML (Identity Manager 2) server, you get the error “UniqueSPIException error -783.”
- ♦ No server is listed next to the driver in the DirXML Overview
- ♦ A server is listed next to the driver in the DirXML Overview, but the name is garbled text

To resolve this issue, you must disassociate the driver object and the server, and then reassociate them.

Log into iManager and go to the Driver object in the DirXML Overview. Use the icons to remove and then add a server to the server name list next to the driver icon. Removing and then adding reassociates the server and the Driver object.

Adding Driver Heartbeat

The driver heartbeat is a new feature of DirXML drivers, which ship with Identity Manager 2, and its use is optional. Driver heartbeat is configured using a driver parameter with a time interval specified. If a heartbeat parameter exists and has an interval value other than 0, the driver sends a heartbeat document to the DirXML engine if there is no communication on the Publisher channel for the specified interval of time.

The intent of the driver heartbeat is to give you a trigger to allow you to initiate an action at regular intervals, in case the driver does not communicate on the Publisher channel as often as you want the action to occur. You must customize your driver configuration or other tools if you want to take advantage of the heartbeat. The DirXML engine accepts the heartbeat document but does not take any action because of it.

For most drivers, a driver parameter for heartbeat is not used in the sample configurations, but you can add it.

A custom driver that is not provided with Identity Manager can also provide a heartbeat document, if the driver developer has written the driver to support it.

To configure the heartbeat, do the following:

- 1** In iManager, click DirXML Management > Overview. Search for your driver, and click on the driver icon.
- 2** In the graphical view of the driver configuration, click the driver icon.
- 3** On the DirXML page, scroll down to Driver Parameters, and look for Heart Beat or a similar display name.

If a driver parameter already exists for heartbeat, you can change the interval and save the changes, and configuration is complete.

The value of the interval cannot be less than 1. A value of 0 means the feature is turned off.

The unit of time is usually minutes; however, some drivers might choose to implement it differently, such as using seconds.

- 4** If a driver parameter does not exist for heartbeat, click Edit XML.
- 5** Add a driver parameter entry like the following example, as a child of <publisher-options>. (For an AD driver, make it a child of <driver-options>.)

```
<pub-heartbeat-interval display-name="Heart Beat">10</pub-heartbeat-interval>
```

TIP: If the driver does not produce a heartbeat document after being restarted, check the placement of the driver parameter in the XML.

- 6** Save the changes, and make sure the driver is stopped and restarted.

After you have added the driver parameter, you can edit the time interval using the graphical view. Another option is to create a reference to a global configuration value (GCV) for the time interval.

Like other global configuration values, the driver heartbeat can be set at the driver set level instead of on each individual driver object. If a driver does not have a particular global configuration value, and the driver set does have it, the driver inherits the value from the driver set.

The following is an example heartbeat status document sent by the Notes driver:

```
<nds dtdversion="2.0" ndsversion="8.x">
  <source>
    <product build="20031112_1037" instance="blackcap" version="2.0">DirXML
Driver for Lotus Notes</product>
    <contact>Novell, Inc.</contact>
  </source>
  <input>
    <status level="success" type="heartbeat"/>
  </input>
</nds>
```


6

Creating Policies

Policies enable you to customize the flow of information into and out of eDirectory, for a particular environment.

For example, one company might use the inetorgperson as the main user class, another company might use User. To handle this, a policy is created which tells the DirXML engine what a user is called in each system. Whenever operations affecting users are passed between connected systems, Identity Manager applies the policy that makes this change.

Policies also create new objects, update attribute values, make schema transformations, define matching criteria, maintain Identity Manager associations, and many other things.

As policies are too large a topic to cover effectively in this manual, a detailed guide to Policies is contained in the *Policy Builder and Driver Customization Guide*. This guide contains:

- ♦ A detailed description of each available policy
- ♦ An in-depth Policy Builder user guide and reference, including examples and syntax for each condition, action, noun, and verb.
- ♦ A discussion on creating policies using XSLT stylesheets.

Please refer to the *Policy Builder and Driver Customization Guide* for information on policies.

7

Managing Passwords Using Password Policies

Using Password Policies, you can increase security by setting rules for how users create their passwords. You can also decrease help desk costs by providing users with self-service options for forgotten passwords and for resetting passwords.

In this section:

- ♦ [“Overview of Password Policy Features” on page 73](#)
- ♦ [“Planning for Password Policies” on page 80](#)
- ♦ [“Prerequisites for Using Password Policies” on page 81](#)
- ♦ [“Creating Password Policies” on page 84](#)
- ♦ [“Assigning Password Policies to Users” on page 84](#)
- ♦ [“Making Sure Password Policies Are Correct for Identity Manager” on page 85](#)
- ♦ [“Finding Out Which Policy a User Has” on page 86](#)
- ♦ [“Setting A User’s Password” on page 86](#)
- ♦ [“Creating or Editing Challenge Sets” on page 86](#)
- ♦ [“Configuring Notification for Password Features” on page 87](#)
- ♦ [“Troubleshooting Password Policies” on page 87](#)

For information about Forgotten Password Self-Service and Reset Password Self-Service, see [Chapter 8, “Password Self-Service,” on page 89](#).

Overview of Password Policy Features

A password policy is a collection of administrator-defined rules that specify the criteria for creating and replacing end-user passwords. Identity Manager takes advantage of NMA to enforce Password Policies that you assign to users in eDirectory. Using Password Synchronization, you can also enforce Password Policies on connected systems, as explained in [Chapter 9, “Password Synchronization across Connected Systems,” on page 113](#).

Password Policies also include Forgotten Password Self-Service features, to reduce help desk calls for forgotten passwords. Another self-service feature is Reset Password Self-Service, which lets users change their passwords while viewing the rules the administrator has specified in the Password Policy. Users access these features through the iManager self-service console.

Most features of password management require Universal Password to be enabled. Ideally, you would also integrate the iManager self-service console into your existing company portal, if you have one, to give users easy access to Forgotten Password Self-Service and Reset Password Self-service.

You create Password Policies using a wizard: in iManager, Password Management > Manage Password Policies > New.

The new Password Management features let you do the following:

- ◆ “Set Advanced Password Rules” on page 74
- ◆ “Enable Universal Password” on page 75
- ◆ “Provide Users with Forgotten Password Self-Service” on page 76
- ◆ “Provide Users with Reset Password Self-Service” on page 76
- ◆ “Assign Policies to eDirectory Users” on page 76
- ◆ “Enforce Policies in eDirectory” on page 77
- ◆ “Enforce Policies on Connected Systems” on page 79
- ◆ “View Which Password Policy Is in Effect for a User” on page 79
- ◆ “Set Universal Password for a User” on page 79

Set Advanced Password Rules

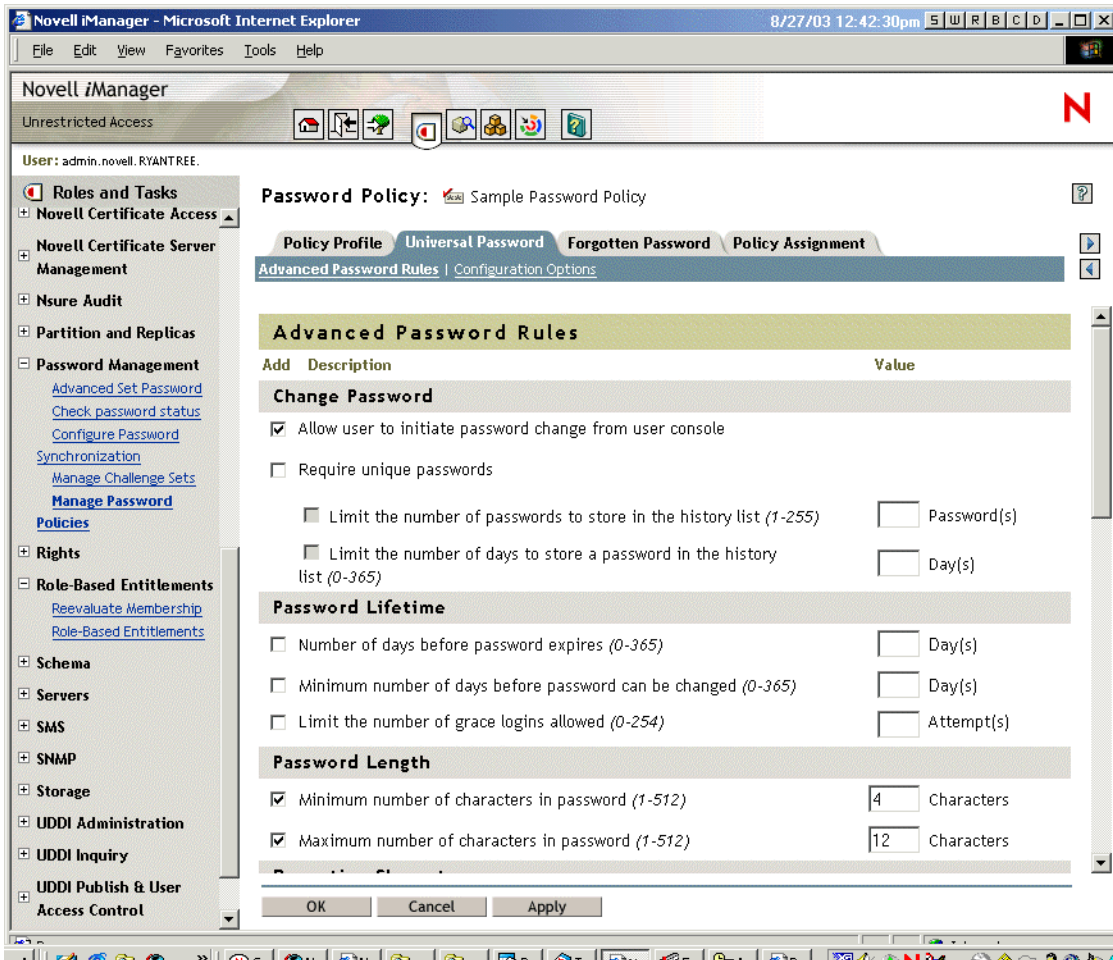
Advanced Password Rules let you define the criteria for acceptable passwords such as:

- ◆ Password syntax
- ◆ Password properties
- ◆ Password lifetime
- ◆ Use of special characters
- ◆ Password exclusions

NOTE: Keep in mind that password exclusions can be useful for a few words that you think would be security risks. However, it is not intended to be used to import a dictionary or other long list of words, because the password must be validated against every word in the list, and a long list would slow down users’ ability to change passwords.

To use Advanced Password Rules in a Password Policy, you must enable Universal Password. (If you don’t enable Universal Password for a policy, the password restrictions set for NDS Password are enforced instead.)

The following figure shows an example of the property page where you specify Advanced Password Rules for a Password Policy.



Enable Universal Password

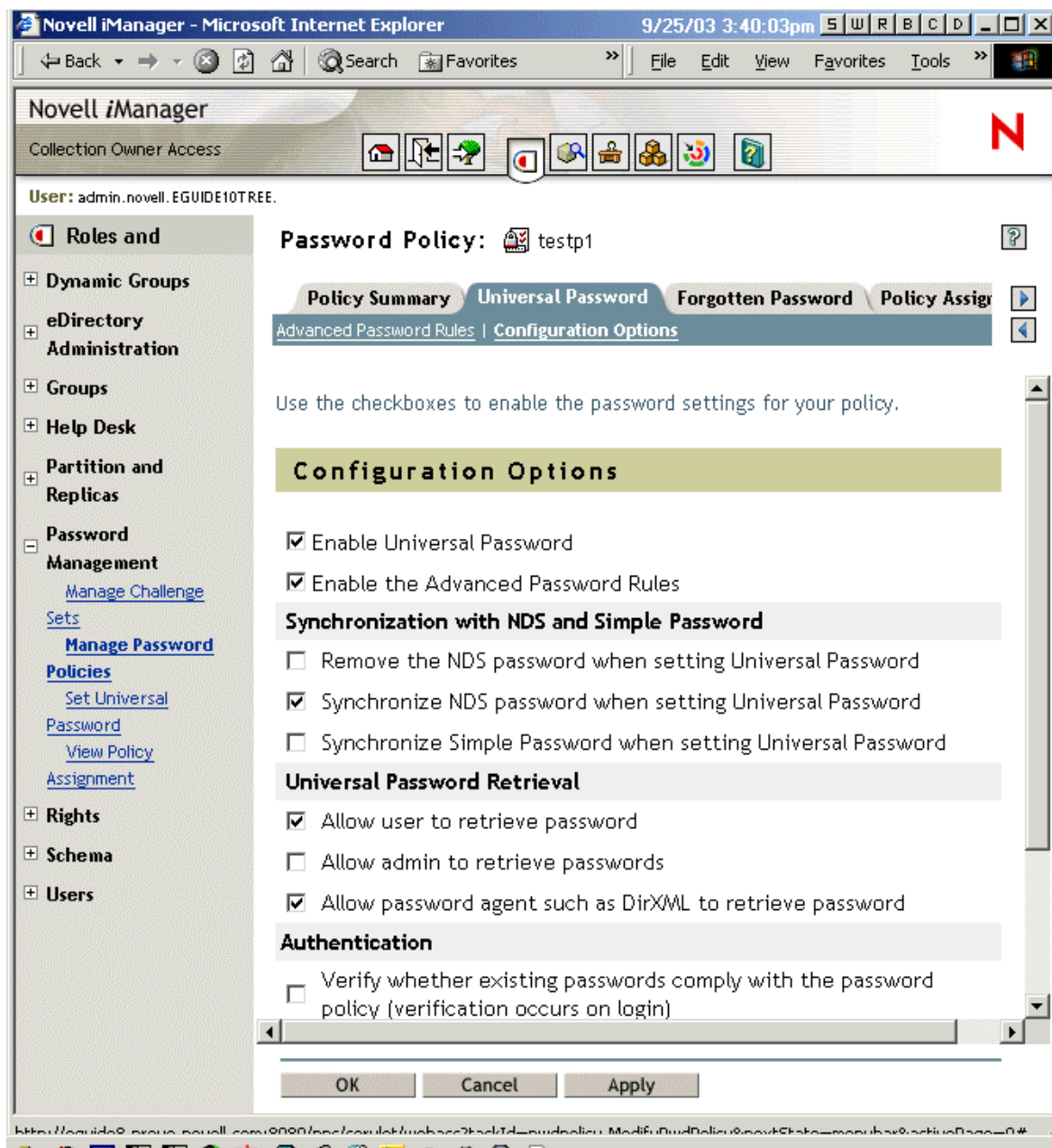
Universal Password is the new password capability in eDirectory 8.7.1. You must enable Universal Password for your users if you want to use Advanced Password Rules, Password Synchronization, and many of the Forgotten Password features.

A Password Policy lets you specify whether Universal Password is enabled. You can then assign the Password Policy to users (the whole tree, a container or partition, or specific user). Universal Password does not have to be on for the whole tree. Using different Password Policies, you can tailor your use of Universal Password to your needs. We recommend assigning Password Policies as high in the tree as possible to simplify administration.

Some additional planning is required to prepare your environment for Universal Password, such as upgrading Novell Client if you use it, and eDirectory.

You can also edit other Universal Password and NMAS settings in a Password Policy, such as whether NDS or Simple Password are synchronized with Universal Password.

The following figure shows an example of the property page where you specify Universal Password configuration options for a Password Policy.



Provide Users with Forgotten Password Self-Service

See “Provide Users with Forgotten Password Self-Service” on page 89.

Provide Users with Reset Password Self-Service

See “Provide Users with Reset Password Self-Service” on page 90.

Assign Policies to eDirectory Users

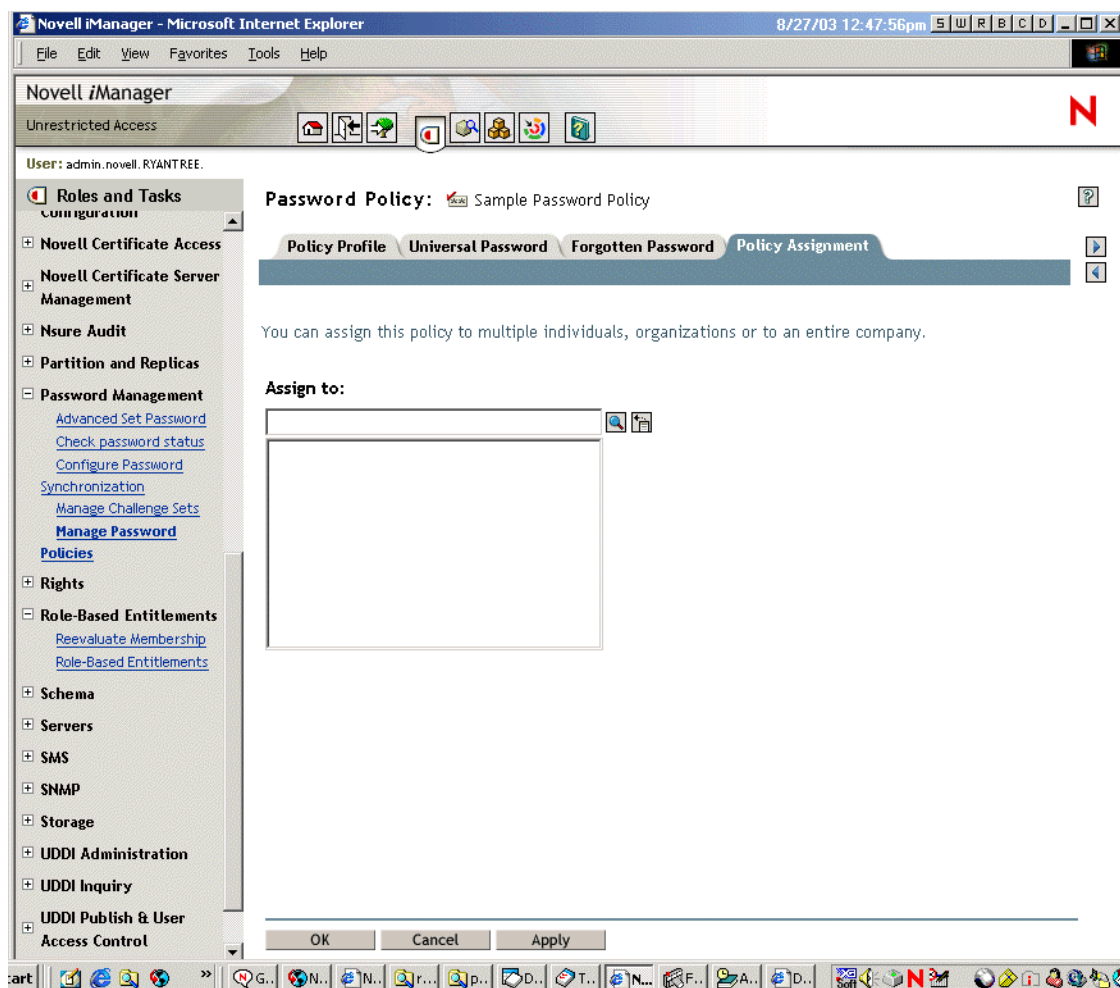
You can assign a Password Policy to users in eDirectory by assigning the policy to the whole tree (using the Login Policy object), specific partitions or containers, or specific users.

Novell recommends that you assign a default policy to the whole tree, and assign any other policies you use as high up in the tree as possible, to simplify administration.

NMAS determines which Password Policy is in effect for a user. See “[Assigning Password Policies to Users](#)” on page 84 for more information on how to assign password policies to users.

If you are using Password Synchronization, keep in mind that you must make sure that the users who are assigned Password Policies match up with the users you want to participate in Password Synchronization for connected systems. Password Policies are assigned with a tree-centric perspective. By contrast, Password Synchronization is set up per driver, on a per-server basis. To get the results you expect from Password Synchronization, make sure the users that are in a read/write or master replica on the server running the drivers for Password Synchronization match with the containers where you have assigned Password Policies with Universal Password enabled. Assigning a Password Policy to a partition root container ensures that all users in that container and subcontainers are assigned the Password Policy.

The following figure shows an example of the property page where you specify which object Password Policy is assigned to.



Enforce Policies in eDirectory

When you assign a Password Policy to users in the tree, any password changes going forward must comply with the Advanced Password Rules in that policy. In the browser, the password rules are

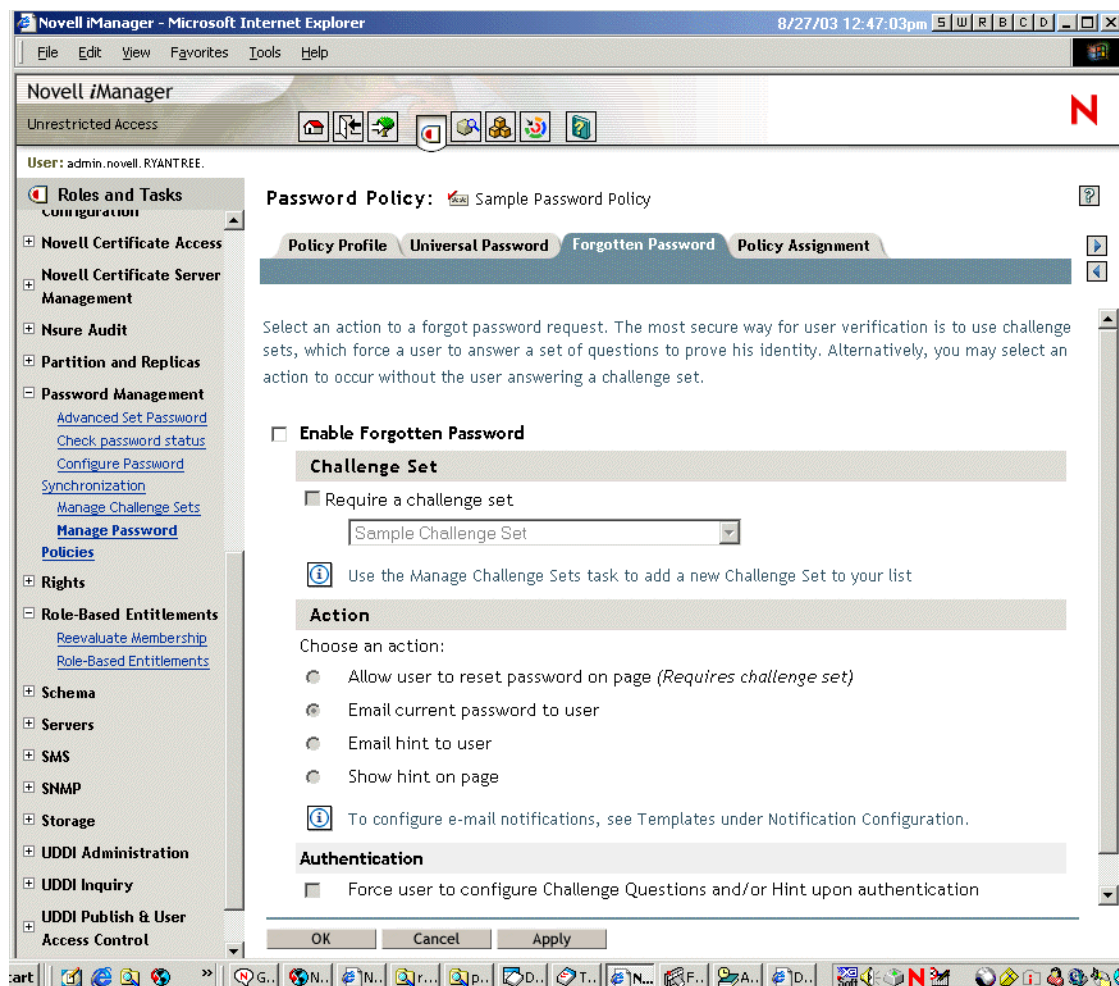
displayed in the page where the user changes the password. In the Novell Client, the rules are not displayed, but they are enforced, so a noncompliant password is rejected. NMAS is the application that enforces these rules.

For the best user experience, have users change their passwords in the iManager self-service console, since this is where the rules are displayed. This way, users can see the rules to help them create compliant passwords.

In addition, you can specify that existing passwords are checked for compliance and users are required to change existing noncompliant passwords.

You can also specify that when users authenticate through the iManager or portal login in the browser, they are prompted to set up any Forgotten Password features you have enabled. This is called post-authentication setup. For example, if you want users to create a Password Hint that can be e-mailed to them when they forget a password, you can use post-authentication setup to prompt users to create a hint at login time.

The post-authentication setting is the last option in the Forgotten Password property page, as shown in the following figure.



Enforce Policies on Connected Systems

If you are using Password Synchronization, settings are provided for each driver to let you enforce the Advanced Password Rules in a Password Policy.

You can do the following:

- ♦ Decide whether Identity Manager should accept passwords published by a connected system, as a general rule.
- ♦ Enforce policy on passwords coming in from a connected system. If the password doesn't comply, Identity Manager does not accept it.
- ♦ Enforce policies on the connected system by resetting noncompliant passwords. If the password coming in to Identity Manager is noncompliant, Identity Manager can not only not accept it as password change to the identity vault, it can also use the existing Identity Manager password to reset the password on the connected system.
- ♦ Notify users when password synchronization is not successful. For example, if the user created a noncompliant password on a connected system and Identity Manager did not accept it, the user could be informed by e-mail so she knows the password change was not synchronized.

If you are using Advanced Password Rules and are using Identity Manager Password Synchronization, to help ensure that passwords can be synchronized successfully we recommend that you research the password policies for all the connected systems to make sure the Advanced Password Rules in the eDirectory Password Policy are compatible.

Keep in mind that you must make sure that the users who are assigned Password Policies match up with the users you want to participate in Password Synchronization for connected systems.

Password Policies are assigned with a tree-centric perspective. By contrast, Password Synchronization is set up per driver, and drivers are installed on a per-server basis and can manage only those users who are in a master or read/write replica. To get the results you expect from Password Synchronization, make sure the users that are in a master or read/write replica on the server running the drivers for Password Synchronization match with the containers where you have assigned Password Policies with Universal Password enabled. Assigning a Password Policy to a partition root container ensures that all users in that container and subcontainers are assigned the Password Policy.

For more information on how you specify password flow, see [“Password Synchronization Settings You Create Using Global Configuration Values” on page 121](#).

View Which Password Policy Is in Effect for a User

In iManager, you can check to see which policy is in effect for a user. See [“Finding Out Which Policy a User Has” on page 86](#).

Set Universal Password for a User

Ability to set the password for a particular user

When you begin using Universal Password, there are more rules that you can enforce.

To support these rules a new plugin is provided, so that administrators or help desk personnel can set a password for a user with the Advanced Password Rules displayed. This helps you create a compliant password.

Planning for Password Policies

In this section:

- ♦ “Planning How to Assign Password Policies in the Tree” on page 80
- ♦ “Planning the Rules for Your Password Policies” on page 80
- ♦ “Planning How to Manage Login Methods for your Users” on page 80

Planning How to Assign Password Policies in the Tree

Novell recommends that you assign a default policy to the whole tree, and assign any other policies you use as high up in the tree as possible, to simplify administration.

NMAS determines which Password Policy is in effect for a user. See “[Assigning Password Policies to Users](#)” on page 84 for more information on how to assign password policies to users.

Planning the Rules for Your Password Policies

You can use the Advanced Password Rules in a password policy to enforce your business policies for passwords.

Keep in mind that only the iManager self-service console displays the password rules, currently. If your users will be changing their passwords using the Novell Client or a login through the LDAP server or on a connected system, you will need to make the password rules readily available to users to help them be successful in creating a compliant password.

If you are using Password Synchronization, keep in mind that you must make sure that the users who are assigned Password Policies match up with the users you want to participate in Password Synchronization for connected systems. Password Policies are assigned with a tree-centric perspective. By contrast, Password Synchronization is set up per driver, on a per-server basis. To get the results you expect from Password Synchronization, make sure the users that are in a read/write or master replica on the server running the drivers for Password Synchronization match with the containers where you have assigned Password Policies with Universal Password enabled. Assigning a Password Policy to a partition root container ensures that all users in that container and subcontainers are assigned the Password Policy.

Planning How to Manage Login Methods for your Users

The iManager self-service console is accessible to end users on your iManager server using a URL such as <https://www.servername.com/nps>. For example, <https://www.myiManager.com/nps>.

Upgrade your client environment to support Universal Password.

- ♦ Novell Client (If you are using it, you must upgrade it. However, using the Novell Client is not required.)
- ♦ NT Client
- ♦ Web browser. If you use no client at all, and users log in through a web browser, you need to make sure they have one of the browsers that supports iManager 2.0.2 or later.

For more information, see the [Universal Password Deployment Guide](http://www.novell.com/documentation/lg/nw65/universal_password/data/front.html) (http://www.novell.com/documentation/lg/nw65/universal_password/data/front.html).

Prerequisites for Using Password Policies

If you want to take advantage of all the features of Password Policies, you need to complete some steps to prepare your environment.

- 1 Upgrade your environment to support Universal Password.

For more information, see the *Universal Password Deployment Guide* (http://www.novell.com/documentation/lg/nw65/universal_password/data/front.html).

If you are not yet ready to deploy Universal Password, or you have eDirectory 8.6.2, find out which Password Policy features you can use without Universal Password in “**Feature Support for eDirectory 8.6.2 and eDirectory 8.7.3**” on page 43.

- 2 Upgrade your client environment to support Universal Password.

- ♦ Novell Client (If you are using it, you must upgrade it. However, using the Novell Client is not required.)
- ♦ NT Client
- ♦ Web browser. If you use no client at all, and users log in through a Web browser, you need to make sure they have one of the browsers that supports iManager 2.0.2 or later.

For more information, see the *Universal Password Deployment Guide* (http://www.novell.com/documentation/lg/nw65/universal_password/data/front.html).

- 3 If you have not run the iManager Configuration Wizard previously when you set up iManager (either as part of the iManager install or post-installation), you must run it.

IMPORTANT: After you run the iManager Configuration Wizard, iManager runs in RBS mode. This means that administrators do not see any tasks unless they have assigned themselves to specific roles. Make sure you assign administrators to roles to give them access to all the iManager tasks.

- 4 Install Identity Manager, as explained in **Chapter 4, “Installation,”** on page 39.

The Password Management plugins are part of this installation on the iManager web server.

For Password Policies, no changes are necessary to driver configurations unless you are using Password Synchronization to enforce Password Policies when synchronizing passwords between Identity Manager and connected systems.

- 5 Make sure that SSL is configured between the iManager Web server and eDirectory, even if they are running on the same machine.

This is a requirement of NMAS 2.3.

- 6 To ensure security for the Forgotten Password gadgets, check your LDAP SSL port number.

If you are using an LDAP SSL port other than 636, the following configuration step must be completed:

Add the following key pair into the `portalservlet.properties` file:

`LDAPSSLPort=your_port_number`

For example, if your Web server is running Active Directory you will need to make this change, because AD uses port 636. If you are running Tomcat, you would change the setting in the `PortalServlet.properties` file in the `tomcat\webapps\nps\WEB_INF` directory.

This setting will take a higher precedence than the default value of 636 if that value exists in the file.

After changing the setting, restart the Web server.

- 7** Make sure the LDAP server group object in eDirectory is configured to require TLS on simple binds. You set this option by editing the LDAP server object properties in iManager.

This step is necessary for the iManager task, Password Management > Set Universal Password.

- 8** To enable e-mail notification for Forgotten Password features, complete the steps in [“Configuring E-Mail Notification” on page 170](#).

You must set up the SMTP server, and customize the e-mail templates.

- 9** (NetWare 6.5 users only) If you have previously set up Universal Password for use with NetWare 6.5, complete the steps in [“\(NetWare 6.5 only\) Re-Creating Universal Password Assignments” on page 82](#).

You are now ready to use all the features of Password Policies. Create policies as described in [“Creating Password Policies” on page 84](#).

Deploying Password Policies without Universal Password

We recommend that you prepare your environment and turn on Universal Password so you can use all the features of Password Policies and Password Synchronization. However, if you are not ready to do so, there are some features you can use without deploying Universal Password.

For a list, see [“Feature Support for eDirectory 8.6.2 and eDirectory 8.7.3” on page 43](#). It explains what you can use with 8.6.2, or with 8.7.3 if you have Universal Password disabled.

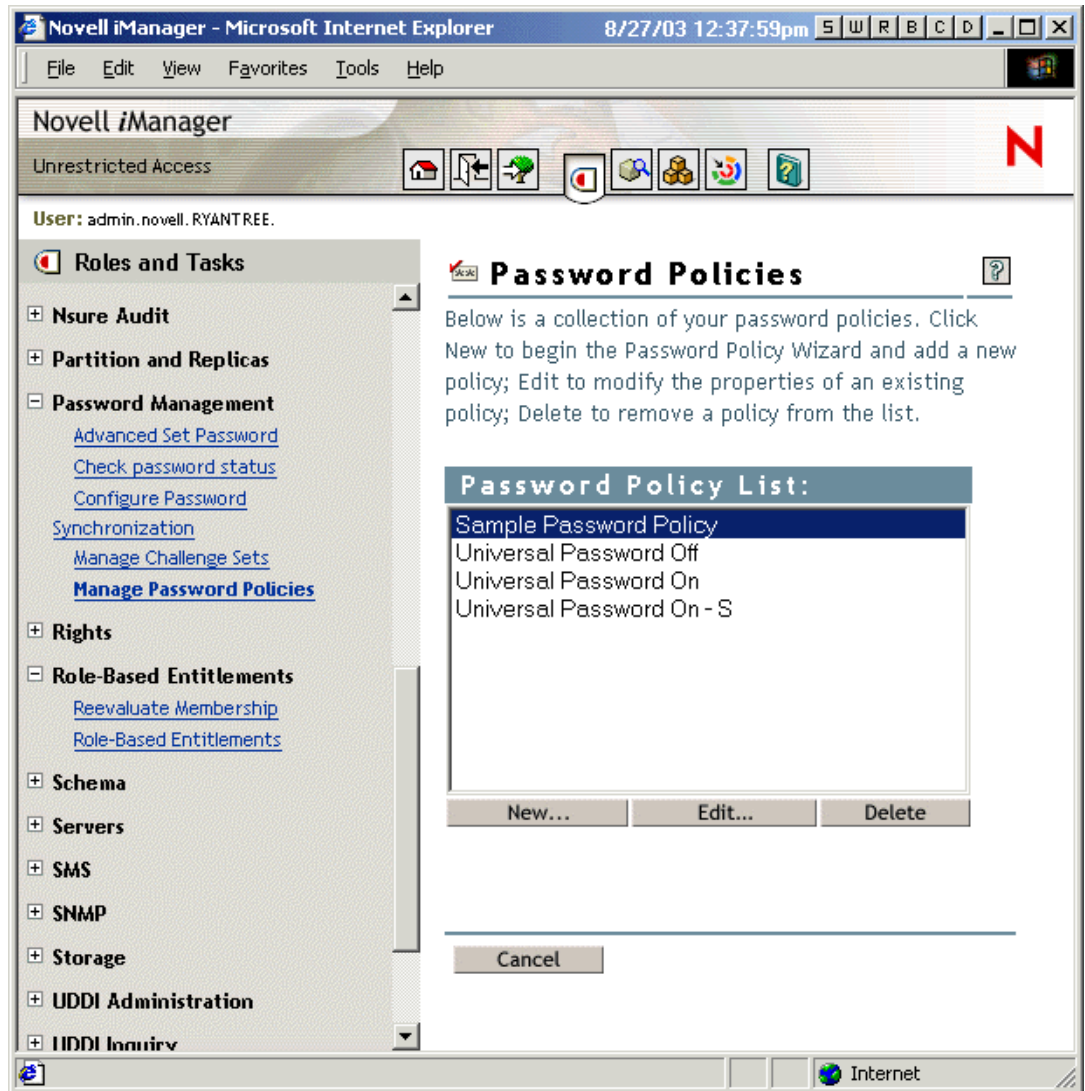
(NetWare 6.5 only) Re-Creating Universal Password Assignments

If you have previously set up Universal Password for use with NetWare 6.5, you must remove the old password policies, and use the new plugins and password policies.

- ◆ After you install Identity Manager, the NMAS plugins that were used in NetWare 6.5 for Universal Password will no longer be available. Instead you use Password Management > Manage Password Policies, which offers more features.
- ◆ The first time you use the Manage Password Policies in the new plugins, you will see three policy objects in the list that cannot be edited: Universal Password On, Universal Password Off, and Universal Password On - S.

These objects were used for the NetWare 6.5 implementation of Universal Password. To take advantage of the additional benefits of Password Policies provided by Identity Manager, you need to remove them.

The following figure shows an example:



To remove the old policy objects and re-create your policies using Password Policies:

- 1** Decide where you want Universal Password enabled in your tree.
 - ♦ Do you want it turned on for the same containers as when you set up Universal Password the first time with the NetWare 6.5 plugins? If so, continue with **Step 2** to find out where Universal Password is turned on.
 - ♦ Do you want it turned on everywhere in your tree? If so, simply create a new Password Policy with Universal Password enabled, and assign it to the Login Policy object. Then continue with **Step 4** to remove the old policies.

- 2** Find out where in the tree you had previously enabled Universal Password when you set it up using the plugins that shipped with NetWare 6.5.

This step is necessary because the plugins do not display where the assignments were made using the old plugins. Instead, you find out by searching the tree.

- 2a** Search the tree for objects that have the `nspmPasswordPolicyDN` populated with one of the following values:
 - ♦ Universal Password On

- ♦ Universal Password On - S

2b Make a note of all the containers that are the results of the search. These are the containers where Universal Password is turned on.

- 3** If you want Universal Password assigned in the same containers where you had assigned it previously, create one or more new Password Policies with Universal Password enabled, and assign them to the same containers.

Refer to the list of containers from the previous step, to make sure your assignments match.

- 4** Go into Password Management > Manage Password Policies, and remove the policy objects that remain from the first NetWare 6.5 implementation:

- ♦ Universal Password Off
- ♦ Universal Password On
- ♦ Universal Password On - S

After removing the old policy objects, you can use new Password Policies to meet your password needs.

Creating Password Policies

- 1** Make sure you have completed the steps in [“Prerequisites for Using Password Policies” on page 81](#). These steps prepare you to use all the features of Password Policies.
- 2** In iManager, click Password Management > Manage Password Policies.
- 3** Click New to create a new Password Policy.
- 4** Follow the steps in the wizard to create Advanced Password Rules, Universal Password Configuration Options, and Forgotten Password selections for the policy.

See the online help for information about each step, as well as the information in [Chapter 7, “Managing Passwords Using Password Policies,” on page 73](#) and in [Chapter 8, “Password Self-Service,” on page 89](#).

Assigning Password Policies to Users

We encourage you to set Password Policies as high up in the tree as you can, to simplify administration.

A policy is not in effect until you assign it to one or more objects. You can assign a password policy to the following objects:

- ♦ Login Policy object

We recommend that you create a default Password Policy for all users in the tree, which you do by creating a policy and assigning it to the Login Policy object. The Login Policy object is located in the Security container just below the root of the tree.

- ♦ A container that is a partition root

If you assign a policy to a container that is the root of a partition, the policy assignment is inherited by all users in that partition, including users in subcontainers. To determine whether a container is a partition root, browse for the container and note whether a partition icon displayed beside it, like the following example:

- ♦ A container that is not a partition root

If you assign a policy to a container that is not the root of a partition, the policy assignment is inherited only by users held in that specific container. It is not inherited by users that are held in subcontainers. If you want the policy to apply to all users below a container that is not a partition root, you must assign the policy to each subcontainer individually.

- ♦ A specific user

IMPORTANT: Make sure policies are assigned correctly for Driver Set objects. See [“Making Sure Password Policies Are Correct for Identity Manager” on page 85](#).

Only one policy is effective for a user at a time. Novell Modular Authentication Services™ (NMA) determines which policy is effective for a user by looking for policies in this order, and applying the first one it finds.

1. Specific user assignment: If a password policy has been assigned specifically to the user, that policy is applied.
2. Container: If the user has no specific assignment, NMA applies the policy that is assigned to the container which holds the user.
3. Partition root container: If no policy is assigned to the user or to the container directly above the user, the policy assigned to the partition root container is applied.
4. Login Policy object: If no policy is assigned to the user or other containers, the policy assigned to the Login Policy object is applied. It is the default policy for all users in the tree.

Making Sure Password Policies Are Correct for Identity Manager

One feature of Identity Manager is the ability to authenticate as a Driver object in order to carry out eDirectory tasks such as creating home directories, granting file permissions, or setting volume restrictions. This feature is available regardless of the platform on which Identity Manager is running; for example, Identity Manager running on Linux can create eDirectory users on NetWare with home directories. Another case where this feature is used is when a move occurs on the Publisher channel and the Identity Manager DirXML server does not hold the master replica of the source or destination partition.

When Identity Manager authenticates as a Driver object, it uses passwords which contain numeric characters and special characters and which can be up to 35 characters long.

To make sure that Identity Manager can carry out eDirectory tasks, you must do one of the following:

- ♦ Specifically to the Driver Set object assign a Password Policy, one that does not have Universal Password enabled.
- ♦ Make sure that the Password Policy that is in effect for the driver set does not restrict passwords in a way that would prevent Identity Manager passwords from being used.

The Password Policy in effect is the Password Policy assigned to the partition root container that holds the Driver Set object, or if there is none, the Password Policy that is assigned to the Login Policy object in the Security container.

If there is no policy on either of those objects, no policy applies to the driver set, so this is not an issue. Also, if there is a policy assigned, but Universal Password and Advanced Password Rules are not enabled, this not an issue.

If a Password Policy is on either the partition root or the Login Policy object, and Universal Password and Advanced Password Rules are turned on in that policy, the Advanced Password Rules must have the following settings:

- ♦ No maximum password length, or a maximum of 35 characters or more. (The default settings for a new Password Policy do not meet this requirement.)
- ♦ No restrictions on repeating characters.
- ♦ Numeric characters allowed, with no restrictions on placement or number of numeric characters.
- ♦ Special characters allowed, with no restrictions on placement or number of special characters.

Here's an example of a situation where you might encounter issues: If you are running Identity Manager on NetWare, and you have a connected system driver that creates users in eDirectory based on a template, you can specify in the template that home directories are created for those users. When Identity Manager creates a user with a home directory, it uses a password to do so, and the password contains numeric characters. If the Password Policy that applies to a driver set does not allow numeric characters, creating home directories does not work.

Finding Out Which Policy a User Has

Only one policy is in effect for a user at a time. To find out which policy is in effect for a particular user or container, go to iManager > Password Management > View Policy Assignment.

If there are multiple policies in the tree, NMAS determines which policy to apply to a user as described in [“Assigning Password Policies to Users” on page 84](#).

Setting A User's Password

Administrators or help desk personnel can set a user's Universal Password using a new task in iManager. The task shows the password rules for the Password Policy that is in effect for the user.

- 1 In iManager, click Password Management > Set Universal Password.

If the user has a password policy assigned, and Universal Password enabled, you are allowed to change the password using this task.

If the Advanced Password Rules are enabled in the policy, you see a list of rules that must be followed.

NOTE: If Universal Password is not enabled for a user, the Advanced Password Set task displays an error and the password is not changed. You must either assign a policy to the user and then return to this task, or change the user's NDS password using the eDirectory Administration > Modify Object task.

- 2 Create a password for the user, making sure it is compliant with all password rules that are displayed.

The Universal Password is changed for the user.

If Password Synchronization is set up in your environment, the user's new password is distributed to the connected systems that are configured to accept it.

Creating or Editing Challenge Sets

See [“Creating or Editing Challenge Sets” on page 110](#).

Configuring Notification for Password Features

Follow the instructions in “[Configuring E-Mail Notification](#)” on page 170.

Troubleshooting Password Policies

- ♦ iManager self-service login requiring full DN: If you have to type a full DN at the login prompt, most likely the user object does not reside under the container(s) specified during iManager/Portal configuration. You need to run the Portal servlet configuration wizard (using `http://your_server/nps/servlet/`), and specify additional login containers for the contextless login. The Forgotten Password feature also uses this setting to resolve a user's DN.
- ♦ Errors about Password Policy not assigned to a user: If you see an error saying that a Password Policy is not assigned to a user from the Set Universal Password task, and you know that the user does have a Password Policy assigned, SSL might be the issue.
 - ♦ To help confirm that SSL configuration is the problem, use the View Policy Assignment task to check the policy for that user. If the View Policy Assignment task displays an NMAS Transport error, this also can be an indicator that SSL is not configured properly.
 - ♦ Make sure that SSL is configured correctly between the Web server running iManager and the primary eDirectory tree. Confirm that you have a certificate configured between the Web server and eDirectory.

This can be a problem if you are running iManager on Windows 2000 machine with IIS as the web server, since iManager install doesn't automatically configure the certificate for you in that scenario.

- ♦ If you are using a nonstandard SSL port for eDirectory, make sure you follow the instructions in **Step 6** in “[Prerequisites for Using Password Policies](#)” on page 81.

For example, if you are running eDirectory and Active Directory on the same Windows 2000 machine, then eDirectory SSL is running on a nonstandard port.
- ♦ Using Challenge Response questions: Make sure that you are using a browser that iManager supports.
- ♦ Giving access to users in new containers: When you set up iManager, or one of Novell's portal products such as exteNd Director Standard Edition, you specify the portal users container. Usually you specify a container at a high level in the tree, so that all users in the tree can access portal features. If all your users are below that container, then all users have access to Forgotten Password and Reset Password Self-Service.

If you later create a container with users that is outside the portal users container, and these users can't access Forgotten Password and Reset Password features, you'll need to specifically assign rights to these gadgets for that new container: Challenge Response Setup, Change Universal Password, Hint Setup.

For instructions on adding new users to the portal users container, see Portal User in the [Novell exteNd Director Platform Edition Installation and Configuration Guide](http://www.novell.com/documentation/lg/nedpe41/configure/data/ajhotzv.html#ajhotzv) (<http://www.novell.com/documentation/lg/nedpe41/configure/data/ajhotzv.html#ajhotzv>).

- ♦ If you see a 222 error about a bad password when trying to create home directories for new users, it might be an issue with the Password Policy that applies to the driver set.

Determine which Password Policy is in effect for the container the driver set is in (for help, see the criteria in “[Assigning Password Policies to Users](#)” on page 84). Then make sure the

Advanced Password Rules for that Password Policy follow the guidelines in “[Making Sure Password Policies Are Correct for Identity Manager](#)” on page 85.

- ◆ NMAS LDAP Transport Error: If you are installing Identity Manager in a multi-server environment, and use some of the Password Management plug-ins in iManager, you might see an error that begins with "NMAS LDAP Transport Error."

One common cause of this error is that the PortalServlet.properties file is pointing to an LDAP server that does not have the NMASTM extensions that are needed for Identity Manager. Open the PortalServlet.properties file and make sure the address for the LDAP server is the same server where you installed Identity Manager.

Other possible causes:

- ◆ The LDAP server is not running.
- ◆ SSL is not configured for LDAP between the iManager server running the plug-ins and the LDAP server.
- ◆ When logging into other trees with iManager to manage remote Identity Manager DirXML servers, you might encounter errors if you use the server name instead of the IP address for the remote server.
- ◆ The trusted root certificate of the tree you authenticate to must be imported as a trusted certificate onto the Web server. You can use keytool.exe to export the certificate to the Web server. (If you install eGuide, the certificate is exported to the Web server during the configuration process.)
- ◆ The LDAP server group object in eDirectory must be configured to require TLS on simple binds. You set this option by editing the LDAP server object properties in iManager.

8

Password Self-Service

Using Password Policies, you can decrease help desk costs by providing users with self-service options for forgotten passwords and for resetting passwords.

Before using Password Self-Service, review the information about Password Policies in [Chapter 7, “Managing Passwords Using Password Policies,” on page 73](#).

In this section:

- ◆ [“Overview of Self-Service Features” on page 89](#)
- ◆ [“Prerequisites for Using Password Self-Service” on page 91](#)
- ◆ [“Planning for Login Methods for Password Self-Service” on page 92](#)
- ◆ [“Providing End Users with Forgotten Password Self-Service” on page 92](#)
- ◆ [“Providing End Users with Password Reset Self-Service” on page 109](#)
- ◆ [“Creating or Editing Challenge Sets” on page 110](#)
- ◆ [“Configuring Notification for Password Self-Service” on page 111](#)
- ◆ [“Troubleshooting Password Self-Service” on page 111](#)

Overview of Self-Service Features

Password Policies include Forgotten Password Self-Service features, to reduce help desk calls for forgotten passwords. Another self-service feature is Reset Password Self-Service, which lets users change their passwords while viewing the rules the administrator has specified in the Password Policy. Users access these features through the iManager self-service console.

Most features of password management require Universal Password to be enabled. Ideally, you would also integrate the iManager self-service console into your existing company portal, if you have one, to give users easy access to Forgotten Password Self-Service and Reset Password Self-service.

The new Password Self-Service features let you do the following:

- ◆ [“Provide Users with Forgotten Password Self-Service” on page 76](#)
- ◆ [“Provide Users with Reset Password Self-Service” on page 76](#)

Provide Users with Forgotten Password Self-Service

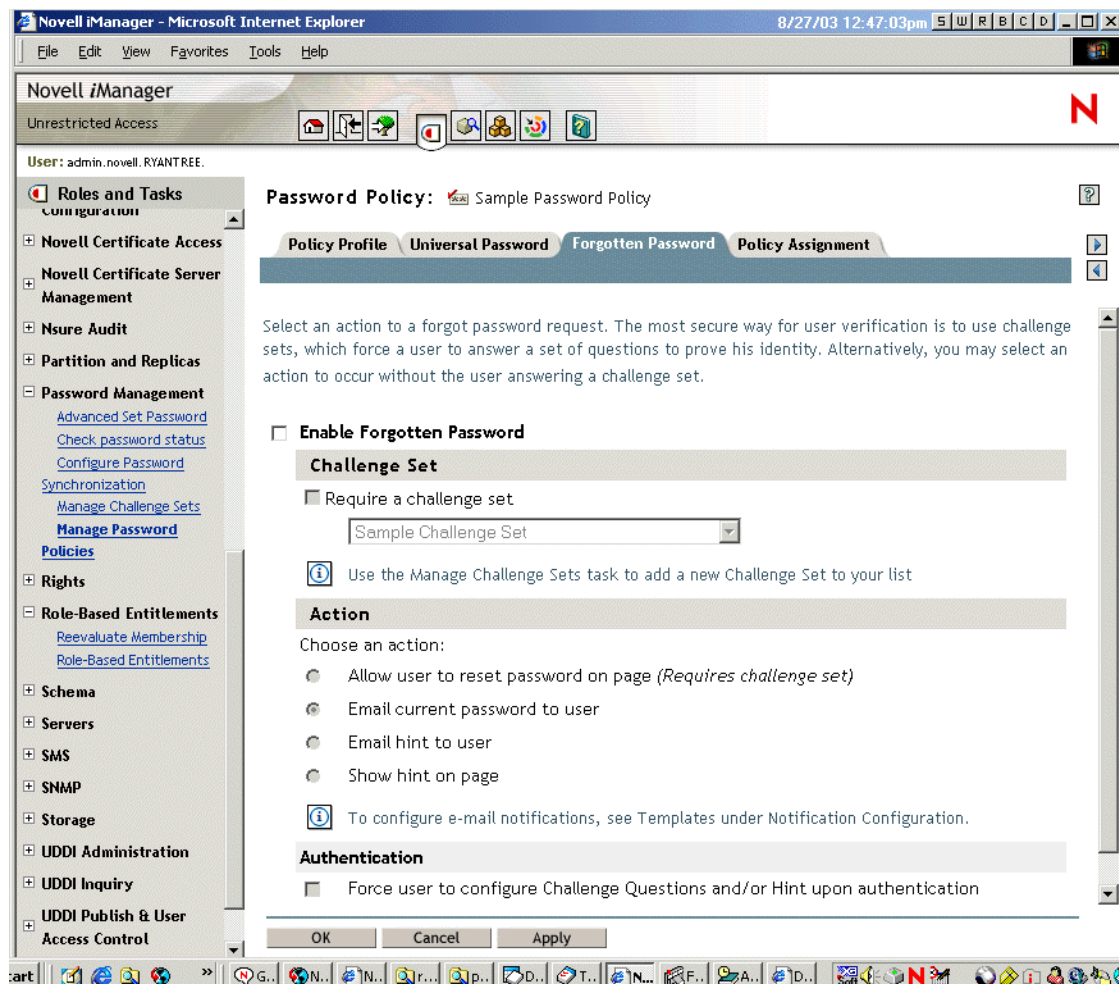
Using a Password Policy, you can provide users with the ability to recover from a forgotten password without contacting the help desk. The “Forgot your password?” link is available when users log in to the iManager self-service console.

The Forgotten Password Self-Service features include the following:

- ♦ Challenge Sets, to let user answer questions to prove identity
- ♦ Ability to e-mail a password hint or the forgotten password to the user
- ♦ Ability to let user reset password in the browser during a forgotten password request

To see examples of what the user will experience using the “Forgot your password?” link, see [“Providing End Users with Forgotten Password Self-Service” on page 92.](#)

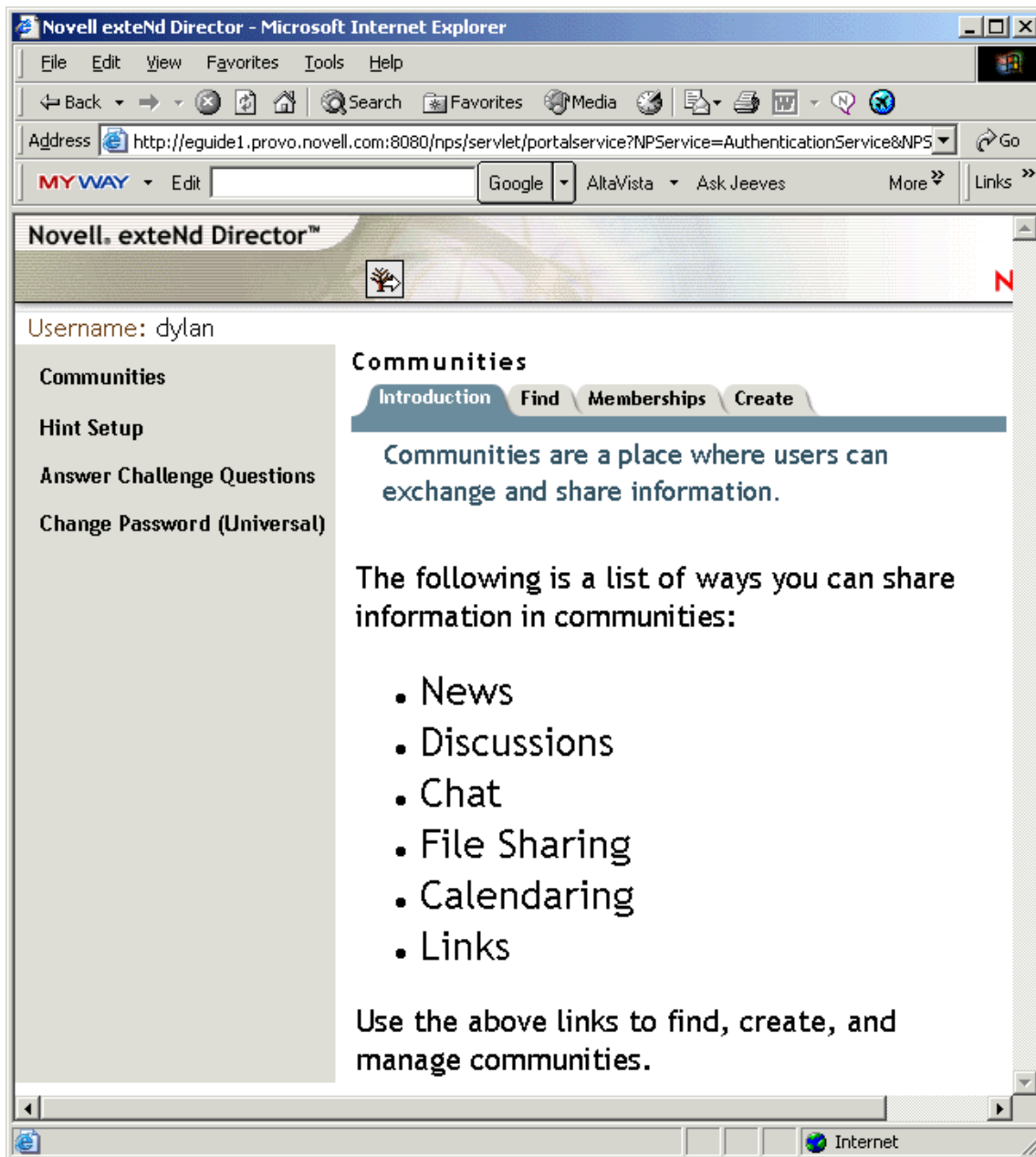
The following figure shows an example of the property page where you specify Forgotten Password settings for a Password Policy.



Provide Users with Reset Password Self-Service

Using the iManager self-service console, users can reset their passwords while viewing the Advanced Password Rules. They do this using the Change Password (Universal) gadget.

Here's an example of the screen they see when they log in to the iManager self-service console on your iManager web server, using a URL like <https://www.servername.com/nps>.



To see examples of what the user will experience using the Change Password (Universal) link, see [“Providing End Users with Password Reset Self-Service” on page 109](#).

Prerequisites for Using Password Self-Service

Review the information in [Chapter 7, “Managing Passwords Using Password Policies,” on page 73](#), including [“Prerequisites for Using Password Policies” on page 81](#).

We recommend that you prepare your environment and turn on Universal Password so you can use all the features of Password Policies. However, if you are not ready to do so, there are some features you can use without deploying Universal Password.

For a list, see [“Feature Support for eDirectory 8.6.2 and eDirectory 8.7.3” on page 43](#). It explains what you can use with 8.6.2, or with 8.7.3 if you have Universal Password disabled.

Planning for Login Methods for Password Self-Service

The iManager self-service console is accessible to end users on your iManager server using a URL such as <https://www.servername.com/nps>. For example, <https://www.myiManager.com/nps>.

Upgrade your client environment to support Universal Password.

- ♦ Novell Client (If you are using it, you must upgrade it. However, using the Novell Client is not required.)
- ♦ NT Client
- ♦ Web browser. If you use no client at all, and users log in through a web browser, you need to make sure they have one of the browsers that supports iManager 2.0.2 or later.

For more information, see the *Universal Password Deployment Guide* (http://www.novell.com/documentation/lg/nw65/universal_password/data/front.html).

Providing End Users with Forgotten Password Self-Service

When you use the New Password Policy wizard to create a Password Policy, you are prompted to decide what Forgotten Password features you want to provide to your end users.

This section gives more detail about your options and shows examples of the end-user experience when using the “Forgot your password?” link.

In this section:

- ♦ “Challenge Sets” on page 92
- ♦ “Forgotten Password Actions” on page 94
- ♦ “Password Hints” on page 94
- ♦ “Prompting End Users to Set Up Forgotten Password” on page 94
- ♦ “How the End User Sets Up Forgotten Password Self-Service” on page 95
- ♦ “What End Users See When They Forget Passwords” on page 104
- ♦ “Turning Off the Forgotten Password Link” on page 108

Challenge Sets

A Challenge Set is a set of questions that can be answered by a user to prove his or her identity, instead of using a password. A Challenge Set is a group of questions assigned to a password policy that is used as part of a password policy’s method of authentication. Challenge Sets are a feature of Password Policies that you can use as part of providing Forgotten Password self-service for your users. Requiring a user to answer Challenge Set questions before receiving forgotten password help provides an additional level of security. To use a Challenge Set, use the Manage Password Policies task to create a Password Policy and set up Forgotten Password.

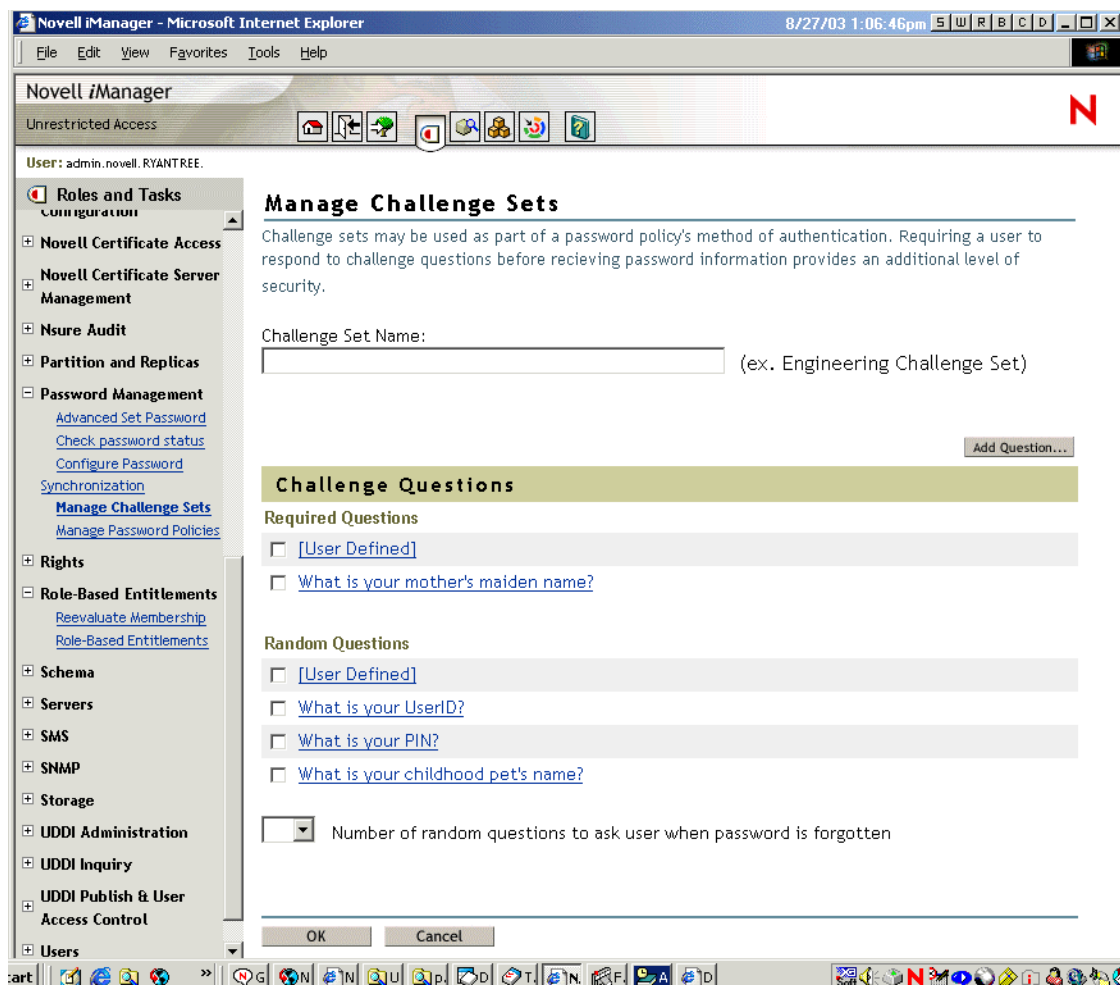
When you create a Password Policy, you can enable Forgotten Password self-service so that users can get help without calling the help desk. To make self-service more secure, you can create a Challenge Set and specify that users must answer the Challenge Set questions before obtaining forgotten password help. You also specify what action takes place to help users after they answer the questions, such as displaying a Password Hint to the user. These self-service features are available to users through the Novell® iManager self-service console. Your choices are explained in “Forgotten Password Actions” on page 94.

You define the structure of the Challenge Set questions, using the following choices:

- ♦ Administrator defined: The administrator can create questions that are presented to every user. Each user's answer, however, will be unique.
- ♦ User defined: The administrator can specify that one or more questions are created by the user. In this case, both the questions and the answers for each user will be unique.
- ♦ Required: Questions in this list are always presented to users when they use the Forgotten Password self-service feature.
- ♦ Random: Questions in this list will be presented to the user as a complete set only once, when the user sets up Forgotten Password by answering the Challenge Set questions the first time. When the user needs to access the Forgotten Password, only a few of the questions will be presented for the user to answer. The number of random questions presented is determined by the administrator.

A user's responses and user-defined questions are stored in Novell eDirectory™ by Novell Modular Authentication Services™ (NMAS™).

Here's an example of the screen where you create a new Challenge Set. You can choose from some sample questions that are provided by default, or add your own.



Forgotten Password Actions

The following Forgotten Password Actions are provided in a Password Policy, if you enable Forgotten Password:

- ◆ Allow user to reset password on page: After answering the Challenge Set questions to prove his or her identity, the user is allowed to change to a new password. Because the user has authenticated through answering the Challenge Questions, the user is allowed to change the password without being required to provide the old password. To use this option, the administrator must require a Challenge Set, and the user must have previously set up Forgotten Password in the iManager self-service console by answering the Challenge Set questions.
- ◆ E-mail current password to user: After answering the Challenge Set questions to prove his or her identity, the user receives the current password in an e-mail. To use this option, the administrator must enable Universal Password for the policy and enable the option “Allow user to retrieve password” (both are found in Universal Password > Configuration Options). Also, the user must have previously set up Forgotten Password in the iManager self-service console by answering the Challenge Set questions.
- ◆ E-mail hint to user: The user receives the Password Hint in an e-mail. To use this option, the user must have previously set up Forgotten Password in the iManager self-service console by providing a Password Hint.
- ◆ Show hint on page: The user is shown the Password Hint in the iManager self-service console. To use this option, the user must have previously set up Forgotten Password in the iManager self-service console by providing a Password Hint.

Password Hints

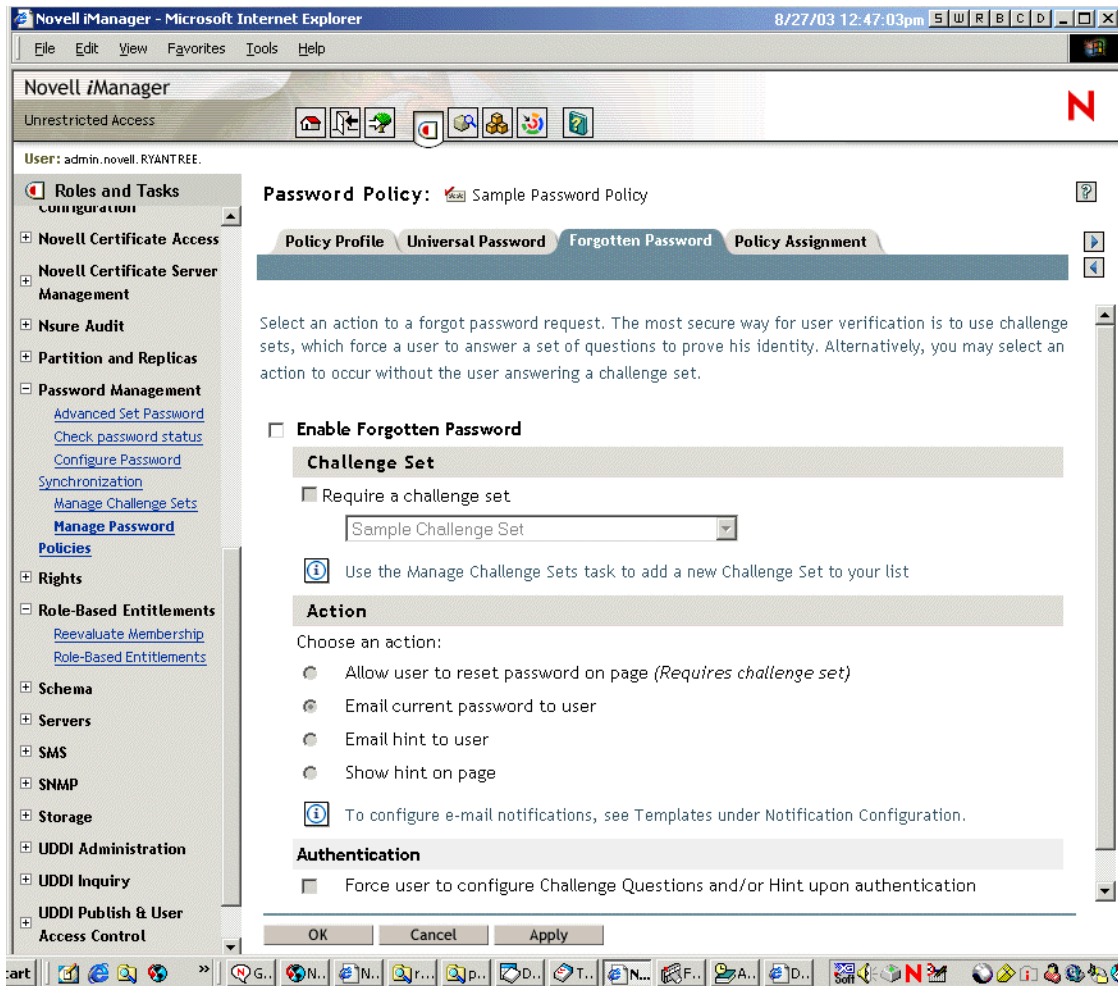
If you specify a Forgotten Password Action that requires Password Hint, the user can enter a hint that is a reminder of the password. The Password Hint is checked to make sure that it does not contain the users’s password.

Prompting End Users to Set Up Forgotten Password

For some Forgotten Password actions, the end user must do some setup before he or she can use the Forgotten Password self-service. For example, if the Password Policy specifies that a Challenge Set is used to allow a user to prove identity, and if the forgotten password action is to e-mail a Password Hint to the user, then the user must first answer Challenge Set questions and create a Password Hint before being able to use Forgotten Password Self-Service.

Users set up these features in the iManager Self-Service Console, or during a post-authentication sequence that you can require.

To prompt users to set up these features, check the option in the Password Policies interface at the bottom of the Forgotten Password tab, named “Force users to configure Challenge Questions and/or Hint upon authentication.” This is checked by default when you create a policy.



To let them set it up at any time initiated by the user, you need to give them the URL for the iManager self-service console, such as <https://www.servername.com/nps>.

How the End User Sets Up Forgotten Password Self-Service

Clicking the “Forgot your password?” link when logging in to the iManager self-service console (such as <https://www.servername.com/nps>) does not work for the user unless the following conditions are met:

- ♦ The administrator has set up a Password Policy with forgotten password enabled.
- ♦ The user has set up Challenge Questions or Password Hint, if either of them are specified in the Forgotten Password setting.

There are two ways the user’s part of the configuration can be accomplished:

- ♦ “User Setup for Forgotten Password, Post Authentication” on page 96
- ♦ “User Setup for Forgotten Password in iManager Self-Service Console” on page 99

User Setup for Forgotten Password, Post Authentication

The administrator can require the user to set up Forgotten password features after a successful login, by checking the Forgotten Password option “Force user to configure Challenge Questions and/or Hint upon authentication.” If this option is checked, and a user does not have them set up, then Forgotten Password configuration gadgets are displayed to the user the next time he or she logs in through the iManager self-service console (such as <https://www.servername.com/nps>). This is called post-authentication setup.

The following screen shows Challenge Set setup, post-authentication.

CRSetup - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print Mail News RSS Feeds Links

Answer Challenge Questions

Notice: Password policy requires that you set up your Challenge Questions before authentication.

These questions are assigned to your password policy. For all Admin-Defined Questions, provide a response. For all User-Defined Questions, create your own question and provide a response.

Admin-Defined Questions

Challenge Question: What is your mother's maiden name?

Challenge Response:

Challenge Question: What is your childhood pet's name?

Challenge Response:

User-Defined Questions

Challenge Question: ?

Challenge Response:

Challenge Question: ?

Challenge Response:

Done Local intranet

The following screen shows Password Hint setup, post-authentication.

The screenshot shows a Microsoft Internet Explorer window titled "HintSetup - Microsoft Internet Explorer". The address bar is empty. The menu bar includes File, Edit, View, Favorites, Tools, and Help. The toolbar contains Back, Forward, Stop, Home, Search, Favorites, Media, Print, and Links. The main content area has a heading "Define Password Hint" followed by a notice: "Notice: Password policy requires that you set up your Password Hint before authentication." Below the notice is a text prompt: "Please enter a password hint to help you remember your password." A form titled "Create a Password Hint" contains a "Username:" field with the value "dylan" and a "Password Hint:" field with a text input box. A "Submit" button is located below the input box. The status bar at the bottom shows "Done" and "Local intranet".

HintSetup - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Media Print Links »

Define Password Hint

Notice: Password policy requires that you set up your Password Hint before authentication.

Please enter a password hint to help you remember your password.

Create a Password Hint

Username: dylan

Password Hint:

Submit

Done Local intranet

User Setup for Forgotten Password in iManager Self-Service Console

When users log in through the portal, they enter the iManager self-service console.

Here the user has access to the gadgets for setting up or changing Challenge Sets and Password Hints for Forgotten Password Self-Service. This is the same place where the user can initiate a password change. The names of the gadgets the user can access here are

- ♦ Hint Setup
- ♦ Answer Challenge Questions
- ♦ Change Password (Universal)

The user can initiate changing these at any time. If a hint or Challenge Set is not required for the user's Password Policy, then they are unable to set them up. The page will display a message indicating that they are not accessible.

The following figure shows the Hint Setup page:

The screenshot shows a web browser window titled "Novell exteNd Director - Microsoft Internet Explorer" with the address bar showing "9/25/03 6:39:54pm". The browser's menu bar includes File, Edit, View, Favorites, Tools, and Help. The main content area displays the "Novell exteNd Director™" logo and a navigation sidebar on the left. The sidebar lists four options: "Communities", "Hint Setup", "Answer Challenge Questions", and "Change Password (Universal)". The "Hint Setup" option is selected. The main content area is titled "Define Password Hint" and contains a message: "Please enter a password hint to help you remember your password." Below this message is a section titled "Create a Password Hint" which includes a "Username:" field with the value "dylan", a "Password" field, and a "Hint:" field with the value "My dog's name and birthday". A "Submit" button is located at the bottom of the form. The browser's status bar at the bottom shows "Done" and "Internet".

The following figure shows the Answer Challenge Questions page:

The screenshot shows a web browser window titled "Novell exteNd Director - Microsoft Internet Explorer" with the address bar displaying "9/25/03 6:44:49pm". The browser's menu bar includes File, Edit, View, Favorites, Tools, and Help. The main content area is titled "Novell exteNd Director™" and features a Novell logo in the top right corner. On the left side, there is a navigation menu with the following items: "Communities", "Hint Setup", "Answer Challenge Questions" (which is highlighted), and "Change Password (Universal)". The main content area is titled "Answer Challenge Questions" and contains the following text: "These questions are assigned to your password policy. For all Admin-Defined Questions, provide a response. For all User-Defined Questions, create your own question and provide a response." Below this text, there are two sections: "Admin-Defined Questions" and "User-Defined Questions". The "Admin-Defined Questions" section contains two sets of "Challenge Question" and "Challenge Response" fields. The first set has the question "What is your mother's maiden name?" and the second set has the question "What is your childhood pet's name?". The "User-Defined Questions" section contains two sets of "Challenge Question" and "Challenge Response" fields, each with a question mark icon next to the question field. A "Submit" button is located at the bottom right of the form. The browser's status bar at the bottom shows "Done" and "Internet".

Novell exteNd Director™

Novell.

Username: dylan

Communities

Hint Setup

Answer Challenge Questions

Change Password (Universal)

Answer Challenge Questions

These questions are assigned to your password policy. For all Admin-Defined Questions, provide a response. For all User-Defined Questions, create your own question and provide a response.

Admin-Defined Questions

Challenge Question: What is your mother's maiden name?

Challenge Response:

Challenge Question: What is your childhood pet's name?

Challenge Response:

User-Defined Questions

Challenge Question: ?

Challenge Response:

Challenge Question: ?

Challenge Response:

Submit

The first questions listed in this example are administrator-defined, and the others are user-defined. The user answers the administrator questions, and creates both a question and answer for the user-defined questions, as in the following example:

Novell exteNd Director - Microsoft Internet Explorer 9/25/03 6:46:56pm S W R B C D

File Edit View Favorites Tools Help

Novell exteNd Director™ Novell.

Username: dylan

Communities

Hint Setup

Answer Challenge Questions

Change Password (Universal)

Answer Challenge Questions

These questions are assigned to your password policy. For all Admin-Defined Questions, provide a response. For all User-Defined Questions, create your own question and provide a response.

Admin-Defined Questions

Challenge Question:	What is your mother's maiden name?
Challenge Response:	<input type="text" value="Ranadive"/>
Challenge Question:	What is your childhood pet's name?
Challenge Response:	<input type="text" value="Cocoa"/>

User-Defined Questions

Challenge Question:	<input type="text" value="What street did you grow up on?"/>
Challenge Response:	<input type="text" value="Van Dorn Street"/>
Challenge Question:	<input type="text" value="What is your favorite food?"/>
Challenge Response:	<input type="text" value="rice"/>

The following figure shows the Change Password (Universal) page:

The screenshot shows a Microsoft Internet Explorer browser window titled "Novell exteNd Director - Microsoft Internet Explorer". The address bar shows the Novell logo and "Novell exteNd Director™". The browser's menu bar includes File, Edit, View, Favorites, Tools, and Help. The toolbar contains buttons for Back, Forward, Stop, Home, Search, Favorites, Media, and a Links button. The main content area displays the "Novell exteNd Director™" header with a Novell logo. Below the header, the username "dylan" is shown. A left sidebar contains a list of links: Communities, Hint Setup, Answer Challenge Questions, and Change Password (Universal). The main content area is titled "Change Password" and contains the following text: "You can now change your password. Type in your new password twice and make sure the password conforms to the displayed rules." Below this, it states "Your password must have the following properties:" followed by a bulleted list: "Minimum number of characters in password: 4" and "Maximum number of characters in password: 12". It then lists three rules: "You may use numbers in your password", "The password is case-sensitive", and "The password may use special characters". Below these rules, it states "You cannot use the following character combinations as passwords:" followed by a bulleted list: "novell" and "admin". At the bottom of the form, there are three input fields labeled "Old password:", "New password:", and "Retype password:". A "Submit" button is located below the "Retype password:" field. The status bar at the bottom of the browser window shows "Done" and "Local intranet".

Novell exteNd Director™

Novell

Username: dylan

Communities

Hint Setup

Answer Challenge Questions

Change Password (Universal)

Change Password

You can now change your password. Type in your new password twice and make sure the password conforms to the displayed rules.

Your password must have the following properties:

- Minimum number of characters in password: 4
- Maximum number of characters in password: 12

You may use numbers in your password

The password is case-sensitive

The password may use special characters

You cannot use the following character combinations as passwords:

- novell
- admin

Old password:

New password:

Retype password:

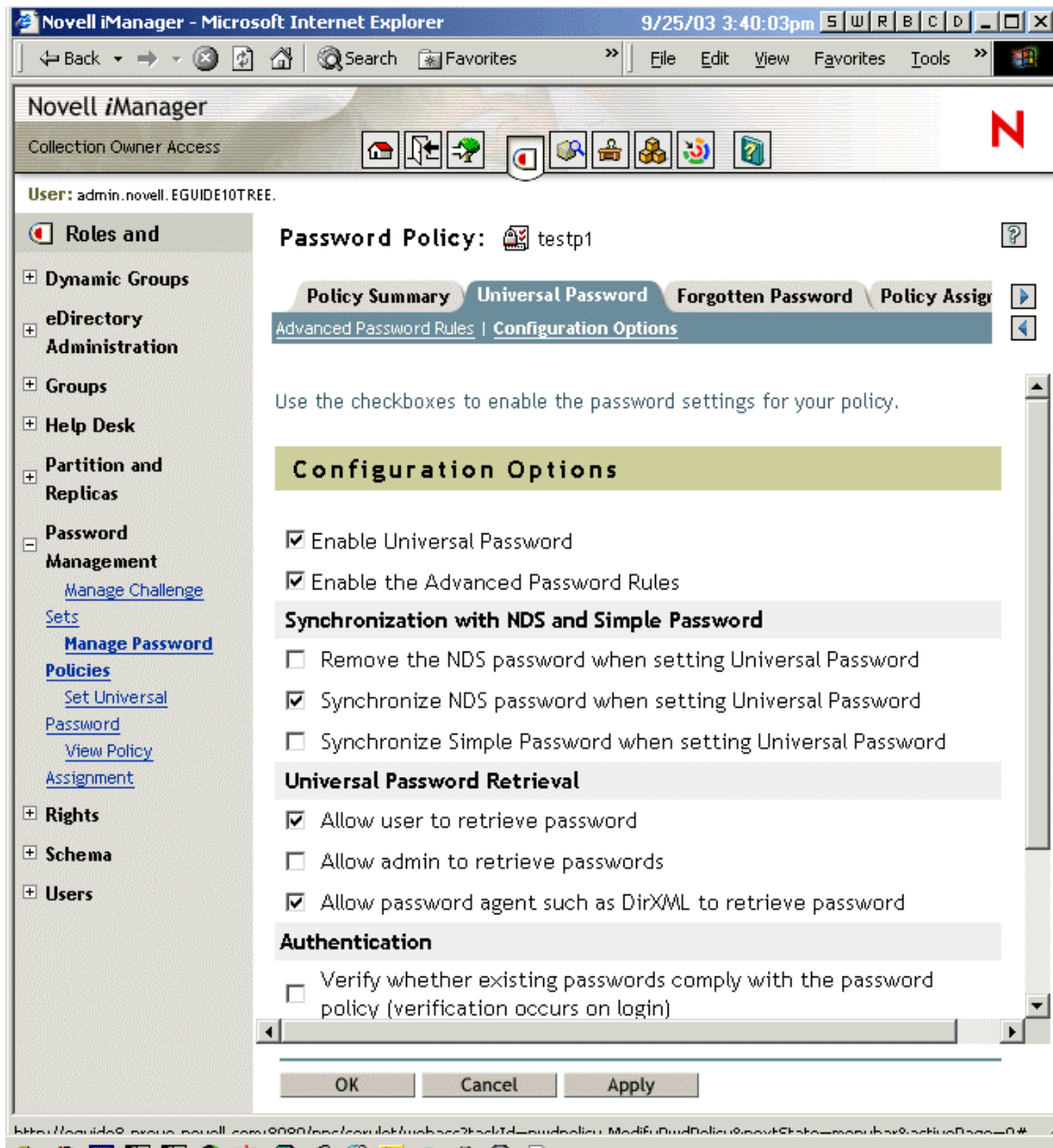
Submit

Done Local intranet

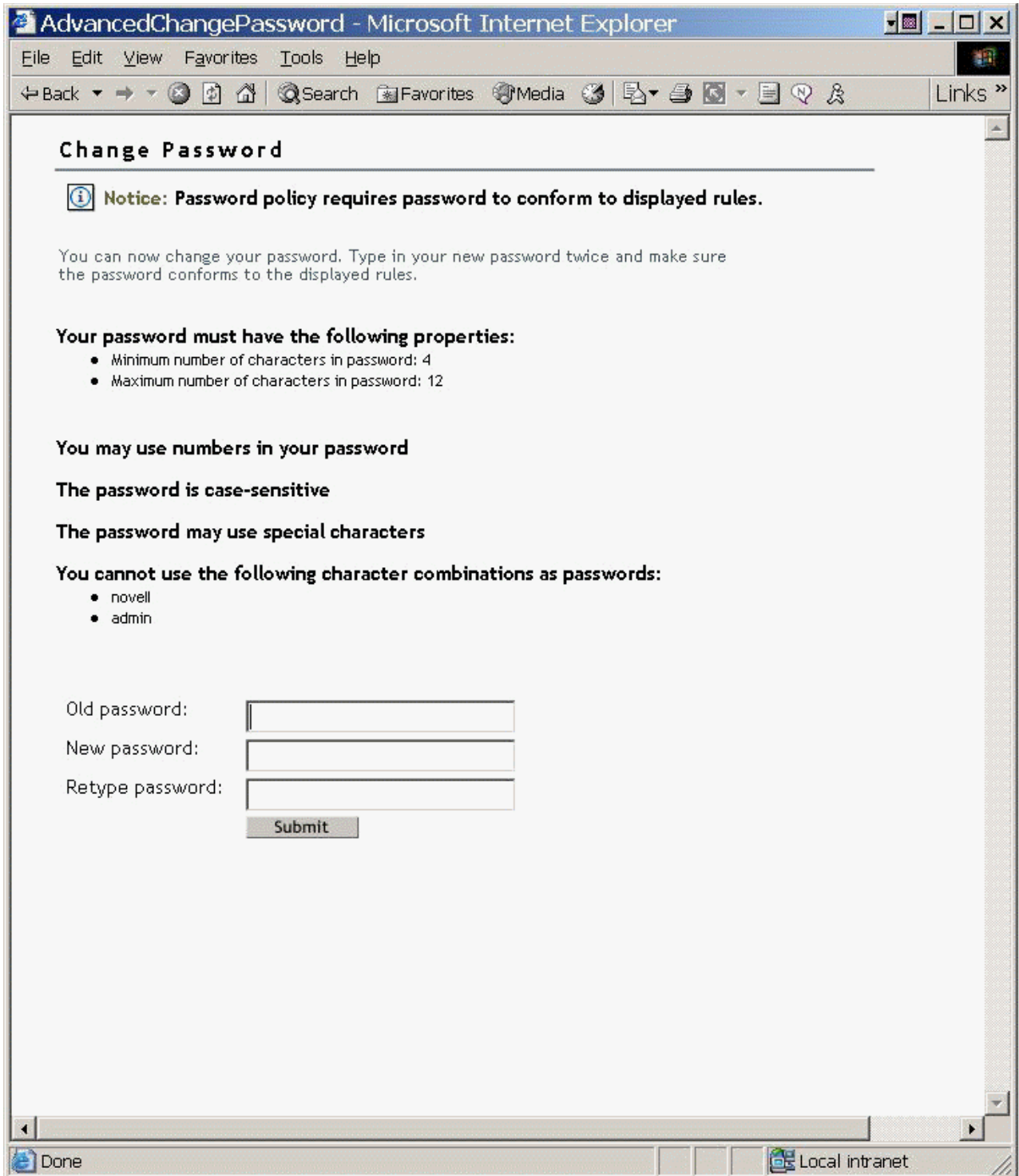
Requiring Existing Passwords to Comply

If an administrator creates or changes a Password Policy, he or she can require users to change existing passwords that don't comply, the next time they log in through the portal.

This is done by setting an option in the Password Policy, in the Universal Password tab under Configuration Options. The option is called "Verify whether existing passwords comply with the password policy (verification occurs on login)." By default, this option is turned off when you create a new Password Policy. The following figure shows the page where you set this option:



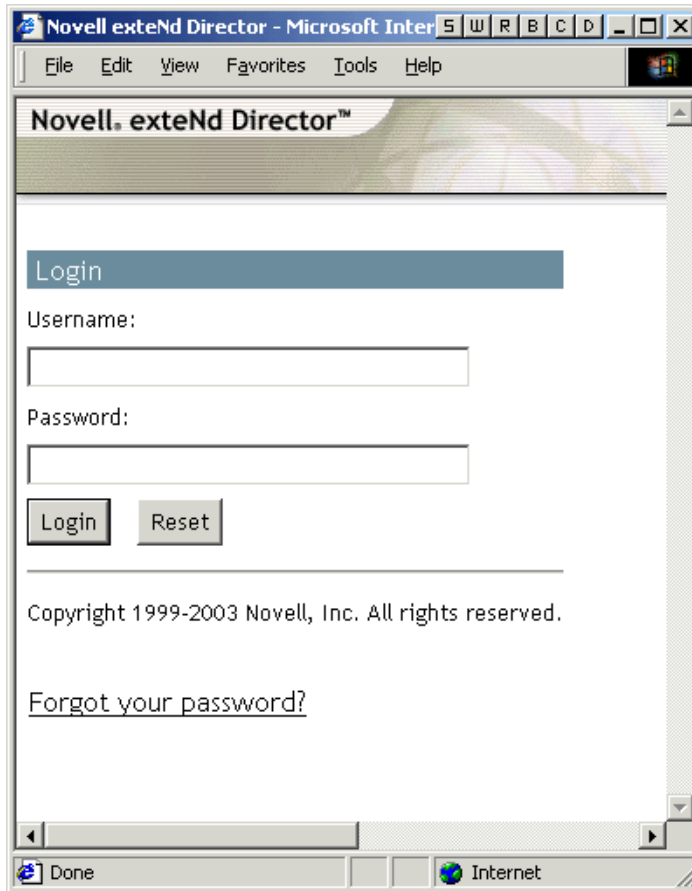
If this option is set, the next time users log in through the portal, their passwords will be checked for compliance with the Password Policy. If the password does not comply, a page like the following is displayed, and the user is not allowed to log in without changing the password.



What End Users See When They Forget Passwords

This section explains the user's experience when using Forgotten Password Self-Service.

After you have installed DirXML plug-ins, which shipped with Identity Manager, on your iManager server, the Forgotten Password link shows up in the iManager self-service console (such as <https://www.servername.com/nps>), as shown in the following figure.



If a user clicks this link, the following page is displayed, asking for the username:



After the username is entered, the Forgotten Password settings determine what the user sees.

For example, if the administrator specified in the Password Policy that a Challenge Set is used, then a page like the following is displayed, and the user must answer Challenge Set questions to prove his or her identity.

Challenge Response

Please provide a response for each presented challenge.

Challenge Questions

Challenge What street did you grow up on?
Response

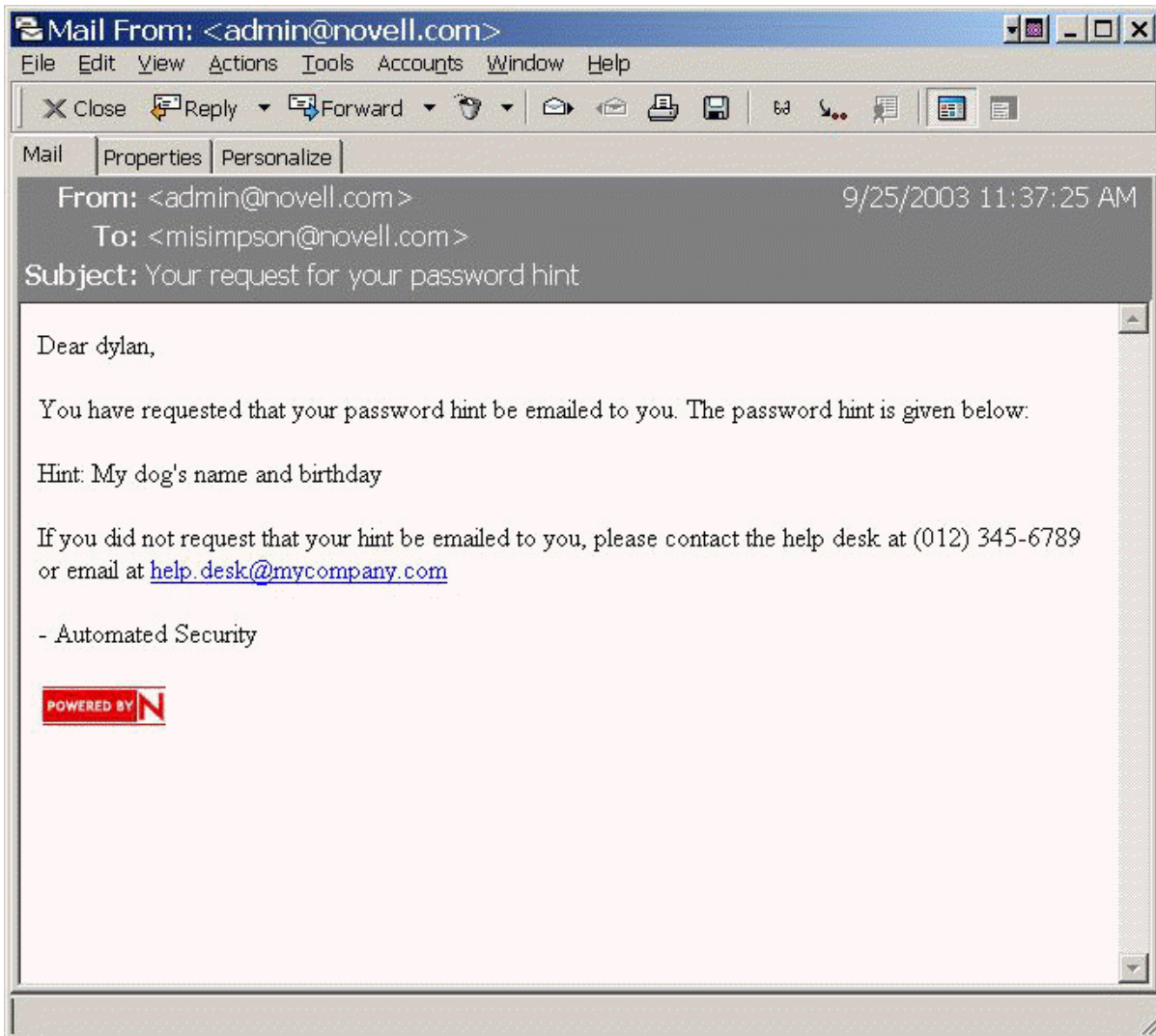
Challenge What is your mother's maiden name?
Response

Challenge What is your childhood pet's name?
Response

If the administrator specified that the Forgotten Password action is “Show hint on page,” a page like the following is displayed:




If the administrator specified that the Forgotten Password action is “E-mail current password to user,” or “E-mail hint to user,” a message is displayed on the page saying that the password or hint has been e-mailed. The user will then receive an e-mail like the following:



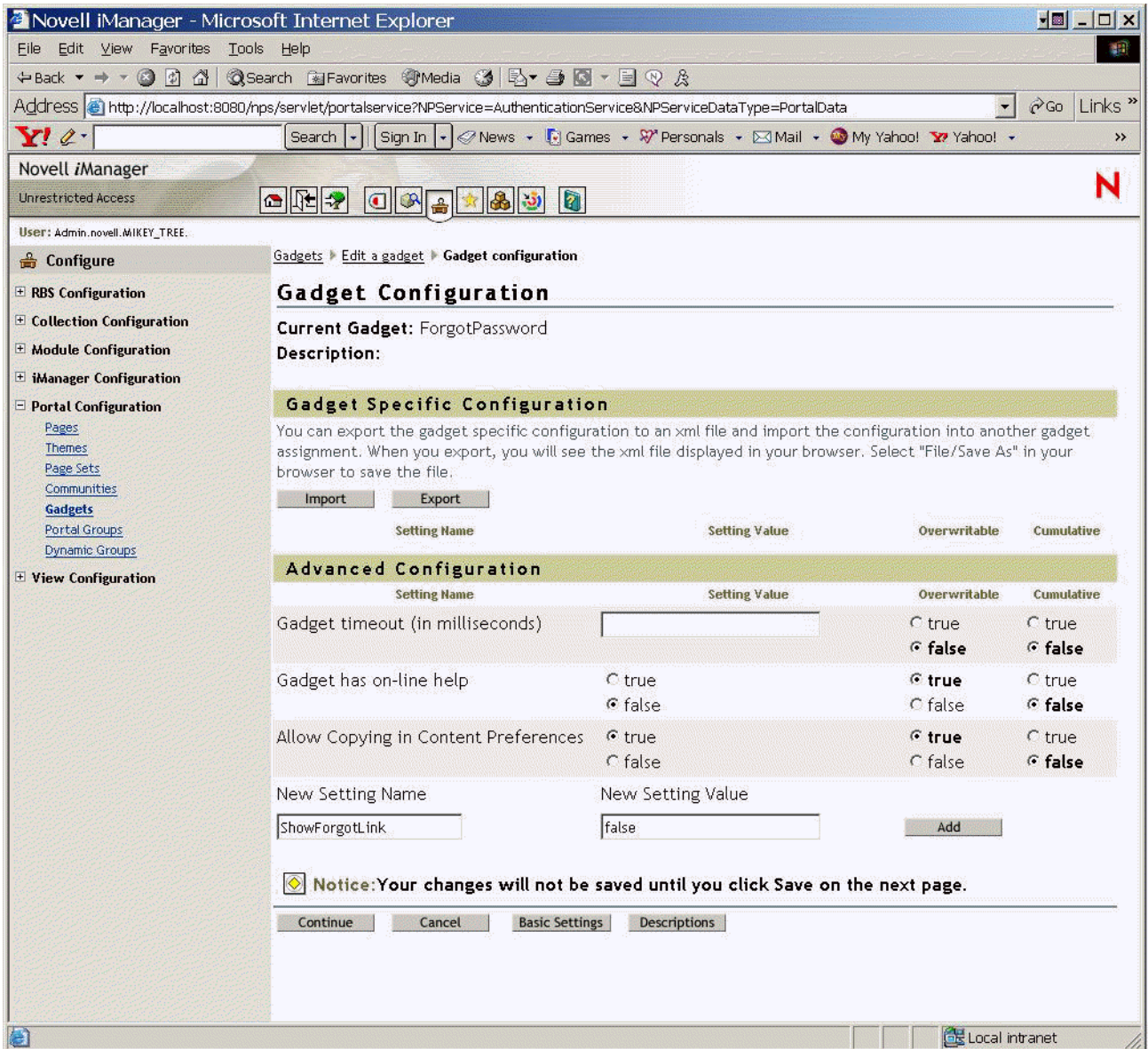
Turning Off the Forgotten Password Link

If you don't want the "Forgot your password?" link to appear in the portal, you can turn it off using the following steps:

- 1 In iManager, click the Configure icon  to enter the Administration gadget.
- 2 Click Portal Platform Configuration > Gadgets.
- 3 In the list of Gadgets, select the Forgot Password gadget.
- 4 Click the Edit button, then click Configuration. Click the All Settings button.
- 5 Add a keypair in the gadget settings, as shown in the figure.

```
ShowForgotLink=false
```

If this keypair does not exist at all in the gadget settings, the default behavior is true.

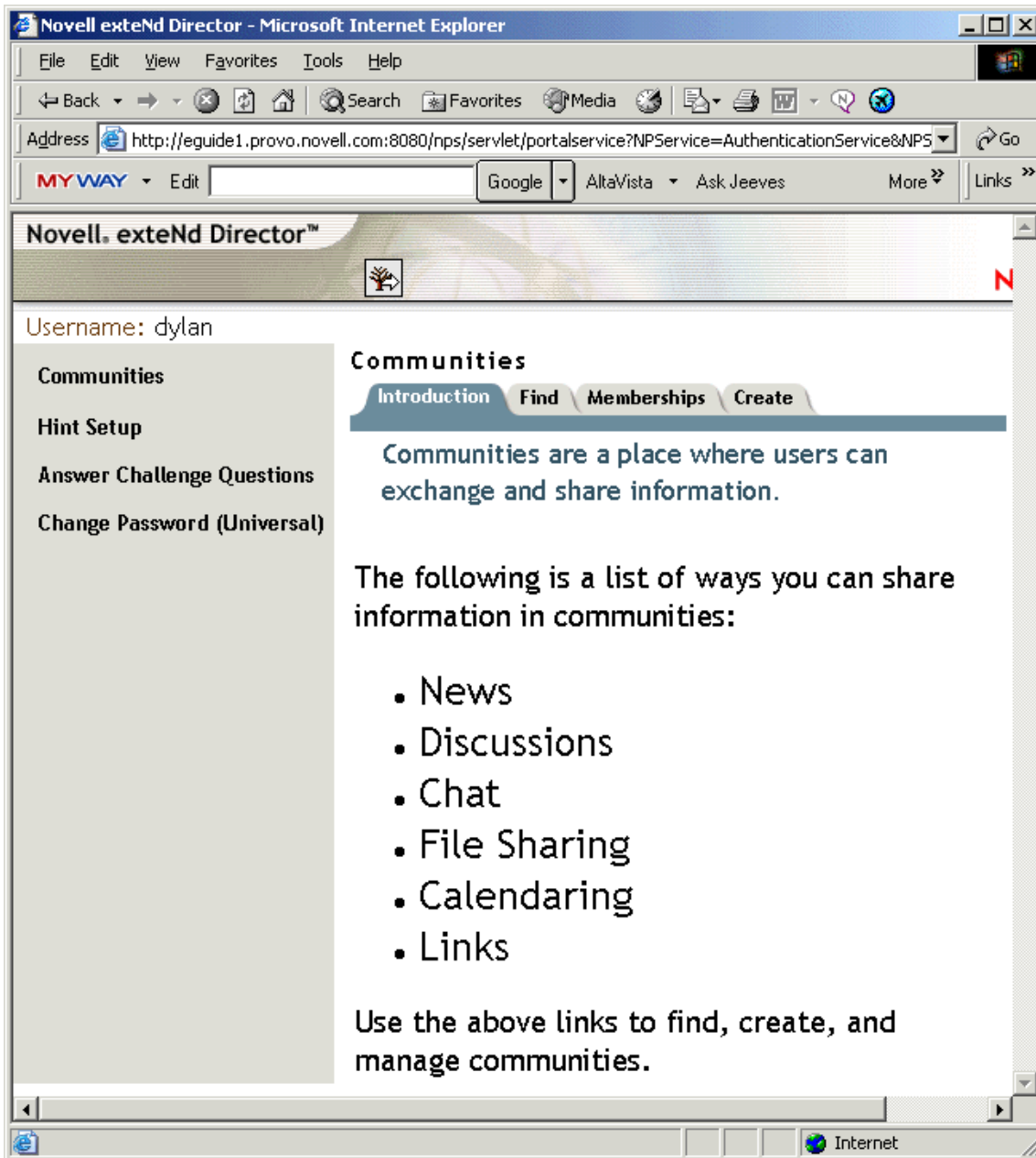


- 6 Click Continue, and on the next page click Save to save the changes.
- 7 Restart the web server so it will take effect.

Providing End Users with Password Reset Self-Service

Users can reset their passwords in the iManager self-service console, accessed using a URL such as <https://www.servername.com/nps>. For example, <https://www.myiManager.com/nps>.

Here's an example of the iManager self-service console after login.



Creating or Editing Challenge Sets

Challenge Sets are a feature of Password Policies that can help you set up Forgotten Password self-service for your users. A Challenge Set is a set of questions that can be answered by a user to prove his or her identity, instead of using a password.

When you create a Password Policy, you can enable Forgotten Password self-service so that users can get help without calling the help desk. To make self-service more secure, you can create a Challenge Set and specify that users must answer the Challenge Set questions before obtaining forgotten password help.

You can create a Challenge Set when you are creating a Password Policy: in iManager, Password Management > Manage Password Policies > New.

You can also manage them as a separate task: in iManager, Password Management > Manage Challenge Sets.

Before a user can use Challenge Sets, he or she must set up the questions and answers. You can require that users set them up the next time they log in to iManager or the iManager self-service console using an option in the Password Policy on the Forgotten Password tab: “Force user to configure Challenge Questions and/or Hint upon authentication.” A user can initiate this setup or change it in the iManager self-service console.

You define the structure of the Challenge Set questions, using the following choices:

Admin-Defined: The administrator can create questions that are presented to every user. Each user's answer, however, will be unique.

User-Defined: The administrator can specify that one or more questions are created by the user. In this case, both the questions and the answers for each user will be unique.

Required: Questions in this list are always presented to users when they use the Forgotten Password self-service feature.

Random: Questions in this list will be presented to the user as a complete set only once, when the user sets up Forgotten Password by answering the Challenge Set questions the first time. When the user needs to access the Forgotten Password, only a few of the questions will be presented for the user to answer. The number of random questions presented is determined by the administrator.

A user's responses and user-defined questions are stored in Novell eDirectory by Novell Modular Authentication Services (NMAS).

Configuring Notification for Password Self-Service

Follow the instructions in [“Configuring E-Mail Notification” on page 170](#).

Troubleshooting Password Self-Service

- ♦ iManager self-service login requiring full DN: If you have to type a full DN at the login prompt, most likely the user object does not reside under the container(s) specified during iManager/Portal configuration. You need to run the Portal servlet configuration wizard (using `http://your_server/nps/servlet/`), and specify additional login containers for the contextless login. The Forgotten Password feature also uses this setting to resolve a user's DN.
- ♦ Errors about Password Policy not assigned to a user: If you see an error saying that a Password Policy is not assigned to a user from the Set Universal Password task, and you know that the user does have a Password Policy assigned, SSL might be the issue.
 - ♦ To help confirm that SSL configuration is the problem, use the View Policy Assignment task to check the policy for that user. If the View Policy Assignment task displays an NMAS Transport error, this also can be an indicator that SSL is not configured properly.
 - ♦ Make sure that SSL is configured correctly between the Web server running iManager and the primary eDirectory tree. Confirm that you have a certificate configured between the Web server and eDirectory.

This can be a problem if you are running iManager on Windows 2000 machine with IIS as the web server, since iManager install doesn't automatically configure the certificate for you in that scenario.

- ♦ If you are using a nonstandard SSL port for eDirectory, make sure you follow the instructions in **Step 6** in “**Prerequisites for Using Password Policies**” on page 81.

For example, if you are running eDirectory and Active Directory on the same Windows 2000 machine, then eDirectory SSL is running on a nonstandard port.

- ♦ Using Challenge Response questions: Make sure that you are using a browser that iManager supports.
- ♦ Giving access to users in new containers: When you set up iManager, or one of Novell’s portal products such as exteNd Director Standard Edition, you specify the portal users container. Usually you specify a container at a high level in the tree, so that all users in the tree can access portal features. If all your users are below that container, then all users have access to Forgotten Password and Reset Password Self-Service.

If you later create a container with users that is outside the portal users container, and these users can't access Forgotten Password and Reset Password features, you'll need to specifically assign rights to these gadgets for that new container: Challenge Response Setup, Change Universal Password, Hint Setup.

For instructions on adding new users to the portal users container, see Portal User in the *Novell exteNd Director Platform Edition Installation and Configuration Guide* (<http://www.novell.com/documentation/lg/nedpe41/configure/data/ajhotzv.html#ajhotzv>).

- ♦ If you see a 222 error about a bad password when trying to create home directories for new users, it might be an issue with the Password Policy that applies to the driver set.
Determine which Password Policy is in effect for the container the driver set is in (for help, see the criteria in “**Assigning Password Policies to Users**” on page 84). Then make sure the Advanced Password Rules for that Password Policy follow the guidelines in “**Making Sure Password Policies Are Correct for Identity Manager**” on page 85.
- ♦ NMAST LDAP Transport Error: If you are installing Identity Manager in a multi-server environment, and use some of the Password Management plug-ins in iManager, you might see an error that begins with “NMAST LDAP Transport Error.”

One common cause of this error is that the PortalServlet.properties file is pointing to an LDAP server that does not have the NMASTM extensions that are needed for Identity Manager. Open the PortalServlet.properties file and make sure the address for the LDAP server is the same server where you installed Identity Manager.

Other possible causes:

- ♦ The LDAP server is not running.
- ♦ SSL is not configured for LDAP between the iManager server running the plug-ins and the LDAP server.
- ♦ When logging into other trees with iManager to manage remote Identity Manager DirXML servers, you might encounter errors if you use the server name instead of the IP address for the remote server.
- ♦ The trusted root certificate of the tree you authenticate to must be imported as a trusted certificate onto the Web server. You can use keytool.exe to export the certificate to the Web server. (If you install eGuide, the certificate is exported to the Web server during the configuration process.)
- ♦ The LDAP server group object in eDirectory must be configured to require TLS on simple binds. You set this option by editing the LDAP server object properties in iManager.

9

Password Synchronization across Connected Systems

Nsure™ Identity Manager Password Synchronization offers several new benefits:

- ♦ Bidirectional password synchronization
- ♦ Enforcement of Password Policies on connected systems
- ♦ E-mail notification when synchronization fails
- ♦ Ability to check password synchronization status for a user

To help you understand your options, some scenarios are described in [“Implementing Password Synchronization” on page 138](#).

In this section:

- ♦ [“Overview” on page 113](#)
- ♦ [“Connected System Support for Password Synchronization” on page 119](#)
- ♦ [“Pieces Required for Password Synchronization to Work” on page 120](#)
- ♦ [“Preparing to Use Identity Manager Password Synchronization and Universal Password” on page 127](#)
- ♦ [“New Driver Configuration and Identity Manager Password Synchronization” on page 131](#)
- ♦ [“Upgrading Password Synchronization 1.0 to Identity Manager Password Synchronization” on page 132](#)
- ♦ [“Upgrading Existing Driver Configurations to Support Identity Manager Password Synchronization” on page 132](#)
- ♦ [“Implementing Password Synchronization” on page 138](#)
- ♦ [“Setting Up Password Filters” on page 165](#)
- ♦ [“Managing Password Synchronization” on page 166](#)
- ♦ [“Checking the Password Synchronization Status for a User” on page 170](#)
- ♦ [“Configuring E-Mail Notification” on page 170](#)
- ♦ [“Troubleshooting Password Synchronization” on page 180](#)

Overview

Identity Manager introduces bidirectional password synchronization, by taking advantage of Universal Password and connected system support for publishing or subscribing to passwords.

As with other attributes for a user account, you can choose your authoritative data sources.

- ♦ [“What is Bidirectional Password Synchronization?” on page 114](#)
- ♦ [“Features of Identity Manager Password Synchronization” on page 114](#)
- ♦ [“Diagrams of Password Synchronization Flow” on page 118](#)

What is Bidirectional Password Synchronization?

Bidirectional password synchronization is the combination of Identity Manager accepting passwords from the connected systems you specify, and distributing passwords to the connected systems you specify.

The ability to have bidirectional password synchronization with a particular connected system depends on what the connected system supports.

Some connected systems can accept new and modified passwords from Identity Manager, and can also provide the user’s actual password to Identity Manager. These connected systems are the ones that support bidirectional password synchronization with Identity Manager. They are the following:

- ♦ Active Directory
- ♦ Novell® eDirectory™
- ♦ NIS
- ♦ NT Domain

For these connected systems, the user can change a password in one of the systems and have that password synchronized to the other systems through Identity Manager. However, if you are using Advanced Password Rules in your Password Policies, it’s best to have users make password changes in the iManager self-service console. This is the best place for password changes because it lists all the rules that the user’s password must comply with.

Other connected systems can’t provide the user’s actual password, so they can’t support full bidirectional password synchronization. But they can provide data that can be used to create passwords and send them to Identity Manager, by defining policies within the driver configuration.

Several other systems can accept passwords from Identity Manager, whether initial password set for a new user, modifying a password, or both.

See [“Connected System Support for Password Synchronization” on page 119](#).

Features of Identity Manager Password Synchronization

To explain the features offered by Identity Manager Password Synchronization, we can divide the subject of bidirectional password synchronization into the two directions: passwords sent from connected systems and accepted by Identity Manager, and passwords distributed by Identity Manager and accepted by connected systems.

The following sections explain the password synchronization features of Identity Manager:

- ♦ [“Identity Manager Can Accept Passwords from Connected Systems” on page 115](#)
- ♦ [“Identity Manager Can Distribute Passwords to Connected Systems” on page 115](#)
- ♦ [“Identity Manager Can Enforce Password Policies, in the Data Store and on Connected Systems” on page 116](#)
- ♦ [“Identity Manager Offers Several Scenarios for Synchronizing Passwords” on page 116](#)

- ♦ [“Identity Manager Can Notify Users of Password Synchronization Failures” on page 117](#)
- ♦ [“Identity Manager Can Check the Password Synchronization Status for a User” on page 117](#)

Identity Manager Can Accept Passwords from Connected Systems

As in previous versions of DirXML, any connected system can publish a password to the identity vault.

You can specify which connected system applications Identity Manager will accept passwords from. You can even choose whether Identity Manager updates the password for users in the same eDirectory tree where Identity Manager is running, or whether Identity Manager simply acts as a conduit or “tunnel,” synchronizing passwords only between connected systems; thus making it possible to keep the eDirectory password separate from the password that Identity Manager distributes to connected systems, if desired.

Some connected systems (AD, other eDirectory trees, NT, and NIS) can provide the user’s actual password, which means that when a user changes a password on a connected system, the change can be synchronized to Identity Manager and back out to other connected systems.

Other connected systems don’t support providing the user’s actual password, but you can configure them to provide a password to Identity Manager that is manufactured in a style sheet, such as an initial password based on last name or employee ID.

Identity Manager Can Distribute Passwords to Connected Systems

Identity Manager Password Synchronization introduces the ability to distribute a common password to connected systems.

In previous versions of DirXML, a driver could send passwords to DirXML from a user account on a connected system, and the password could be used to update the corresponding user in eDirectory. But because the NDS password in eDirectory is non-reversible, you couldn’t push a password out from the central Identity Manager identity vault to multiple connected systems. You could obtain the eDirectory password only by capturing the password before it was stored in eDirectory, such as through the Novell Client.

The new Universal Password provided by eDirectory 8.7.3 is reversible, so it can be distributed.

Identity Manager can accept a password from a connected system, and because Universal Password is reversible, Identity Manager can distribute the password from the identity vault to connected systems that support setting initial passwords for new accounts and modifying password.

Regardless of where the password comes from, Identity Manager uses the Distribution Password as the repository from which it distribute passwords to connected systems. The Distribution Password, like the Universal Password, lets you enforce Password Policies.

For information about using Universal Password and Distribution Password in when synchronizing passwords, see [“Implementing Password Synchronization” on page 138](#).

As with other attributes of a user, you can decide which systems are authoritative sources for passwords, and Identity Manager will distribute the passwords from the authoritative source to the other connected systems.

You can set up bidirectional password synchronization among connected systems that support it.

Identity Manager Can Enforce Password Policies, in the Data Store and on Connected Systems

By making calls to NMAS, Identity Manager lets you enforce Password Policies on incoming passwords. If the password being published from a connected system to Identity Manager does not comply, you can specify that Identity Manager does not accept the password into the identity vault. This also means that passwords that don't comply with your policies are not distributed to other connected systems.

In addition, Identity Manager lets you enforce Password Policies on connected systems. If the password being published to Identity Manager does not comply, you can specify that Identity Manager not only does not accept the password for distribution, but actually resets the noncompliant password on the connected system using the current Distribution Password in the identity vault.

For example, if you want to require passwords to include at least one numeric character, but the connected system does not itself have the ability to enforce such a policy, you can specify that Identity Manager resets passwords from that connected system that don't comply.

If you are using Advanced Password Rules and are using Identity Manager Password Synchronization, to help ensure that passwords are synchronized successfully we recommend that you research the password policies for all the connected systems to make sure the Advanced Password Rules in the eDirectory Password Policy are compatible.

Keep in mind that you must make sure that the users who are assigned Password Policies match up with the users you want to participate in Password Synchronization for connected systems.

Password Policies are assigned with a tree-centric perspective. By contrast, Password Synchronization is set up per driver, and drivers are installed on a per-server basis and can manage only those users who are in a master or read/write replica. To get the results you expect from Password Synchronization, make sure the containers that are in a master or read/write replica on the server running the drivers for Password Synchronization match the containers where you have assigned Password Policies with Universal Password enabled. Assigning a Password Policy to a partition root container ensures that all users in that container and subcontainers are assigned the Password Policy.

For information about how Password Policies are assigned to users, see [“Assigning Password Policies to Users” on page 84](#).

Identity Manager Offers Several Scenarios for Synchronizing Passwords

As with other object attributes, Identity Manager lets you decide which systems should be authoritative sources for passwords. Identity Manager gives you flexibility in deciding how you want passwords to flow.

Much of the new functionality of Identity Manager Password Synchronization relies on Universal Password, the new reversible password functionality provided by eDirectory.

However, there are scenarios that don't require you to deploy Universal Password.

Identity Manager Password Synchronization also relies on the Distribution Password, which is the repository from which Identity Manager distributes passwords to connected systems. Like Universal Password, policy can be enforced on the Distribution Password.

For a basic list of the ways you can implement password synchronization, see [“Implementing Password Synchronization” on page 138](#). These scenarios can be combined to meet the needs of your environment.

Identity Manager Can Sync Passwords on Windows without the Novell Client

A Novell Client is no longer required for password synchronization with Active Directory and NT Domain.

Identity Manager Can Notify Users of Password Synchronization Failures

The previous section, [“Identity Manager Can Enforce Password Policies, in the Data Store and on Connected Systems” on page 116](#), explains that Identity Manager can enforce Password Policies by not accepting passwords that don’t comply from connected systems.

Using the new e-mail notification feature, you can specify that Identity Manager notifies the user when a password change they made was not successful.

For example, suppose you have set Identity Manager to not accept an incoming password from NT Domain if it doesn’t comply with your Password Policy, and you have enabled e-mail notification. One rule in your Password Policy says that the company name can’t be used as a password, and a user changes the password on the NT Domain connected system to be the company name. In this case, NMAS would not accept the password, and Identity Manager would send an e-mail message to the user stating that the password change was not synchronized.

You must set up the e-mail server and templates before you can use this feature. You can customize the text of the messages that Identity Manager sends, and you can customize the notification to send a copy to the administrator. For more information, see [“Configuring E-Mail Notification” on page 170](#).

Identity Manager Can Check the Password Synchronization Status for a User

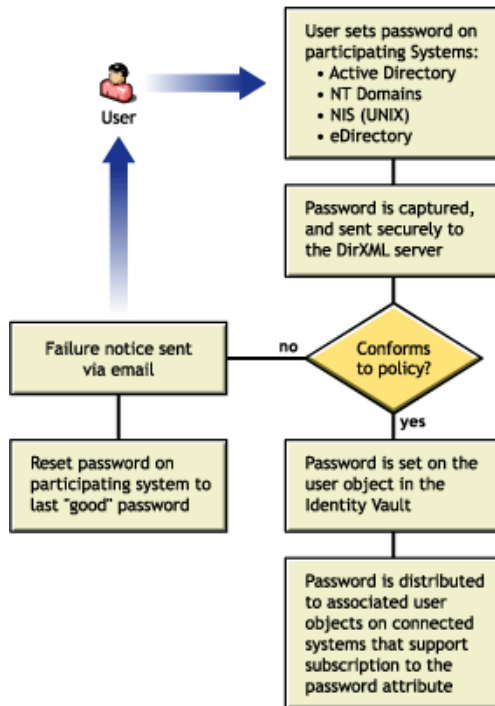
Identity Manager lets you query connected systems to check the password synchronization status for a user. If the connected system supports the check password feature, you can see whether passwords are synchronizing successfully.

For how to check passwords, see [“Checking the Password Synchronization Status for a User” on page 170](#).

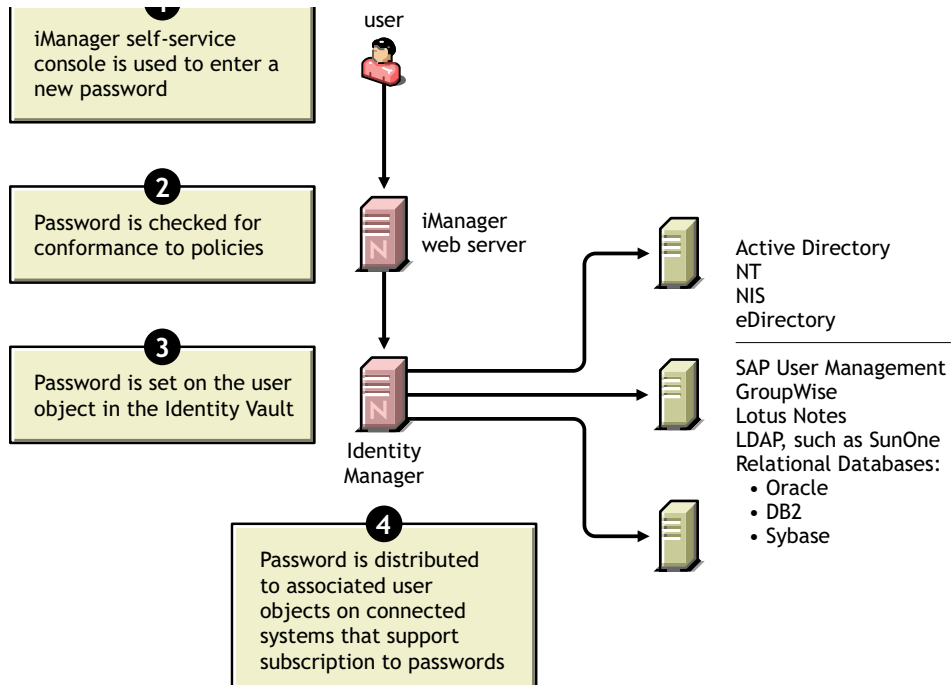
For a list of which systems support checking passwords, see [“Connected System Support for Password Synchronization” on page 119](#).

Diagrams of Password Synchronization Flow

Here's an overview of connected systems publishing passwords to Identity Manager.



Here's an overview of Identity Manager distributing passwords to connected systems.



Connected System Support for Password Synchronization

Identity Manager is always capable of accepting a password from a connected system, even if the connected system does not support providing the user's actual password from that system.

AD, NT, eDir, and NIS can accept a password from Identity Manager and also support sending the user's actual password to Identity Manager. This means they offer full support for bidirectional password synchronization.

Other systems can provide data that can be used to create passwords, by defining policy within the driver configuration on the publisher channel. The sample driver configurations for most of the drivers show an example of this; a policy is included that provides a default password based on Surname.

Connected systems have varying ability to accept a password from Identity Manager. Some connected systems support initial password set for new accounts, but not password modify events.

This section contains a list of the connected systems and what they support.

These capabilities are noted in the driver manifest for each driver, provided as part of the sample driver configurations. This table provides the following additional information:

- ◆ Regarding the application's ability to accept a password, this table indicates whether an application accepts initial password set for a new account, versus whether it can accept a modification to an existing password.

The manifest indicates only that the connected system is capable of accepting a password, and doesn't indicate this distinction.

- ◆ Grouping the drivers together so you can see which of them have similar abilities.

Connected System Driver	Subscriber Channel	Subscriber Channel	Subscriber Channel	Publisher Channel
	Application Can Accept Setting of Initial Password	Application Can Accept Modification of Password	Application Supports Check Password	Application Can Provide (sync) Password

The following connected systems support bidirectional password synchronization.

They can provide the user's actual password on the connected system, and accept passwords from Identity Manager.

Active Directory	Yes	Yes	Yes	Yes
eDirectory	Yes	Yes	Yes	Yes
NT Domain	Yes	Yes	No	Yes
NIS	Yes	Yes	Yes	Yes

The following connected systems can accept passwords from Identity Manager to some degree. They can't provide a user's password on the connected system to Identity Manager.

Although they can't provide the user's password to Identity Manager, they could be configured to create a password using a policy on the Publisher channel, based on other user data in the connected system. (The sample driver configurations demonstrate default password based on surname.)

Groupwise 2.05	Yes	Yes	No	No ¹
JDBC	Yes ²	No	No	No
LDAP	Yes ³	Yes ³	Yes ³	No

Connected System Driver	Subscriber Channel	Subscriber Channel	Subscriber Channel	Publisher Channel
	Application Can Accept Setting of Initial Password	Application Can Accept Modification of Password	Application Supports Check Password	Application Can Provide (sync) Password
Notes 2.0	Yes	Yes	Yes	No

The following connected systems can't accept passwords or provide a user's password on the connected system.

Although they can't provide the user's password to Identity Manager, they could be configured to create a password using a policy on the Publisher channel, based on other user data in the connected system. (The sample driver configurations demonstrate default password based on surname.)

Delimited Text	No	No	No	No
Exchange 5.5	No	No	No	No
PeopleSoft 3.6	No	No	No	No
PeopleSoft 4.0	No	No	No	No
SAP HR	No	No	No	No
SIF	No	No	No	No

The following connected systems are not intended to be used with password synchronization.

Avaya PBX	No	No	No	No
Entitlements Service Driver	No	No	No	No
LoopBack Service Driver	No	No	No	No
Manual Task Service Driver	No	No	No	No

¹GroupWise supports two authentication methods. 1) GroupWise provides its own authentication and maintains user passwords. 2) GroupWise authenticates against eDirectory using LDAP and does not maintain passwords. When using option 2, driver-synchronized passwords are ignored by GroupWise.

²Initial password set is available on all databases where the OS user account is distinct from the database user account, such as Oracle, MS SQL, MySQL, and Sybase.

³If the target LDAP server allows setting the userpassword attribute.

NOTE: Between eDirectory trees, you can have bidirectional password synchronization for users even if Universal Password is not enabled for those users. See [“Scenario 1: eDirectory to eDirectory Password Synchronization Using NDS Password” on page 140](#).

Pieces Required for Password Synchronization to Work

Password Synchronization depends on the following elements being in place:

- ◆ [“Password Synchronization Capabilities Declared in the Driver Manifest” on page 121](#)
- ◆ [“Password Synchronization Settings You Create Using Global Configuration Values” on page 121](#)
- ◆ [“Policies Required in the Driver Configuration” on page 125](#)

- ♦ [“Filters You Install on the Connected System to Capture Passwords” on page 126](#)
- ♦ [“Password Policies You Create for Your Users” on page 127](#)
- ♦ [“NMA Login Methods” on page 127](#)

Password Synchronization Capabilities Declared in the Driver Manifest

The driver manifest declares whether a connected system supports the following password synchronization functions:

- ♦ Publishing the user’s actual password to Identity Manager
- ♦ Accepting a password from Identity Manager (the manifest does not distinguish between accepting the creation of an initial password versus accepting password modifications)
- ♦ Letting Identity Manager check the password on the connected system, to determine the password synchronization status for a user

NOTE: The driver manifest is written by the driver developer, or the Identity Manager expert who creates the driver configuration. It is not meant to be edited by a network administrator. It represents the true capabilities of the driver shim and configuration, so changing the manifest alone does not change functionality. To add functionality, the driver shim, connected system, or driver configuration would have to be enhanced.

The driver configurations delivered with Identity Manager contain driver manifest entries. To add them to an existing driver, see [“Upgrading Existing Driver Configurations to Support Identity Manager Password Synchronization” on page 132](#).

Password Synchronization Settings You Create Using Global Configuration Values

New in Identity Manager are global configuration values, which let you set a constant value which you can reference in a policy. (They are sometimes called server variables, because they are held in an attribute that is per replica.)

For Password Synchronization, they allow you to create settings for the flow of passwords to and from Identity Manager.

Because the password synchronization policies in the driver configuration are written to behave differently based on your settings in the global configuration value, it’s easy to change the flow of passwords without having to edit policies.

You control the following settings for each connected system separately, using global configuration values. Note that in the interface, Identity Manager is referred to as DirXML.

- ♦ Whether Identity Manager accepts passwords from the connected system.
This setting applies to a password provided by the connected system, as well as a password that could be created by policies in the driver configuration on the publisher channel. If you disable this setting, both kinds of passwords are stripped out so they don’t reach Identity Manager.
- ♦ Which method of synchronization Identity Manager uses, updating Universal Password directly, or Distribution Password directly. Identity Manager controls the entry point, meaning which password Identity Manager updates. NMA controls the flow of passwords between each different kind of password, based on what you have set in the Password Policy in Universal Password > Configuration Options.

See [“Implementing Password Synchronization” on page 138](#) for examples of scenarios using these methods.

- ◆ Whether Password Policies are enforced on passwords coming in to Identity Manager from a connected system.

If they are enforced, this means that passwords coming in are not written to the Identity Manager data store if they don't comply.

- ◆ Whether Identity Manager enforces Password Policies on a connected system by resetting passwords that don't comply, using the Identity Manager password.

This option is dimmed in the interface if the connected system doesn't support it (as declared in the driver manifest).

- ◆ Whether the connected system accepts passwords.

This setting applies to both a password distributed by Identity Manager and a password that could be created by policies in the driver configuration on the subscriber channel. If you disable this setting, both kinds of passwords are stripped out so they don't reach the connected system.

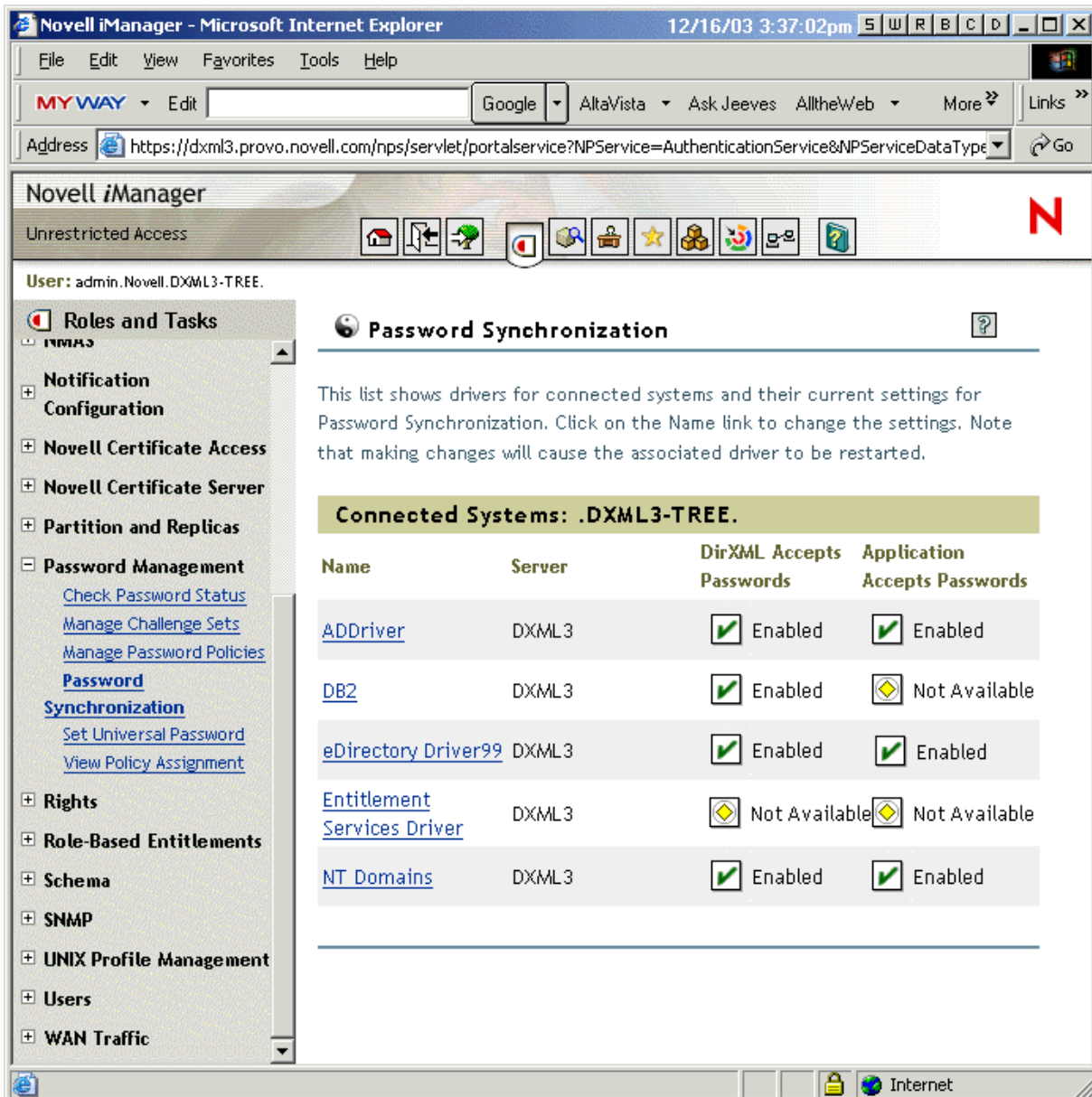
This option is dimmed in the interface if the connected system doesn't support it (as declared in the driver manifest).

- ◆ Whether users are notified by e-mail when a password is not synchronized

The driver configurations delivered with Identity Manager contain driver manifest entries. To add them to an existing driver, see [“Upgrading Existing Driver Configurations to Support Identity Manager Password Synchronization”](#) on page 132.

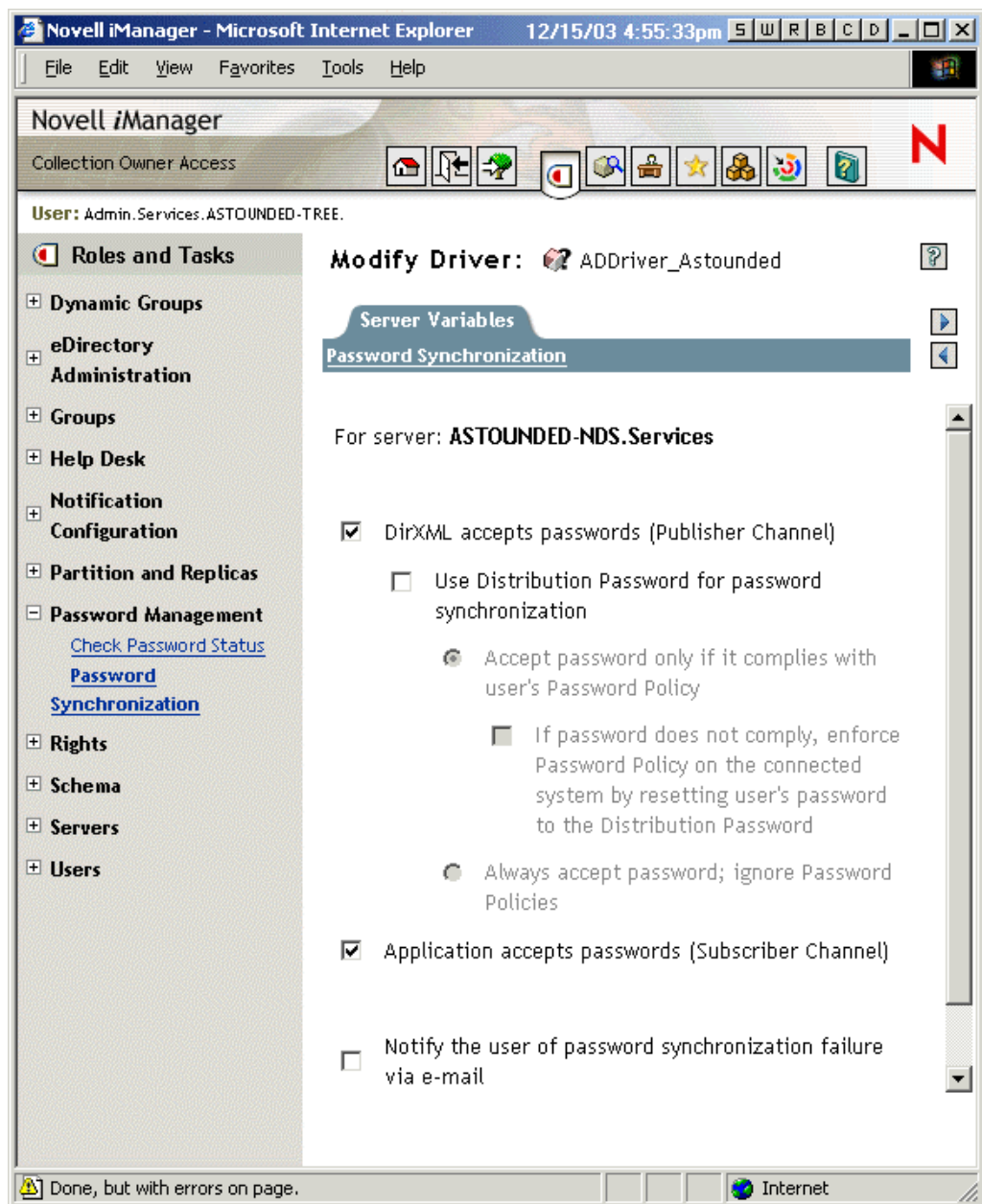
The Password Synchronization task in iManager (Password Management > Password Synchronization) is where you should edit these GCVs. This graphical interface lets you specify how you want passwords to flow among connected systems and Identity Manager.

After you specify where you want to search for connected system drivers, the interface displays an overview of the password flow settings for all the connected system drivers it finds. Here's an example of the overview screen:



On this page, you can click a driver name to drill down and see all the settings you control.

The following figure shows the page that appears. This is the graphical interface for setting the global configuration values for Password Synchronization.



If an option on this page is dimmed, it is because the driver manifest shows that the connected system does not support it.

NOTE: This interface lets you set global configuration values on each driver separately. Global configuration values on a driver override those on the driver set, and setting them on a specific driver gives you more granular control. This page can display only the global configuration values that are present on the individual driver.

Global configuration values can be set on the driver set object, and can be inherited by a driver in that driver set if the driver does not have values of its own. If a driver has no settings of its own and instead inherits the global configuration values from the driver set, this interface does not display them. Although this interface does not display inherited global configuration values, they are still honored by the password synchronization policies.

Policies Required in the Driver Configuration

Policies on the Publisher and Subscriber Channels for each driver govern the password flow, based on your settings in the global configuration variables explained above.

These policies are included in the driver configurations in Identity Manager.

If you are upgrading an existing driver configuration instead of replacing it, you must add these policies to the configuration. (See [“Upgrading Existing Driver Configurations to Support Identity Manager Password Synchronization”](#) on page 132.)

These policies must be in your driver configuration in the correct location for password synchronization to work.

Location in the Driver Configuration	Password Synchronization Policy Name	What the Policy Does
Publisher Command Transformation These policies must be present in this order, and it's required that they be the last policies in the Command Transformation policy set.	Password(Pub)-Default Password Policy	Adds a default password to an add object if the add does not already contain a password. This policy and the Password(Sub)-Default Password Policy are the only policies that you can modify or remove. The others should be used without changes, in order for password synchronization functionality to work properly.
	Password(Pub)-Check Password GCV	Checks the GCV to determine whether you have specified that Identity Manager accepts passwords from this connected system. If not, it strips out all password elements. The name of the GCV is enable-password-publish, and the display name is "DirXML accepts passwords from application."
	Password(Pub)-Publish Distribution Password	Transforms the <password> element to the form that allows it to update Universal Password. This policy references the following GCVs: publish-password-to-dp, and enforce-password-policy.
	Password(Pub)-Publish NDS Password	Allows the <password> element to go through if you have specified that NDS password should be updated. If not, it strips out the <password> element. This policy references the GCV named publish-password-to-nds.
	Password(Pub)-Add Password Payload	Puts in payload data that is passed around in the engine for purposes of e-mail notification.

Location in the Driver Configuration	Password Synchronization Policy Name	What the Policy Does
Publisher Input Transformation Novell recommends that this policy be listed last if there are multiple policies in the Input Transformation.	Password(Pub)-Sub Email Notifications	<p>If the payload information comes through, and the status shows a problem, it sends e-mail to the user. It sends the mail to the user's e-mail address indicated in the Internet EMail Address attribute in eDirectory.</p> <p>This policy references the GCV named notify-user-on-password-dist-failure to determine whether to send notification e-mails.</p>
Subscriber Command Transformation These policies must be present in this order, and it's required that they be the last policies in the Command Transformation.	Password(Sub)-Transform Distribution Password Password(Sub)-Default Password Policy	<p>Transforms the Universal Password to a <password> element.</p> <p>Adds a default password to an add object if the add does not already contain a password.</p> <p>This policy and the Password(Pub)-Default Password Policy are the only policies that you can modify or remove. The others should be used without changes, in order for password synchronization functionality to work properly.</p>
	Password(Sub)-Check Password GCV	<p>Checks the GCV to determine whether you have specified that the connected system accepts passwords. If not, it strips out all password elements.</p> <p>The name of the GCV is enable-password-subscribe, and the display name is "Application accepts passwords from DirXML data store."</p>
	Password(Sub)-Add Password Payload	<p>Puts in payload data that is passed around in the engine for purposes of e-mail notification.</p>
Subscriber Output Transformation Novell recommends that this policy be listed last if there are multiple policies in the Output Transformation.	Password(Sub)-Pub Email Notifications	<p>If the payload information comes through, and the status shows a problem, it sends e-mail to the user.</p> <p>This policy references the GCV named notify-user-on-password-dist-failure to determine whether to send notification e-mails.</p>

Filters You Install on the Connected System to Capture Passwords

For AD, NT Domain, and NIS, filters must be installed to capture the user's password.

See ["Setting Up Password Filters" on page 165](#).

Password Policies You Create for Your Users

Password Policies must be used to enable Universal Password for your users (although you can use some features of Password Synchronization without Universal Password). The Password Policy also lets you specify Advanced Password Rules, and specify whether user's existing passwords are checked for compliance with the rules.

You must understand Password Policies to use Identity Manager Password Synchronization.

Password Policies are explained in [Chapter 7, “Managing Passwords Using Password Policies,” on page 73](#).

NMAS Login Methods

For some situations, you must have the NMAS Simple Password Login Method in place to be able to do password functions. For example, LDAP requires it.

Preparing to Use Identity Manager Password Synchronization and Universal Password

In this section:

- ♦ [“Switching Users from NDS Password to Universal Password” on page 127](#)
- ♦ [“Changing Passwords Using iManager Self-Service Console” on page 128](#)
- ♦ [“Preparing to Use Universal Password” on page 128](#)
- ♦ [“Replica Planning and Password Policies” on page 130](#)
- ♦ [“Setting Up E-Mail Notification” on page 130](#)

Switching Users from NDS Password to Universal Password

When you turn on Universal Password for a group of users by using a Password Policy, the user needs the Universal Password to be populated.

If you have previously been using Password Synchronization to update the NDS password, you need to plan for the transition of user's passwords. To switch to using Universal Password, you can do one of the following things to have your users create a Universal Password:

- ♦ If you use the Novell Client (it is not required for Identity Manager Password Synchronization), Roll out the new Novell Client that supports Universal Password. The next time users log in using the new Novell Client, the client will capture the NDS password before it is hashed, and will use it to populate the Universal Password.
- ♦ If you are not using Novell Client, have users log in to the iManager self-service console. That login method will populate Universal Password. To access the iManager self-service console, go to /nps on your iManager server. For example, <https://www.myiManager.com/nps>.
- ♦ Have users log in using through any service that is authenticating using a Universal Password enabled LDAP server. For example, a company portal.

Changing Passwords Using iManager Self-Service Console

The iManager self-service console displays the Advanced Password Rules you choose in the Password Policy. This allows the user to create a compliant password without having to guess at the rules.

Depending on how your password flow is set up, a user could change a password on a connected system and it would be synchronized to Identity Manager and other connected systems. However, the Novell Client and the connected systems don't display the rules when the user changes a password.

If you want to enforce Advanced Password Rules and avoid noncompliant passwords, it's best to require users to change the password only in the iManager self-service console. If a user changes the password by another means, they might experience these issues:

- ◆ In the Novell Client, the rules are not displayed but the password is rejected if it does not comply with the rules. The user must guess the rules correctly to create a password that complies.
- ◆ On a connected system, the user is allowed to change the password without viewing the Password Policy rules, and might not remember the rules correctly. Only the policies of the connected system itself will be enforced when users first make the change. But frustration might occur for the user depending on your Identity Manager settings:
 - ◆ If you have enabled the setting that enforces the policy on passwords coming in to Identity Manager from connected systems, the user's password won't be synchronized. If you have set Identity Manager to notify users of failure, they will find out by e-mail that their password didn't go through.
 - ◆ If you have set Identity Manager to replace noncompliant passwords on connected systems, the user will not be able to log in with the new password they chose on the connected system.

If you allow users to change their passwords on a connected system or by using the latest version of the Novell Client, help users be successful in creating a compliant password by making sure your Password Policy rules are well publicized for your users.

Preparing to Use Universal Password

Most of the information you need is in the *Universal Password Deployment Guide* (http://www.novell.com/documentation/lg/nw65/universal_password/data/front.html).

In addition, keep in mind the following:

- ◆ eDirectory 8.7.1 or later is the requirement for using Universal Password. NetWare 6.5 is not a requirement, and the NetWare documentation has been updated to reflect this.
- ◆ Identity Manager Password Synchronization relies on both Universal Password and another new kind of password, the Distribution Password, which is the repository from which Identity Manager distributes passwords to connected systems. Like Universal Password, policy can be enforced on the Distribution Password.
- ◆ The DirXML iManager plug-ins, which ship with Identity Manager, include the new Password Management plugins that let you create Password Policies. These plugins let you determine how you want Universal Password to be synchronized with NDS Password, Simple Password, and Distribution Password.

These plugins replace the ones for Universal Password that were shipped with NetWare 6.5. They are described in **Chapter 7, “Managing Passwords Using Password Policies,”** on **page 73**.

- ♦ eDirectory 8.6.2 can’t be used for the tree that Identity Manager is using. However, eDirectory 8.6.2 is supported for a subset of password synchronization features, so it can be used for other trees if you are not yet ready to upgrade your whole environment.
- ♦ One way to reduce the impact when you are upgrading software for deploying Universal Password is to create a separate tree for Identity Manager as an identity vault. Many environments already use an identity vault for DirXML and the drivers.
- ♦ Universal Password gives you new capabilities which were not supported with previous password management tools, such as enforcement of Password Policies and the ability to use special characters.
- ♦ It’s very important to update your Novell Clients and other utilities, to avoid password “drift,” a situation in which the NDS Password gets out of sync with the Universal Password.
- ♦ The latest version of the Novell Client supports Universal Password, can populate it for a user when you first enable Universal Password for that user, and can enforce password policies when users are changing passwords.
- ♦ A connected system does not display the Advanced Password Rules that you create in a Password Policy. At this time, neither does the Novell Client, although it enforces them.

Instead, it’s best to require users to change the password only in the iManager self-service console.

If you allow users to change their passwords on a connected system or by using the latest version of the Novell Client, help users be successful in creating a compliant password by making sure your Password Policy rules are well publicized for your users.

The following general steps help you prepare to use Universal Password.

1 Upgrade your server environment to support Universal Password.

eDirectory 8.7.3 or later is required for full functionality, and must be used for the tree that Identity Manager is using.

- ♦ eDirectory 8.7.3 or later
- ♦ NMAS 2.3 or later
- ♦ NCI 2.6 or later
- ♦ LDAP server 8.7.3 or later
- ♦ iManager 2.0.2 or later

2 Upgrade your client environment to support Universal Password.

- ♦ Novell Client (If you are using it, you must upgrade it. However, using the Novell Client is not required.)
- ♦ NT Client
- ♦ Web browser. If you use no client at all, and users log in through a Web browser, you need to make sure they have a browser that supports iManager 2.0.2 or later.

3 Complete the additional steps notes in the *Universal Password Deployment Guide* (http://www.novell.com/documentation/lg/nw65/universal_password/data/front.html), such as running SDIDiag.exe.

- 4 For administrators and help desk users, make them aware that ConsoleOne® supports Universal Password only if it is used on a NetWare® 6.5 server or later, or is used on a machine that has the latest Novell Client.
- 5 Make sure administrators and help desk users understand the implications of using utilities which support only NDS Password. They can be used to log in, but they should not be used to change passwords. This measure avoid password “drift,” a situation in which the NDS Password gets out of sync with the Universal Password.

The *Universal Password Deployment Guide* (http://www.novell.com/documentation/lg/nw65/universal_password/data/front.html) references a TID which lists utilities and their support for Universal Password.

- 6 Install Identity Manager, as explained in **Chapter 4, “Installation,”** on page 39.

On the Identity Manager DirXML server:

- ♦ DirXML engine
- ♦ DirXML driver shims

IMPORTANT: If you are using NT or AD with Password Synchronization 1.0, see the upgrade information in the *driver implementation guides* (<http://www.novell.com/documentation/lg/dirxml/drivers/index.html>) before you install the new driver shim.

- ♦ Activation credentials

On the iManager Web server:

- ♦ New iManager plug-ins, which include the Password Synchronization plug-ins
- ♦ Driver configuration files, which include the password synchronization enhancements. These are explained in **“Pieces Required for Password Synchronization to Work”** on page 120.
 - ♦ Driver manifest declaring what password functionality is supported for a connected system.
 - ♦ Global configuration values (GCVs) that you use to specify whether you want passwords to flow to and from the connected system, whether Password Policies should be enforced, and whether e-mail notifications should be sent. These settings can be edited in iManager.
 - ♦ New policies for the driver configurations that carry out the settings that you specify for the global configuration values.

Replica Planning and Password Policies

Password Policies are assigned with a tree-centric perspective. By contrast, Password Synchronization is set up per driver, and drivers are installed on a per-server basis and can manage only those users who are in a master or read/write replica. To get the results you expect from Password Synchronization, make sure the containers that are in a master or read/write replica on the server running the drivers for Password Synchronization match the containers where you have assigned Password Policies with Universal Password enabled. Assigning a Password Policy to a partition root container ensures that all users in that container and subcontainers are assigned the Password Policy.

Setting Up E-Mail Notification

To use the e-mail notification feature, you must do the following:

- ♦ Use the Notification Configuration task in iManager to set up the e-mail server
- ♦ Use the Notification Configuration task in iManager to customize the e-mail templates if desired.
- ♦ Make sure that eDirectory users have the Internet EMail Address attribute populated.

Follow the instructions in “[Configuring E-Mail Notification](#)” on page 170.

New Driver Configuration and Identity Manager Password Synchronization

If you have not used Password Synchronization 1.0 in your environment, and you are creating a new driver or replacing an existing configuration with a new Identity Manager configuration, use the following instructions to set up the new Identity Manager Password Synchronization functionality.

- 1** Make sure your environment is ready to use Universal Password. See “[Preparing to Use Identity Manager Password Synchronization and Universal Password](#)” on page 127.
- 2** Create a new driver, or replace an existing driver’s configuration with the Identity Manager 2 configuration.

The Identity Manager configurations contain the policies and other items necessary for Identity Manager Password Synchronization. See the individual [DirXML Driver Guides](#) (<http://www.novell.com/documentation/beta/dirxmldrivers>) for information on importing the new sample driver configurations.

- 3** Turn on Universal Password for users by creating Password Policies with Universal Password enabled.

See “[Creating Password Policies](#)” on page 84. If you previously used Universal Password with NetWare 6.5, note that there are some extra steps described in “[\(NetWare 6.5 only\) Re-Creating Universal Password Assignments](#)” on page 82.

We recommend that you assign Password Policies as high up in the tree as possible.

In the Password Policy, Universal Password > Configuration Options, there are options for how you want NMAST to keep the different kinds of passwords synchronized.

For examples of scenarios for using Password Synchronization, and how Password Policies fit in, see “[Implementing Password Synchronization](#)” on page 138. See also the online help.

- 4** (Active Directory, NIS, or NT Domain only) Install new Password Synchronization filters and configure them if you want the connected systems to provide user passwords to Identity Manager:

For instructions, see the driver implementation guide for each of these drivers, at [DirXML Drivers](#) (<http://www.novell.com/documentation/lg/dirxmldrivers/index.html>).

- 5** Make sure your password flow is set the way you want it for each connected system.
 - 5a** In iManager, click Password Management > Password Synchronization, and search for the drivers for connected systems you want to manage.
 - 5b** View the current settings for password flow. This is a graphical interface for the global configuration values (GCVs). Edit them by clicking on the name of a driver.

You can edit settings for

- ♦ Whether Identity Manager accepts passwords from this system

- ◆ Which password you want Identity Manager to update: Universal Password directly, or Distribution Password directly. Identity Manager controls the entry point, meaning which password Identity Manager updates. NMAS controls the flow of passwords between each different kind of password, based on what you have set in the Password Policy in Universal Password > Configuration Options.
- ◆ Whether the Password Policy for the user is enforced on password changes coming in to Identity Manager
- ◆ Whether the Password Policy for the user is enforced on the connected system by resetting passwords that don't comply
- ◆ Whether passwords are accepted by this connected system
- ◆ Whether e-mail notifications are sent when password synchronization fails

For more information and screen captures for these options, see “[Implementing Password Synchronization](#)” on page 138. See also the online help.

6 Test password synchronization:

- ◆ Confirm that the Identity Manager password is distributed to the systems you specified
- ◆ Confirm that the connected systems you specified are publishing passwords to Identity Manager.

For troubleshooting tips, see “[Implementing Password Synchronization](#)” on page 138.

Upgrading Password Synchronization 1.0 to Identity Manager Password Synchronization

This task applies only to existing DirXML Drivers for Active Directory and NT Domain that are being used with Password Synchronization 1.0.

It's very important that you follow the correct procedure when upgrading from Password Synchronization 1.0.

For instructions, see the driver implementation guides for the DirXML Drivers for Active Directory and NT Domain, at [DirXML Drivers \(http://www.novell.com/documentation/lg/dirxml/drivers/index.html\)](http://www.novell.com/documentation/lg/dirxml/drivers/index.html).

Upgrading Existing Driver Configurations to Support Identity Manager Password Synchronization

This section explains the process for adding support for Identity Manager Password Synchronization to existing driver configurations, instead of replacing your existing driver configurations with the Identity Manager sample configurations.

IMPORTANT: If you are upgrading a DirXML Driver for AD or NT Domain, and it is being used with Password Synchronization 1.0, you should follow the upgrade instructions in the driver implementation guides for the DirXML Drivers for Active Directory and NT Domain, at [DirXML Drivers \(http://www.novell.com/documentation/lg/dirxml/drivers/index.html\)](http://www.novell.com/documentation/lg/dirxml/drivers/index.html).

The following is an overview of the tasks you must complete, using the procedure in this section:

- ◆ Convert the driver to Identity Manager 2 format.

- ♦ Add driver manifest, global configuration values, and password synchronization policies to the driver configuration. For a list of the policies you add, see “[Policies Required in the Driver Configuration](#)” on page 125.
- ♦ Change the Filter settings for the nspmDistributionPassword attribute.
- ♦ Set up password synchronization flow.

Prerequisites

- ☐ Create a backup of your existing driver using the Export Drivers Wizard.
- ☐ Make sure you have installed the new driver shim. Some password synchronization features such as Check Password Status won’t work without the new Identity Manager driver shim.

IMPORTANT: If you are upgrading a DirXML Driver for AD or NT Domain, and it is being used with Password Synchronization 1.0, don’t install the driver shim until you have reviewed the upgrade instructions. Follow the upgrade instructions in the driver implementation guides for the DirXML Drivers for Active Directory and NT Domain, at [DirXML Drivers \(http://www.novell.com/documentation/ig/dirxml/drivers/index.html\)](http://www.novell.com/documentation/ig/dirxml/drivers/index.html).

Procedure

- 1** Make sure your environment is ready to use Universal Password. See “[Preparing to Use Identity Manager Password Synchronization and Universal Password](#)” on page 127.
- 2** Convert the driver to Identity Manager format using a wizard. See “[Upgrading a Driver Configuration from DirXML 1.x to Identity Manager Format](#)” on page 63.
- 3** In iManager, click DirXML Utilities > Import Drivers.

You add support for Identity Manager Password Synchronization to each driver you want to participate in password synchronization, by importing an “overlay” configuration file to add the policies, driver manifest, and the GCVs, all at once.

After adding them, you must also add the nspmDistributionPassword attribute to the Filter.

These tasks are described in subsequent steps.

- 4** Select the driver set where your existing driver resides.
- 5** In the list of driver configurations that appears, click only the item labeled “Password Synchronization 2.0 Policies.” It is listed under Additional Policies. Click Next.
A list of import prompts appears.
- 6** Select your existing driver to update.
- 7** Answer three prompts about the capabilities of the driver configuration and the connected system.
 - ♦ Whether the connected system can provide passwords to Identity Manager. This refers to the users’s actual password on the connected system, not to a password that can be created using a style sheet. Only AD, eDirectory, and NIS can do this.
 - ♦ Whether the connected system can accept passwords from Identity Manager
 - ♦ Whether the connected system can check a password to see if it matches the password in Identity Manager.

The Import Driver Wizard makes entries into the driver manifest based on your answers. This is required for Password Synchronization policies to work.

The driver manifest indicates the combined ability of the connected system, the Identity Manager DirXML driver shim, and the driver configuration policies, and usually should not be edited by the network administrator.

The following table shows how to answer for each driver when you are importing the policies that support Identity Manager Password Synchronization.

Driver	Connected System Provides Password	Connected System Accepts Password	Connected System Checks Password
Active Directory	Yes	Yes	Yes
eDirectory	Yes	Yes	Yes
Exchange (applicable only if you use NT with it)	No (This system can't provide the user's actual password; however, a policy is provided that creates a default password.)	No	No
GroupWise	No (This system can't provide the user's actual password; however, a policy is provided that creates a default password.)	No	No
JDBC	No (This system can't provide the user's actual password; however, a policy is provided that creates a default password.)	No	No
LDAP	No	Yes	Yes
NIS	Yes	Yes	Yes
Lotus Notes	No	Yes	Yes
NT	Yes	Yes	No
PeopleSoft	No (This system can't provide the user's actual password; however, a policy is provided that creates a default password.)	No	No
SAP	No (This system can't provide the user's actual password; however, a policy is provided that creates a default password.)	No	No

Driver	Connected System Provides Password	Connected System Accepts Password	Connected System Checks Password
SAP User	No (This system can't provide the user's actual password; however, a policy is provided that creates a default password.)	No	No
SIF	No (This system can't provide the user's actual password; however, a policy is provided that creates a default password.)	No	No

For the following drivers, the Password Synchronization policies are not applicable.

Delimited Text

Entitlement Service

Loopback Service or Move
Proxy Service

Manual Task Service

8 Click Next. Choose to update everything about the driver.

This option gives you the driver manifest, global configuration values (GCVs), and policies necessary for password synchronization.

The driver manifest and GCVs overwrite any values that already exist, but these kinds of driver parameters are new in Identity Manager 2, so for a DirXML 1.x driver there should be no existing values to be overwritten.

The password synchronization policies don't overwrite any existing policy objects. They are simply added to the Driver object.

NOTE: If you do have driver manifest or GCV values that you want to save, choose the option named Update Only Selected Policies in That Driver, and check the checkboxes for all the policies. This option imports the password policies but does not change the driver manifest or GCVs. You will need to manually paste in any additional values.

9 Click Next, then click Finish to complete the wizard.



At this point, the new policies have been created as policy objects under the driver object, but are not yet part of the driver configuration. To link them in, you must manually insert each of them at the right point in the driver configuration on the Subscriber and Publisher channels.

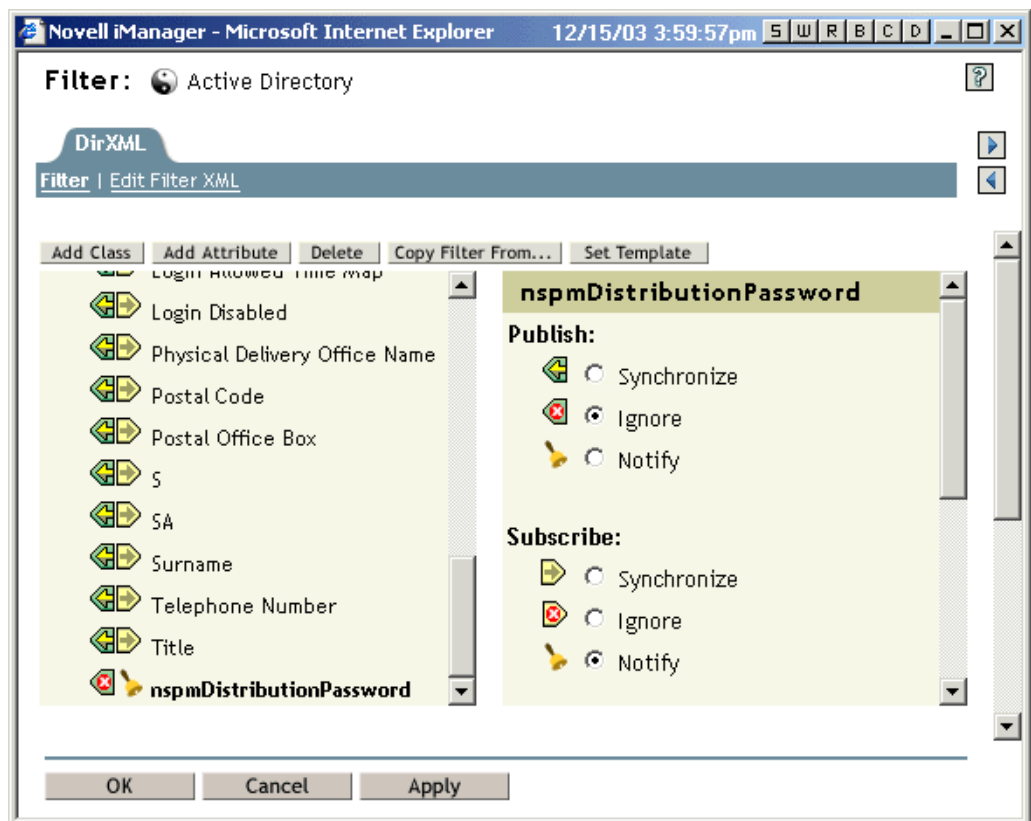
10 Insert each of the new policies into the correct place on your existing driver configuration. If there are multiple policies in a policy set, make sure these password synchronization policies are listed last.

The list of the policies and where to insert them is in [“Policies Required in the Driver Configuration” on page 125](#).

Repeat the following steps for each policy.

10a Click DirXML Management > Overview. Select the driver set for the driver you are updating.

- 10b** Click the driver you just updated. A page opens showing a graphical representation of the driver configuration.
- 10c** Click the icon for the place where you need to add one of the new policies.
- 10d** Click Insert to add the new policy. In the Insert page that appears, click Use an Existing Policy and browse for the new policy object. Click OK.
- 10e** If you have more than one policy in the list for any of the new policies, use the arrow buttons   to move the new policies to the correct location in the list. Make sure the policies are in the order listed in **“Policies Required in the Driver Configuration”** on [page 125](#).
- 11** For the object classes that you want to synchronize passwords for (such as User), make sure that nspmDistributionPassword attribute is in the filter and has the following settings:
 - ◆ For the Publisher channel, set the Filter to Ignore for the nspmDistributionPassword attribute.
 - ◆ For the Subscriber channel, set the Filter to Notify for the nspmDistribution Password attribute.



- 12** Ignore both the Public Key and Private Key attributes in the driver Filter for all objects that have Notify set for the nspmDistributionPassword attribute.



- 13** Repeat [Step 2](#) through [Step 12](#) for all the drivers that you want to upgrade to participate in password synchronization.

At this point, the driver has the new driver shim, Identity Manager format, and the other pieces that are necessary in the driver configuration to support password synchronization: driver manifest, GCVs, password synchronization policies, and filter settings.

- 14** For any additional steps or information for setting up Identity Manager Password Synchronization for each kind of driver, see the driver implementation guides at [DirXML Drivers](http://www.novell.com/documentation/lg/dirxml/drivers/index.html) (<http://www.novell.com/documentation/lg/dirxml/drivers/index.html>).
- 15** Turn on Universal Password for users by creating Password Policies with Universal Password enabled.

See [“Creating Password Policies” on page 84](#). If you previously used Universal Password with NetWare 6.5, note that there are some extra steps described in [“\(NetWare 6.5 only\) Re-Creating Universal Password Assignments” on page 82](#).

We recommend that you assign Password Policies as high up in the tree as possible.

In the Password Policy, Universal Password > Configuration Options, there are options for how you want NMAS to keep the different kinds of passwords synchronized. The default settings should work for most implementations. See the online help for that page for more information.

For examples of scenarios for using Password Synchronization, and how Password Policies fit in, see [“Implementing Password Synchronization” on page 138](#).

Password Policies are assigned with a tree-centric perspective. By contrast, Password Synchronization is set up per driver, and drivers are installed on a per-server basis and can manage only those users who are in a master or read/write replica. To get the results you

expect from Password Synchronization, make sure the containers that are in a master or read/write replica on the server running the drivers for Password Synchronization match the containers where you have assigned Password Policies with Universal Password enabled. Assigning a Password Policy to a partition root container ensures that all users in that container and subcontainers are assigned the Password Policy.

16 Make sure your password flow is set the way you want it for each connected system.

16a In iManager, click Password Management > Password Synchronization, and search for the drivers for connected systems you want to manage.

16b View the current settings for password flow. This is a graphical interface for the global configuration values (GCVs). Edit them by clicking the name of a driver.

You can edit settings for

- ♦ Whether Identity Manager accepts passwords from this system
- ♦ Which password you want Identity Manager to update: Universal Password directly, or Distribution Password directly. Identity Manager controls the entry point, meaning which password Identity Manager updates. NMAS controls the flow of passwords between each different kind of password, based on what you have set in the Password Policy in Universal Password > Configuration Options.
- ♦ Whether the Password Policy for the user is enforced on password changes coming in to Identity Manager
- ♦ Whether the Password Policy for the user is enforced on the connected system by resetting passwords that don't comply
- ♦ Whether passwords are accepted by this connected system
- ♦ Whether e-mail notifications are sent when password synchronization fails

For more information and screen captures for these options, [“Implementing Password Synchronization” on page 138](#). See also the online help.

17 Test password synchronization:

- ♦ Confirm that the Identity Manager password is distributed to the systems you specified
- ♦ Confirm that the connected systems you specified are publishing passwords to Identity Manager.

For troubleshooting tips, see [“Implementing Password Synchronization” on page 138](#).

Implementing Password Synchronization

The Password Synchronization functionality provided in Identity Manager allows you to implement several different scenarios. This section outlines some basic scenarios, to help you understand how the settings in Password Synchronization and Password Policies affect the way passwords are synchronized. You can use one or more of the scenarios to meet the needs of your environment.

In this section:

- ♦ [“Overview of Identity Manager Relationship to NMAS” on page 139](#)
- ♦ [“Scenario 1: eDirectory to eDirectory Password Synchronization Using NDS Password” on page 140](#)
- ♦ [“Scenario 2: Synchronizing Universal Password” on page 142](#)

- ♦ “Scenario 3: Synchronizing eDirectory and Connected Systems with Identity Manager Updating the Distribution Password” on page 152
- ♦ “Scenario 4: Tunneling — Synchronizing Connected Systems but not eDirectory, with Identity Manager Updating the Distribution Password” on page 160

Overview of Identity Manager Relationship to NMAS

In this section:

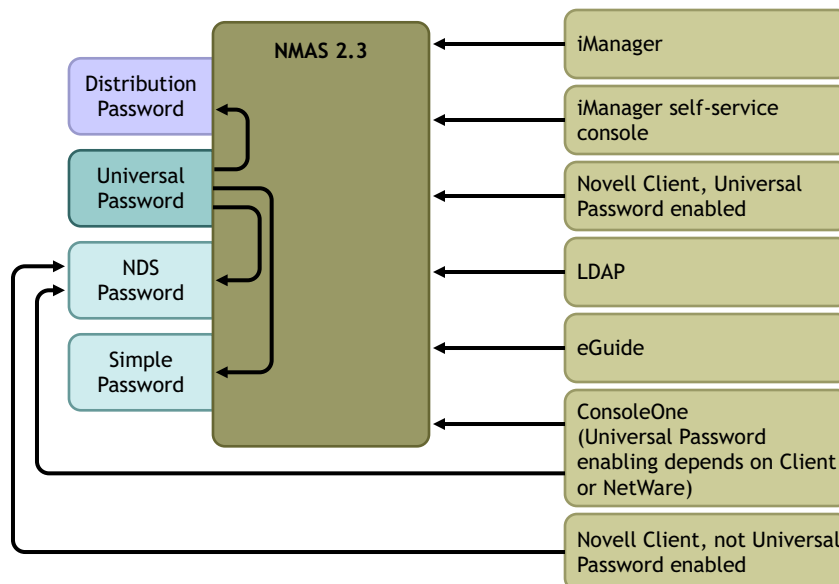
- ♦ “Utilities and NMAS” on page 139
- ♦ “Identity Manager and NMAS” on page 139

Utilities and NMAS

The latest utilities such as iManager and the Novell Client communicate with NMAS rather than directly updating a specific password, and NMAS is the entity that determines which passwords are updated.

NMAS synchronizes passwords within eDirectory, based on your settings in Password Policies.

Legacy utilities that are not Universal Password enabled update the NDS password directly, instead of communicating with NMAS and letting NMAS determine which passwords are updated. Be aware of how legacy utilities are used in your environment by users and help desk administrators; they can cause password “drift” if you are using Universal Password and NMAS 2.3, because legacy utilities update the NDS password directly. For example, you should make sure users upgrade the Novell Client, and make sure that help desk users use ConsoleOne only with the latest Novell Client or NetWare release to ensure support of Universal Password.



Identity Manager and NMAS

Identity Manager controls the “entry point” (updating either Universal or Distribution Password directly), while NMAS controls the flow of synchronizing passwords inside eDirectory.

In [Scenario 1](#), the DirXML Driver for eDirectory can be used to update the NDS password directly. This scenario is basically the same as the one provided in DirXML 1.x.

In [Scenario 2](#), [Scenario 3](#), and [Scenario 4](#), described later in the section, Identity Manager is used to update either Universal Password or Distribution Password, and Identity Manager goes through NMAS to make password changes. This allows NMAS to update other eDirectory passwords as determined by Password Policy settings, and allows NMAS to enforce Advanced Password Rules from Password Policies for passwords being synchronized with connected systems. In these scenarios, the password that Identity Manager distributes to connected systems is always the Distribution Password. The difference between the scenarios lies in the different combinations of NMAS Password Policy settings, and Identity Manager Password Synchronization settings for each connected system driver.

Scenario 1: eDirectory to eDirectory Password Synchronization Using NDS Password

As in Password Synchronization 1.0, you can synchronize NDS Password between two eDirectory trees using the eDirectory driver. This scenario does not require Universal Password to be implemented, and can be used with eDirectory 8.6.2 or later. Another name for this kind of password synchronization is synchronizing the public/private key pair.

This method should only be used to synchronize passwords from eDirectory to eDirectory. It does not use NMAS and therefore cannot be used to synchronize passwords to connected applications.

In this section:

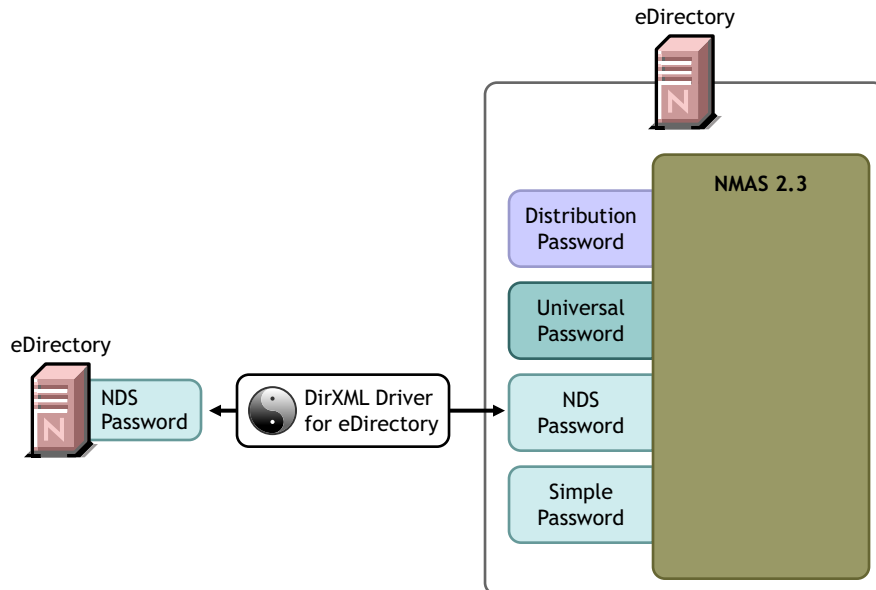
- ◆ [“Advantages and Disadvantages of Scenario 1” on page 140](#)
- ◆ [“Scenario 1 Diagram” on page 141](#)
- ◆ [“Setting Up Scenario 1” on page 141](#)
- ◆ [“Troubleshooting Scenario 1” on page 142](#)

Advantages and Disadvantages of Scenario 1

Advantages	Disadvantages
Simple configuration – Just include the correct attributes in the driver filter.	This method synchronizes passwords between eDirectory trees. Passwords cannot be synchronized to other connected systems.
If you are deploying Identity Manager 2 and eDirectory 8.7.3 in stages, this method can help you deploy gradually.	Does not update Universal Password or Distribution Password.
◆ You don’t need to add the new password synchronization policies to driver configurations.	Because this method does not use NMAS, you can’t validate passwords against Advanced Password Rules in Password Policies for passwords coming from another tree.
◆ Does not require Universal Password to be implemented in the Identity Manager 2 tree.	Because this method does not use NMAS, you can’t reset passwords on the connected eDirectory tree if they don’t comply with Password Policy.
◆ Can be used with connected trees running eDirectory 8.6.2 or later.	E-mail notifications are not provided for password synchronization failures.
◆ Does not require NMAS 2.3.	Check Password Status operations from the iManager task are not supported (Distribution Password is required for this feature).
Enforces the basic password restrictions you can set for NDS Password.	

Scenario 1 Diagram

The figure shows that, as in DirXML 1.x, the DirXML Driver for eDirectory can be used to synchronize the NDS password between two eDirectory trees. This scenario does not go through NMAS.



Setting Up Scenario 1

To set up this kind of password synchronization:

Universal Password Deployment

Not necessary.

Password Policy Configuration

None.

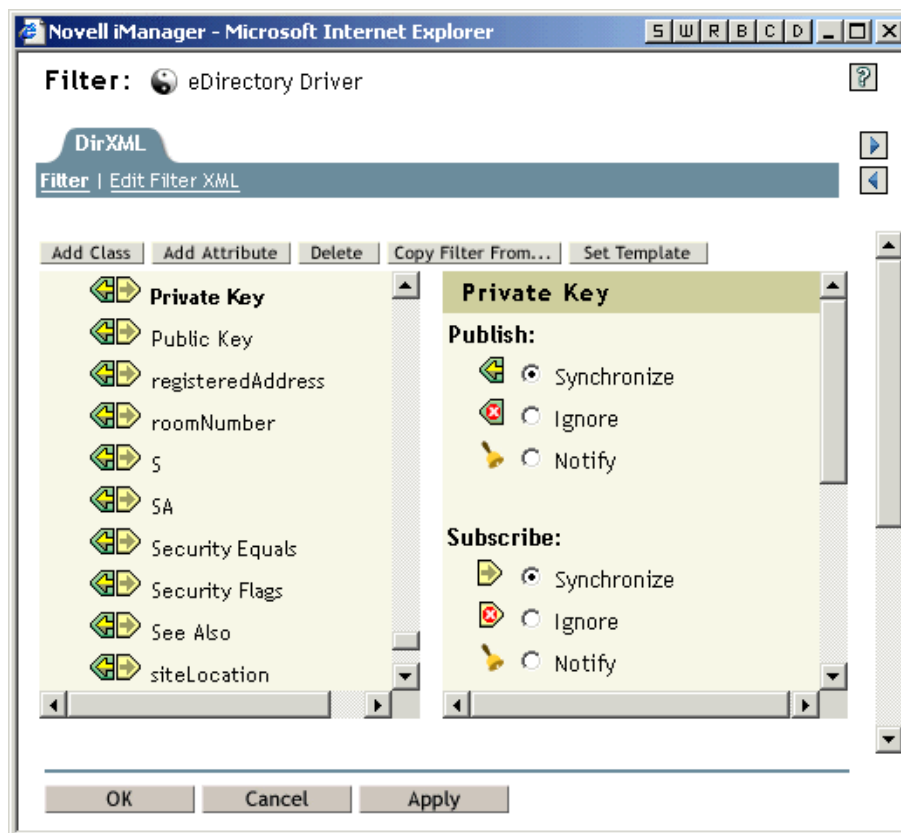
Password Synchronization Settings

None. The settings on the Password Synchronization page for a driver have no effect on this method of synchronizing NDS Password.

Driver Configuration

No password synchronization policies need to be added to upgrade drivers for this kind of password synchronization.

Make sure that the Filter for both eDirectory drivers is synchronizing the Public Key and Private Key attributes for all object classes for which passwords should be synchronized. The following figure shows an example.



Troubleshooting Scenario 1

- ◆ Turn on the DXML Dstrace option.
- ◆ Check the driver Filter to make sure the Public Key and Private Key attributes are being synchronized, not ignored.
- ◆ See also the tips in [“Troubleshooting Password Synchronization” on page 180](#).

Scenario 2: Synchronizing Universal Password

With Identity Manager, you can synchronize a connected system password with the Universal Password in eDirectory.

When Universal Password is updated, the NDS Password, Distribution Password, or Simple Password can also be updated, depending on your settings in the Password Policy.

Any connected system can publish passwords to Identity Manager, though not all connected systems can provide the user’s actual password. For example, Active Directory can publish a user’s actual password to Identity Manager, while PeopleSoft can provide an initial password created in a policy in the driver configuration, such as a password based on the user’s employee ID or last name. Not all drivers can subscribe to password changes from Identity Manager. See [“Connected System Support for Password Synchronization” on page 119](#).

In this section:

- ◆ [“Advantages and Disadvantages of Scenario 2” on page 143](#)
- ◆ [“Scenario 2 Diagram” on page 143](#)

- ♦ “Setting Up Scenario 2” on page 144
- ♦ “Troubleshooting Scenario 2” on page 149

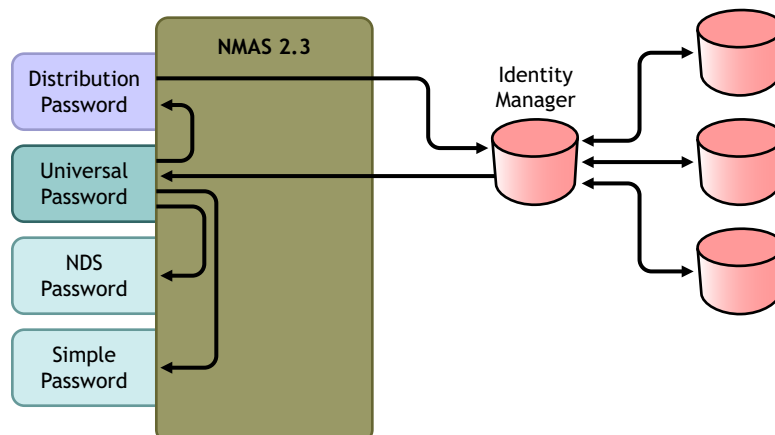
Advantages and Disadvantages of Scenario 2

Advantages	Disadvantages
Allows synchronization of passwords to and from eDirectory and the connected system.	By design, resetting passwords in the connected system is not supported with this method because the Distribution Password and Universal passwords might not be the same, depending on your settings in the Password Policies.
Allows passwords to be validated against the NMAS Password Policy.	
Allows e-mail notifications for failed password operations, such as when a password coming from a connected system does not comply with Password .	
Supports the Check Password Status task in iManager, if Universal Password is being synchronized with Distribution Password and if the connected system supports checking passwords.	
NMAS enforces the Advanced Password Rules in your Password Policies, if you have the rules enabled. If a password coming from a connected system does not comply, an error is generated, and an e-mail notification is sent if you have specified that option.	
If you don't want Password Policy rules enforced, you can uncheck Enable Advanced Password Rules in the Password Policy.	

Scenario 2 Diagram

The figure shows that in this scenario, passwords come in through Identity Manager, which goes through NMAS to directly update Universal Password. Then, NMAS synchronizes the Universal Password with the Distribution Password and other passwords according to the Password Policy settings. Finally, Identity Manager retrieves the Distribution Password to distribute to connected systems that are set to accept passwords.

Although multiple connected systems are shown as connecting to Identity Manager in this figure, keep in mind that you create the settings for each connected system driver individually.



Setting Up Scenario 2

Here's what is required to set up this kind of password synchronization.

In this section:

- ◆ [“Universal Password Deployment” on page 144](#)
- ◆ [“Password Policy Configuration” on page 144](#)
- ◆ [“Password Synchronization Settings” on page 145](#)
- ◆ [“Driver Configuration” on page 147](#)

Universal Password Deployment

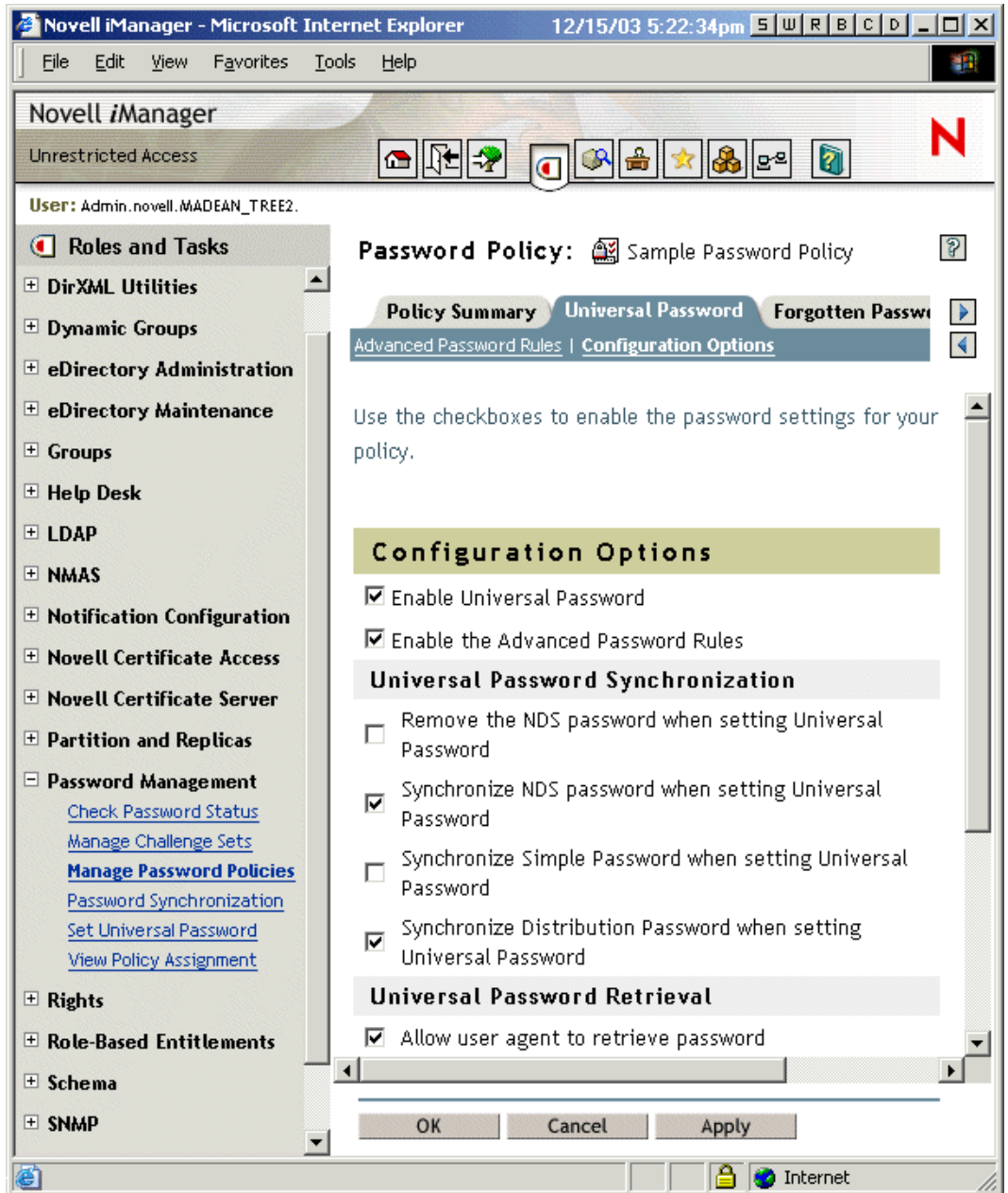
Make sure your environment is ready to use Universal Password. See [“Preparing to Use Identity Manager Password Synchronization and Universal Password” on page 127](#).

Password Policy Configuration

In Password Management > Manage Password Policies, do the following:

- 1** Make sure a Password Policy is assigned to the parts of the eDirectory tree that you want to have this kind of password synchronization. You can assign it to the whole tree (using Login Policy object), a partition root container, a container, or a specific user. We recommend that you assign Password Policies as high in the tree as possible to simplify management.
- 2** In the Password Policy, make sure the following are checked:
 - ◆ Enable Universal Password
 - ◆ Synchronize NDS Password when Setting Universal Password
 - ◆ Synchronize Distribution Password when Setting Universal Password

Because Identity Manager retrieves the Distribution Password to distribute passwords to connected systems, it's important that this option be checked to allow bidirectional password sync.



3 Complete your Password Policy as desired.

NMAS enforces the Advanced Password Rules in your Password Policies, if you have the rules enabled. If you don't want Password Policy rules enforced, uncheck Enable Advanced Password Rules.

4 If you are using Advanced Password Rules, make sure they don't conflict with the password policies on any connected systems that are subscribing to passwords.

Password Synchronization Settings

In Password Management > Password Synchronization, create these settings for the driver for the connected system:

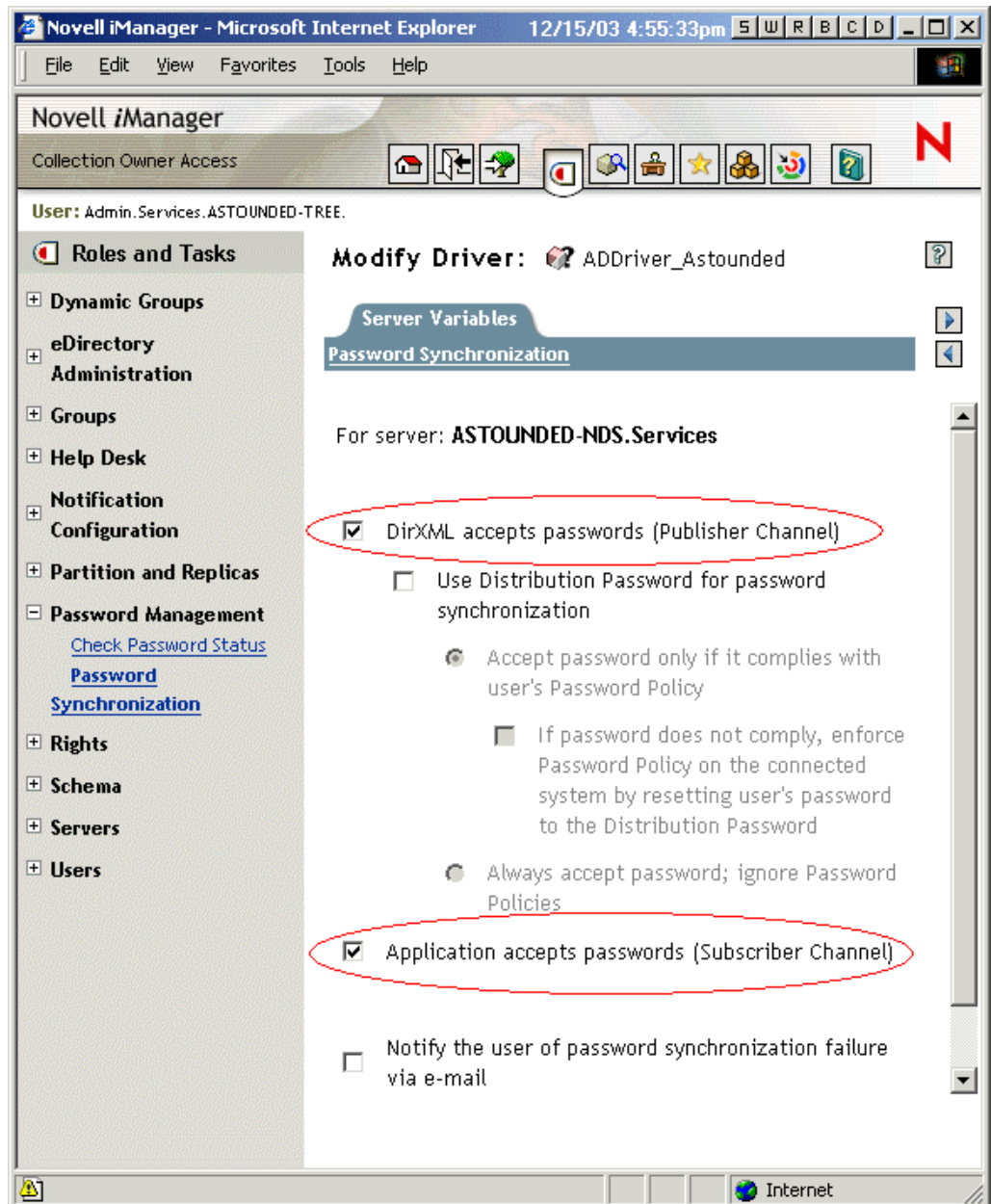
1 Make sure the following are checked:

- ◆ DirXML Accepts Passwords (Publisher Channel)

A message is displayed on the page if the driver manifest does not contain a “password-publish” capability. This is to inform users that passwords cannot be retrieved from the application and can only be published by creating a password in the driver configuration using a policy.

- ◆ Application Accepts Passwords (Subscriber Channel)

If the connected system does not support accepting passwords, the option is dimmed.



These settings allow for bidirectional password synchronization if it is supported by the connected system.

You can adjust the settings to match your business policies for the authoritative source for passwords. For example, if a connected system should subscribe to passwords but not publish, only Application Accepts Passwords (Subscriber Channel) should be selected.

2 Make sure the following is not checked:

- ♦ Use Distribution Password for Password Synchronization

In this scenario, Identity Manager updates the Universal Password directly. The Distribution Password is still used to distribute passwords to connected systems, but is updated from the Universal Password by NMAS instead of by Identity Manager.

3 (Optional) Check the following if desired:

- ♦ Notify the User of Password Synchronization Failure via E-mail

Keep in mind that e-mail notifications require the Internet EMail Address attribute on the eDirectory user object to be populated.

E-mail notifications are non-invasive. They do not affect the processing of the XML document that triggered the e-mail, and if they fail they will not be retried unless the operation itself is retried.

However, debug messages for e-mail notifications are written to the trace file.

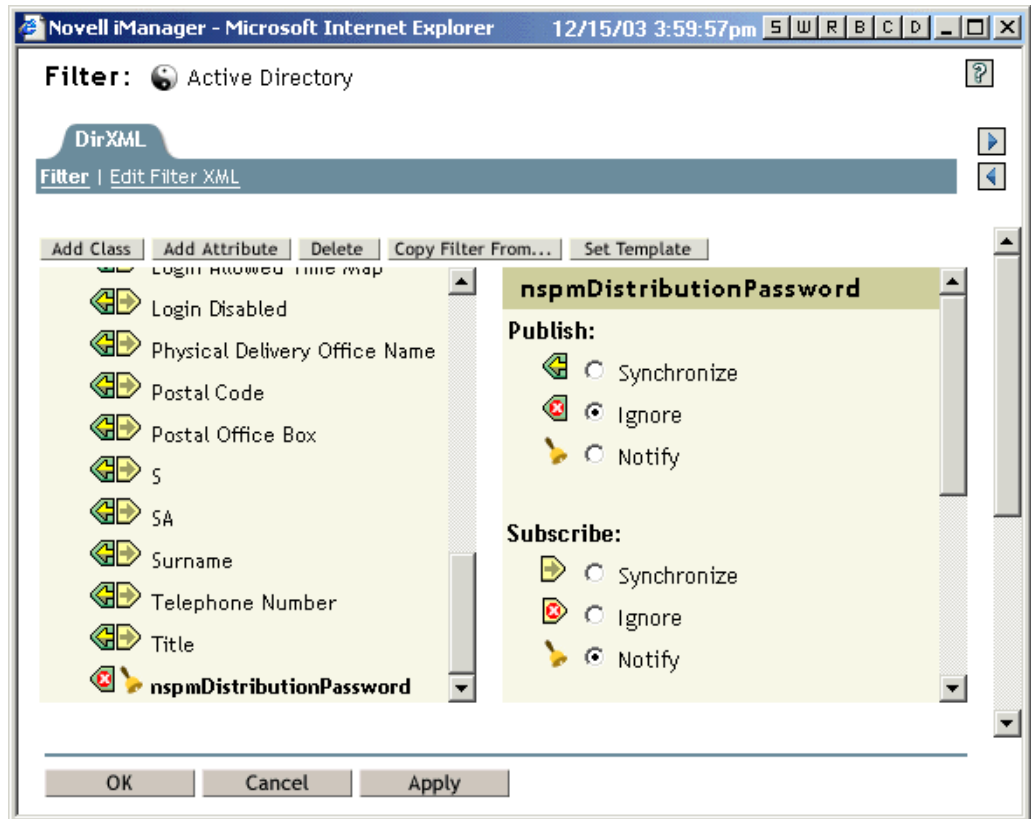
Driver Configuration

- 1** Make sure the required Identity Manager script password synchronization policies are included in the driver configurations for each driver that should participate in password synchronization. The policies must be in the correct location and the correct order in the driver configuration. For the list of policies, see [“Policies Required in the Driver Configuration” on page 125](#).

The Identity Manager sample configurations already contain the policies. If you are upgrading an existing driver, you can add the policies using the instructions in [“Upgrading Existing Driver Configurations to Support Identity Manager Password Synchronization” on page 132](#).

- 2** Set the filter correctly for nspmDistributionPassword attribute:

- ♦ For the Publisher channel, set the Filter to Ignore for the nspmDistributionPassword attribute for all object classes.
- ♦ For the Subscriber channel, set the Filter to Notify for the nspmDistribution Password attribute for all object classes that should subscribe to password changes.



- 3 Ignore both the Public Key and Private Key attributes in the driver Filter for all objects that have Notify set for the nspmDistributionPassword attribute.



- 4 To ensure password security, make sure you control who has rights to Identity Manager objects.

Troubleshooting Scenario 2

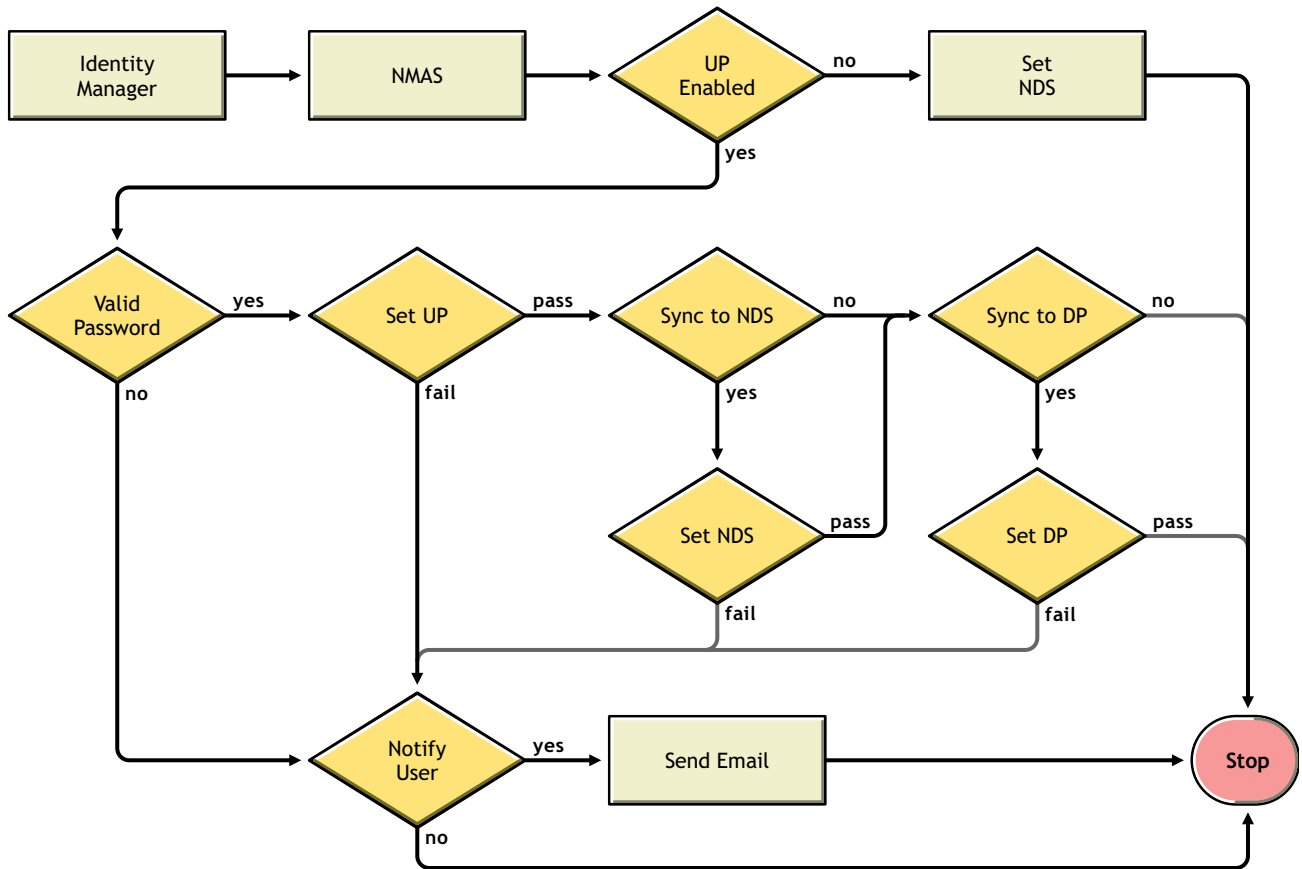
In this section:

- ◆ [“Flowchart” on page 149](#)
- ◆ [“Trouble Logging in to eDirectory” on page 150](#)
- ◆ [“Trouble Logging in to Another Connected System that Subscribes to Passwords” on page 150](#)
- ◆ [“E-Mail Not Generated on Password Failure” on page 151](#)
- ◆ [“Error When Using Check the Object Password” on page 151](#)
- ◆ [“Helpful DSTrace Commands” on page 151](#)

See also the tips in [“Troubleshooting Password Synchronization” on page 180](#).

Flowchart

The following figure shows how NMAS handles the password it receives from Identity Manager. The password is synchronized to Universal Password in this scenario, and NMAS decides how to handle the password based on whether Universal Password is enabled in the Password Policy, whether Advanced Password Rules are enabled that incoming passwords must comply with, and what the other settings are in the Password Policy for synchronizing Universal Password with the other passwords.



Trouble Logging in to eDirectory

- ◆ Turn on the +AUTH, +DCLN, +DXML, and +DVRS settings in DSTrace
- ◆ Verify the <password> or <modify-password> elements are being passed to Identity Manager. To verify that they are being passed, watch the trace screen with those options turned on.
- ◆ Verify that the password is valid according to the rules of the Password Policy.
- ◆ Check NMA Password Policy configuration and assignment. Try assigning the policy directly to user make sure correct policy is being used.
- ◆ On the Password Synchronization page for the driver, make sure “DirXML Accepts Passwords” is checked.
- ◆ In the Password Policy, make sure Synchronize Distribution Password when Setting Universal Password is checked.

Trouble Logging in to Another Connected System that Subscribes to Passwords

This section is for troubleshooting cases where this connected system is publishing passwords to Identity Manager, and another connected system that is subscribing to passwords does not appear to be receiving the changes from this system. Another name for this relationship is a secondary connected system, secondary in the sense that it receives passwords from the first connected system through Identity Manager.

- ◆ Turn on the +DXML and +DVRS settings in DSTrace to see Identity Manager rule processing
- ◆ Set the DirXML trace level for the driver to 3.

- ◆ Check the Password Synchronization “DirXML Accepts Passwords” option.
- ◆ Check the driver filter for nspmDistributionPassword attribute.
- ◆ Verify the <password> for an add or <modify-password> element is being sent to the connected system. To verify, watch the DSTRACE screen or file with the trace options turned on as noted in the first items.
- ◆ Verify that the driver configuration includes the DirXML script password policies in the correct location and correct order, as described in [“Policies Required in the Driver Configuration” on page 125](#).
- ◆ Compare the Password Policy in eDirectory with any password policies enforced by the connected system, to make sure they are compatible.

E-Mail Not Generated on Password Failure

- ◆ Turn on the +DXML in DSTrace to see DirXML rule processing
- ◆ Set the DirXML trace level for the driver to 3.
- ◆ Verify that the rule to generate e-mail is selected.
- ◆ Verify that the eDirectory object contains an Internet EMail Address.
- ◆ In the Notification Configuration task, check the SMTP server and the e-mail template. See [“Configuring E-Mail Notification” on page 170](#).

Error When Using Check the Object Password

The Check Password Status task in iManager causes the driver to be given a check object password action to perform. If you have problems

- ◆ If the Check Object Password returns -603, the eDirectory object does not contain an nspmDistributionPassword attribute; check the DirXML attribute filter along with the “Synchronize Universal to Distribution” option within the NMAS policy.
- ◆ If the Check Object Password returns “Not Synchronized,” verify that the driver configuration contains the appropriate DirXML password synchronization policies.
- ◆ Compare the Password Policy in eDirectory with any password policies enforced by the connected system, to make sure they are compatible.
- ◆ Check Object Password operates from the Distribution Password. If the distribution password is not being updated, Check Object Password might not report that passwords are synchronized.
- ◆ Keep in mind that for eDirectory, Check Password Status is checking the NDS Password.

Helpful DSTrace Commands

+DXML – To view DirXML rule processing and potential error message

+DVRS – To view DirXML driver messages

+AUTH – To view NDS password modifications

+DCLN – To view NDS DCLient messages

Scenario 3: Synchronizing eDirectory and Connected Systems with Identity Manager Updating the Distribution Password

In this method, Identity Manager updates the Distribution Password directly, and allows NMAS to determine how the other eDirectory passwords are synchronized.

Any connected system can publish passwords to Identity Manager, though not all connected systems can provide the user’s actual password. For example, Active Directory can publish a user’s actual password to Identity Manager, while PeopleSoft can provide an initial password created in a policy in the driver configuration, such as a password based on the user’s employee ID or last name. Not all drivers can subscribe to password changes from Identity Manager. See [“Connected System Support for Password Synchronization” on page 119](#).

In this section:

- ◆ [“Advantages and Disadvantages of Scenario 3” on page 152](#)
- ◆ [“Scenario 3 Diagram” on page 152](#)
- ◆ [“Setting Up Scenario 3” on page 153](#)
- ◆ [“Troubleshooting Scenario 3” on page 157](#)

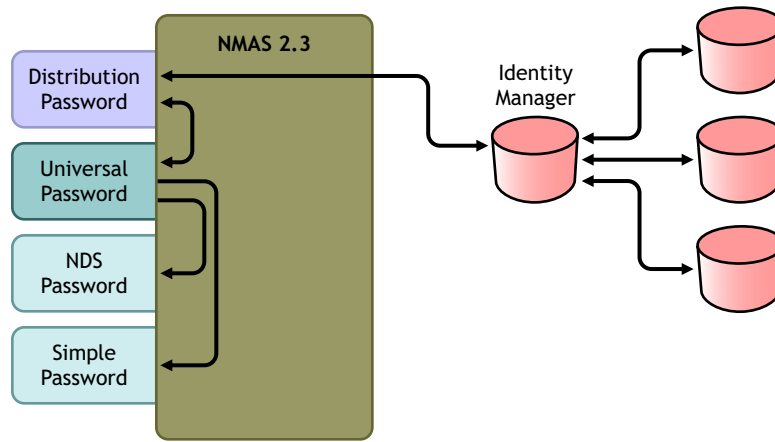
Advantages and Disadvantages of Scenario 3

Advantages	Disadvantages
Allows synchronization of passwords between eDirectory and connected systems.	
Lets you choose whether or not to enforce Password Policies for passwords coming from connected systems.	
You can specify that notification be sent if password synchronization fails.	
If you are enforcing Password Policies, you can choose to reset a password on the connected system to the Distribution Password if the password doesn’t comply.	

Scenario 3 Diagram

The figure shows that in this scenario, passwords come in through Identity Manager, which goes through NMAS to directly update Distribution Password. Identity Manager also uses the Distribution Password to distribute to connected systems that you have specified should accept passwords. NMAS synchronizes Universal Password with the Distribution Password, and with other passwords according to the Password Policy settings.

Although multiple connected systems are shown as connecting to Identity Manager in this figure, keep in mind that you individually create the settings for each connected system driver.



Setting Up Scenario 3

In this section:

- ♦ [“Universal Password Deployment” on page 153](#)
- ♦ [“Password Policy Configuration” on page 153](#)
- ♦ [“Password Synchronization Settings” on page 154](#)
- ♦ [“Driver Configuration” on page 156](#)

Universal Password Deployment

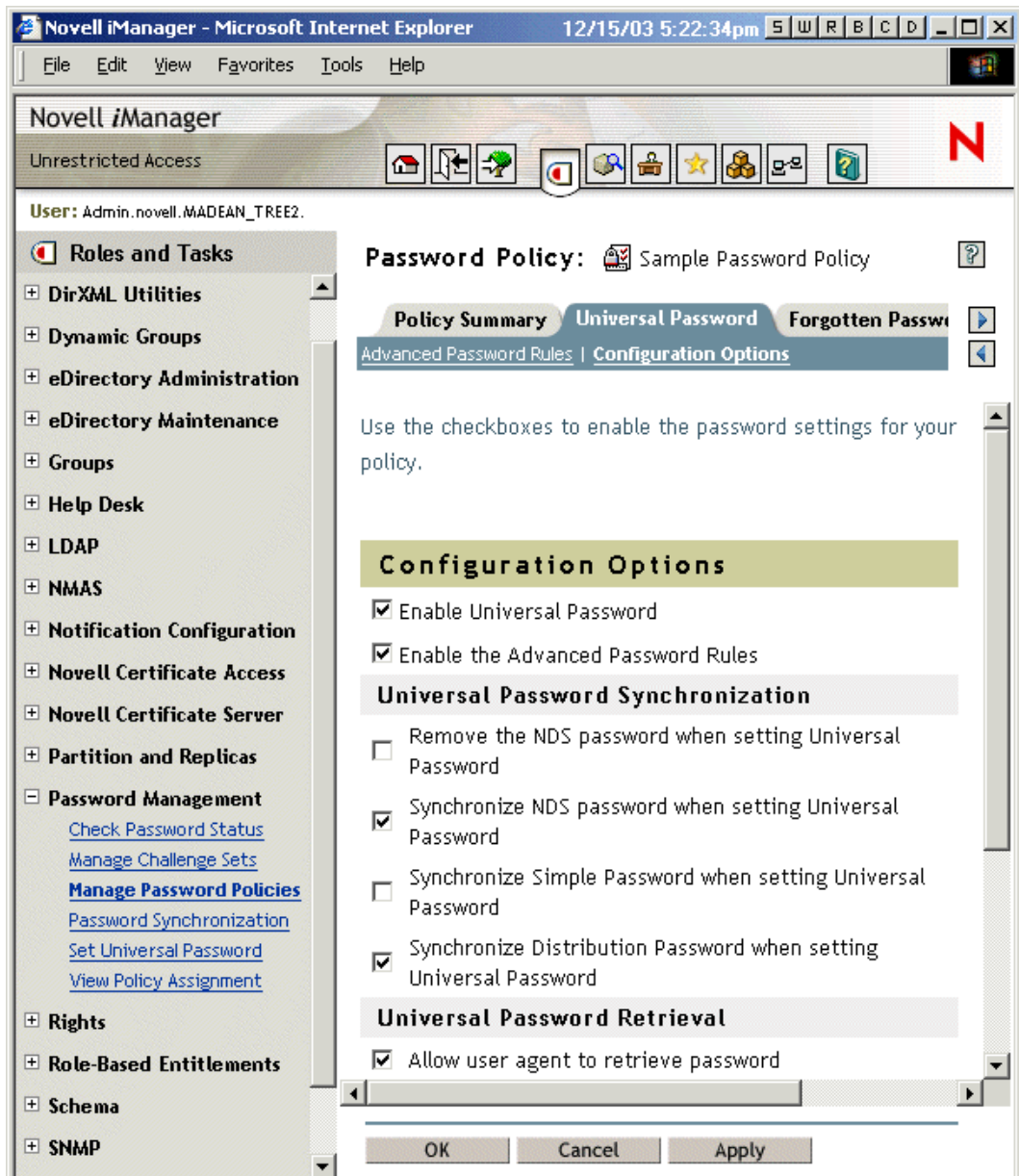
Make sure your environment is ready to use Universal Password. See [“Preparing to Use Identity Manager Password Synchronization and Universal Password” on page 127](#).

Password Policy Configuration

In Password Management > Manage Password Policies, do the following:

- 1** Make sure a Password Policy is assigned to the parts of the eDirectory tree that you want to have this kind of password synchronization. You can assign it to the whole tree, a partition root container, a container, or a specific user. We recommend that you assign Password Policies as high in the tree as possible to simplify management.
- 2** In the Password Policy, make sure the following are checked:
 - ♦ Enable Universal Password
 - ♦ Synchronize NDS Password when Setting Universal Password
 - ♦ Synchronize Distribution Password when Setting Universal Password

Because Identity Manager retrieves the Distribution Password to distribute passwords to connected systems, it's important that this option be checked to allow bidirectional password sync.



- 3 If you are using Advanced Password Rules, make sure they don't conflict with the password policies on any connected systems that are subscribing to passwords.

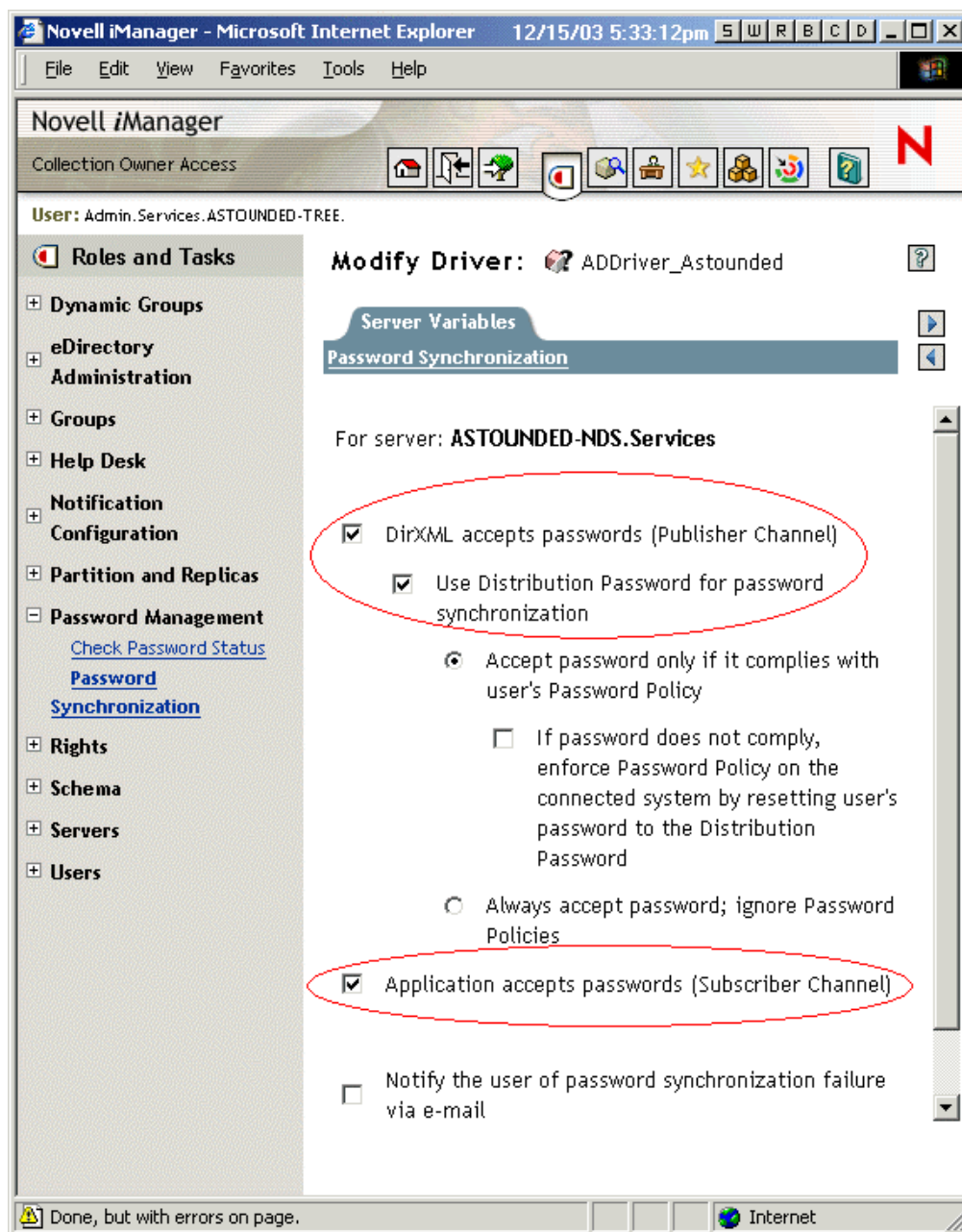
Password Synchronization Settings

In Password Management > Password Synchronization, use these settings:

- 1 Make sure the following are checked:
 - ♦ DirXML Accepts Passwords (Publisher Channel)
 - ♦ Use Distribution Password for Password Synchronization

A message is displayed on the page if the driver manifest does not contain a "password-publish" capability. This is to inform users that passwords cannot be retrieved from the application and can only be published by creating a password in the driver configuration using a policy.

- ♦ Application accepts passwords (Subscriber Channel)



These settings allow for bidirectional password synchronization if it is supported by the connected system.

You can adjust the settings to match your business policies for the authoritative source for password. For example, if a connected system should subscribe to passwords but not publish, only Application Accepts Passwords (Subscriber Channel) should be selected.

- 2 Specify whether you want Password Policies to be enforced or ignored, using the options under Use Distribution Password for password synchronization.
- 3 (Conditional) If you have specified that you want Password Policies to be enforced, also specify whether you want Identity Manager to reset the connected system password if it does not comply.

4 (Optional) Check the following if desired:

- ◆ Notify the User of Password Synchronization Failure via E-mail

Keep in mind that e-mail notifications require the Internet EMail Address attribute on the eDirectory user object to be populated.

E-mail notifications are noninvasive. They do not affect the processing of the XML document that triggered the email and if they fail they will not be retried unless the operation itself is retried.

However, debug messages for e-mail notifications are written to the trace file.

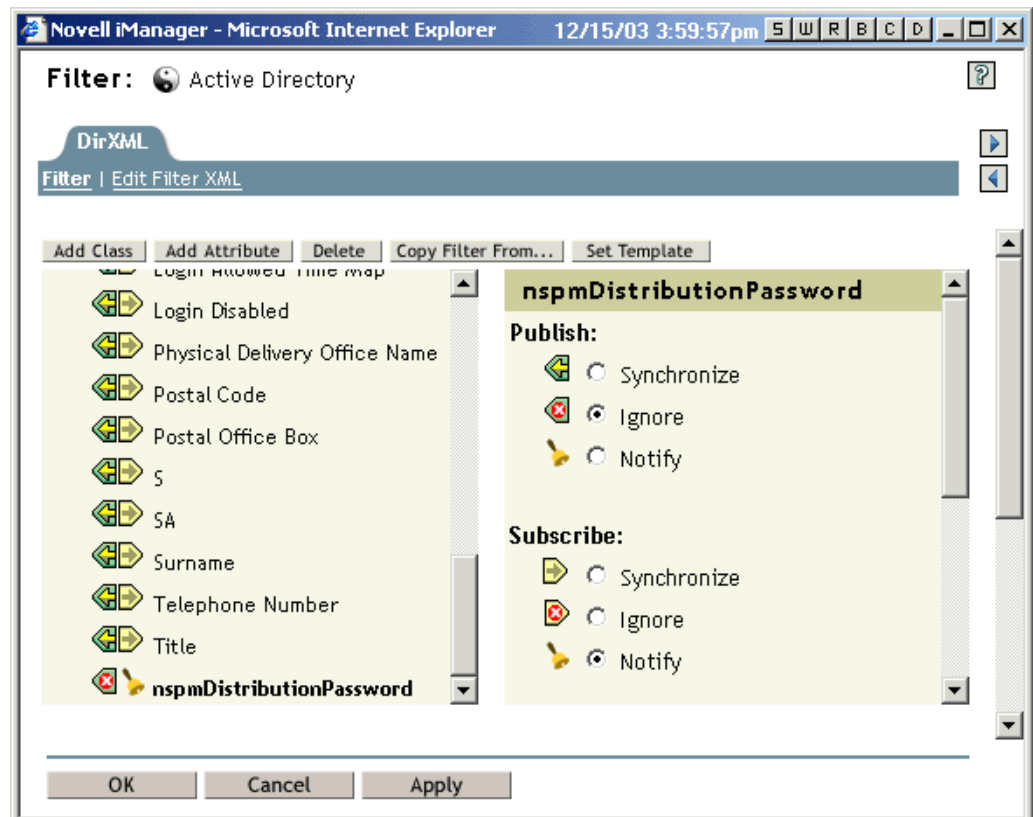
Driver Configuration

- 1** Make sure the required DirXML script password synchronization policies are included in the driver configurations for each driver that should participate in password synchronization. The policies must be in the correct location and the correct order in the driver configuration. For the list of policies, see [“Policies Required in the Driver Configuration” on page 125](#).

The Identity Manager sample configurations already contain them. If you are upgrading an existing driver, you can add the policies using the instructions in [“Upgrading Existing Driver Configurations to Support Identity Manager Password Synchronization” on page 132](#).

- 2** Set the filter correctly for nspmDistributionPassword attribute:

- ◆ For the Publisher channel, set the Filter to Ignore for the nspmDistributionPassword attribute for all object classes.
- ◆ For the Subscriber channel, set the Filter to Notify for the nspmDistribution Password attribute for all object classes that should subscribe to password changes.



- 3** Ignore both the Public Key and Private Key attributes in the driver Filter for all objects that have Notify set for the nspmDistributionPassword attribute.



- 4** To ensure password security, make sure you control who has rights to DirXML objects.

Troubleshooting Scenario 3

In this section:

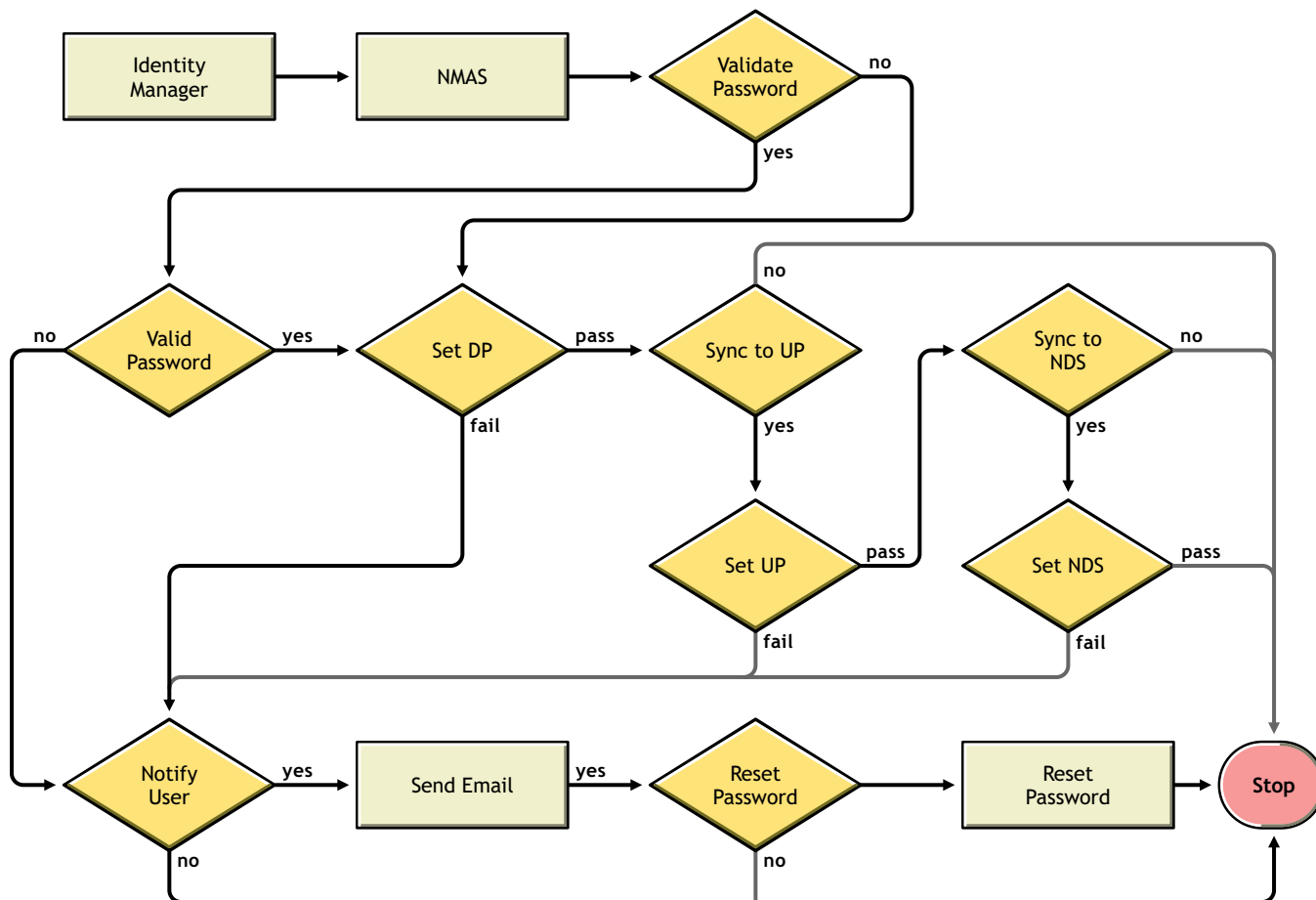
- ♦ [“Flowchart for Scenario 3” on page 157](#)
- ♦ [“Trouble Logging in to eDirectory” on page 158](#)
- ♦ [“Trouble Logging in to Another Connected System that Subscribes to Passwords” on page 159](#)
- ♦ [“E-mail Not Generated on Password Failure” on page 159](#)
- ♦ [“Error When Using Check Password Status” on page 160](#)
- ♦ [“Helpful DSTrace Commands” on page 160](#)

See also the tips in [“Troubleshooting Password Synchronization” on page 180](#).

Flowchart for Scenario 3

The following figure shows how NMAS handles the password it receives from Identity Manager. The password is synchronized to the Distribution Password in this scenario, and NMAS decides how to handle the password based on whether you have specified that incoming passwords should be validated against Password Policy rules (if Universal Password and Advanced Password Rules

are enabled), and what the other settings are in the Password Policy for synchronizing Universal Password with the other passwords.



Trouble Logging in to eDirectory

- ◆ Turn on the +AUTH, +DCLN, +DXML, and +DVRS settings in DSTrace
- ◆ Verify that the <password> or <modify-password> elements are being passed to Identity Manager. To verify, watch the DSTTRACE screen or file with the trace options turned on as noted in the first item.
- ◆ Verify that the password is valid according to the rules of the Password Policy.
- ◆ Check Password Policy configuration and assignment. Try assigning the policy directly to the user to make sure correct policy is being used.
- ◆ On the Password Synchronization page for the driver, make sure that DirXML Accepts Passwords (Publisher Channel) is checked.
- ◆ In the Password Policy, make sure that Synchronize Distribution Password when Setting Universal Password is checked.
- ◆ In the Password Policy, make sure that Synchronize NDS Password when Setting Universal Password is checked, if this is desired.
- ◆ If users are logging in through the Novell Client or ConsoleOne, check the version. Legacy Novell Clients and ConsoleOne cannot log in to eDirectory if the Universal Password is not synchronized with the NDS Password.

Versions of the Novell Client and ConsoleOne are available that are aware of the Universal Password.

- ♦ Some legacy utilities authenticate using the NDS Password, and also cannot log in to eDirectory if the Universal Password is not synchronized with the NDS Password. If you don't want to use the NDS Password for most users, but you have administrator or help desk users that need to authenticate with legacy utilities, try using a different Password Policy for help desk users so you can specify different Universal Password synchronization options for them.

Trouble Logging in to Another Connected System that Subscribes to Passwords

This section is for troubleshooting cases where this connected system is publishing passwords to Identity Manager, and another connected system that is subscribing to passwords does not appear to be receiving the changes from this system. Another name for this relationship is a secondary connected system, secondary in the sense that it receives passwords from the first connected system through Identity Manager.

- ♦ Turn on the +DXML and +DVRs settings to see DirXML rule processing and potential errors
- ♦ Set the DirXML trace level for driver to 3.
- ♦ Make sure that the DirXML Accepts Passwords (Publisher Channel) option is checked in Password Synchronization page.
- ♦ In the Password Policy, make sure that Synchronize Distribution Password when Setting Universal Password is checked.

Identity Manager uses the Distribution Password to synchronize passwords to connected systems. Universal Password must be synchronized with the Distribution Password for this synchronization method.

- ♦ Check the driver filter for nspmDistributionPassword attribute.
- ♦ Verify that the <password> element for an add or a <modify-password> element have been converted to add and modify attribute operations for the nspmDistributionPassword. To verify, watch the DSTRACE screen or file with the options turned on as noted in the first item.
- ♦ Verify that the driver configuration includes the Identity Manager script password policies in the correct location and correct order, as described in [“Policies Required in the Driver Configuration” on page 125](#).
- ♦ Compare the Password Policy in eDirectory with any password policies enforced by the connected system, to make sure they are compatible.

E-mail Not Generated on Password Failure

- ♦ Turn on the +DXML setting in DSTrace to see DirXML rule processing
- ♦ Set the DirXML trace level for the driver to 3.
- ♦ Verify that the rule to generate e-mail is selected.
- ♦ Verify that the eDirectory object contains an Internet EMail Address.
- ♦ In the Notification Configuration task, check the SMTP server and the e-mail template. See [“Configuring E-Mail Notification” on page 170](#).

E-mail notifications are non-invasive. They do not affect the processing of the XML document that triggered the e-mail, and if they fail they will not be retried unless the operation itself is retried.

However, debug messages for e-mail notifications are written to the trace file.

Error When Using Check Password Status

The Check Password Status task in iManager causes the driver to be given a check object password action to perform.

- ♦ Make sure the connected system supports checking passwords. See [“Connected System Support for Password Synchronization” on page 119](#).

This operation is not available through iManager if the driver manifest does not indicate that the connected system supports password-check capability.

- ♦ If the Check Object Password returns -603, the eDirectory object does not contain an nspmDistributionPassword attribute. Check the DirXML attribute filter, and the “Synchronize Universal to Distribution” option within the Password Policy.
- ♦ If the Check Object Password returns “Not Synchronized”, verify that the driver configuration contains the appropriate Identity Manager Password Synchronization policies.
- ♦ Compare the Password Policy in eDirectory with any password policies enforced by the connected system, to make sure they are compatible.
- ♦ Check Object Password checks the Distribution Password. If the Distribution Password is not being updated, Check Object Password might not report that passwords are synchronized
- ♦ Keep in mind that for eDirectory, the Check Password Status checks the NDS Password instead of the Universal Password. This means that if the user’s Password Policy does not specify to synchronize the NDS Password with the Universal Password, the passwords are always reported as being not synchronized. In fact, the Distribution Password and the password on the connected system might be in sync, but Check Password Status won’t be accurate unless both the NDS Password and the Distribution Password are synchronized with the Universal Password.

Helpful DSTrace Commands

+DXML – To view DirXML rule processing and potential error message.

+DVRs – To view DirXML driver messages

+AUTH – To view NDS password modifications

+DCLN – To view NDS DCLient messages

Scenario 4: Tunneling — Synchronizing Connected Systems but not eDirectory, with Identity Manager Updating the Distribution Password

Identity Manager allows you to synchronize passwords among connected systems while keeping the eDirectory password separate. In this documentation, it is referred to as “tunneling.”

In this method, Identity Manager updates the Distribution Password directly. This method is almost the same as the previous one, [“Scenario 3: Synchronizing eDirectory and Connected Systems with Identity Manager Updating the Distribution Password” on page 152](#). The difference is that you make sure the Universal Password and the Distribution Password are not being synchronized. You do this either by not using Password Policies, or by using Password Policies with the option disabled for Synchronize Distribution Password when Setting Universal Password.

In this section:

- ♦ [“Advantages and Disadvantages of Scenario 4” on page 161](#)
- ♦ [“Setting Up Scenario 4” on page 162](#)

- ◆ “Troubleshooting Scenario 4” on page 164

Advantages and Disadvantages of Scenario 4

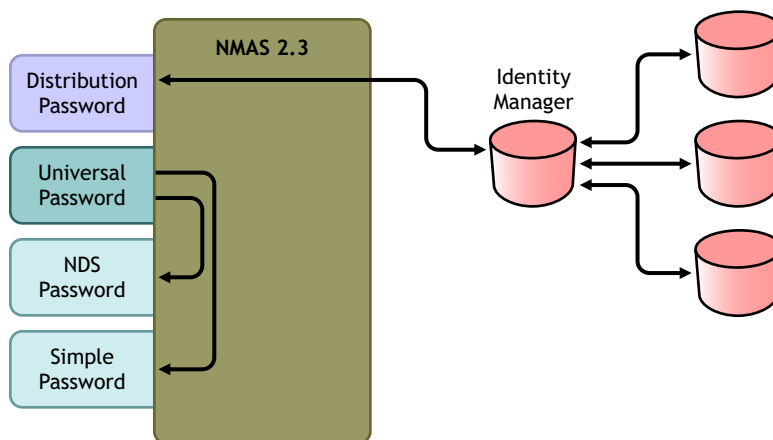
Advantages	Disadvantages
<p>Allows synchronization of passwords among connected systems, while keeping eDirectory password separate.</p> <p>Password Policies are not required.</p> <p>If you are using a Password Policy, the policy does not need to have Universal Password enabled. However, the environment must support Universal Password.</p> <p>Supports the Check Password Status task in iManager, if the connected system supports it.</p> <p>You can specify that notification be sent if password synchronization fails.</p> <p>You can reset a connected system password that does not comply with Password Policy.</p> <p>If Universal Password and Advanced Password Rules are enabled, Password Policies are enforced if you specify that they should be enforced, and passwords on connected systems can be reset.</p>	<p>If Universal Password or Advanced Password Rules are not enabled, Password Policies are not enforced, and passwords on connected systems cannot be reset.</p>

Scenario 4 Diagram

The figure shows that in this scenario, passwords come in through Identity Manager, which goes through NMAS to directly update the Distribution Password. Identity Manager also uses the Distribution Password to distribute to connected systems that you have specified should accept passwords.

The key to this scenario is that in the Password Policy, the setting is disabled for Synchronize Universal Password with Distribution Password. Because the Distribution Password is not synchronized with the Universal Password, Identity Manager synchronizes passwords among connected systems without affecting passwords in eDirectory.

Although multiple connected systems are shown as connecting to Identity Manager in this figure, keep in mind that you create the settings for each connected system driver individually.



Setting Up Scenario 4

In this section:

- ◆ “Universal Password Deployment” on page 162
- ◆ “Password Policy Configuration” on page 162
- ◆ “Password Synchronization Settings” on page 163
- ◆ “Driver Configuration” on page 163

Universal Password Deployment

Although you don’t have to have Password Policies with Universal Password enabled, your environment must still be using eDirectory 8.7.3, which supports Universal Password. See [“Preparing to Use Identity Manager Password Synchronization and Universal Password” on page 127](#).

Password Policy Configuration

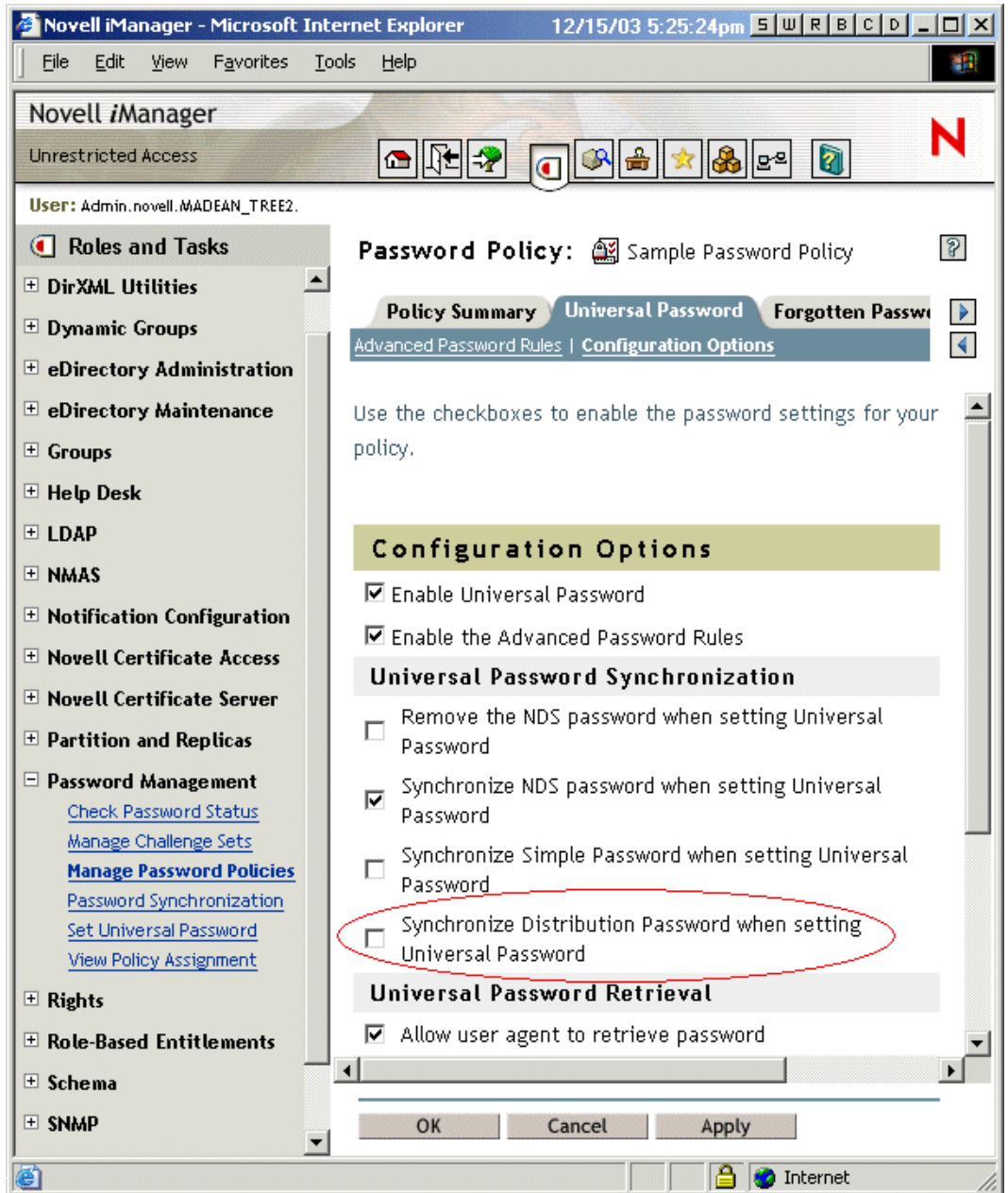
No Password Policy is required for eDirectory users for this method.

However, if you use a Password Policy, you must do the following:

- 1 Make sure the following is NOT checked:

- ◆ Synchronize Distribution Password when Setting Universal Password

This is the key to tunneling passwords without the eDirectory password being affected. By not synchronizing the Universal Password with the Distribution Password, you keep the Distribution Password separate, for use only by Identity Manager for connected systems. Identity Manager acts as a conduit, distributing passwords to and from other connected systems, without affecting the eDirectory password.



2 Complete the other Password Policy settings as desired.

The other password settings in the Password Policy are up to you.

Password Synchronization Settings

Use the same settings as **Password Synchronization Settings** in “Scenario 3: Synchronizing eDirectory and Connected Systems with Identity Manager Updating the Distribution Password” on page 152.

Driver Configuration

Use the same settings as **Driver Configuration** in “Scenario 3: Synchronizing eDirectory and Connected Systems with Identity Manager Updating the Distribution Password” on page 152.

Troubleshooting Scenario 4

If password synchronization is set up for tunneling, the Distribution Password is different than the Universal Password and the NDS Password.

In this section:

- ♦ [“Trouble Logging in to Another Connected System that Subscribes to Passwords” on page 164](#)
- ♦ [“E-mails Not Generated on Password Failure” on page 164](#)
- ♦ [“Error When Using Check Password Status” on page 165](#)
- ♦ [“Helpful DSTrace Commands” on page 165](#)

See also the tips in [“Troubleshooting Password Synchronization” on page 180](#).

Trouble Logging in to Another Connected System that Subscribes to Passwords

This section is for troubleshooting cases where this connected system is publishing passwords to Identity Manager, and another connected system that is subscribing to passwords does not appear to be receiving the changes from this system. Another name for this relationship is a secondary connected system, secondary in the sense that it receives passwords from the first connected system through Identity Manager.

- ♦ Turn on the +DXML and +DVRS settings in DSTrace to see DirXML rule processing and potential errors
- ♦ Set the DirXML trace level for the driver to 3.
- ♦ Make sure that the DirXML Accepts Passwords (Publisher Channel) option is checked on the Password Synchronization page.
- ♦ In the Password Policy, make sure “Synchronize Distribution Password when setting Universal Password” is checked.

Identity Manager uses the Distribution Password to synchronize passwords to connected systems. The Universal Password must be synchronized with the Distribution Password for this synchronization method.

- ♦ Check the driver filter for the nspmDistributionPassword attribute.
- ♦ Verify that the <password> element for an add or a <modify-password> element have been converted to add and modify attribute operations for the nspmDistributionPassword. To verify, watch the DSTRACE screen or file with the trace options turned on as noted in the first item.
- ♦ Verify that the driver configuration includes the DirXML script password policies in the correct location and correct order, as described in [“Policies Required in the Driver Configuration” on page 125](#).
- ♦ Compare the Password Policy in eDirectory with any password policies enforced by the connected system, to make sure they are compatible.

E-mails Not Generated on Password Failure

- ♦ Turn on the +DXML setting in DSTrace to see DirXML rule processing
- ♦ Set the DirXML trace level for driver to 3.
- ♦ Verify that the rule to generate email is selected.

- ♦ Verify that the eDirectory object contains an Internet EMail Address.
- ♦ In Notification Configuration task, check the SMTP server and the e-mail template. See [“Configuring E-Mail Notification” on page 170](#).

E-mail notifications are non-invasive. They do not affect the processing of the XML document that triggered the e-mail, and if they fail they will not be retried unless the operation itself is retried.

However, debug messages for e-mail notifications are written to the trace file.

Error When Using Check Password Status

The Check Password Status task in iManager causes the driver to be given a Check Object Password action to perform.

- ♦ Make sure the connected system supports checking passwords. See [“Connected System Support for Password Synchronization” on page 119](#).

This operation is not available through iManager if the driver manifest does not indicate that the connected system supports password-check capability.

- ♦ If the Check Object Password returns -603, the eDirectory object does not contain an nspmDistributionPassword attribute. Check the DirXML attribute filter, and the Synchronize Universal to Distribution option within the Password Policy.
- ♦ If the Check Object Password returns Not Synchronized, verify that the driver configuration contains the appropriate DirXML password synchronization policies.
- ♦ Compare the Password Policy in eDirectory with any password policies enforced by the connected system, to make sure they are compatible.
- ♦ Check Object Password checks the Distribution Password. If the Distribution Password is not being updated, Check Object Password might not report that passwords are synchronized

Helpful DSTrace Commands

+DXML – To view DirXML rule processing and potential error messages.

+DVRS – To view DirXML driver messages

+AUTH – To view NDS password modifications

+DCLN – To view NDS DCLient messages

Setting Up Password Filters

Some connected systems can provide the user’s actual password to Identity Manager.

To capture passwords on Active Directory, NIS, and NT Domain, you must do some minor setup to install password filters on connected systems.

- ♦ [“Setting Up Password Synchronization Filters for Active Directory and NT Domain” on page 166](#)
- ♦ [“Setting Up Password Synchronization Filters for NIS” on page 166](#)

Setting Up Password Synchronization Filters for Active Directory and NT Domain

This information is in the “Password Synchronization” sections in the driver implementation guides for the DirXML Drivers for Active Directory and NT Domain, at [DirXML Drivers \(http://www.novell.com/documentation/lg/dirxmldrivers/index.html\)](http://www.novell.com/documentation/lg/dirxmldrivers/index.html).

The DirXML driver for AD or NT Domain needs to be installed on only one Windows machine. The other domain controllers don’t need the driver installed, but each domain controller does need a pwfilter.dll file installed to capture passwords so they can be sent to Identity Manager. To simplify your setup and administration, a utility is provided that lets you do this for all domain controllers from the Windows machine where the driver is installed.

Setting Up Password Synchronization Filters for NIS

The DirXML Driver for NIS 2.0 can operate with three UNIX authentication data stores: files, NIS and NIS+. A PAM module is provided to capture passwords and send them to the DirXML Driver for NIS.

The deployment of the PAM module for the NIS Driver is explained in the *DirXML Driver for NIS Implementation Guide*, at [DirXML Drivers \(http://www.novell.com/documentation/lg/dirxmldrivers/index.html\)](http://www.novell.com/documentation/lg/dirxmldrivers/index.html).

Managing Password Synchronization

In this section:

- ♦ “Setting the Flow of Passwords Across Systems” on page 166
- ♦ “Enforcing Password Policies on Connected Systems” on page 169
- ♦ “Keeping the eDirectory Password Separate from the Synchronized Password” on page 170

Setting the Flow of Passwords Across Systems

The following interface lets you see how your systems are set up to accept or publish passwords. It’s available in the Password Synchronization task under the Password Management role.

The first page you see lets you search for drivers for connected systems.



The search results show the settings for password flow to and from Identity Manager and the connected systems.

Novell iManager - Microsoft Internet Explorer 12/16/03 3:37:02pm

File Edit View Favorites Tools Help

MYWAY Edit Google AltaVista Ask Jeeves AlltheWeb More Links

Address https://dxml3.provo.novell.com/nps/servlet/portalservice?NPService=AuthenticationService&NPServiceDataType

Novell iManager

Unrestricted Access

User: admin.Novell.DXML3-TREE.

Roles and Tasks

- Notification
- Configuration
- Novell Certificate Access
- Novell Certificate Server
- Partition and Replicas
- Password Management
 - Check Password Status
 - Manage Challenge Sets
 - Manage Password Policies
 - Password Synchronization
 - Set Universal Password
 - View Policy Assignment
- Rights
- Role-Based Entitlements
- Schema
- SNMP
- UNIX Profile Management
- Users
- WAN Traffic

Password Synchronization

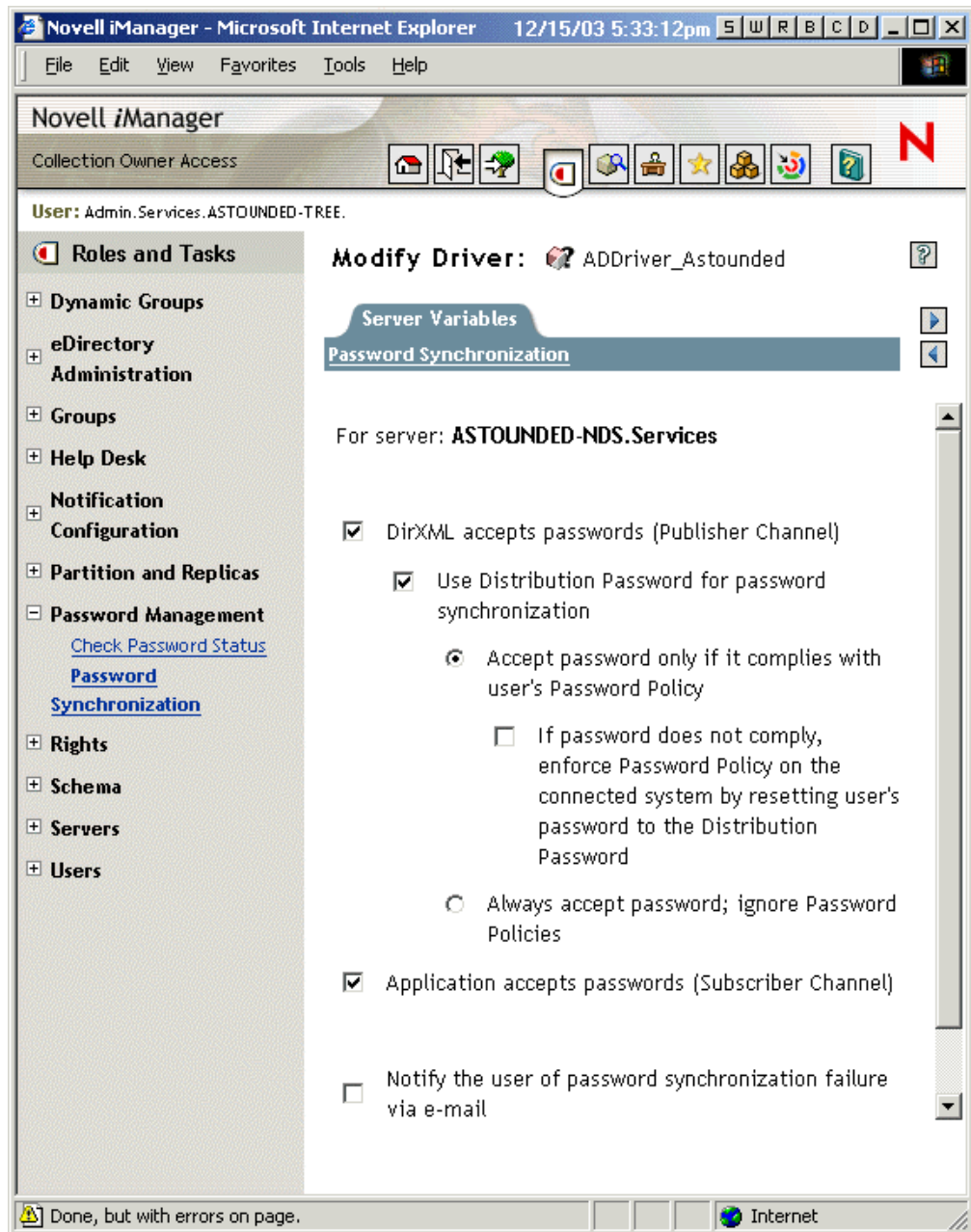
This list shows drivers for connected systems and their current settings for Password Synchronization. Click on the Name link to change the settings. Note that making changes will cause the associated driver to be restarted.

Connected Systems: .DXML3-TREE.

Name	Server	DirXML Accepts Passwords	Application Accepts Passwords
ADDriver	DXML3	✓ Enabled	✓ Enabled
DB2	DXML3	✓ Enabled	✗ Not Available
eDirectory Driver99	DXML3	✓ Enabled	✓ Enabled
Entitlement Services Driver	DXML3	✗ Not Available	✗ Not Available
NT Domains	DXML3	✓ Enabled	✓ Enabled

Internet

To make changes to these settings, you click a connected system driver name. The following page appears, where you can see more detail and change the settings:



On this page, you can set whether Password Policy is enforced for passwords coming in to Identity Manager, and whether Password Policy is enforced on the connected system by resetting the connected system password.

The settings on this page are global configuration values (GCVs), which are stored per server. See [“Password Synchronization Settings You Create Using Global Configuration Values”](#) on page 121.

Enforcing Password Policies on Connected Systems

If you are using Advanced Password Rules and are using Identity Manager Password Synchronization, We recommend that you research the password policies for all the connected systems, and then make sure the Advanced Password Rules are compatible.

Keeping the eDirectory Password Separate from the Synchronized Password

This scenario is described in “[Scenario 4: Tunneling — Synchronizing Connected Systems but not eDirectory, with Identity Manager Updating the Distribution Password](#)” on page 160.

Checking the Password Synchronization Status for a User

An iManager task is provided to let you determine whether the Distribution Password for a specific user is the same as the password in the connected system.

In iManager, click Password Management > Check Password Status.

The Check Password Status task in iManager causes the driver to be given a Check Object Password action to perform.

Not all drivers support password check. Those that do must contain a password-check capability in the driver’s manifest. iManager does not allow password check operations to be sent to drivers that do not contain this capability in the manifest.

Check Object Password checks the Distribution Password. If the Distribution Password is not being updated, Check Object Password might report that passwords are not synchronized.

The Distribution Password is not updated if

- ◆ You are using the synchronization method described in “[Scenario 1: eDirectory to eDirectory Password Synchronization Using NDS Password](#)” on page 140.
- ◆ You are synchronizing Universal Password (as in “[Scenario 2: Synchronizing Universal Password](#)” on page 142), but you have not enabled the Password Policy configuration option to synchronize Universal Password to Distribution Password.

NOTE: Keep in mind that for eDirectory, the Check Password Status checks the NDS Password instead of the Universal Password. This means that if the user’s Password Policy does not specify to synchronize the NDS Password with the Universal Password, the passwords are always reported as being not synchronized. In fact, the Distribution Password and the password on the connected system might be in sync, but Check Password Status won’t be accurate unless both the NDS Password and the Distribution Password are synchronized with the Universal Password.

Configuring E-Mail Notification

The iManager role named Notification Configuration lets you specify the e-mail server and customize the templates for e-mail notifications.

E-mail templates are provided to allow Password Synchronization and Password Self-Service to send automated e-mails to users.

You don’t create the templates; they are provided by the application that uses them. The e-mail templates are Template objects in eDirectory that are created the first time you go into the Notification Configuration task, and they are placed in the Security container, usually found at the root of your tree. Although they are eDirectory objects, you should edit them only through the iManager interface.

This is a modular framework; as new applications are added that use e-mail templates, the templates can be installed along with the applications that use them.

Identity Manager provides templates for Password Synchronization and Forgotten Password notifications. You control whether e-mail messages are sent, based on your choices in the iManager interface.

Password Synchronization is configured to send e-mail only for failed password sync operations, and only for the drivers you specify. The page you use to set this option is shown in the last figure in [“Password Synchronization Settings You Create Using Global Configuration Values” on page 121](#).

For Forgotten Password, e-mail notifications are sent only if you choose to use one of the Forgotten Password actions that causes an e-mail to be sent: e-mail password to user, or e-mail password hint to user. The page you use to set this option is shown in [“Provide Users with Forgotten Password Self-Service” on page 76](#).

In this section:

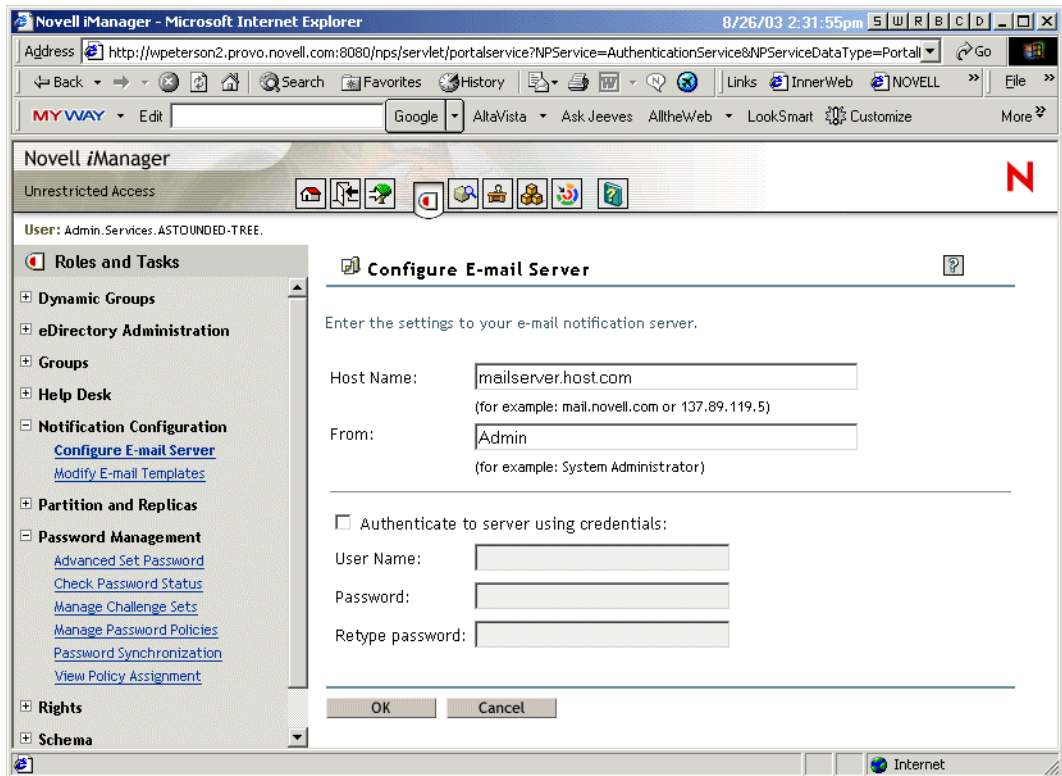
- ◆ [“Prerequisites” on page 171](#)
- ◆ [“Setting Up the SMTP Server To Send E-Mail Notification” on page 171](#)
- ◆ [“Setting Up E-Mail Templates for Notification” on page 172](#)
- ◆ [“Adding Your Own Replacement Tags to E-Mail Notification Templates” on page 173](#)
- ◆ [“Sending E-Mail Notifications to the Administrator” on page 180](#)
- ◆ [“Localizing E-Mail Notification Templates” on page 180](#)

Prerequisites

- ◆ Make sure that your eDirectory users have the Internet EMail Address attribute populated.
- ◆ If you are concerned that some users might not have the e-mail address populated, or you want an e-mail record of all failure notifications, consider choosing a password administrator account that all e-mail notifications are sent to, in addition to the user. This e-mail address should be in the To field of the DirXML script policy. For more information, see [“Sending E-Mail Notifications to the Administrator” on page 180](#).
- ◆ If eDirectory and Identity Manager are on a UNIX server, the server must hold a replica of the e-mail template objects. They are located in the Security container, at the root, which means the server would need a replica of the root partition.

Setting Up the SMTP Server To Send E-Mail Notification

- 1 In iManager, click Notification Configuration > Configure E-mail Server.
The following page appears.



2 Type the following information:

- ♦ The host name
- ♦ The name you want to appear in the From field of the e-mail message, such as Administrator
- ♦ The username and password for authenticating to the server, if necessary

3 Click Close.

4 Customize the e-mail templates as described in [“Setting Up E-Mail Templates for Notification” on page 172](#).

5 Restart DirXML drivers that need to be updated with the changes.

The driver reads the templates and SMTP server information only at startup time.

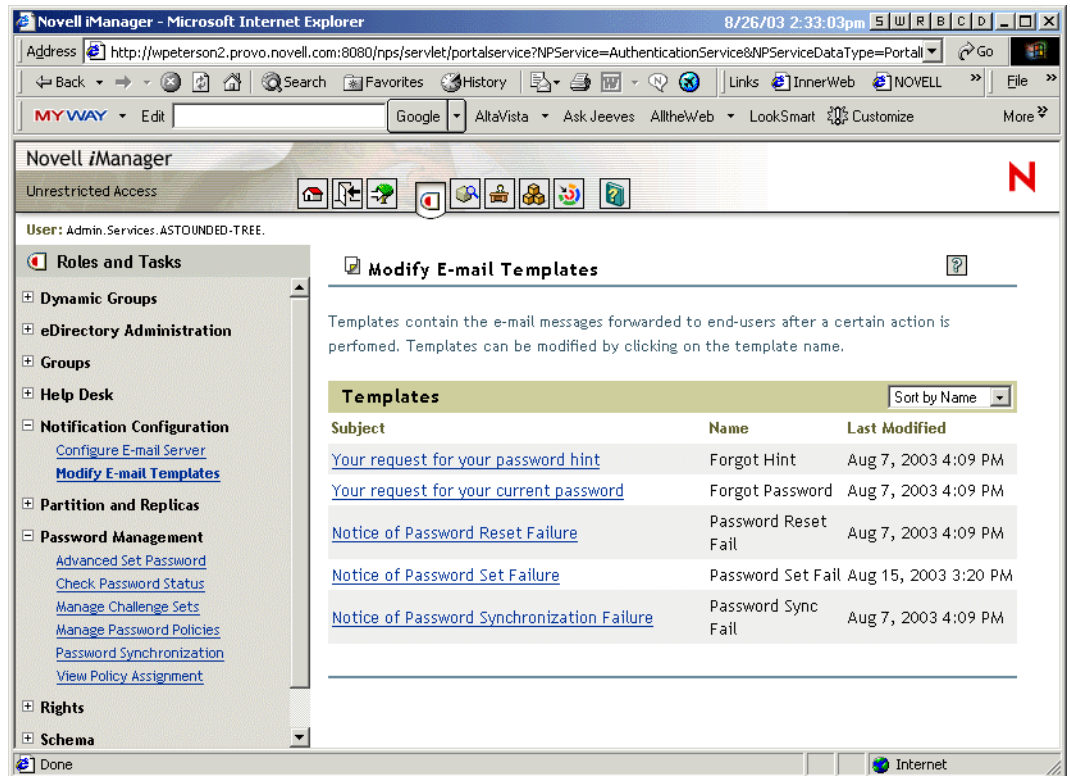
After the e-mail server is set up, e-mail messages can be sent by the applications that use them, if you are using the features that cause messages to be sent.

Setting Up E-Mail Templates for Notification

You can customize these templates with your own text. The name of the template indicates what it is used for.

1 In iManager, click Notification Configuration > Modify E-mail Templates.

A list of templates appears, like the following example.



- 2 Edit the templates as desired. Keep in mind that if you want to add any replacement tags, some additional tasks might be required. Follow the instructions in [“Adding Your Own Replacement Tags to E-Mail Notification Templates”](#) on page 173.
- 3 Restart DirXML drivers that need to be updated with the changes.
The driver reads the templates and SMTP server information only at startup time.

Adding Your Own Replacement Tags to E-Mail Notification Templates

The e-mail notification templates have some tags defined by default, to help you personalize the message for the user. You can also add your own tags.

Your ability to add tags is dependent on the application that is using the e-mail template.

In this section:

- ♦ [“Adding Replacement Tags to Password Synchronization E-Mail Notification Templates”](#) on page 173
- ♦ [“Adding Replacement Tags to Forgotten Password E-Mail Notification Templates”](#) on page 179

Adding Replacement Tags to Password Synchronization E-Mail Notification Templates

You can add replacement tags to the e-mail notification templates for Password Synchronization, but these tags won’t work unless you also define them in every password synchronization policy rule that refers to the e-mail notification template. When using a DoSendEmailFromTemplate action, all replacement tags declared within the template must be defined as child arg-strings elements of the action.

For example, Identity Manager provides default replacement tags that are included with the e-mail notification templates. Identity Manager also provides default password synchronization policies in the driver configurations. Each default tag provided with the e-mail template is also defined in each rule of the password synchronization policy that uses that e-mail template. For example, the UserGivenName tag is one of the default tags defined in the e-mail template named Password Set Fail. A policy rule named “Send e-mail on a failure when subscribing to passwords” refers to that e-mail template in a DoSendEmailFromTemplate action. This rule is used in a policy to send notification to a user about a password synchronization failure. The same UserGivenName tag is defined as an arg-string element in that rule.

Like this example, each new tag you add must be defined in both the e-mail template and the policy rules that refer to the e-mail template, so that the DirXML engine knows how to insert the correct data in place of the replacement tag when sending the e-mail to the user.

You can refer to the tags in the DirXML driver configurations that shipped with Identity Manager as examples to follow.

Keep in mind the following guidelines:

- ◆ The items called replacement tags in the e-mail templates are called tokens in the context of Policy Builder.
- ◆ You should use Policy Builder to make it easier to define the argument strings for the replacement tags, as explained in the steps in this section.
- ◆ The tags you add might be defined to be any of the following:

- ◆ Any Source or Destination attribute for the user

Unlike adding tags for the e-mail templates for Forgotten Password, simply adding a tag that has the same name as an attribute on the user object in eDirectory does not cause the tag to work. As with all tags used in password synchronization e-mail notification templates, you must also define the tag in the policy that is referring to the e-mail template.

- ◆ A global configuration value
- ◆ An XPATH expression

This is in contrast to tags for the e-mail templates for Forgotten Password, which are limited to eDirectory user attributes.

- ◆ Unlike adding tags for the e-mail templates for Forgotten Password (which require you to use the exact name of an eDirectory user attribute), you can name the replacement tags any name you choose, as long as it matches the name used to define the tag in the policies that reference the e-mail template.

To define the tags in a policy, find all the policies that refer to the e-mail notification template, and use Policy Builder to add the tags to them:

- 1** Find all the policies that refer to the e-mail notification template.

One way to make sure you find all the policies that refer to the e-mail notification templates is to export your driver configurations, and search the XML for a do-send-e-mail action that has the template equal to the name of the e-mail notification template.

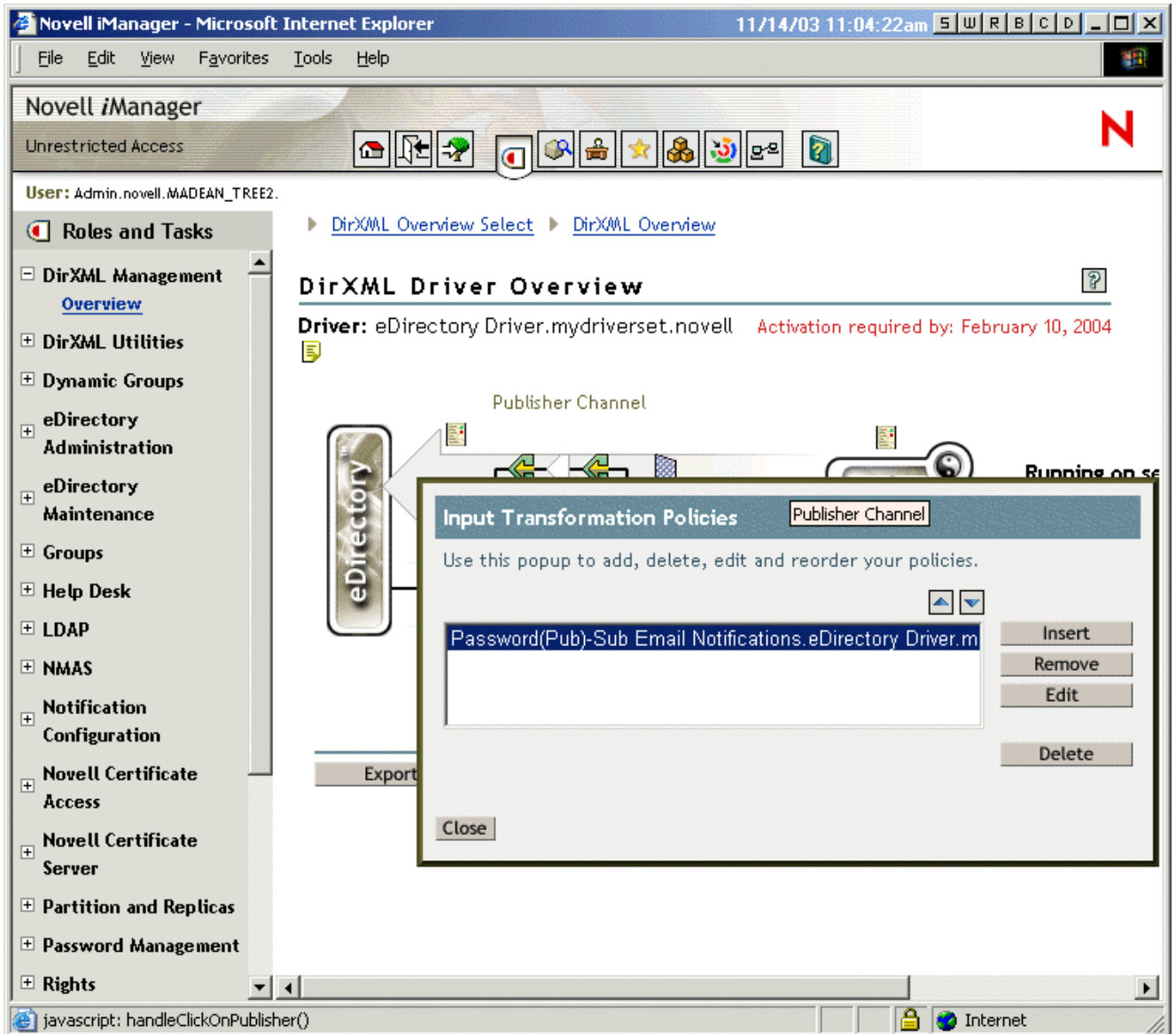
- 2** In each policy, edit each rule that refers to the template. In iManager, click DirXML Management > Overview. Select the driver set that contains the driver with the policy you want to edit.
- 3** Click the icon for the driver that has the policy you want to edit.

- 4 Click the set of policies that contains the policy you want to edit.

For example, the driver configuration for the eDirectory driver that ships with Identity Manager contains a policy in the Input Transformation policy set which references both password synchronization e-mail notification templates.

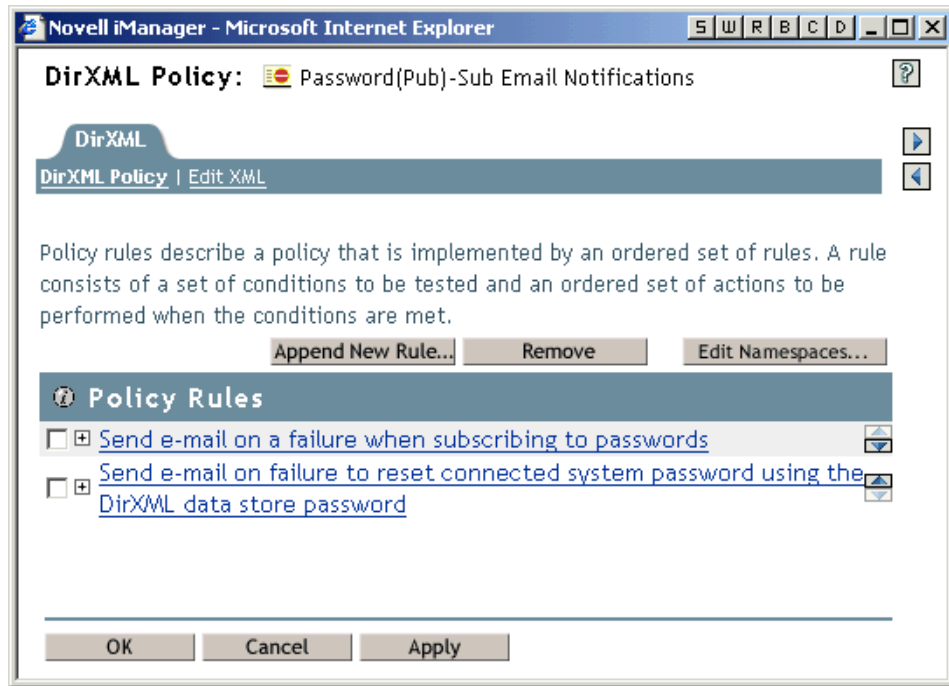
- 5 Click the policy, and then click Edit.

For example, if you were editing the Password(Pub)-Sub Email Notifications policy for the eDirectory driver, this is the page where you would click Edit:

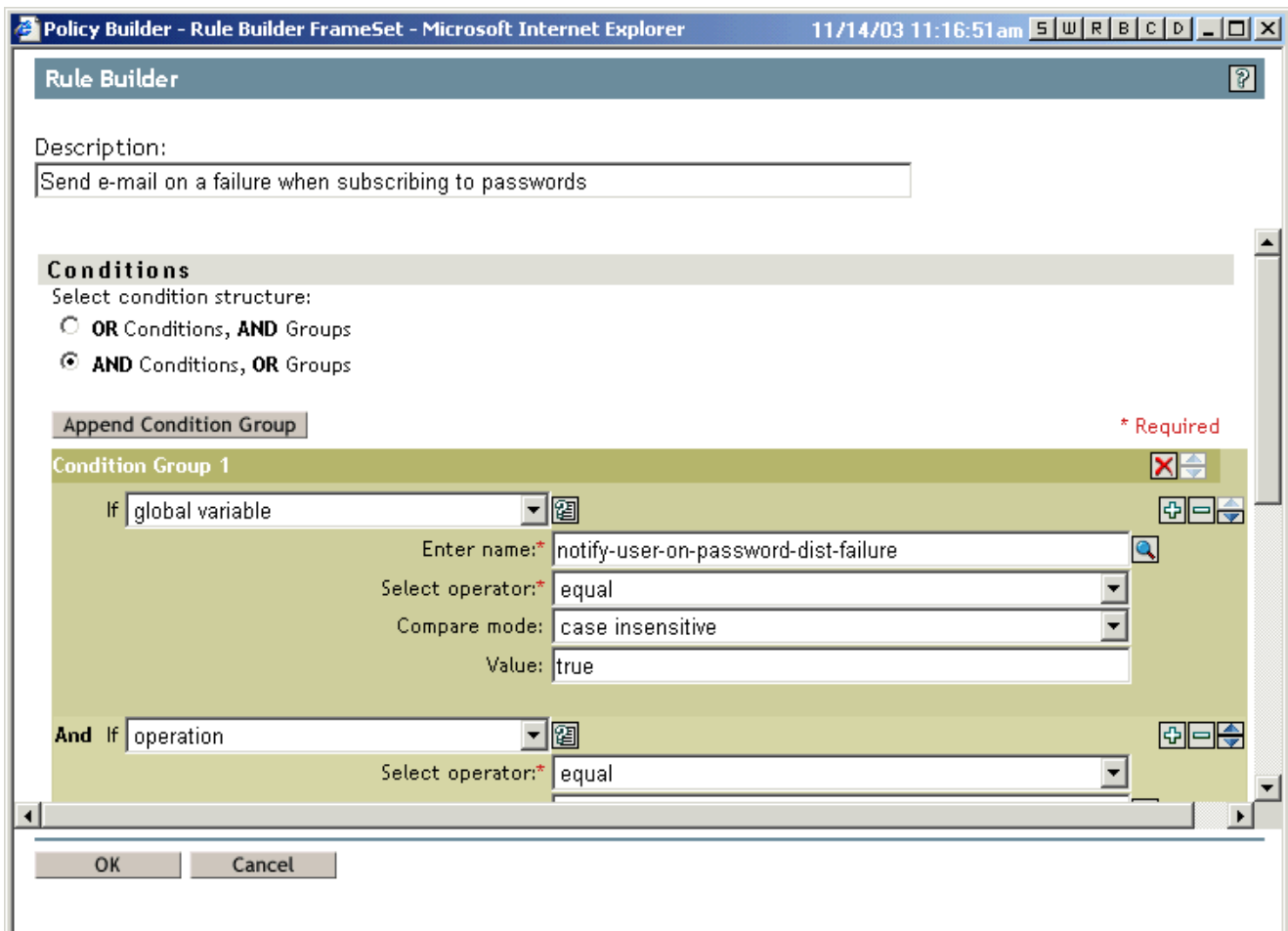


- 6 In the list of rules that opens, click the rule that refers to the e-mail notification template.

For example, in the Password(Pub)-Sub Email Notifications policy, you would see this list of rules. Both of these rules reference one of the password synchronization e-mail templates. You need to edit both rules if you are adding tags to both templates.




If you click the first rule, the following page appears:



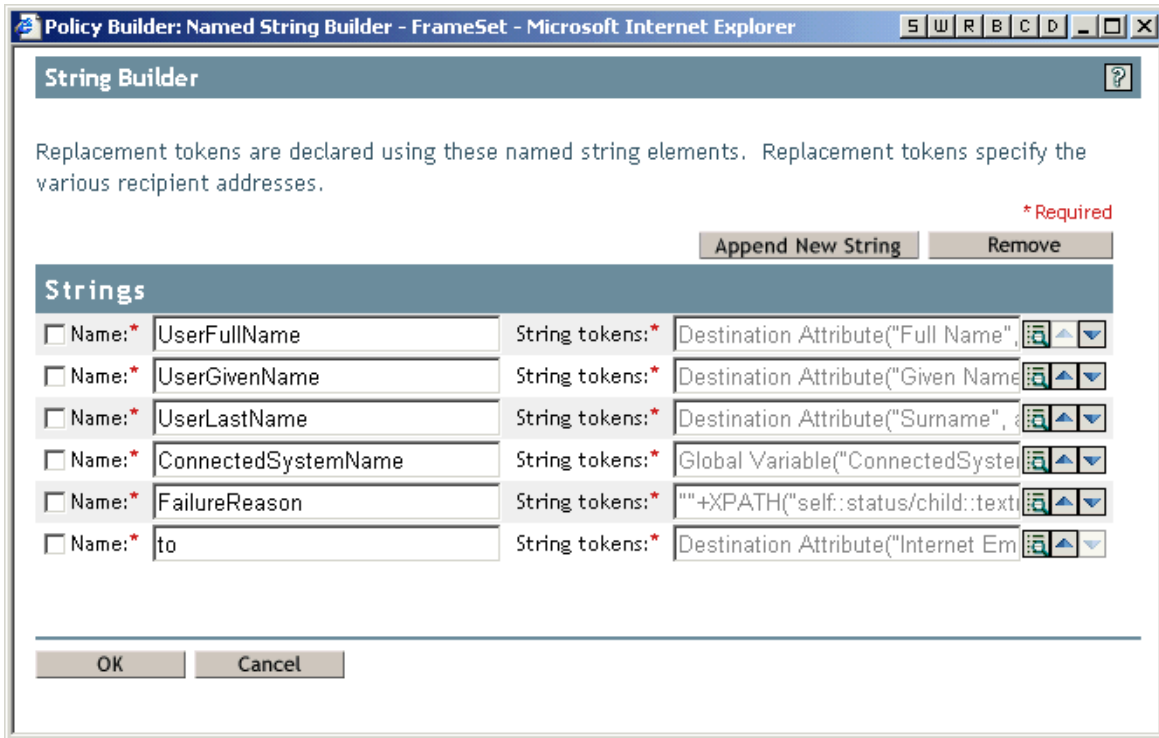
- 7** Scroll down to the section of the rules that displays the actions.

For the example rule, you would scroll down to this section:

The screenshot shows the 'Policy Builder - Rule Builder FrameSet' window in Microsoft Internet Explorer. The window title bar includes the date and time: 11/14/03 11:31:56am. The main area is titled 'Rule Builder' and contains a 'Description' field with the text 'Send e-mail on a failure when subscribing to passwords'. Below the description is a section for defining the rule's conditions. It starts with 'And If' followed by a dropdown menu set to 'XPath expression'. The 'Select operator' is set to 'true', and the 'Value' is set to 'self::status[@level != 'success']/operation-data/password-s'. Below this is the 'Actions' section, which contains an 'Action List' with a single action: 'Do send email from template'. This action has several fields: 'Enter notification DN' set to '\\cn=security\\cn=Default Notification Collection', 'Enter template DN' set to '\\cn=security\\cn=Default Notification Collection\\cn=Password', 'Enter password' (empty), and 'Enter strings' set to 'UserFullName, UserGivenName, UserLastName, Connect'. There is an 'Edit the strings' button next to the 'Enter strings' field. At the bottom of the window are 'OK' and 'Cancel' buttons.

- 8** For the Do Send Email from Template rule, click the browse button  for the Enter strings field. This opens the string builder.

For the example rule, the following figure shows the list of strings you would see. Note that the default tags that are used in the e-mail notification templates are already defined in the password synchronization policies that are part of the DirXML driver configurations, like this one. You can use the default tags as an example.



9 Click Append New String to define a tag that you could use in an e-mail notification template. Enter a name for the tag, making sure it is exactly the same name you use in the e-mail notification template.

10 In the String tokens field, click the browse button  to help you define the tag.

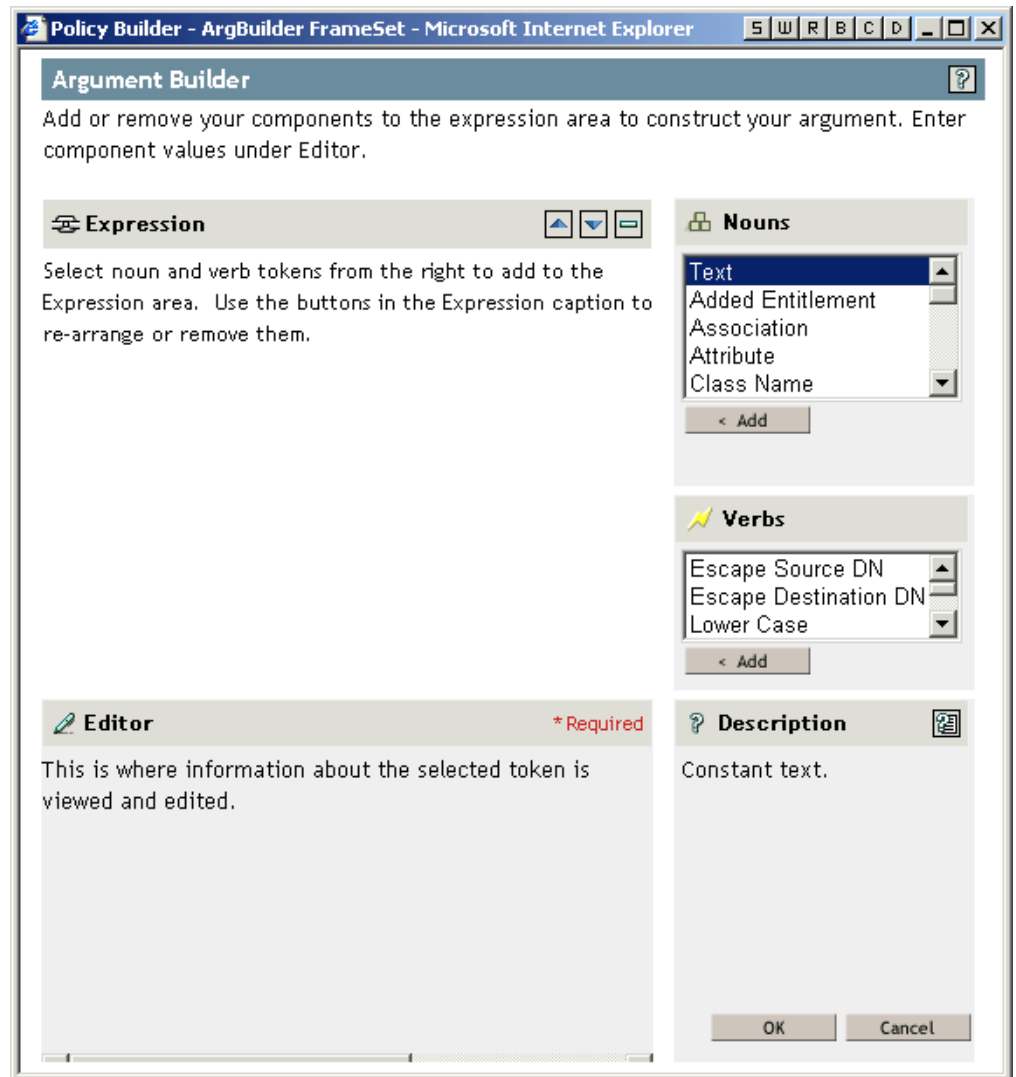
The Argument Builder page appears. This is where you specify what value should be brought in when this tag is used in an e-mail notification template. You can define the tag to be any of the following:

- ♦ Any Source or Destination attribute for the user

Unlike adding tags for the e-mail templates for Forgotten Password, simply adding a tag that has the same name as an attribute on the user object in eDirectory does not cause the tag to work. As with all tags used in password synchronization e-mail notification templates, you must also define the tag in the policy that is referring to the e-mail template.

- ♦ A global configuration value
- ♦ An XPATH expression

The following figure shows an example of the page that helps you define the tag.



After you define the tag and click OK, it shows up as one of the strings in the String Builder page.

- 11** Make sure you click OK to complete all the pages, so that your changes to the policy are saved.
- 12** Repeat the steps to edit the rules in all the policies that refer to the e-mail notification template.
- 13** Add the tag you defined in the policy to the e-mail notification template, using the exact name you used in the policies.

At this point, you can use the tag name in the body of the e-mail notification template.

- 14** Save the changes and restart the driver.

Adding Replacement Tags to Forgotten Password E-Mail Notification Templates

You can add tags to the e-mail notification templates for Forgotten Password, using the following guidelines:

- ♦ You can add only tags that correspond to LDAP attributes on the user object that the message is being sent to.

- ♦ The name of the tag you add must be exactly the same as the LDAP attribute name on the user object.

To see how LDAP attributes correspond to eDirectory attribute names, you can refer to the Schema Mapping Policy that is provided in the DirXML Driver for LDAP.

- ♦ No other configuration is necessary.

Sending E-Mail Notifications to the Administrator

The default configuration is for the e-mail notification to go only to the user. The policies shipped with Identity Manager use the e-mail address from the eDirectory object for the user that is affected.

However, you can configure the password synchronization policies so that e-mail notifications also go to the administrator. To do this, you must modify the DirXML script for one of the policies.

Send a Blind Copy to the administrator by defining the token with the administrator's e-mail address.

To copy an administrator, you would have to modify the policy that generates the e-mail (such as PublishPasswordEmails.xml, in which the policy looks up the e-mail address to send notifications) and add an additional <arg-string> element with the administrator's e-mail address. Here's an example of the additional arg-string element.

```
<arg-string name="to">
    <token-text>Admin@company.com</token-text>
</arg-string>
```

Make sure to restart the driver after making these changes.

Localizing E-Mail Notification Templates

The default templates are in English, but you can edit the text to use other languages.

Keep in mind that the names and the definitions of the replacement tags must remain in English, so that the arg-string token definitions in the policies match the names of the replacement tags.

Troubleshooting Password Synchronization

- ♦ See the tips in [“Implementing Password Synchronization” on page 138](#).
- ♦ Make sure you have the Simple Password Login Method installed with NMAS.
- ♦ Make sure you have a copy of the root of the tree on the servers where you need to NMAS to enforce Password Policies on eDirectory login methods or on passwords from connected systems being synchronized by Identity Manager.
- ♦ Make sure the users that you want to do password synchronization for are replicated on the same server with the driver that is synchronizing the passwords. As with other driver functions, the driver can manage only the users that are in a master or read/write replica on the same server.
- ♦ Make sure SSL is configured properly between the web server and eDirectory.

- ♦ If you see an error about a password not complying when a user is initially created, but the password is set correctly in eDirectory, this might be an issue with the default password in the driver policy not conforming to the Password Policy that applies to that user.

Here's an example using the Active Directory driver, although the same issue could occur for another driver.

Example: Suppose you want the Active Directory driver to provide the initial password for user when it creates a new user object in eDirectory to match a user in Active Directory. The sample configuration for the Active Directory driver sends the initial password as a separate operation than adding the user, and the sample configuration also includes a policy that provides a default password for a user if no password is provided by Active Directory. Because adding the user and setting the password are done separately, in this case a new user always receives the default password, even if only momentarily, and it is soon updated because the Active Directory driver sends the password immediately after adding the user. If the default password does not comply with the eDirectory Password Policy for the user, an error is displayed. For example, if a default password created using the user's surname is too short to comply with the Password Policy, you might see a -216 error saying the password is too short. However, the situation is soon rectified if the Active Directory driver then sends an initial password that does comply.

Regardless of the driver you are using, if you want a connected system that is creating user objects to provide the initial password, consider doing one of the following. These measures are especially important if the initial password does not come with the add event and instead comes in a subsequent event.

- ♦ Change the policy on the Publisher channel that creates default password, so that the default password conforms to the Password Policies (created using Password Management > Manage Password Policies) that have been defined for your organization in eDirectory. When the initial password comes from the authoritative application, it replaces the default password.

This option is preferable because we recommend that a default Password Policy exists in order to maintain a high level of security within the system.

or

- ♦ Remove the policy on the Publisher channel that creates default password. In the sample configuration, this policy is provided in the Command Transformation policy set. Adding a user without a password is allowed in eDirectory. The assumption for this option is that the password for the newly created user object eventually comes through the Publisher channel, so the user object exists without a password for only a short time.
- ♦ Password Policies are assigned with a tree-centric perspective. By contrast, Password Synchronization is set up per driver, and drivers are installed on a per-server basis and can manage only those users who are in a master or read/write replica. To get the results you expect from Password Synchronization, make sure the containers that are in a master or read/write replica on the server running the drivers for Password Synchronization match the containers where you have assigned Password Policies with Universal Password enabled. Assigning a Password Policy to a partition root container ensures that all users in that container and subcontainers are assigned the Password Policy.
- ♦ Helpful DSTrace commands:
 - +DXML – To view DirXML rule processing and potential error messages.
 - +DVRS – To view DirXML driver messages
 - +AUTH – To view NDS password modifications
 - +DCLN – To view NDS DCLient messages

10

Using Role-Based Entitlements

Role-Based Entitlements let you grant entitlements on connected systems to a group of Novell® eDirectory™ users. Using Entitlement Policies, you can streamline management of business policies and reduce the need to configure your DirXML® drivers.

In this section:

- ♦ [“Overview” on page 183](#)
- ♦ [“How it Works” on page 185](#)
- ♦ [“Prerequisites” on page 185](#)
- ♦ [“Getting Started” on page 186](#)
- ♦ [“Creating Entitlement Policies” on page 187](#)
- ♦ [“Keeping Accounts Safe” on page 194](#)
- ♦ [“Controlling Adding or Removing Entitlements” on page 195](#)
- ♦ [“Conflict Resolution between Entitlement Policies” on page 196](#)
- ♦ [“Password Synchronization and Role-Based Entitlements” on page 199](#)
- ♦ [“Troubleshooting Role-Based Entitlements” on page 199](#)

Overview

Role-Based Entitlements let you define business policies about who should be granted entitlements in your environment. Using an Entitlement Policy (an enhanced eDirectory dynamic group), you define the users who should be granted entitlements based on dynamic search criteria, such as a job title of “Tester.” You can manage exceptions using a static inclusion and exclusion list for the policy.

After defining the users for whom the policy should apply, you specify the entitlements you want those users to be granted on connected systems. You can also grant rights in eDirectory, as you can for any dynamic group.

Entitlements on connected systems are granted by DirXML drivers that are configured to support Role-Based Entitlements.

This model of administering business policies is different from the traditional method of provisioning with Identity Manager, because you specify the business policies upstream from the DirXML driver configuration.

Traditionally, entitlements on connected systems are administered on a per-driver basis, solely by creating and editing driver configuration policies such as the ones you create with Policy Builder. In this traditional distributed model, a different administrator often controls each DirXML driver and connected system, and the business policies that determine whether a user gets resources on

that system are “hard-coded” in the driver configuration policies for each connected system driver separately.

The Role-Based Entitlement model fits an environment where one or a few administrators have authority to control business policies. This kind of administrator needs to understand Identity Manager in general but does not necessarily need a lot of Identity Manager or XSLT expertise to use the Role-Based Entitlements interface.

Scenario

Business Policy Example

Suppose you want to automatically provision new employees with the job title of “Tester” by giving them two things:

- ♦ A GroupWise® e-mail account
- ♦ An account in an Oracle database used for tracking defects

Setting Up the Business Policy

Traditional: Using the traditional model, an Identity Manager developer would use Policy Builder or a style sheet to “hard-code” the business policy in the driver configurations for the DirXML Driver for JDBC and the DirXML Driver for GroupWise.

Role-Based Entitlements: Using Role-Based Entitlements in this example, you would create an Entitlement Policy and define dynamic membership for the job title of “Tester.” An Identity Manager developer would also need to have the DirXML Driver for GroupWise and the DirXML Driver for JDBC configured to support Role-Based Entitlements. The drivers would grant accounts to users who met the dynamic membership criteria.

The result is the same up to this point in the example. Using either method, users with the job title of “Tester” automatically are granted an account.

However, if you are using the Role-Based Entitlements model, changing this business policy requires less Identity Manager expertise.

Changing the Business Policy

Suppose that after setting up the business policy, you find out that you also need to grant the same kinds of accounts to users with the title of “Testing Manager.”

Traditional: Using the traditional model, an Identity Manager developer would use Policy Builder to “hard-code” the changes to business policy in two places:

- ♦ The driver configuration for GroupWise
- ♦ The driver configuration for JDBC

Role-Based Entitlements: Using the Role-Based Entitlements model, a network administrator with knowledge of LDAP filters could easily add the additional user criteria to the dynamic membership in the Entitlement Policy, without having to edit DirXML Script. Both the JDBC Driver and the GroupWise driver would grant the accounts to the correct users without changes to the driver configurations.

Another difference between Role-Based Entitlements and traditional Identity Manager administration is that Entitlement Policies make changes directly in a production environment. Traditionally, changes to driver configurations are tested in a lab first. Entitlement Policies make it easier to make changes to business policies, but you should use caution when making these changes in your production environment. For suggestions, see [“Keeping Accounts Safe” on page 194](#).

How it Works

The Role-Based Entitlements feature relies on the Entitlements Service Driver, which is an engine service that monitors whether users have membership in an Entitlement Policy. If a user meets the dynamic membership criteria of an Entitlement Policy dynamic group, or is statically included, the Entitlements Service Driver adds information to the DirXML-SPEntitlements attribute on the user. The entitlement a user should receive is written to the attribute.

For the systems listed in [“Configuring Drivers to Use Entitlement Policies” on page 186](#), you can choose the Role-Based Entitlements option when importing the Identity Manager sample driver configuration. You can then review the policies provided, which support Role-Based Entitlements by monitoring the DirXML-SPEntitlements attribute and granting or revoking entitlements.

The DirXML-SPEntitlements attribute is updated by the Entitlements Service Driver only when one of the following happens:

- ◆ You use the Reevaluate Membership task
You can specify in which part of the tree users should be reevaluated.
- ◆ User is moved
- ◆ User is renamed
- ◆ Any attribute used for membership in an Entitlement Policy is modified

Entitlement Policies let you grant entitlements on connected systems and rights in eDirectory. Entitlements on connected systems can be any of the following:

- ◆ Accounts
- ◆ Membership in e-mail distribution lists
- ◆ Group membership
- ◆ Attributes for the corresponding objects in connected systems, populated with values you specify
- ◆ Placement
- ◆ Other entitlements that you customize

Some of the options are demonstrated in the sample driver configurations.

Because one Entitlements Service Driver is used per driver set, an Entitlement Policy can manage only users that are in a read/write or master replica on the server that is associated with that driver set.

Because Role-Based Entitlements functionality is based on Identity Manager, you must have DirXML drivers installed and configured properly in order to be able to administer connected systems.

In addition, to avoid possible conflicts between Entitlement Policy assignments and DirXML driver configurations, you should be aware of your business policies and how they are administered through Identity Manager. An attribute should not be managed in an overlapping or conflicting way by DirXML Entitlement Policies and policies in a driver configuration.

Prerequisites

- ◆ eDirectory 8.7.3

Getting Started

You must have an Entitlements Service Driver in each driver set where you want to use Role-Based Entitlements. This requires a very simple, one-time setup for each driver set.

Before you can use Role-Based Entitlements with a connected system, you must import the Identity Manager driver configuration for the driver and specify that the driver is to be used with Role-Based Entitlements.

To create an Entitlement Policy, you can use the wizard provided.

In this section:

- ♦ “Creating a Driver Object for the Entitlements Driver” on page 186
- ♦ “Configuring Drivers to Use Entitlement Policies” on page 186

Creating a Driver Object for the Entitlements Driver

Before you can create Entitlement Policies, you need an Entitlements Service Driver object. You must create one for each driver set.

If you don’t have one, you are prompted to create one when you click the Role-Based Entitlements role and task.

- 1** To find out whether you already have an Entitlements Service Driver, in iManager, click Role-Based Entitlements > Role-Based Entitlements. Select the driver set.
 - ♦ If the No Entitlements Driver page appears, continue with **Step 2** to create an Entitlements Service Driver object.
 - ♦ If a Role-Based Entitlements page appears with a list of Entitlement Policies, you already have an Entitlements Service Driver object. You don’t need to complete this procedure.
- 2** In the No Entitlements Driver page, click OK. This opens the Import Driver Wizard (which is also accessible by clicking DirXML Utilities > Import Drivers).
- 3** Follow the steps to create a driver object, choosing Entitlements Service from the list of drivers.

The correct driver configuration file is chosen automatically. Just choose a name for the driver object; no additional configuration or information is necessary.

The driver shim for the Entitlements Driver is installed by default when DirXML is installed. The Entitlements Driver configuration file is installed by default when you install the DirXML plug-ins on your iManager server.

- 4** After you have completed the Wizard, you can access the plug-ins for Role-Based Entitlements and begin creating Entitlement Policies for this driver set.

Configuring Drivers to Use Entitlement Policies

To use Role-Based Entitlements with a connected system, you must have a Identity Manager driver shim installed.

The driver must be configured to support Role-Based Entitlements, which includes having the correct entries in the driver manifest.

You can either import a Identity Manager sample driver configuration for the driver and choose the option to use Role-Based Entitlements, or customize a driver configuration yourself by following the examples in the sample driver configuration.

The following sample driver configurations include support for Role-Based Entitlements as an option:

- ♦ Active Directory
- ♦ Exchange
- ♦ GroupWise
- ♦ LDAP
- ♦ NIS
- ♦ Notes
- ♦ NT Domain

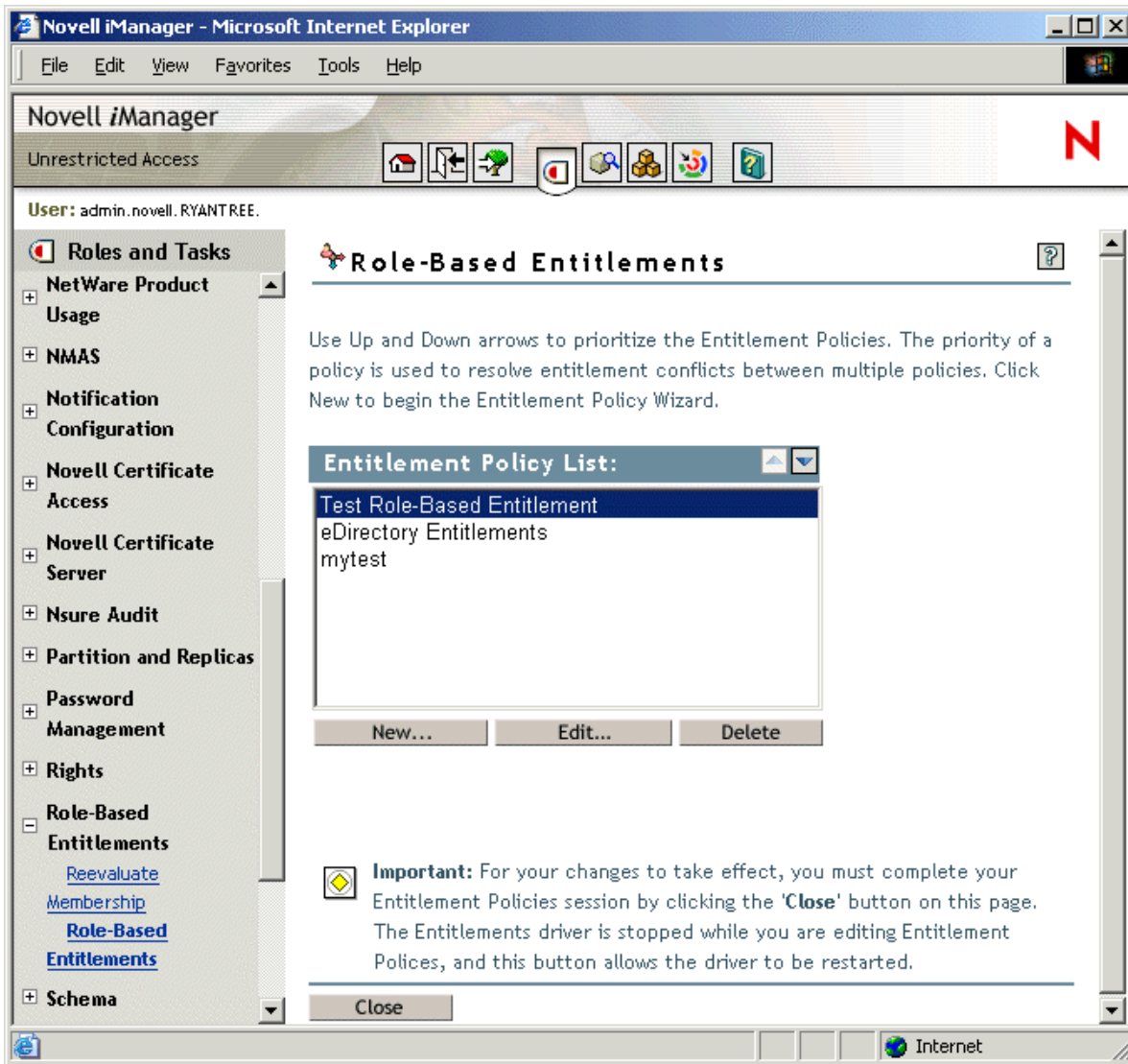
These driver configurations demonstrate just a sample of what you can do with Role-Based Entitlements. You can configure other connected system drivers, and other kinds of entitlements and interpretive variables.

Creating Entitlement Policies

- 1** Make sure you have set up the Entitlements Service Driver and created the driver configurations that are necessary.
- 2** In iManager, click the Role-Based Entitlements role > Role-Based Entitlements task.
- 3** Select a driver set.

Entitlement Policies are per driver set.

The list of existing Entitlement Policies opens, like the page in the following figure. If you are using Role-Based Entitlements for the first time, no policies are listed.



4 Click New.

The Create New Entitlement Policy Wizard opens.

5 Follow the steps in the wizard to create a new policy.

Refer to the online help for information about each step in the wizard.

Defining Membership for an Entitlement Policy

Like a DirXML driver, each Entitlement Policy can manage only objects that are in a master or read/write replica on the server to which is it assigned. Each Entitlement Policy is associated with a single Driver Set object, which is assigned to a particular server.

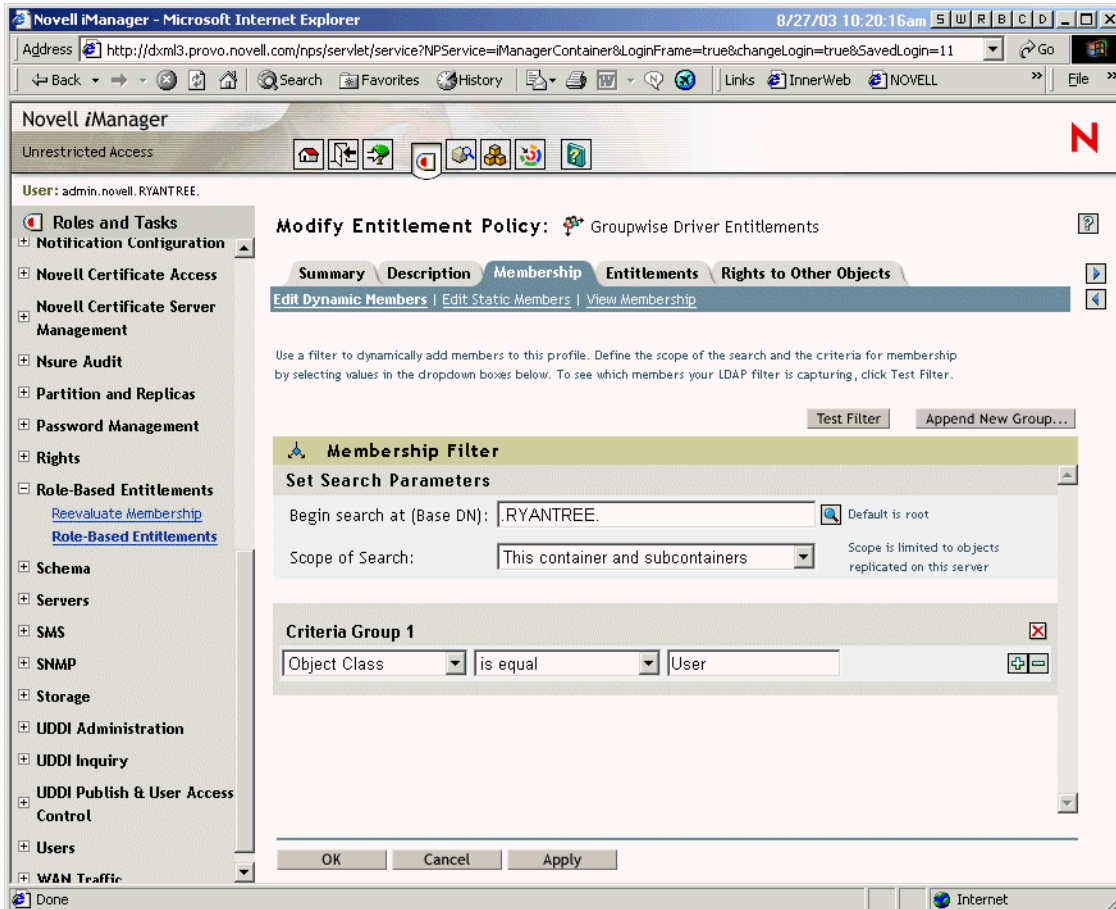
Only User objects (and other object types derived from the class of User) can be members of an Entitlement Policy.

An Entitlement Policy is a dynamic group object. You can define membership for an Entitlement Policy using two methods, dynamic and static. You can use both methods in the same Entitlement Policy.

- ♦ **Dynamic:** You can define criteria for membership based on values of attributes of the object, such as whether the job title includes the word “Manager.” The criteria you specify are converted into an LDAP filter.

Users who meet the criteria are automatically part of the Entitlement Policy, without requiring you to specifically add each user to the policy. The dynamic membership is the same as a Dynamic Group object.

If an object changes so that it no longer meets the criteria for dynamic membership, the entitlements are automatically revoked.



- ♦ **Static:** In addition to creating criteria for dynamic membership (an LDAP filter), you can include or exclude specific users.

You can add statically members who don't meet the criteria of the filter. You can exclude members who meet the filter's criteria but should not be included in the Entitlement Policy.

Choosing Entitlements for an Entitlement Policy

Role-Based Entitlements lets you grant entitlements on connected systems and rights in eDirectory.

Drivers that support Role-Based Entitlements offer a list of entitlements that can be assigned using an Entitlement Policy. The entitlements that the driver can provide are listed in the driver manifest, which is created by the driver developer to represent the capability of the driver and connected system. (The driver manifest should not be edited by an Identity Manager administrator.)

Trustee rights to objects in eDirectory are immediately granted to members of the Entitlement Policy. By default, entitlements in connected systems are granted to each member of the Entitlement Policy the next time an attribute used for Entitlement Policy membership is modified for that user, or when a user is moved to a different container or renamed.

Entitlements on connected systems can be any of the following:

- ◆ Accounts
- ◆ Membership in e-mail distribution lists
- ◆ Group membership in NOS lists
- ◆ Attributes for the corresponding objects in connected systems, populated with values you specify
- ◆ Other entitlements that you customize

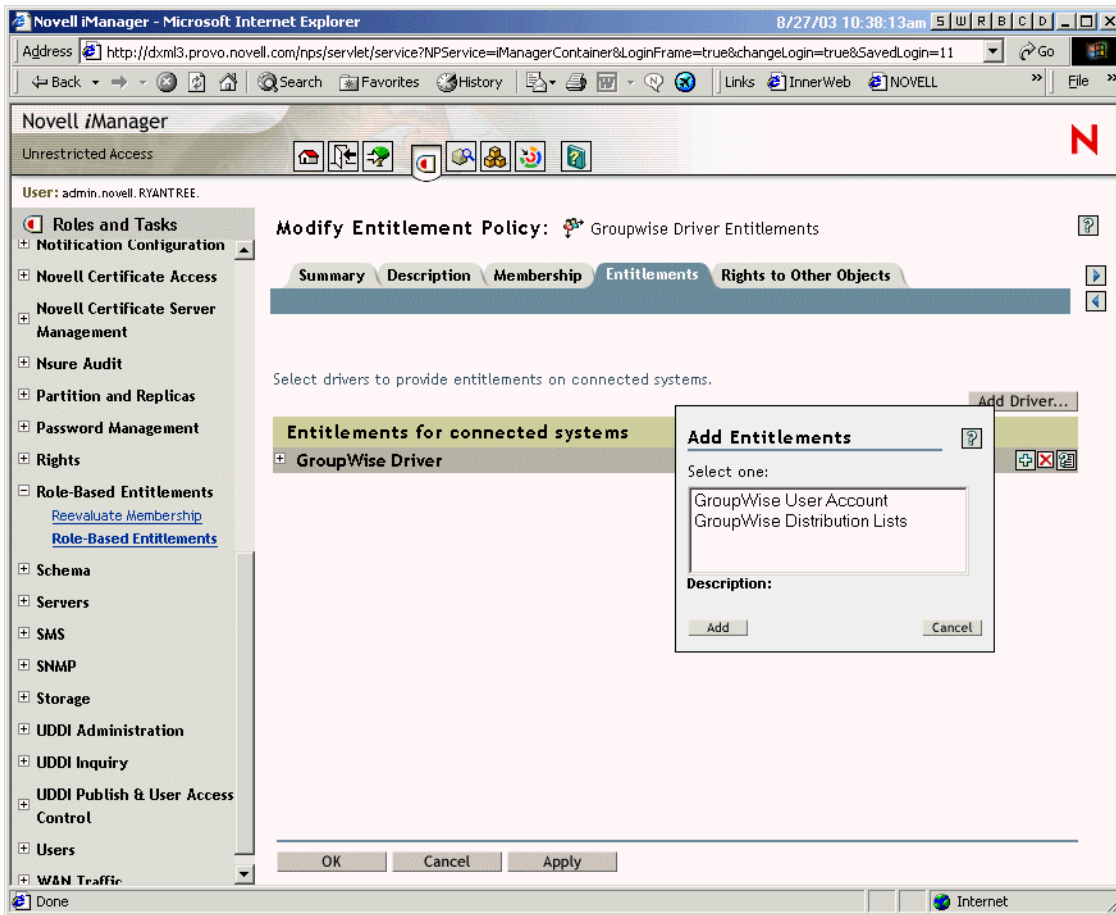
In this section:

- ◆ [“Accounts on Connected Systems” on page 190](#)
- ◆ [“Membership in E-Mail Distribution Lists and NOS Lists” on page 191](#)
- ◆ [“Attribute Values on Connected Systems” on page 193](#)

Accounts on Connected Systems

To add entitlements to an Entitlement Policy, go to the Entitlements page and select a driver. A pop-up window displays what entitlements that driver offers.

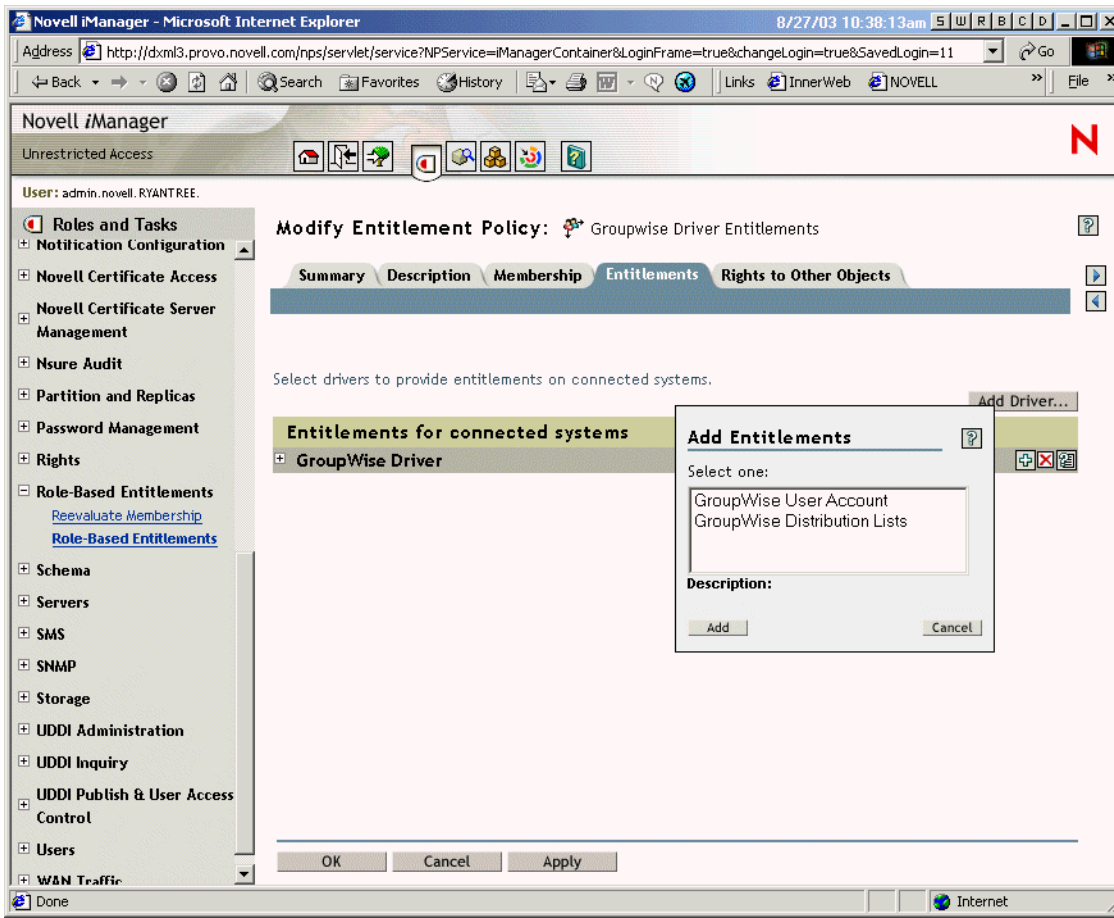
For example, in the following figure, you can see two kinds of entitlements being offered by a GroupWise driver, and the first one in the list is a GroupWise User Account.



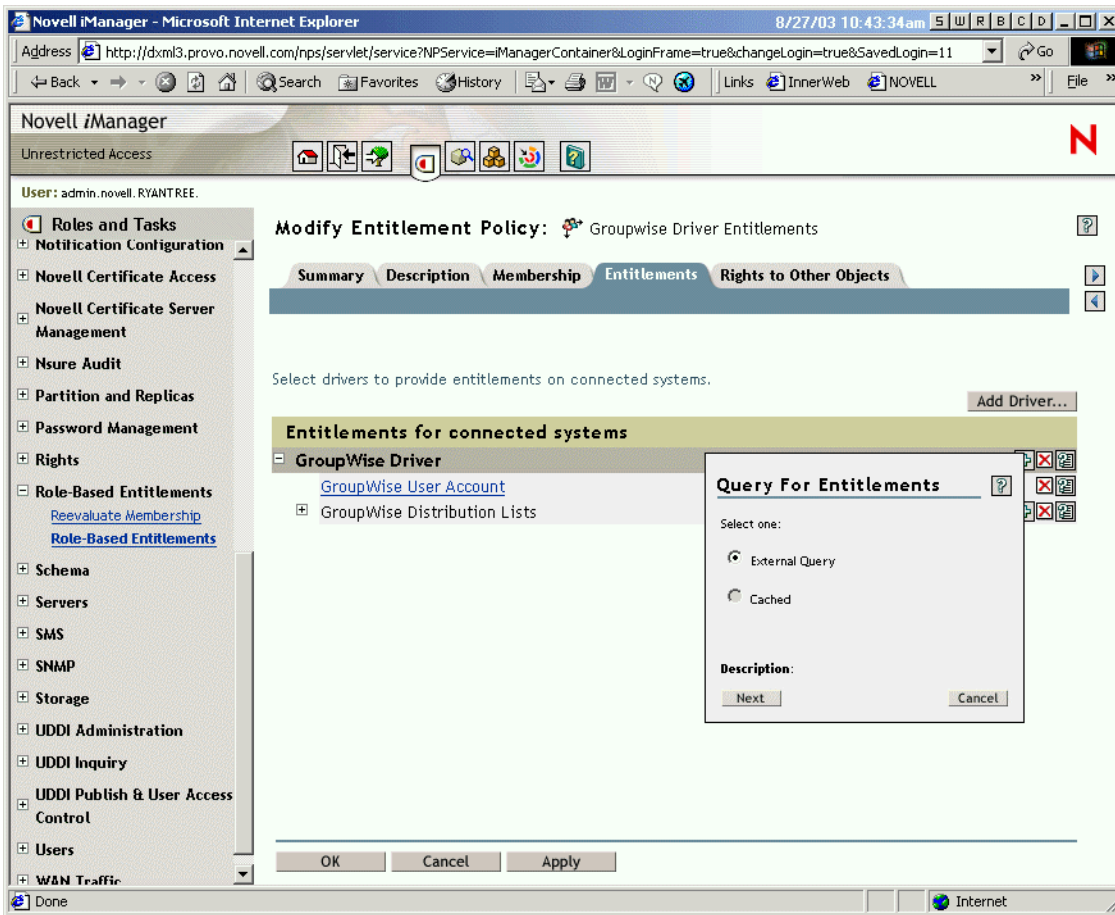
Membership in E-Mail Distribution Lists and NOS Lists

To assign membership in groups on connected systems, you choose the membership entitlement from the list of entitlements offered by a driver.

The following figure shows an example, with GroupWise Distribution Lists shown second in the list.



If you choose GroupWise Distribution Lists in this example, a query pop-up is displayed, like the example in the following figure.



The Entitlement Policy interface lets you query for the list of e-mail distribution lists or NOS lists. After a query has been performed, you can choose to view the cached list.

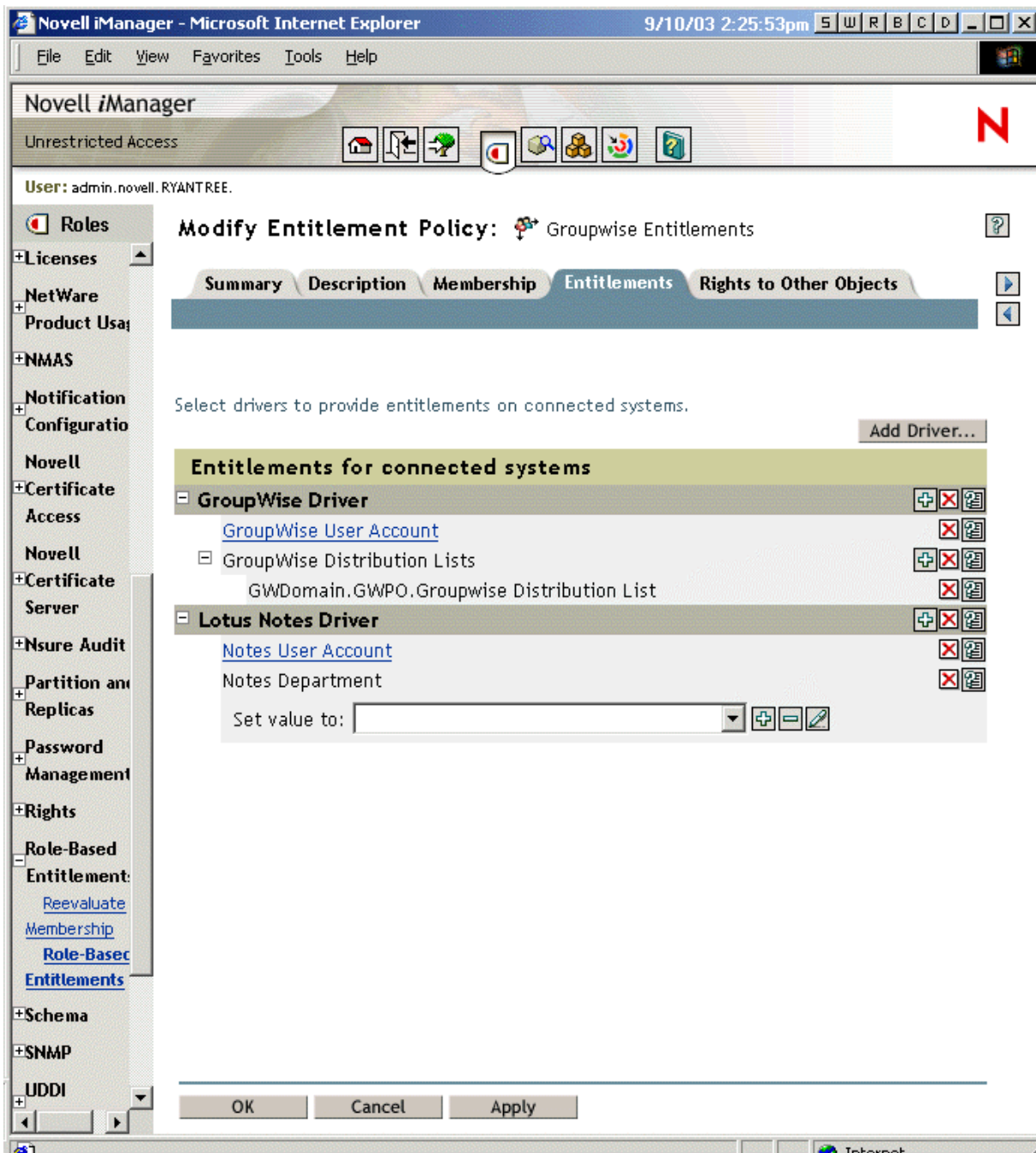
The drivers are configured to return the complete list, so you can choose from the lists that exist on the connected system.

NOTE: A driver could be customized to limit the list to group names you type in, rather than a query that returns the complete list.

Attribute Values on Connected Systems

You can assign attribute values for user accounts on connected systems. The interface provided lets you type in the value you want the user accounts to have.

The following figure shows an example of adding an attribute value for a Notes attribute, Department.



Keeping Accounts Safe

Role-Based Entitlements is designed to allow you to make sweeping changes to entitlements such as accounts, based on membership in the policy. This means, however, that mistakes made in changing policies could be a concern. The driver configurations that ship with Identity Manager use the most benign settings. You should understand which settings help to avoid loss of data.

The two kinds of settings that make the most difference are interpretive variables and conflict resolution. See [“Controlling Adding or Removing Entitlements” on page 195](#) and [“Conflict Resolution between Entitlement Policies” on page 196](#).

For example, Novell recommends that you never use delete as the value for the interpretive variable for removing an account. Role-Based Entitlements allows you to make major changes in your production environment without going through a test cycle, and it's possible you could make a mistake which would remove an account entitlement from someone without meaning to.

An administrator could safeguard data by making sure the interpretive variable for revoking accounts is set to disable instead of delete.

As another measure to protect your data, when you edit or create a new entitlement policy, the driver is turned off so that changes are not made while your editing of policies is incomplete. You can then manually restart the driver when you are finished, using the Restart button in the Entitlement Policies interface. Similarly, if another user appears to be editing Entitlement Policies, and you try to restart the driver using the Restart button, you are prompted not to restart the driver until the other user is finished making changes.

Controlling Adding or Removing Entitlements

You can control the consequences of granting or revoking an entitlement. Each driver provides a list of supported choices that control the meaning of “add” or “remove.”

For example, when adding a GroupWise account, you could specify that add actually means to grant the user an account in a disabled state, so that the administrator must intervene before the user can access the account. Or, you could choose to enable the account, which is the default.

By default, the driver configurations use the option that is most likely to preserve data. For example, by default, the meaning of remove for a GroupWise account is set to “disable,” to avoid unintentionally losing accounts if a mistake is made when the administrator is making changes to policies. As another example, the DirXML driver configurations don't remove entitlements that have values from a user account in another system. If a user is granted membership in an e-mail distribution list, and if later the user no longer meets the criteria for the Entitlement Policy, he or she is simply dropped from the policy membership. Accounts are disabled, but group membership and attribute values are not removed. A Identity Manager expert could customize the driver configurations if you wanted a different result.

The interpretation of removing an entitlement is especially important because Role-Based Entitlements functionality gives you the ability to make sweeping changes in an organization's entitlements in a production environment, without testing the results in a lab.

You can change the settings for interpreting add or remove by clicking the account entitlement on the Entitlements page in Entitlement Policy. The page that appears lets you edit the global configuration values, which are part of the driver parameters. Keep in mind that although you can edit the interpretation settings on the Entitlement page for an individual Entitlement Policy, the change affects all Entitlement Policies that grant that particular entitlement from that particular DirXML driver and connected system, not just the Entitlement Policy you were editing when you made the change. The settings are per entitlement and driver, not per Entitlement Policy.

See also [“Conflict Resolution between Entitlement Policies” on page 196](#).

In the Identity Manager 2 driver configurations, interpretive variables are used only on account entitlements. However, you could configure the driver to have interpretive variables for other types of entitlements.

NOTE: The actions that a driver supports are declared in the driver manifest. The manifest is created by the driver developer to represent the capability of the driver configuration. These options should not be edited by a network administrator. Changing the driver manifest alone does not cause the driver to support a new interpretation; the driver or connected system needs to be enhanced as well.

Conflict Resolution between Entitlement Policies

In this section:

- ♦ [“Overview” on page 196](#)
- ♦ [“Changing the Conflict Resolution Method for an Individual Entitlement” on page 197](#)
- ♦ [“Prioritizing Entitlement Policies” on page 198](#)

Overview

When you are creating Entitlement Policies, it’s possible that the policies that affect a particular user might conflict in assigning entitlements to that user.

Here’s how those conflicts are resolved. For some entitlements, you can change the conflict resolution.

- ♦ **Entitlements that don’t have values are additive.** In most cases an account is an entitlement that doesn’t have values. If a user is granted an account on a connected system by any Entitlement Policy, the user will receive an account on that system. It does not matter whether another Entitlement Policy conflicts; the result is additive.

This is always true; the method of conflict resolution for granting accounts cannot be changed.

One metaphor for entitlements that don’t have values is a light switch; it’s either “on” or “off,” granted or not granted.

For example, if the Manager Entitlement Policy grants Jean Chandler an Exchange account, but Jean Chandler is excluded from the Mail Room Employees Entitlement Policy that also grants Exchange accounts, Jean still gets an Exchange account.

- ♦ **Entitlements that have values are additive by default, but you can choose to resolve by priority.** These are entitlements such as group membership with a list of group names for the values, or an attribute with a value. By default, these kinds of entitlements are also additive.

You can change the conflict resolution for these kinds of entitlements, if desired.

The setting that governs conflict resolution for each entitlement type is in the driver manifest for a driver. Each kind of entitlement that a driver offers is listed separately in the manifest. Entitlements that have values have a conflict-resolution attribute, which is set for each entitlement independently. The default setting is `conflict-resolution="union"`. The other possible value is `conflict-resolution="priority"`.

- ♦ **conflict-resolution="union"** — A value of “union” means the entitlements are additive. A user is granted all the entitlements that he or she is assigned by membership in any policy. The differing entitlement values are simply added together and the user gets them all.

For example, if Jameel is a member of the Trade Show Contractors Policy that grants membership in a GroupWise e-mail distribution list named Trade Show Mailing List, and he is excluded from membership in the Trade Show Managers Policy that also assigns the e-mail distribution list named Trade Show Mailing List, he will still receive membership in the e-mail distribution list.

As another example, if Consuela is granted membership in the AD group named Mailroom Staff by the Mailroom Policy, and also granted membership in the AD group named Emergency Response by the Emergency Volunteers Policy, she is granted membership in both groups in AD.

With this setting, the order of an Entitlement Policy in the list of policies is not important for the entitlement.

- ♦ **conflict-resolution="priority"** — By contrast, a value of “priority” means that if the values in two different policies conflict, or if one policy includes the user and another excludes the user, the entitlements granted to the user are only those in the Entitlement Policy that is listed higher in the list of Entitlement Policies.

The previous examples would have a different result with this setting.

In the example above for Jameel, if the GroupWise e-mail distribution list entitlement had a value of “priority,” and the Trade Show Managers Policy were higher in the list than the Trade Show Contractors Policy, Jameel would not be granted membership in the Trade Show Mailing List.

In the example above for Consuela, if the AD NOS group membership entitlement had a value of “priority,” and the Mailroom Policy were higher in the list than the Emergency Volunteers Policy, Consuela would be granted membership in only the Mailroom Staff group. She would not be granted membership in the Emergency Response group because the conflict resolution is by priority, not additive.

This functionality would be useful if, for example, you configured your environment to use Role-Based Entitlements to place users in a hierarchical structure on another system. You would want the user to be placed in either one place or another, not two places at the same time.

Keep in mind that the setting is independent for each entitlement offered by each driver.

As a general rule, if you use the “priority” setting, you should place administrator or manager policies higher in the list than policies for end users or individual contributors. You should put groups with narrower membership higher than groups with broader membership.

Changing the Conflict Resolution Method for an Individual Entitlement

- 1** In iManager, click DirXML Management > Overview, then select a driver set.
A page appears with a graphical representation of all the drivers in the driver set.
- 2** Stop the driver.
- 3** Click the driver icon for the driver that offers the entitlement you want to change.
A page appears showing icons for the driver’s policies and the driver.
- 4** Click the driver icon to open the driver parameters page.
- 5** Click Driver Manifest.

The driver manifest is displayed in XML, but it is dimmed because it is not in editable mode.

- 6** Check the check box for Enable XML editing.
- 7** In the XML, find the definition of the entitlement you want to change.

Here’s an example of the line you should look for:

```
<entitlement conflict-resolution="union" description="Grants membership  
to GroupWise Distribution lists" display-name="GroupWise Distribution  
Lists" name="gwDistLists">
```

- 8** Change the conflict-resolution value. The two possible values are the following:

`conflict-resolution="union"`

`conflict-resolution="priority"`

- 9 Restart the Entitlements Service driver.

Prioritizing Entitlement Policies

By default, the order of the list of Entitlement Policies does not matter. This is because the driver configurations shipped with Identity Manager 2 have `conflict-resolution="union"` as the method of conflict resolution for each entitlement.

If you change any of the entitlements to `conflict-resolution="priority"`, then the order of the list of Entitlement Policies matters, but only for those entitlements you changed.

You change the order of the Entitlement Policies by using the arrow buttons next to the list of Entitlement Policies. The policy first in the list is the highest priority.

- 1 In iManager, click Role-Based Entitlements > Role-Based Entitlements.

- 2 Search for and select a driver set.

A page appears with a list of the Entitlement Policies.

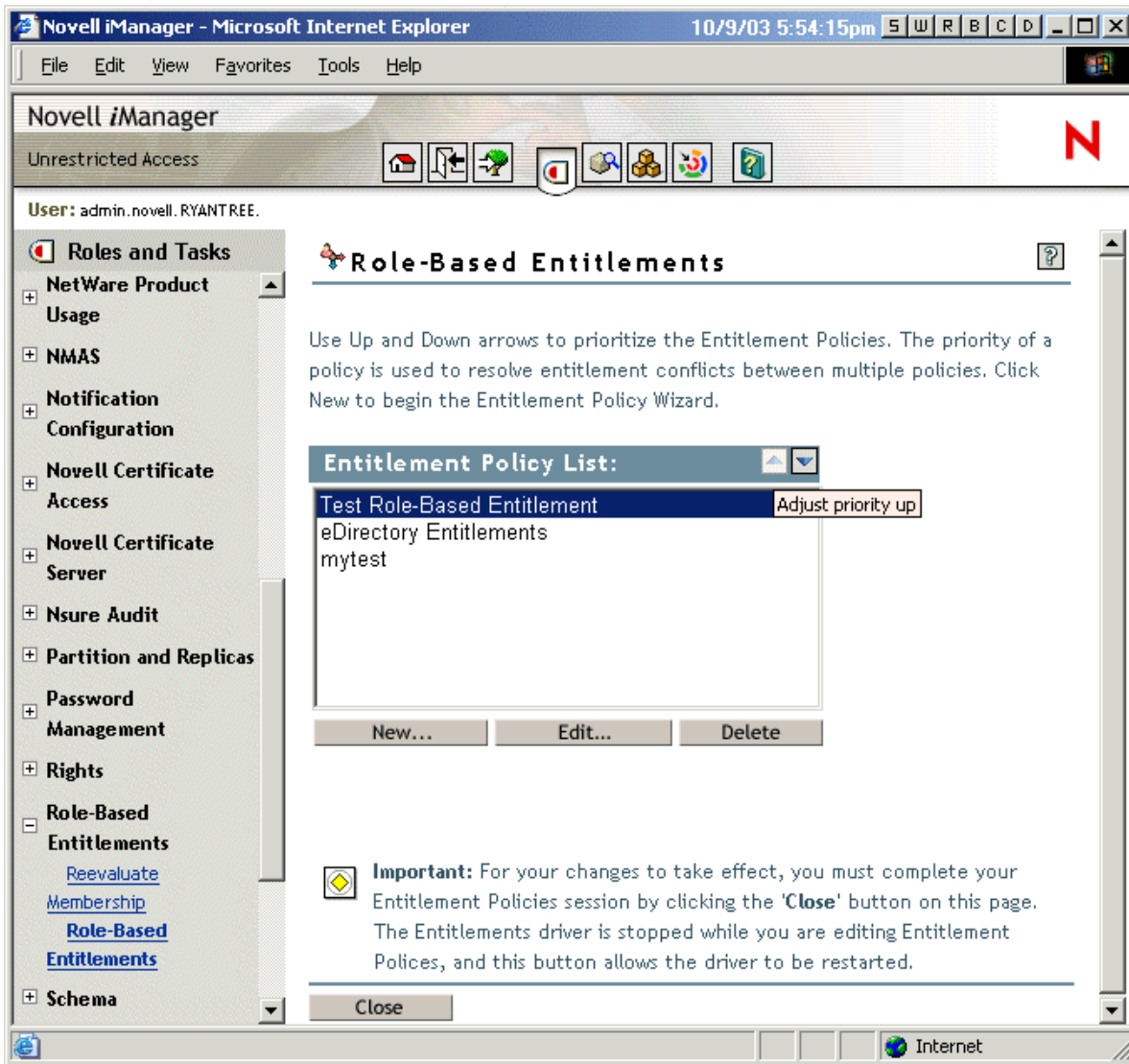
- 3 Change the priority of the Entitlement Policies by using the arrow buttons to move the policies up and down in the list.

Moving an Entitlement Policy higher in the list gives it a higher priority.

- 4 Click the Close button to restart the driver.

Changes in priority don't take effect until the driver is restarted.

See the figure below for an example of the policy list page showing the arrow buttons.



Password Synchronization and Role-Based Entitlements

Password Synchronization is managed the same way for drivers that are using Role-Based Entitlements as it is for other drivers, as described in [Chapter 9, “Password Synchronization across Connected Systems,”](#) on page 113.

Troubleshooting Role-Based Entitlements

When troubleshooting, keep in mind these issues:

- ♦ When you make any changes to policies by clicking New, Edit, or Remove on the page where the policies are listed, the Entitlements Service Driver is stopped. The driver is not restarted unless you click the Close button on that page.

This feature prevents the driver from granting or revoking entitlements in your production environment while your changes to policies are incomplete.

- ◆ Similarly, the Entitlements Service Driver won't start if more than one person appears to be editing Entitlement Policies at the same time.
- ◆ The Entitlements Service Driver won't start if the driver object is associated with more than one server. This configuration is not supported.
- ◆ The Entitlement Policy grants entitlements on connected systems through a DirXML driver, and the driver is identified by the GUID of the driver object in eDirectory, not the name of the object. This means that if you replace a driver object with another driver object of the same name, the Entitlement Policy won't work for entitlements on that driver because the object has a new GUID.
- ◆ Because one Entitlements Service Driver is used per driver set, an Entitlement Policy can manage only users that are in a read/write or master replica on the server that is associated with that driver set.

11

Managing Engine Services

Some drivers are used only for DirXML[®] Engine services, not for external connected systems.

In this section:

- ♦ [“Entitlements Service Driver” on page 201](#)
- ♦ [“Loopback Service Driver: Facilitating Moving Objects Using the Move Proxy Service” on page 201](#)
- ♦ [“Manual Task Service Driver \(Workflow Service Request Driver\)” on page 204](#)

Entitlements Service Driver

See [Chapter 10, “Using Role-Based Entitlements,” on page 183](#).

Loopback Service Driver: Facilitating Moving Objects Using the Move Proxy Service

DirXML[®] drivers can synchronize objects that are replicated on the same server, in either a master or read/write replica. One of the things a driver can do is to move objects from one container to another. For example, you can set up a driver to place users in Novell[®] eDirectory[™] based on the organization they are assigned to in a human resources application. When the organization for a user is changed in the human resources application, the driver can move the eDirectory user object to the corresponding container.

If you want a driver to be able to move objects from one container to another, you need to do one of the following:

- ♦ Place the driver on a server which holds master replicas of all the source or destination containers.
- ♦ Place the driver on a server with read/write replicas, and set up the Move Proxy service on the servers with the master replicas, to facilitate moving objects. Then configure the driver to delegate moves to the Move Proxy service.

The Move Proxy service is a particular configuration that you can run with the Loopback Service Driver shim. This section explains the Move Proxy service and how to set it up and configure other connected system drivers to take advantage of the service.

In this section:

- ♦ [“Understanding the Move Proxy Service” on page 202](#)
- ♦ [“Setting Up the Move Proxy Service” on page 202](#)
- ♦ [“Configuring Other Drivers To Delegate Moves to the Move Proxy Service” on page 203](#)

Understanding the Move Proxy Service

With Identity Manager and eDirectory, moving an object is best done on the master replica, especially if any other modifications are being made to the object at the same time.

If you want a driver to be able to move objects from one container to another, you need to do one of the following:

- ◆ Place the driver on a server which holds master replicas of all the source or destination containers.
- ◆ Place the driver on a server with read/write replicas, and set up the Move Proxy service on the servers with the master replicas, to facilitate moving objects. Then configure the driver to delegate moves to the Move Proxy service.

The Move Proxy service is a Driver object with a special configuration that you run on the server with the master replica. The purpose of the Move Proxy service is to move objects on behalf of DirXML drivers that are running on servers that hold read/write replicas. Delegation of the move allows object modifications that were performed by the delegating driver to replicate to the master server before the move is performed.

The following steps take place when a move is delegated from a driver to the Move Proxy service:

1. A driver delegates the move by setting a value for the moveProxyTrigger attribute of the object that needs to be moved. The driver sets the moveProxyTrigger attribute to the DN of the destination container to which the object should be moved.
2. The Move Proxy service monitors "add value" events for the moveProxyTrigger attribute, and converts the events into a custom commands that specify the source DN of the object to be moved and the DN of the destination container.

The custom command is created by the Subscriber Event Transformation Rule of the Move Proxy service driver.

3. The Move Proxy service driver initiates the actual object move on its Publisher channel. Then the Move Proxy service driver removes the destination DN value from the object moveProxyTrigger attribute.

If the move fails with a "retry" status (usually because a previous move of the same object has not yet completed), the status is returned to Identity Manager via the Subscriber channel. Identity Manager will resubmit the original event every 30 seconds or until the move succeeds or fails for other reasons.

Setting Up the Move Proxy Service

Set up the Move Proxy service on the server that holds the master replica. For an overview of when you might need this service, see [“Understanding the Move Proxy Service” on page 202](#).

After completing this procedure, configure drivers that are running on other servers to delegate their moves to the Move Proxy driver, so the moves can be performed on the master replica.

- 1** Install Identity Manager on the server with the master replicas, if it is not already installed.
- 2** Confirm that the following files for the Move Proxy service have been installed with Identity Manager. If they have not been installed, obtain them from your product distribution or from [Novell Support \(http://support.novell.com/cgi-bin/search/searchtid.cgi?/2964749.htm\)](http://support.novell.com/cgi-bin/search/searchtid.cgi?/2964749.htm).
 - ◆ loopback.jar

This is the Loopback Driver shim file needed to run the Move Proxy service. This driver shim file should be placed in the /lib directory of the respective OS.

- ♦ moveproxy.xml

This is the driver configuration file. If it is not placed in the default location where the other driver configuration files are located, you will have to browse to it when you are creating the Driver object in [Step 4](#).

- ♦ moveproxy.xlf

This file creates the prompts you see when importing the driver configuration moveproxy.xml.

- ♦ mvproxy_client_publisher_command_transformation.xsl

This file provides the Command Transformation style sheet that you add to each driver that delegates moves to the Move Proxy service, as explained in [“Configuring Other Drivers To Delegate Moves to the Move Proxy Service” on page 203](#).

- 3 Confirm that your eDirectory schema includes the attribute named DirXML-moveProxyTrigger. If it does not, extend the eDirectory schema using the mvproxy.sch file and the appropriate utility depending on your platform (nwconfig on NetWare, install.dlm on Win32, and ndssch on UNIX).

Obtain the mvproxy.sch file from [Novell Support \(http://support.novell.com/cgi-bin/search/searchtid.cgi?/2964749.htm\)](http://support.novell.com/cgi-bin/search/searchtid.cgi?/2964749.htm).

NOTE: If the schema already includes the DirXML-moveProxyTrigger attribute, your product distribution also should contain the files listed in [Step 2](#). If your schema does not include the attribute, and you obtain the mvproxy.sch schema extension file and the other files listed in [Step 2](#) from Novell Support, keep in mind that the files from Novell Support use an attribute named moveProxyTrigger instead of DirXML-moveProxyTrigger. The setup is the same although the attribute name is slightly different.

- 4 Create a new DirXML Driver object for the server holding the master replica, importing moveproxy.xml to create the driver configuration.

The DirXML Engine runs this Driver object using the Loopback Driver shim.

- 5 For the new Driver object you just created, edit the Subscriber and Publisher filters to include the object classes for which you want moves to be proxied. Then add the DirXML-moveProxyTrigger (or moveProxyTrigger) attribute to the filter for each of those classes.

Do not add any other attributes for the classes in the filters.

- 6 Set the desired Driver Startup Option for the Driver object, and start the driver.

After the driver has been configured and is operating correctly, Automatic is the preferred Driver Startup Option.

- 7 Make sure the drivers on other servers are set up to take advantage of the Move Proxy service, by setting them up as clients of the Move Proxy service, as explained in [“Configuring Other Drivers To Delegate Moves to the Move Proxy Service” on page 203](#).

Configuring Other Drivers To Delegate Moves to the Move Proxy Service

For an overview of when you might need this service, see [“Understanding the Move Proxy Service” on page 202](#).

- 1 Make sure you have completed [“Setting Up the Move Proxy Service” on page 202](#).
- 2 Create a DirXML-Stylesheet object in the driver’s DirXML-Publisher object.

- 3** Confirm that the file named `mvproxy_client_publisher_command_transformation.xsl` has been installed with Identity Manager.

This style sheet is one of the Move Proxy files that you checked for in [Step 2 on page 202](#). If it has not been installed, obtain it from your product distribution or from [Novell Support \(http://support.novell.com/cgi-bin/search/searchtid.cgi?/2964749.htm\)](http://support.novell.com/cgi-bin/search/searchtid.cgi?/2964749.htm).

- 4** In the new Stylesheet object you just created, go to the Edit XML property page and paste in the contents of the file named `mvproxy_client_publisher_command_transformation.xsl`.
- 5** Make the style sheet part of the Command Transformation Rule by doing one of the following:
 - ♦ If the DirXML-Publisher object does not have a Command Transformation Rule, make the new style sheet the Command Transformation Rule.
 - ♦ If the DirXML-Publisher object already has a Command Transformation Rule, use rule chaining and set the next transformation of the existing Command Transformation Rule to be the new style sheet.
- 6** If any style sheets used in either the Subscriber or Publisher channel are generating and sending moves to eDirectory via the `srcCommandProcessor` or `destCommandProcessor`, modify those style sheets to send a modify event similar to the one generated by the new Command Transformation Rule you created in [Step 5](#).

Manual Task Service Driver (Workflow Service Request Driver)

To better represent the use of the driver, in Identity Manager the name is changed from Workflow Request Service Driver to Manual Task Service Driver.

The Manual Task Service Driver is designed to notify one or more users that a data event has occurred and whether any action is required on the users' part. In an employee provisioning scenario, the data event might be the creation of a new User object and the user action might include assigning an office number by entering data into Novell eDirectory or by entering data in an application. Other scenarios include notifying an administrator that a new user object has been created, notifying an administrator that a user has changed data on an object, etc.

Configuring the Manual Task Service Driver usually consists of configuring two separate but related subsystems: the Subscriber channel rules and e-mail templates, and the Publisher channel Web server templates and rules.

In addition, driver parameters such as SMTP server name, Web server port number, etc., must be configured.

For more information, see the *Manual Task Service Driver Implementation Guide* (<http://www.novell.com/documentation/lg/dirxml/drivers/index.html>).

12 Logging Events Using Nsure Audit

Nsure™ Identity Manager is instrumented to use Novell® Nsure Audit for auditing and reporting.

Nsure Audit is a collection of technologies providing monitoring, logging, reporting and notification capabilities. Through integration with Nsure Audit, Identity Manager provides detailed information about the current and historical status of driver and engine activity. This information is provided by a set of preconfigured reports, standard notification services, and user-defined data logging.

You can to monitor real-time Identity Manager events, send e-mail notifications for any Identity Manager event, and generate reports of Identity Manager activity through Nsure Audit.

The types of messages sent to Nsure Audit are controlled using plug-ins similar to those provided with the Reporting and Notification Service (RNS). Additional levels are added to these plug-ins, to select the type of operations or debug information you would like to track, such as status, add entry, search, and so on.

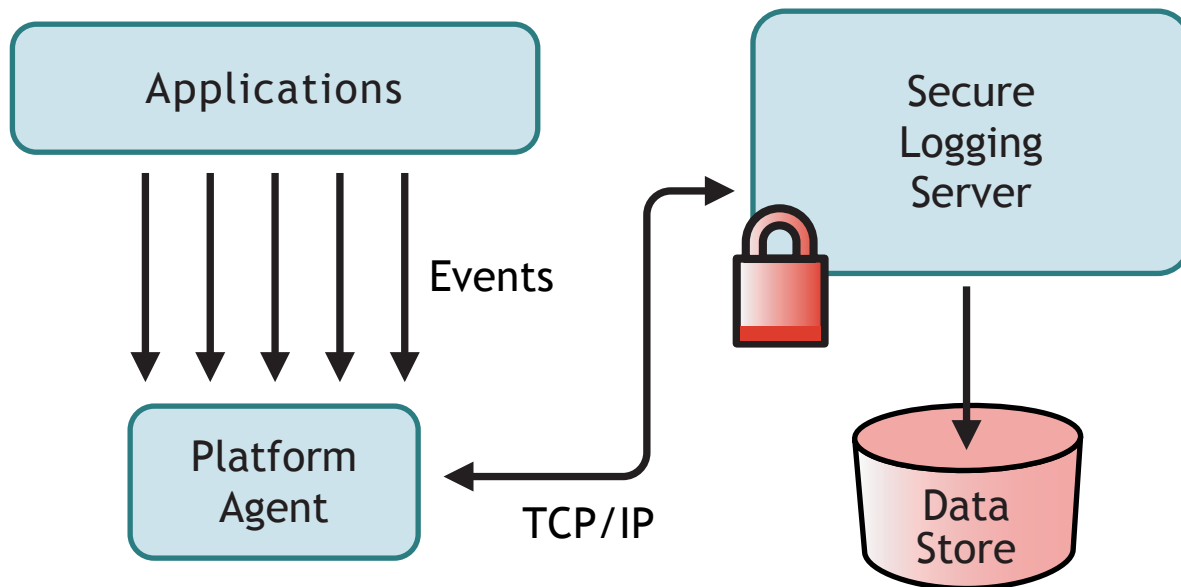
Reporting and Notification Service

The Identity Manager engine that ships with Identity Manager is backward compatible with the Reporting and Notification Service (RNS). The engine continues to process RNS functions if you are currently using RNS. However, we recommend that you move to Nsure Audit because it provides the functionality of RNS with the addition of many new features. RNS can process only status events. For RNS documentation, see the *DirXML 1.1a Administration Guide* (<http://www.novell.com/documentation/lg/dirxml11a/dirxml/data/afae8bz.html>).

Overview

Nsure Audit is a centralized, cross-platform logging service that can log data from multiple applications to a centralized data store. After event data is logged, you can run detailed reports, custom queries, and trigger notifications based on the logged events.

The following figure illustrates the high-level architecture of Nsure Audit:



In this illustration, Identity Manager is one of the applications which uses the Platform Agent to report events to the Nsure Audit Secure Logging Server.

Setting Up Nsure Audit

As described in the Overview, Nsure Audit consists of two fundamental components:

- ♦ Platform Agent
- ♦ Secure Logging Server

The Platform Agent is the component that runs with Identity Manager to communicate events to the Secure Logging Server. It is installed with Identity Manager. The Secure Logging Server is the component that receives event data from Identity Manager and other applications, and is installed separately from Identity Manager as part of Nsure Audit 1.0.1.

Setting Up the Platform Agent

The platform agent is installed by selecting the Novell Nsure Audit System Components for DirXML option during install. The platform agent can be installed with Identity Manager, or it can be installed later.

NOTE: If you install the Platform Agent after the DirXML engine has been started, Identity Manager must be restarted before the Platform Agent and Identity Manager are linked. Identity Manager attempts to connect to the Platform Agent during startup only.

Once the platform agent is installed, complete the following steps to configure the Platform Agent:

- 1 Open the Nsure Audit configuration file, `logevent.cfg`, in a text editor. The default location for this file is:

Operating System	Path
NetWare®	<code>sys:\etc\logevent.cfg</code>
Windows	<code>windows_directory\logevent.cfg</code>

Operating System	Path
Linux\Solaris	/etc/logevent.conf

- 2** Change the value of the *LogHost* parameter to the IP address or DNS name of your Secure Logging Server.
- 3** Restart Identity Manager.

Setting Up the Secure Logging Server

NOTE: The Nsure Audit Secure Logging Server is not included with DirXML. The Secure Logging Server is part of Nsure Audit 1.0.1. For information on downloading Nsure Audit 1.0.1, see the [Nsure Audit Product Page](http://www.novell.com/products/nsureaudit) (<http://www.novell.com/products/nsureaudit>).

The Secure Logging Server runs on NetWare 5.1 or later, Windows NT 4.0 or 2000, Solaris 8 or 9, and several versions of Linux, including SUSE Enterprise Linux Server 8.

The Secure Logging Server can log events to MySQL*, Oracle*, Java* Applications, and several other locations, including a flat file. Nsure Audit includes a custom application designed to query databases for event data, called Nsure Audit Report. A datastore that has an ODBC connector is required to use this advanced reporting tool.

A Quick Start Guide containing Secure Logging Server setup instructions is available for each platform, and is included with the Nsure Audit 1.0.1 installation. The Quick Start guides are also viewable on the Web along with the *Nsure Audit 1.0.1 Administration Guide* on the [Novell Nsure Audit Documentation Web site](http://www.novell.com/documentation/lg/nsureaudit) (<http://www.novell.com/documentation/lg/nsureaudit>).

Logging Configuration

Identity Manager enables you to configure the events that are logged using several predefined levels, or by individually selecting each event you want to log. Changes to the configuration settings are also logged.

User-defined events, discussed in “[User-Defined Events](#)” on [page 209](#), are logged any time logging is enabled, and are never filtered by the DirXML engine.


Logging is configured on a driver set or on an individual driver. Drivers can inherit logging configuration from the driver set. For information on the eDirectory attributes containing log information, see “[eDirectory Objects](#)” on [page 211](#).

By default, only critical and user-defined events are logged.

Selecting Events to Log

On the Driver Set:

- 1** In iManager, open the DirXML Driver Management role, then select the Overview task.
- 2** Click the Driver Set name link. The Modify Object window appears.
- 3** Click the log Level link on the DirXML tab. The following logging options are available:

Option	Description
Log Errors	This is the default log level. This option logs all events with an error status, and also user-defined events.
Log Errors and Warnings	This option logs all events with an error or warning status, and also user-defined events.
Log Specific Events	This option enables you to select specific events to log from a list. Click the  icon to select events. User-defined events are always logged.
Only Update the Last Log Time	Only user-defined events are logged. When an event occurs, the last log time is updated, so you can view the time and date of the last error in the status log.
Logging Off	Only user-defined events are logged.
Maximum Number of Entries in the Log	This setting allows you to specify the maximum number of entries to log in the status logs. See “Viewing Status Logs” on page 213 for details.

4 After you have selected the events you would like to log, click OK.


On the Driver:

- 1** In iManager, open the DirXML Driver Management role, then select the Overview Task.
- 2** Click the driver status icon, then select Edit properties.
- 3** Click the Log Level link on the DirXML tab. By default, the driver is configured to inherit log settings from the driver set. To select logged events for this driver only, uncheck the following:

☒ Use log settings from the DriverSet, DS.Novell

The following log settings are from the DriverSet and cannot be changed on this page. To modify the DriverSet's settings, [click here](#).

4 The following logging options are available:

Option	Description
Log Errors	This is the default log level. This option logs all events with an error status, and also user-defined events.
Log Errors and Warnings	This option logs all events with an error or warning status, and also user-defined events.
Log Specific Events	This option enables you to select specific events to log from a list. Click the  icon to select events. User-defined events are always logged.
Only Update the Last Log Time	Only user-defined events are logged. When an event occurs, the last log time is updated, so you can view the time and date of the last error in the status log.
Logging Off	Only user-defined events are logged.

Option	Description
Maximum Number of Entries in the Log	This setting allows you to specify the maximum number of entries to log in the status logs. See “Viewing Status Logs” on page 213 for details.

5 Once you have selected the events you would like to log, click OK.

Driver Start and Stop Events

Identity Manager generates an event whenever a driver starts or stops. The following table contains details on these events:

Event	Log Level	Information
EV_LOG_DRIVER_START (34)	LOG_INFO	To log driver starts, you must use the Log Specific Events option and select this event.
EV_LOG_DRIVER_STOP (35)	LOG_WARNING	To log driver stops, select Log Errors and Warnings, or use the Log Specific Events option and select this event.

For details on creating Nsure Audit notifications based on these events, see [“Sending Notifications Based on Events” on page 212](#).

Error and Warning Events

Identity Manager generates an event whenever an error or warning is encountered. The following table contains details on these events:

Event	Log Level	Information
DirXML_Error(40)	LOG_ERROR	<p>All DirXML errors log this event. The actual error code encountered is stored in the event.</p> <p>To log errors, select Log Errors, Log Errors and Warnings, or use the Log Specific Events option and select this event.</p>
DirXML_Warning(41)	LOG_WARNING	<p>All DirXML warnings log this event. The actual warning code encountered is stored in the event.</p> <p>To log driver stops, select Log Errors and Warnings, or use the Log Specific Events option and select this event.</p>

For details on creating Nsure Audit notifications based on these events, see [“Sending Notifications Based on Events” on page 212](#).

User-Defined Events

Identity Manager enables you to configure your own events to log to Nsure Audit. Events can be logged using an action in Policy Builder, or within a style sheet. Any information you have access to when defining policies can be logged.

Event IDs

Event IDs between 1000 and 1999 are allotted for user-defined events. You must specify a value within this range for the event ID when defining your own events. In Nsure Audit, this ID is combined with the DirXML application ID of 003.


Log Levels

Log levels enable you to group events based on the type of event being logged. The following pre-defined log levels are available:

Log Level	Description
log-emergency	Events that cause the DirXML engine or driver to shut down.
log-alert	Events that require immediate attention.
log-critical	Events that can cause parts of the DirXML engine or driver to malfunction.
log-error	Events describing errors that can be handled by the DirXML engine or driver.
log-warning	Negative events not representing a problem.
log-notice	Positive or negative events an administrator can use to understand or improve use and operation.
log-info	Positive events of any importance.
log-debug	Events of relevance for support or engineers to debug operation of the DirXML engine or driver.

Generating Events Using Policy Builder

In Policy Builder, events are logged by selecting the Generate Event action.

- 1 Select the condition to be met before the event is generated, then select the Generate Event action.
- 2 Specify an **event ID**.
- 3 Select a **log level**.
- 4 Click the  icon next to the Enter Strings field to launch the Named String Builder.
- 5 Use the Named String Builder to construct named strings corresponding to the custom data fields:

Strings			
<input type="checkbox"/> Name:*	text1	String tokens:*	Operation Attribute("Given Name") 
<input type="checkbox"/> Name:*	text2	String tokens:*	Operation() 
<input type="checkbox"/> Name:*	value	String tokens:*	"1000" 

- 6 Click OK to return to the Policy Builder to construct the remainder of your policy.

Generating Events Using Status Documents

Status documents generated through style sheets using the `<xsl:message>` element are sent to Nsure Audit with an event ID corresponding to the level attribute of the status document as specified in the following table:

Status Level	Status Event ID
Success	EV_LOG_STATUS_SUCCESS (1)
Retry	EV_LOG_STATUS_RETRY (2)
Warning	EV_LOG_STATUS_WARNING (3)
Error	EV_LOG_STATUS_ERROR (4)
Fatal	EV_LOG_STATUS_FATAL (5)
User Defined	EV_LOG_STATUS_OTHER (6)

The following example generates an Nsure Audit event 0x004 and value1=7777, with a level of EV_LOG_STATUS_ERROR:

```
<xsl:message>
  <status level="error" text1="This would be text1" value="7777">This data
would be in the blob and in text 2, since no value is specified for text2 in
the attributes.</status>
</xsl:message>
```

The following example generates an Nsure Audit event 0x004 and value1=7778, with a level of EV_LOG_STATUS_ERROR:

```
<xsl:message>
  <status level="error" text1="This would be text1" text2="This would be
text2" value="7778">This data would be in the blob only for this case, since
a value for text2 is specified in the attributes.</status>
</xsl:message>
```

eDirectory Objects

This section provides details on the eDirectory attributes that store log data. You do not need to modify these attributes directly, because these objects are automatically configured based on your selections in iManager.

The Identity Manager events you want to log are stored in the DirXML-LogEvent attribute on the Driver Set object or Driver object. The attribute is a multi-valued integer with each value identifying an event ID to be logged.

Before logging an event, the engine checks the current event type against the contents of this attribute to determine whether the event should be logged.

Previous versions of Identity Manager used the DirXML-DriverTraceLevel attribute to set up logging levels. The logging level was specified on each Driver object, and did not support inheritance. For Identity Manager 2, Driver objects can inherit this information from the Driver Set object.

The DirXML-DriverTraceLevel attribute of a driver object has the highest precedence when determining log settings. If a Driver object does not contain a DirXML-DriverTraceLevel attribute, the engine uses the log settings from the parent driver set object.

Viewing Events in Nsure Audit Report

Nsure 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).

Follow the instructions in the *Nsure Audit 1.0.1 Administration Guide* to set up Nsure Audit Report. This guide is available on the [Novell Nsure Audit Documentation Web site \(http://www.novell.com/documentation/lg/nsureaudit\)](http://www.novell.com/documentation/lg/nsureaudit)

After Audit Report is configured, continue with “[Importing the Identity Manager Application Schema](#)” on [page 212](#).

Importing the Identity Manager Application Schema

Each application that logs data to Nsure Audit defines events and specifies the contents of each event field in a log schema file (.lsc). By importing the Identity Manager log schema file, Audit Report can display event names rather than numbers, and provide text descriptions of each event field in logged events.

To import the Identity Manager log schema file:

- 1 In Audit Report, click File > Import > Application Schemata.
- 2 Specify the hostname of your Secure Logging Server, select a language, then click OK.
- 3 Restart Audit Report.

Viewing Identity Manager Events

- 1 In the Nsure Audit Report Workspace, click the Events tab, then expand the DirXML folder. This list contains all predefined DirXML events. Double-click any event in the list to view event properties.
- 2 To query for a DirXML event, right-click the event in the Workspace and select Define Query.
- 3 When the Query Expert appears, specify a time frame and verify the event.
- 4 To run this query, select the Query tab in the Workspace, right-click the query name, then select Run.

Queries can also be created using SQL statements. All DirXML events have an Event ID between 109608 and 262144.

Sending Notifications Based on Events

Nsure Audit provides the ability to send a notification when a specific event occurs, or does not occur. Notifications can be sent based on one or more events and any values contained within these events. Notifications can be sent to any logging channel, enabling you to log notifications to a database, a Java application or SNMP management system, or several other locations.

For information on creating notifications, see “[Configuring Filters and Event Notifications](http://www.novell.com/documentation/lg/nsureaudit/nsureaudit/data/al0lg08.html#al0lg08)” in the *Nsure Audit 1.0.1 Administration Guide* (<http://www.novell.com/documentation/lg/nsureaudit/nsureaudit/data/al0lg08.html#al0lg08>)

Using Status Logs

In addition to the functionality provided by Nsure Audit, Identity Manager logs a specified number of events on the Driver Set object and Driver object. These status logs provide a view of recent Identity Manager activity. Once the log reaches the set size, the oldest half of the log is permanently removed to clear room for more recent events. Therefore, any events you want to track over time should be logged to Nsure Audit or the Report and Notification Service.

Setting the Maximum Log Size

Status logs can be configured to hold between 50 and 500 events. This setting can be configured on the Driver Set object to be inherited by all drivers in the set, or configured for each driver in the set. The maximum log size operates independently of the events you have selected to log, so you can configure the events you want to log on the Driver Set, then specify a different log size for each driver in the set.

Setting the Log Size on the Driver Set:

- 1 in iManager, open the DirXML Driver Management role, then select the Overview Task.
- 2 Click the Driver Set name link. The Modify Object window appears.
- 3 Click the Log Level link on the DirXML tab. Specify the maximum log size in the Maximum number of entries in the log field:

Maximum number of entries in the log (50 - 500):

- 4 After you have specified the maximum number, click OK.


Setting the Log Size on the Driver:

- 1 In iManager, open the DirXML Driver Management Role and select the Overview task.
- 2 Click the driver status icon, then select Edit properties.
- 3 Click the Log Level link on the DirXML tab. Specify the maximum log size in the Maximum number of entries in the log field:

Maximum number of entries in the log (50 - 500):

- 4 After you have specified the maximum number, click OK.

Viewing Status Logs

Status log entries are represented in iManager with a status log icon . Anywhere you see this icon in iManager, you can view a short-term log. The following status logs are available:

- ♦ On the driver set.
- ♦ On the Publisher Channel for each driver in the set.
- ♦ On the Subscriber Channel for each driver in the set.

The status logs for the Publisher and Subscriber channels report channel-specific messages generated by the driver, such as an operation veto for an un-associated object.

The status log for the driver set contains only messages generated by the engine, such as state changes for any drivers in the driver set. All engine messages are logged.

A

Activating Novell Identity Manager Products

The following information explains how activation works for products based on Novell Nsure Identity Manager®. To activate your products you must:

- ♦ Generate a Product Activation Request
- ♦ Submit the Product Activation Request
- ♦ Install the Product Activation Credential received from Novell

The Identity Manager Suite (or Server) and driver groups must be activated within 90 days of installation, otherwise they will shut down. At any time during the 90 days, or afterward, you can choose to activate Identity Manager products.

If you are installing a Identity Manager product on multiple trees, as would be the case if you use the DirXML Driver for eDirectory, you must install a unique Product Activation Credential on each tree. You use the same license to get both Product Activation Credentials.

NOTE: Activating a driver does not change your current configuration or install a newer version of the driver shim. It simply changes the driver to an activated state.

Activation procedures are the same regardless of the Identity Manager products you purchase.

After you complete the activation procedures, you can view your current activations through Novell iManager. For more information, refer to “[Viewing Product Activations for Identity Manager and DirXML Drivers](#)” on page 219.

For more information about activation, refer to [Activation Basics \(http://www.novell.com/partners/partnerplace/epd/product_activation_basics.html\)](http://www.novell.com/partners/partnerplace/epd/product_activation_basics.html) and [Activation Troubleshooting \(http://www.novell.com/partners/partnerplace/epd/troubleshooting_activation.html\)](http://www.novell.com/partners/partnerplace/epd/troubleshooting_activation.html).

Generating a Product Activation Request

You will use your Customer ID to generate a Product Activation Request. When you purchase your Identity Manager product, Novell will send an e-mail to your company's primary contact (the person who purchased the product license) that includes a customer ID.

If you do not remember or do not receive your Customer ID, please call the Novell Activation Center at 1-800-418-8373 in the U.S. In all other locations, call 1-801-861-8373. (If applicable, you will be charged long distance fees for calls made using the 801 area code.)

NOTE: The individual who purchases the product license will receive an e-mail containing the Customer ID. If your company uses its purchasing agent to handle this transaction, you might need to check with this individual to obtain your Customer ID.

You should create a Driver Set object before you generate a Product Activation Request to activate Identity Manager.

- 1 Launch iManager by going to <http://serveripaddress/nps/iManager.html>.

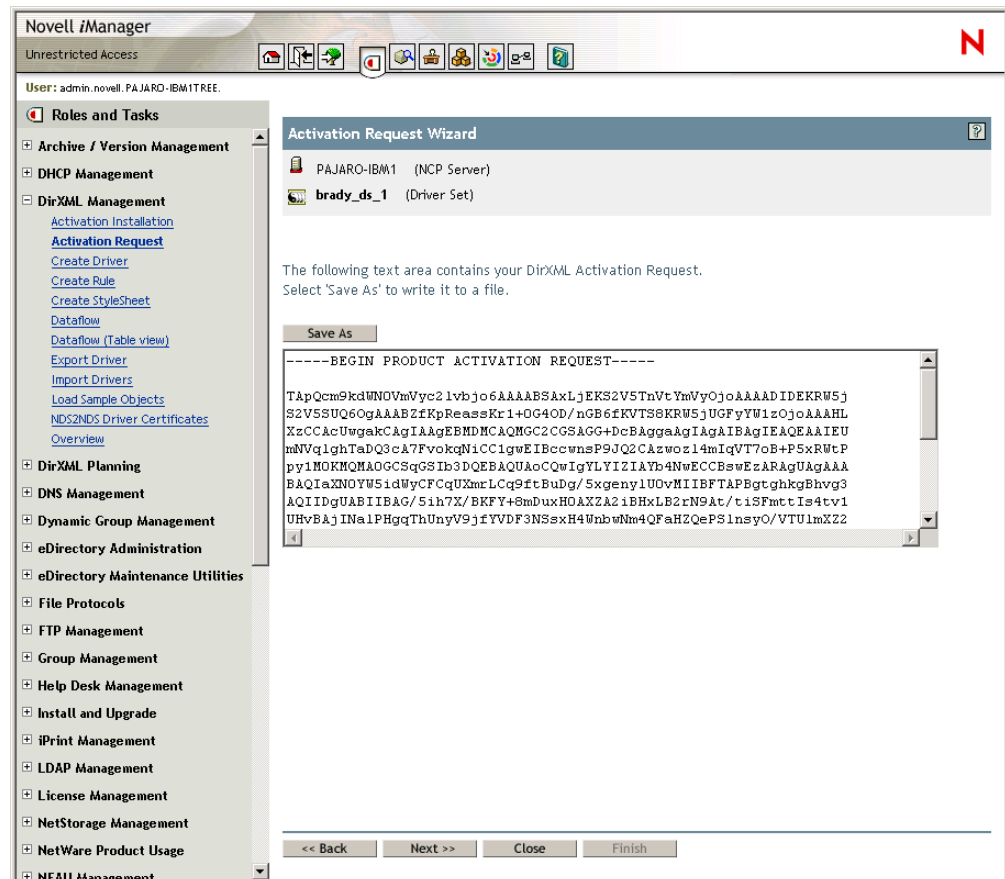
2 Click DirXML Management > Activation Request.

3 Browse to the driver set that you want to activate > click Next.

NOTE: If the driver set is not associated with a server or is associated with multiple servers, you will be prompted to select a server to associate with a driver set.

4 Enter your Novell Customer ID, then click Next to build your Activation Request file.

Your customer ID and identifying information about the server's tree are stored in the Product Activation Request.



5 Copy the Product Activation Request that is in the text area to the clipboard or save the request directly to a file, then click Next.

You will need this information later at the Novell Product Activation Web site.

IMPORTANT: Do not edit the content of the Product Activation Request.

6 Click the hyperlink to launch the [Novell Product Activation Web site \(http://www.novell.com/products/activation\)](http://www.novell.com/products/activation).

or

Click Finish to return to the main menu of iManager.

NOTE: To continue the activation process, you need to submit this Product Activation Request to Novell at the [Novell Product Activation Web site \(http://www.novell.com/products/activation\)](http://www.novell.com/products/activation). For information, see "Submitting a Product Activation Request" on page 217.

Submitting a Product Activation Request

After you create a Product Activation Request, you submit it to Novell through the [Novell Product Activation Web site \(http://www.novell.com/products/activation\)](http://www.novell.com/products/activation). Novell will then send an e-mail containing a Product Activation Credential. Use this credential to activate the suite or driver groups.

- 1 Go the [Product Activation Web site \(http://www.novell.com/products/activation\)](http://www.novell.com/products/activation) site, then click the Identity Manager product(s).
- 2 Follow through the introductory screens, then when prompted, log in to your MyNovell account.

You must have a MyNovell account to access the Product Activation Web site. If you don't already have an account, you can create this free account when you visit the Product Activation site.

- 3 Click Browse to specify the path to the Product Activation Request file or paste the text of the Product Activation Request into the text area.

If you copied the Product Activation Request to a diskette, make sure you have the request available on the computer you are working on.

IMPORTANT: Do not edit the content of the Product Activation Request.

- 4 Click Submit.

Your product purchases available for activation are displayed.

Novell

choose a language | HOW TO BUY | DOWNLOAD | SEARCH

products a-z
product activation
evaluation software
support and training

Novell Product Activator

Company Information
Company Name: ACME Design
Address: 1800 SOUTH NOVELL
PROVO, UT 84606
US
Contact Email: jnoble@acme.com
If the company information is incorrect, please contact activator@novell.com

New Activation Request
Your following purchases of Novell product licenses have been found in Novell's Orders Database. The figures under the heading "Licensed Units" are calculated based on the license model of the relevant product; for example, the purchase of 2 licenses for a 250-user product would result in 500 "Licensed Units." The information in the "Licensed Units" column is only for the specific license purchase and is not a cumulative total of all license purchases that have been made over time.

Because for certain Novell products you may be allowed to activate the product more than once, a record of previous activations is provided below; for example, for some products, you may obtain a unique Product Activation Credential for each tree in which the product is installed. Keep in mind that you may rightfully use Product Activation Credentials only to activate products to the extent you have obtained valid license rights in sufficient quantities.

Please select the purchases for which you want to obtain a Product Activation Certificate.

Product Description	Date Purchased	Quantity Purchased	Licensed Units	Previous Activations
<input checked="" type="checkbox"/> DirXML Starter Pack -DirXML Engine -DirXML Driver for eDirectory -DirXML Driver for Active Directory and Exchange 2000 -DirXML Driver for NT Domain	05/06/2003	1	1	5 (details)

Submit

Legal | Privacy | Accessibility | Subscribe | Feedback | © 2003 Novell, Inc. All Rights Reserved. | powered by N

- 5 Mark the product purchase you are activating.

You can activate only one purchase at a time. Mark the purchase you are currently activating. If you need to activate any of the other products listed and they will be used in the same tree, submit the Product Activation Request again. If they will be used in a different tree, you must create a new Product Activation Request and submit that request to obtain a credential.

6 Click Submit.

Novell generates a Product Activation Credential based on the Product Activation Request you submitted and sends that credential to you via e-mail. A copy of the credential will be sent to the primary contact as well.

NOTE: Some companies limit the list of employees authorized to receive credentials. You might not have rights to use the customer ID. In this case, after you click Submit, a notification is sent to the primary contact. The primary contact must approve your usage of the customer ID before you will receive the credential by e-mail.

Installing a Product Activation Credential

You should install the Product Activation Credential via iManager. The following procedures explain how to install the Product Activation Credential.

1 Open the Novell e-mail that contains the Product Activation Credential.

2 Do one of these steps:

- ◆ Save the Product Activation Credential file.
- or
- ◆ Open the Product Activation Credential file > copy the contents of the Product Activation Credential to your clipboard.

IMPORTANT: Do not edit the contents of the Product Activation Credential.

3 Open iManager.

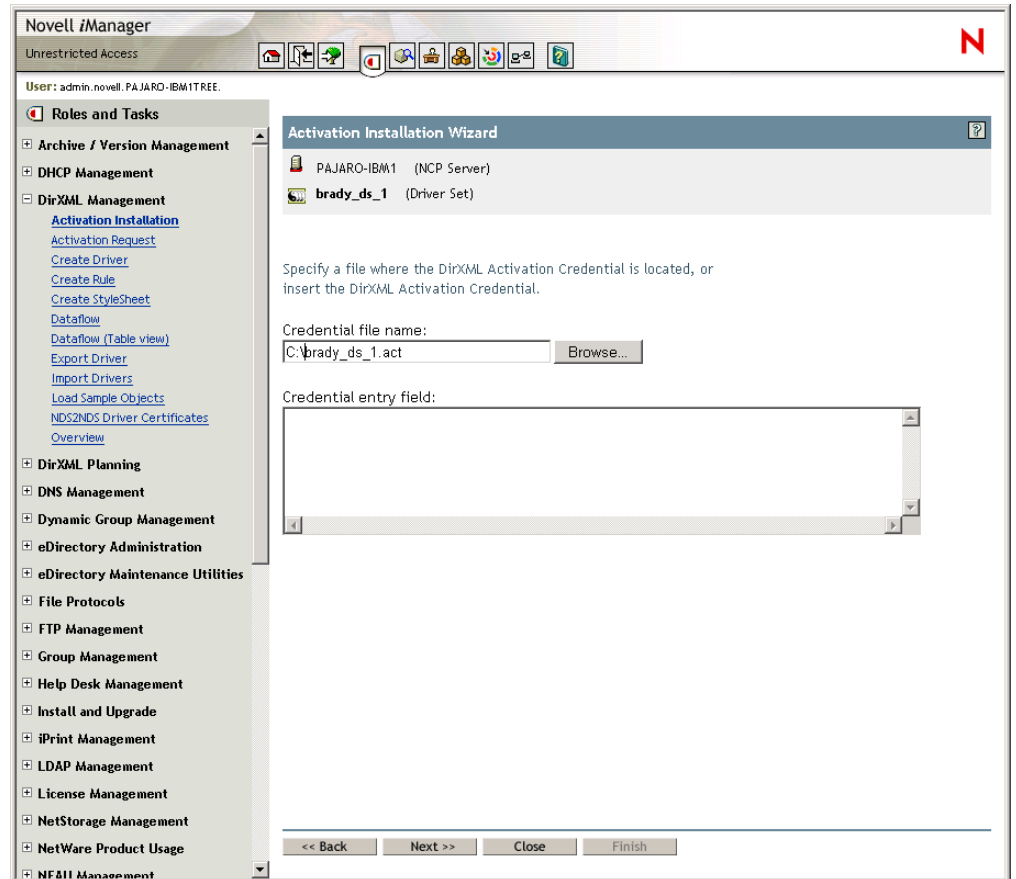
4 Choose DirXML Utilities > Activation Installation.

5 Select the driver set or browse to a driver set > click Next.

IMPORTANT: Make sure you choose a driver set that is in the same tree that the Product Activation Request was created from initially.

6 If the driver set is not associated with a server or is associated with multiple servers, select a server to associate with a driver set > click Next.

The installation dialog box appears:



7 Do one of these steps:

- ◆ Specify where you saved the DirXML Activation Credential > click Next.
- or
- ◆ Paste the contents of the DirXML Activation Credential into the text area > click Next.

8 Click Finish.

NOTE: You need to activate each driver set that has a driver. You can use the same Product Activation Credential to activate other driver sets as long as the driver sets are in the same tree. A Product Activation Credential can only be used in the tree from which the Product Activation Request was created.

Viewing Product Activations for Identity Manager and DirXML Drivers

For each of your driver sets, you can see the Product Activation Credentials you have installed for the DirXML engine and drivers. To view Product Activation Credentials:

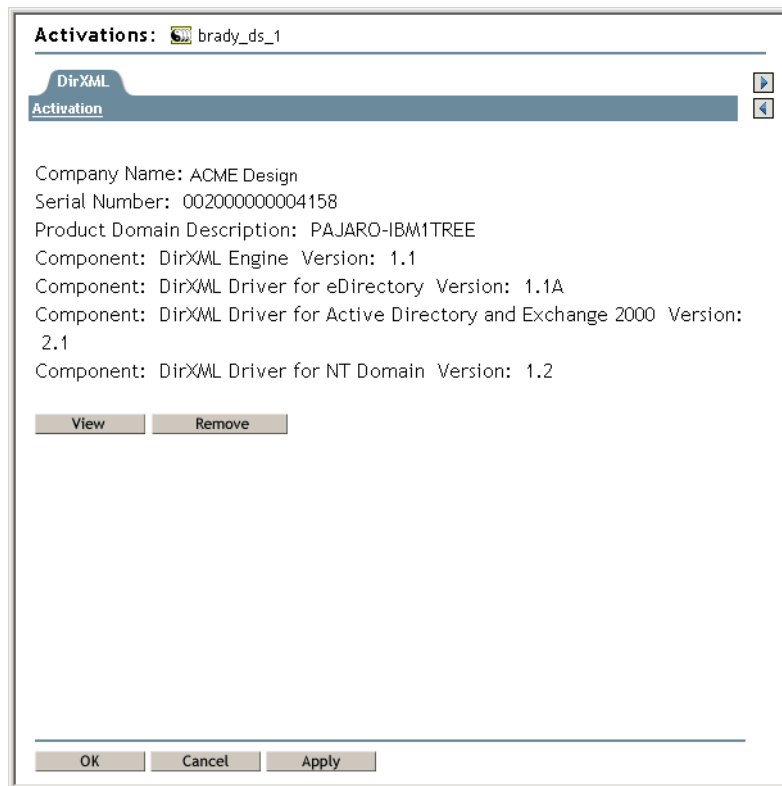
- 1** Open iManager.
- 2** Click eDirectory Administration > Modify Object.
- 3** Enter the driver set or the driver you want to view activation information for in the object name field.

or

Browse to the driver set or the driver you want to view activation information on.

4 From the DirXML tab, select Activation.

DirXML and DirXML Driver activation credentials display on this page.



You can view the text of the activation credential or, if an error is reported, you can remove an activation credential.

NOTE: After installing a valid Product Activation Credential for a driver set, you might still see “Activation Required” next to the driver name. If this is the case, restart the driver and the message should then disappear.

B

Updates

- ♦ “March 2004” on page 221
- ♦ “April 1, 2004” on page 221

March 2004

- ♦ The following new sections were added:
 - ♦ “Using the DirXML Command Line Utility” on page 64.
 - ♦ “Using Named Passwords” on page 64.
 - ♦ “Global Configuration Values” on page 17 and “Driver Heartbeat” on page 17 in the section about new features.
- ♦ References to Password Synchronization 2.0 have been changed to Identity Manager Password Synchronization, to indicate that the new Password Synchronization functionality is not a separate product, but is a feature of Identity Manager.
- ♦ References to DirXML 2.0 have been changed to Identity Manager 2. The engine and drivers are still referred to as the DirXML engine and DirXML drivers.

April 1, 2004

- ♦ The Password Self-Service information was put into a separate chapter, to make it easier to find. Now the information about NMAS Password Policies is in two chapters:
 - ♦ Chapter 7, “Managing Passwords Using Password Policies,” on page 73
 - ♦ Chapter 8, “Password Self-Service,” on page 89

