NetIQ® Identity Manager

Driver for Active Directory Implementation Guide

November 2022

Legal Notice

For information about NetIQ trademarks, see https://www.netiq.com/company/legal/.

Contents

About this Book and the Library

The *Identity Manager Driver for Active Directory Implementation Guide* explains how to install, configure, and manage the Identity Manager Driver for Active Directory.

Intended Audience

This book provides information for individuals responsible for understanding administration concepts and implementing a secure, distributed administration model.

Other Information in the Library

For more information about the library for Identity Manager, see the following resources:

- Identity Manager documentation website (https://www.netiq.com/documentation/identity-manager-48/)
- Identity Manager drivers documentation website (https://www.netiq.com/documentation/identity-manager-48-drivers/)

About NetIQ Corporation

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

Our Viewpoint

Adapting to change and managing complexity and risk are nothing new

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

Enabling critical business services, better and faster

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

Our Philosophy

Selling intelligent solutions, not just software

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

Driving your success is our passion

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

Our Solutions

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

Contacting Sales Support

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

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

Contacting Technical Support

For specific product issues, contact our Technical Support team.

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

Contacting Documentation Support

Our goal is to provide documentation that meets your needs. The documentation for this product is available on the NetIQ Web site in HTML and PDF formats on a page that does not require you to log in. If you have suggestions for documentation improvements, click **Add Comment** at the bottom of any page in the HTML version of the documentation posted at www.netiq.com/documentation. You can also email Documentation-Feedback@netiq.com. We value your input and look forward to hearing from you.

Contacting the Online User Community

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

Other Information in the Library

For more information about the library for Identity Manager, see the Identity Manager documentation website.

1 Understanding the Active Directory Driver

This section contains high-level information about how the Active Directory driver functions.

Key Terms

Identity Manager

NetIQ Identity Manager is a service that synchronizes data among servers in a set of connected systems by using a robust set of configurable policies. Identity Manager uses the Identity Vault to store shared information, and uses the Identity Manager engine for policy-based management of the information as it changes in the vault or connected system. Identity Manager runs on the server where the Identity Vault and the Identity Manager engine are located.

Connected System

A connected system is any system that can share data with Identity Manager through a driver. Active Directory is a connected system.

Identity Vault

The Identity Vault is a persistent database powered by eDirectory and used by Identity Manager to hold data for synchronization with a connected system. The vault can be viewed narrowly as a private data store for Identity Manager or more broadly as a metadirectory that holds enterprise-wide data. Data in the vault is available to any protocol supported by eDirectory, including the NetWare Core Protocol (NCP), which is the traditional protocol used by iManager, LDAP, and DSML.

Because the vault is powered by eDirectory, Identity Manager can be easily integrated into your corporate directory infrastructure by using your existing directory tree as the vault.

Identity Manager Engine

The Identity Manager engine is the core server that implements the event management and policies of Identity Manager. The engine runs on the Java Virtual Machine in eDirectory.

Active Directory Driver

A driver implements a data sharing policy for a connected system. You control the actions of the driver by using iManager to define the filters and the policy. For Active Directory, a driver implements the policy for a single domain.

Driver Shim

A driver shim is the component of a driver that converts the XML-based Identity Manager command and event language (XDS) to the protocols and API calls needed to interact with a connected system. The shim is called to execute commands on the connected system after the Output Transformation runs. Commands are usually generated on the Subscriber channel but can be generated by command write-back on the Publisher channel.

The shim also generates events from the connected system for the Input Transformation policy. A driver shim can be implemented either in Java class or as a native Windows DLL file. The shim for Active Directory is ADD`river.dll`.

ADD`river.dll` is implemented as a native Windows DLL file. It uses several different Windows APIs to integrate with Active Directory. These APIs typically require some type of login and authentication to succeed. Also, the APIs might require that the login account have certain rights and privileges within Active Directory and on the machine where ADD`river.dll` executes.

If you use the Remote Loader, ADD`river.dll` executes on the server where the Remote Loader is running. Otherwise, it executes on the server where the Identity Manager engine is running.

Remote Loader

A Remote Loader enables a driver shim to execute outside of the Identity Manager engine (perhaps remotely on a different machine). The Remote Loader is typically used when a requirement of the driver shim is not met by the Identity Manager server. For example, if the Identity Manager engine is running on Linux, the Remote Loader is used to execute the Active Directory driver shim on a Windows server.

The Remote Loader is a service that executes the driver shim and passes information between the shim and the Identity Manager engine. When you use a Remote Loader, you install the driver shim on the server where the Remote Loader is running, not on the server where the Identity Manager engine is running. You can choose to use SSL to encrypt the connection between the Identity Manager engine and the Remote Loader.

When you use the Remote Loader with the Active Directory driver shim, two network connections exist:

- Between the domain controller and the Remote Loader
- Between Active Directory and the Active Directory driver shim

Data Transfers Between Systems

Data flows between Active Directory and the Identity Vault by using the Publisher and Subscriber channels.

The Publisher channel does the following:

- Reads events from Active Directory for the domain hosted on the server that the driver shim is connecting to.
- Submits that information to the Identity Vault.

The Subscriber channel does the following:

- Watches for additions and modifications to the Identity Vault objects.
- Makes changes to Active Directory that reflect those changes.

You can configure the driver so that both Active Directory and the Identity Vault are allowed to update a specific attribute. In this configuration, the most recent change determines the attribute value, except for merge operations that are controlled by the filters and merge authority.

NOTE: A single transaction can handle multiple events. When any one of the event fails, the driver fails to execute all the subsequent events in the transaction.

Key Driver Features

The sections below contains information about the key driver features.

- "Local Platforms" on page 13
- "Remote Platforms" on page 14
- "Password Synchronization Support" on page 14
- "Data Synchronization Support" on page 14

Local Platforms

A local installation is an installation of the driver on the Identity Manager server. The Active Directory driver can be installed on the Windows operating systems supported for the Identity Manager server. The following Windows platforms are supported:

- Windows Server 2022
- Windows Server 2019
- Windows Server 2016
- Windows Server 2012 R2 (64-bit)

For more information about local installations, see "Where to Install the Active Directory Driver" on page 20.

Remote Platforms

The Active Directory driver can use the Remote Loader service to run on a Windows server other than the Identity Manager server. The Remote Loader service for the Active Directory driver can be installed on the following Windows platforms:

- Windows Server 2022 (64-bit)
- Windows Server 2019 (64-bit)
- Windows Server 2016 (64-bit)
- Windows Server 2012 (64-bit)
- Windows Server 2012 R2 (64-bit)

For more information about remote installations, see "Where to Install the Active Directory Driver" on page 20.

Password Synchronization Support

The Active Directory driver synchronizes passwords on both the Subscriber channel and the Publisher channel. For more information, see Chapter 6, "Synchronizing Passwords," on page 47.

Data Synchronization Support

The Active Directory driver synchronizes User objects, Group objects, containers, and Exchange mailboxes.

Default Driver Configuration

The Active Directory driver is shipped with packages. When the driver is created with packages in Designer, a set of policies and rules are created suitable for synchronizing with Active Directory. If your requirements for the driver are different from the default policies, you need to modify the default policies to do what you want. Pay close attention to the default Matching policies. The data that you trust to match users usually is different from the default. The policies themselves are commented and you can gain a greater understanding of what they do by creating a test driver and reviewing the policies with Designer or iManager.

User Object Name Mapping

Management utilities for the Identity Vault, such as iManager and Designer, typically name user objects differently than the Users and Computers snap-in for the Microsoft Management Console (MMC). Make sure that you understand the differences so the Matching policy and any Transformation policies you have are implemented properly.

Data Flow

Data flow between Active Directory and the Identity Vault is controlled by the filters, mappings, and policies that are in place for the Active Directory driver.

Filters

The driver filter determines which classes and attributes are synchronized between Active Directory and the Identity Vault, and in which direction synchronization takes place.

Schema Mapping

Table 1-1 through Table 1-6 list Identity Vault user, group, and Organizational Unit attributes that are mapped to Active Directory user and group attributes.

The mappings listed in the tables are default mappings. You can remap same-type attributes.

Table 1-1 *Mapped User Attributes*

eDirectory - User	Active Directory - user
DirXML-ADAliasName	userPrincipalName
CN	sAMAccountName
L	PhysicalDeliveryOfficeName
Physical Delivery Office Name	l
nspmDistributionPassword	nspmDistributionPassword

Table 1-2 *Mapped Group Attributes*

eDirectory - Group	Active Directory - group
DirXML-ADAliasName	userPrincipalName

eDirectory's L attribute is mapped to Active Directory's physicalDeliveryOfficeName attribute, and eDirectory's Physical Delivery Office Name attribute is mapped to Active Directory's L attribute. Because similarly named fields have the same value, mapping the attributes this way enables the attributes to work well with iManager and the Microsoft Management Console.

Table 1-3 *Mapped Organizational Unit Attributes*

eDirectory - Organizational Unit	Active Directory - organizationalUnit
L	physicalDeliveryOfficeName
Physical Delivery Office Name	l

Table 1-4 *Mapped Organization Attributes*

eDirectory - Organization	Active Directory - organization
L	physicalDeliveryOfficeName
Physical Delivery Office Name	l

The driver maps the Locality class, but there are no attributes for the class.

Table 1-5 *Mapped Locality Class*

eDirectory	Active Directory
Locality	locality

Table 1-6 *Mapped Non-Class Specific Attributes*

eDirectory	Active Directory
Description	description
DirXML-EntitlementRef	DirXML-EntitlementRef
DirXML-EntitlementResult	DirXML-EntitlementResult
Facsimile Telephone Number	facsimiletelephoneNumber
Full name	displayName
Given Name	givenName
Group Membership	memberOf
Initials	initials
Internet EMail Address	mail
Login Allowed Time Map	logonHours
Login Disabled	dirxml-uACAccountDisabled
Login Expiration Time	accountExpires

eDirectory	Active Directory
Login Intruder Reset Time	lockoutTime
Member	member
OU	ou
Owner	managedBy
Postal Code	PostalCode
Postal Office Box	postOfficeBox
S	st
SA	streetAddress
See Also	seeAlso
DirXML-SPEntitlements	DirXML-SPEntitlements
Surname	sn
Telephone Number	telephoneNumber
Title	title

Name Mapping Policies

The Active Directory packages includes two name mapping policies that work together to help you reconcile different naming policies between the Identity Vault and Active Directory. When you create a user with the Active Directory Users and Computers tool (a snap-in for the Microsoft Management Console and abbreviated as MMC in this document) you see that the user full name is used as its object name. Attributes of the user object define pre-Windows 2000 Logon Name (also known as the NT Logon Name or sAMAccountName) and the Windows 2000 Logon Name (also known as the userPrincipalName). When you create a user in the Identity Vault with iManager or ConsoleOne, the object name and the user logon name are the same.

If you create some users in Active Directory by using MMC, and then create other objects in the Identity Vault or another connected system that is synchronized with the Identity Vault, the object can look odd in the opposite console and might fail to be created in the opposite system. However, you can use the name mapping policies to avoid this problem.

The Full Name Mapping policy is used to manage objects in Active Directory by using the MMC conventions. When this policy is enabled, the Full Name attribute in the Identity Vault is synchronized with the object name in Active Directory.

The NT Logon Name Mapping policy is used to manage objects in Active Directory by using the Identity Vault conventions. When it is enabled, the Identity Vault object name is used to synchronize both the object name and NT Logon Name in Active Directory. Objects in Active Directory have the same names as the Identity Vault, and the NT Logon Name matches the Identity Vault logon name.

When both of the policies are enabled at the same time, the Active Directory object name is the Identity Vault Full Name, but the NT Logon Name matches the Identity Vault logon name.

When both policies are disabled, no special mapping is made. The object names are synchronized and there are no special rules for creating the NT Logon Name. Because the NT Logon Name is a mandatory attribute in Active Directory, you need some method of generating it during Add operations. The NT Logon Name (sAMAccountName) is mapped to the DirMXL-ADAliasName in the Identity Vault, so you could either use that attribute to control the NT Logon Name in Active Directory or you could build your own policy in the Subscriber Create policies to generate one. With this policy selection, users created with MMC use the object name generated by MMC as the object name in the Identity Vault. However, this name might be inconvenient for login to the Vault.

Using the Name Mapping policies is controlled through Global Configuration Values. For information, see "Global Configuration Values" on page 99.

Active Directory Logon Name Policies

The Windows 2000 Logon name (also known as the userPrincipalName or UPN) does not have a direct counterpart in the Identity Vault. The UPN looks like an e-mail address (user@mycompany.com) and might in fact be the user's e-mail name. The important thing to remember when working with the UPN is that it must use a domain name (the part after the @ sign) that is configured for your domain. You can find out what domain names are allowed by using MMC to create a user and looking at the domain name drop-down box when adding the UPN.

The default configuration offers several choices for managing userPrincipalName. If your domain is set up so that the user's e-mail address can be used as a userPrincipalName, one of the options to track the user's e-mail address is appropriate. You can make userPrincipalName follow either the Identity Vault or Active Directory e-mail address, depending on which side is authoritative for e-mail. If the user e-mail address is not appropriate, you can choose to have a userPrincipalName constructed from the user logon name plus a domain name. If more than one name can be used, update the policy after import to make the selection. If none of these options are appropriate, then you can disable the default policies and write your own.

Use of the Active Directory Logon Name policy is controlled through Global Configuration Values. For information, see "Global Configuration Values" on page 99.

2 Preparing Active Directory

Use the information in this section as you prepare to install the Active Directory driver:

Driver Prerequisites

- ❒ Windows Server 2022(64-bit), Windows Server 2019(64-bit), Windows Server 2016 (64-bit), Windows Server 2012 R2 (64-bit), Windows Server 2012 (64-bit)
- ❒ Internet Explorer 6.0 or later on the server running the Active Directory driver and on the target domain controller
- ❒ Active Directory domain controller DNS name or IP address, depending on the authentication method

---

NOTE: The domain functional level for the Active Directory driver is supported on all supported Windows Server platforms.

Also, we recommend that the server hosting the Active Directory driver be a member of the Active Directory domain. This is required to provision Exchange mailboxes and synchronize passwords. If you don't require these features, the server can be a member of any domain as long as the Simple (simple bind) authentication mode is used. To have bidirectional password synchronization, the Negotiate authentication option must be selected.

If you want to synchronize with an ADAM instance, see Appendix B, "Configuring the Driver for Use with an AD LDS/ADAM Instance," on page 105 for more information.

If you want to synchronize Exchange accounts, see Appendix C, "Provisioning Exchange Accounts," on page 113.

Where to Install the Active Directory Driver

The Active Directory driver shim must run on one of the supported Windows platforms. However, you don't need to install the Identity Manager engine on this same machine. Using a Remote Loader, you can separate the engine and the driver shim, allowing you to balance the load on different machines or accommodate corporate directives.

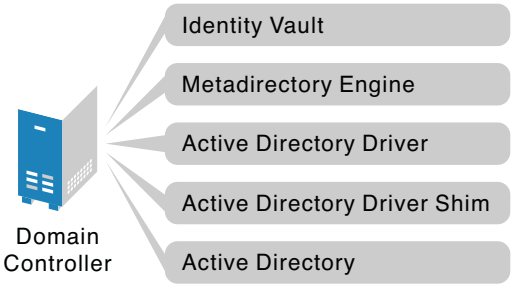
The installation scenario you select determines how the driver shim is installed. If you choose to install the driver shim on the same machine as Identity Manager (where the Identity Manager engine and the Identity Vault are located), Identity Manager calls the driver shim directly. If you choose to install the driver shim on another machine, you must use the Remote Loader.

You can install the Active Directory driver on either the domain controller or a member server. Before you start the driver installation, determine where you want to install the driver.

Local Installation

A single Windows domain controller can host the Identity Vault, the Identity Manager engine, and the driver.

Figure 2-1 *All Components on the Domain Controller*



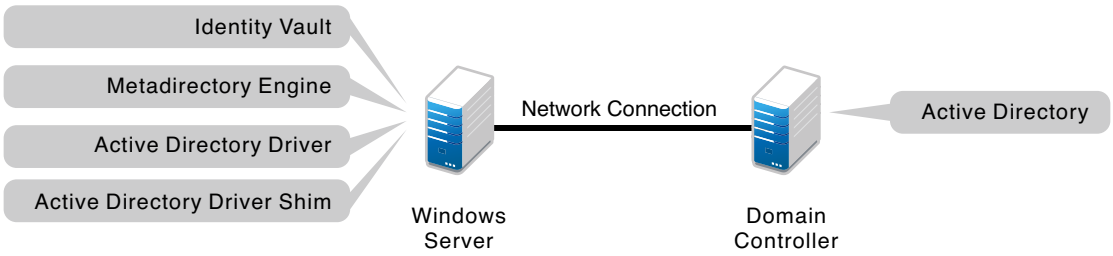
This configuration works well for organizations that want to save on hardware costs. It is also the highest-performance configuration because there is no network traffic between Identity Manager and Active Directory.

However, hosting Identity Vault and the Identity Manager engine on the domain controller increases the overall load on the controller and increases the risk that the controller might fail. Because domain controllers play a critical role in Microsoft networking, many organizations are more concerned about the speed of the domain authentication and the risks associated with a failure on the domain controller than about the cost of additional hardware.

Remote Installation on Windows Server Only

You can install the Identity Vault, the Identity Manager engine, and the driver on a computer other than the Active Directory domain controller.

Figure 2-2 *All Components on a Windows Server*

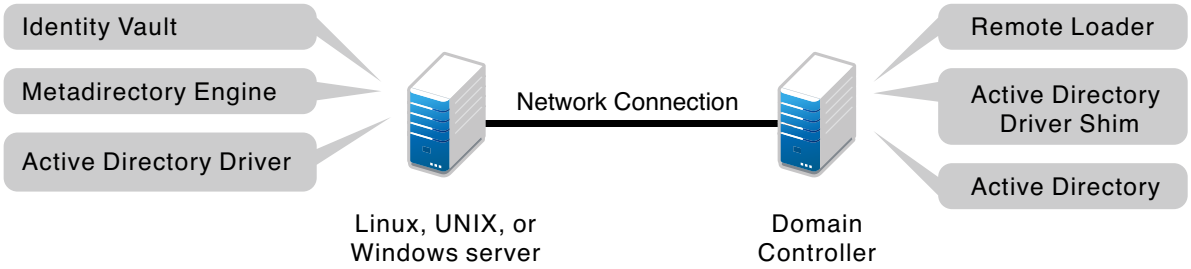


This configuration fits best when your corporate policy disallows running the driver on your domain controller so that there is no Identity Manager software on the domain controller.

Remote Installation on Windows and Other Platforms

You can install the Remote Loader and the driver shim on the Active Directory domain controller, but install the Identity Vault and the Identity Manager engine on a different server.

Figure 2-3 *Remote Loader and Driver on the Domain Controller*



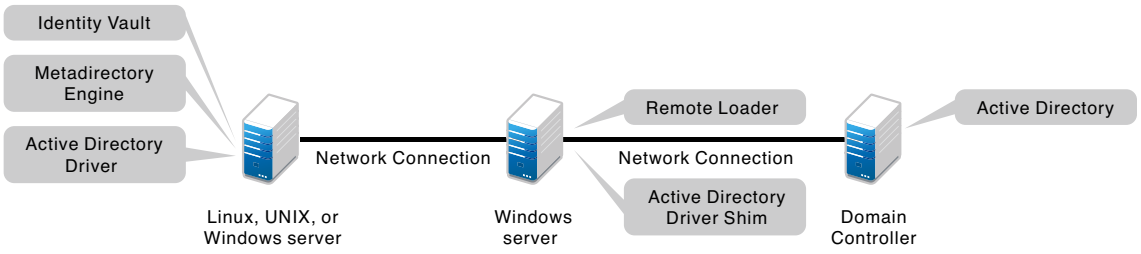
This configuration fits best when you require Identity Vault and Identity Manager engine installations on a platform other than one of the supported versions of Windows.

Both types of remote installations eliminate the performance impact of hosting the Identity Vault and the Identity Manager engine on the domain controller.

Remote Installation on a Windows Member Server

If you have platform requirements and domain controller restrictions in place, you can use a three-server configuration.

Figure 2-4 *Remote Loader and Driver on a Windows Server*



This configuration fits best when your corporate policy disallows running the driver on your domain controller and your Identity Manager engine installation is not on a supported Windows server.

Addressing Security Issues

The major security issues to consider are authentication, encryption, and use of the Remote Loader. You might want to consider a security option called signing. See “Digitally sign communications” on page 72.

A simple prescription for managing security is not possible because the security profile available from Windows varies with the service pack, DNS server infrastructure, domain policy, and local policy settings on the server. The following sections explain your security choices and provide suggested configurations. When you implement your driver and when you upgrade components, pay close attention to security.

Authentication Methods

Authentication identifies the driver shim to Active Directory and, potentially, the local machine. To authenticate to Active Directory, you can use either the Negotiate method or the Simple (simple bind) method.

Table 2-1 *Authentication Methods*

Authentication Method	Description	Advantages	Disadvantages
Negotiate	The preferred method. Uses kerberos, NTLM, or a pluggable authentication scheme if one is installed.	The driver can be installed on any server in the domain.	The server hosting the driver must be a member of the domain.
Simple	Used when the server hosting the driver shim is not a member of the domain.	The driver can be installed on a server that is not a member of the domain.	Some provisioning services are unavailable, such as Exchange mailbox provisioning and password synchronization.

NOTE: Active Directory driver uses Negotiate as the default authentication method. When the Active Directory driver’s basic configuration file is imported to create a new driver, the authentication method is set to Negotiate by default. If you want to use Simple authentication, change the authentication setting on the driver’s property page after the driver is created.

Encryption Using SSL

SSL encrypts data. Depending on your configuration, SSL can be used in two places:

- Between the Active Directory driver and the domain controller
- Between the Identity Vault and the Remote Loader running the Active Directory driver

Password synchronization occurs between Active Directory and the Identity Vault. You need to make sure that you use SSL with any communication that goes across the network.

If the Identity Manager engine, Identity Vault, the Active Directory driver, and Active Directory are on the same machine, you don't need SSL. Communication isn't going across the network.

However, if you are accessing Active Directory remotely by using an Active Directory driver shim on a member server, you need to set up SSL between the Active Directory driver shim and Active Directory. You do this by setting the SSL parameter to **Yes** on the driver configuration. See Step 5, in "SSL Connection between the Active Directory Driver and the Domain Controller" on page 24.

If you are using the Remote Loader on the domain controller, you can set up SSL between the Identity Manager engine and the Remote Loader.

The following table outlines where SSL connections can be used for each of the scenarios discussed in "Where to Install the Active Directory Driver" on page 20:

Table 2-2 *SSL Connections*

Configuration	SSL Connections Available
Single-Server	No SSL connections are necessary.
Two-Server: Identity Manager and the Active Directory driver are on the same server	An SSL connection can be established between the Active Directory driver and the domain controller. See "SSL Connection between the Active Directory Driver and the Domain Controller" on page 24.
Dual-Server: Identity Manager is on one server but the Active Directory driver is on a separate server	An SSL connection can be established between Identity Manager and the Remote Loader running the Active Directory driver. See "SSL Connection Between the Remote Loader and Identity Manager" on page 26.
Three-Server	An SSL connection can be established between the Active Directory driver and the domain controller. See "SSL Connection between the Active Directory Driver and the Domain Controller" on page 24. An SSL connection can also be established between Identity Manager and the Remote Loader running the Active Directory driver. See "SSL Connection Between the Remote Loader and Identity Manager" on page 26.

SSL Connection between the Active Directory Driver and the Domain Controller

To make SSL connections to an Active Directory domain controller, you must be set up to use SSL. This involves setting up a CA, then creating, exporting, and importing the necessary certificates. This is only needed if the Remote Loader is not running on a domain controller.

Setting Up a Certificate Authority

Most organizations already have a CA. If this is the case for your organization, you need to export a valid certificate, then import it to the certificate store on your domain controller. The server hosting the driver shim must trust the root CA that the issuing CA of this certificate chains to.

If you do not have a CA in your organization, you must establish one. NetIQ, Microsoft, and several other third parties provide the tools necessary to do this. Establishing a CA is beyond the scope of this guide. For more information about the NetIQ solution, see the *NetIQ Certificate Server 3.3 Administration Guide* (http://www.novell.com/documentation/lg/crt33/index.html).

Creating, Exporting, and Importing Certificates

After you have a CA, the LDAP server must have the appropriate server authentication certificate installed for LDAP SSL to operate successfully. Also, the server hosting the driver shim must trust the authority that issued those certificates. Both the server and the client must support 128-bit encryption.

1. Generate a certificate that meets the following Active Directory LDAP service requirements:
   - The LDAPS certificate is located in the local computer's personal certificate store (programmatically known as the computer's MY certificate store).
   - A private key matching the certificate is present in the local computer's store and is correctly associated with the certificate.

     The private key must not have strong private-key protection enabled.
   - The Enhanced Key Usage extension includes the Server Authentication (1.3.6.1.5.5.7.3.1) object identifier (also known as the OID).
   - The Active Directory fully qualified domain name (for example, DC01.DOMAIN.COM) of the domain controller appears in one of the following places:
     - The Common Name (CN) in the Subject field
     - The DNS entry in the Subject Alternative Name extension
   - The certificate was issued by a CA that the domain controller and the LDAPS clients trust.

     Trust is established by configuring the clients and the server to trust the root CA that the issuing CA chains to.

   This certificate permits the LDAP service on the domain controller to listen for and automatically accept SSL connections for both LDAP and global catalog traffic.

NOTE: This information appears in the Microsoft Knowledge Base Article 321051, "How to enable LDAP over SSL with a third-party certification authority" (http://support.microsoft.com/kb/321051). Consult this document for the latest requirements and additional information.

2 Export this certificate in one of the following standard certificate file formats:

- Personal Information Exchange (PFX, also called PKCS #12)
- Cryptographic Message Syntax Standard (PKCS #7)
- Distinguished Encoding Rules (DER) Encoded Binary X.509
- Base64 Encoded X.509

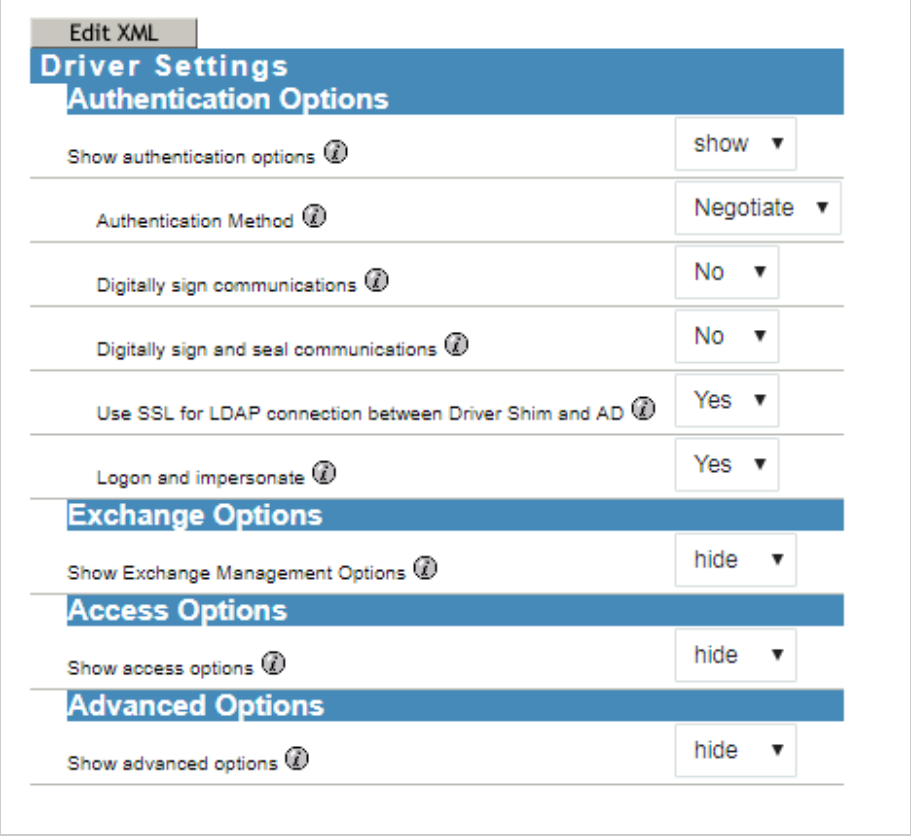
3 Install this certificate on the domain controller.

4 Ensure that a trust relationship is established between the server hosting the driver shim and the root CA that issued the certificate.

The server hosting the driver shim must trust the root CA that the issuing CA chains to.

For more information on establishing trust for certificates, see "Policies to establish trust of root certification authorities" (http://technet.microsoft.com/en-us/library/cc775613(v=ws.10).aspx), in the Microsoft TechNet Library (http://technet.microsoft.com/library/bb625087.aspx).

5 In iManager, edit the driver properties and change the **Use SSL (yes/no)** option to yes.



6 Restart the driver.

When the driver restarts, an SSL connection is negotiated between the domain controller and the server running the Active Directory driver shim.

Verifying the Certificate

To verify the certificate, authenticate to Active Directory via SSL. Use the ldifde command line utility found on Windows servers. To use the ldifde command:

1. Open a command line prompt.
2. Enter `ldifde -f output/input file -t 636 -b administrator domain password -s computerFullName`

For example, you would use the following command if your server is configured for port 636.

```
ldifde -f out.txt -t 636 -b administrator dxad.netiq.com netiq -s
parent1.dxad3.lab.netiq
```

The output is sent to the `out.txt` file. If you open the file and see the objects in Active Directory listed, you made a successful SSL connection to Active Directory and the certificate is valid.

SSL Connection Between the Remote Loader and Identity Manager

If you are using the Remote Loader, you need to set up SSL between the Identity Manager engine and the Remote Loader, and configure the settings between the driver and Active Directory.

For information on establishing an SSL connection between the Remote Loader and Identity Manager, see "Creating a Secure Connection to the Identity Manager Engine" in the *NetIQ Identity Manager Driver Administration Guide*.

IMPORTANT: When the Remote Loader is running on a Windows server, the certificate must be in Base64 format. If you use the DER format, the Remote Loader fails to connect to the Identity Manager engine.

Creating an Administrative Account

In a test environment, use the Administrator account until you get the Active Directory driver working. Then create an administrative account that has the proper rights (including restricted rights) for the Active Directory driver to use exclusively to authenticate to Active Directory.

Doing this keeps the Identity Manager administrative account insulated from changes to other administrative accounts. Advantages to this design are:

- You can use Active Directory auditing to track the activity of the Active Directory driver.
- You can implement a password change policy as with other accounts, then make necessary updates to the driver configuration.

This account name and password are stored in the driver configuration. Therefore, you must change this password whenever the account password changes. If you change the account password without updating the driver configuration, authentication fails the next time the driver is restarted.

At a minimum, this account must have Read and Replicating Directory Changes rights at the root of the domain for the Publisher channel to operate. For more information, see How to grant the Replicating Directory Changes permission in Windows. You also need Write rights to any object modified by the Subscriber channel. Write rights can be restricted to the containers and attributes that are written by the Subscriber channel.

To obtain delete events from Active Directory, you need permission to view the contents of the deleted objects container. By default, only built-in Administrators group has this permission. To grant this permission to a non-administrator such as the driver account, modify the permissions on the deleted object container by following the instructions from Microsoft documentation. Similarly, the driver account needs permissions to view the contents of the Password Sync registry. By default, only local system account has these permissions. If the driver is running as any other account, you must grant "Full Control" over the "Novell" registry keys and sub-keys to the driver account. For more information, see "Configuring System Permissions" on page 27".

To provision Exchange mailboxes, your Identity Manager account must have "Act as part of the Operating System" permission for the logon account.

Configuring System Permissions

In order to retrieve a user's password on the Publisher channel, the driver requires system permissions in addition to Active Directory permissions.

Identity Manager also configures specific permissions for its own internal components. On domain controllers, the PWFilter component runs using SYSTEM privileges, so the local system account should have full permissions to the `HKEY_LOCAL_MACHINE\SOFTWARE\Novell\PwFilter\Data` registry key, as well as any sub-keys.

The driver shim runs using SYSTEM privileges by default, so the system account should also have full permissions to the `HKEY_LOCAL_MACHINE\SOFTWARE\Novell\PassSync\Data` registry key, as well as any sub-keys. If the driver is run using any other account, that account should be given full permissions to the `HKEY_LOCAL_MACHINE\SOFTWARE\Novell\PwFilter\Data` registry key, as well as any sub-keys. The account should also be a member of the `Administrators` group.

NOTE: The driver automatically provides default permissions to both PWFilter and the driver shim. Modifying these permissions can affect the functionality of the driver and should be performed with caution.

Becoming Familiar with Driver Features

This section discusses driver features you should become familiar with before deploying the Active Directory driver.

Multivalue Attributes

The way the Active Directory driver handles multivalue attributes has changed from version 2.

Version 2 treated multivalue attributes as single-valued on the Subscriber channel by ignoring all but the first change value in an Add or Modify operation. Version 3 and later of the Active Directory driver fully supports multivalue attributes.

However, when the Active Directory driver synchronizes a multivalue attribute with a single-value attribute, the multivalue attribute is treated as single-valued. For example, the Telephone Number attribute is single-valued in Active Directory, and multivalued in the Identity Vault. When this attribute is synchronized from Active Directory, only a single value is stored in the Identity Vault.

This creates true synchronization and mapping between the two attributes, but can result in a potential loss of data if you have multiple values in an attribute that is mapped to an attribute with a single value. In most cases, a policy can be implemented to preserve the extra values in another location if this is required in your environment.

Using Custom Boolean Attributes to Manage Account Settings

The Active Directory attribute userAccountControl is an integer whose bits control logon account properties, such as whether logon is allowed, passwords are required, or the account is locked. Synchronizing the Boolean properties individually is difficult because each property is embedded in the integer value.

In version 2, the Active Directory driver took a shortcut that let you map userAccountControl to the eDirectory Login Disabled attribute, but didn't let you map the other property bits within the attribute.

In version 3 and later, each bit within the userAccountControl attribute can be referenced individually as a Boolean value, or userAccountControl can be managed in-total as an integer. The driver recognizes a Boolean alias to each bit within userAccountControl. These alias values are included in the schema for any class that includes userAccountControl. The alias values are accepted on the Subscriber channel and are presented on the Publisher channel.

The advantage of this is that each bit can be used as a Boolean, so the bit can be enabled individually in the Publisher filter and accessed easily. You can also put userAccountControl into the Publisher filter to receive change notification as an integer.

The integer and alias versions of userAccountControl should not be mixed in a single configuration.

The following table lists available aliases and hexadecimal values. Read-only attributes cannot be set on the Subscriber channel:

Table 2-3 *Aliases and Hexadecimal Values*

Alias	Hexadecimal	Notes
dirxml-uACAccountDisable	0x0002	Read-write
dirxml-uACDontExpirePassword	0x10000	Read-write
dirxml-uACDontRequirePreauth	0x400000	Read-write

Alias	Hexadecimal	Notes
dirxml-uACEncryptedTextPasswordAllowed	0x0080	Read-write
dirxml-uACHomedirRequired	0x0008	Read-write
dirxml-uACInterdomainTrustAccount	0x0800	Read-only
dirxml-uACLockout	0x0000	Read-write (only for clearing the lock)
dirxml-uACMNSLogonAccount	0x20000	Read-write
dirxml-uACNormalAccount	0x0200	Read-only
dirxml-uACNotDeleagated	0x100000	Read-write
dirxml-uACPasswordCantChange	0x0040	Read-only
dirxml-uACPasswordNotRequired	0x0020	Read-write
dirxml-uACScript	0x0001	Read-write
dirxml-uACServerTrustAccount	0x2000	Read-only
dirxml-uACSmartcardRequired	0x40000	Read-write
dirxml-uACTrustedForDelegation	0x80000	Read-write
dirxml-uACUseDESKeyOnly	0x200000	Read-write
dirxml-uACWorkstationTrustAccount	0x1000	Read-only

For troubleshooting tips relating to the userAccountControl attribute, see "Active Directory Account Is Disabled After a User Is Added on the Subscriber Channel" on page 83.

Provisioning Exchange Mailboxes

The Active Directory driver can be configured to provision Exchange accounts as well as Active Directory accounts. The Active Directory driver can provision Exchange Server 2019 accounts. For information on configuring the driver to provision the Exchange mailboxes, see Appendix C, "Provisioning Exchange Accounts," on page 113.

Expiring Accounts in Active Directory

If you map the eDirectory attribute of Login Expiration Time to the Active Directory attribute of accountExpires, an account in Active Directory expires a day earlier than the time set in eDirectory.

This happens because Active Directory sets the value of the accountExpires attribute in full-day increments. The eDirectory attribute of Login Expiration Time uses a specific day and time to expire the account.

For example, if you set an account in eDirectory, to expire on July 15, 2007, at 5:00 p.m., the last full day this account is valid in Active Directory is July 14.

If you use the Microsoft Management Console to set the account to expire on July 15, 2007, the eDirectory attribute of Login Expiration Time is set to expire on July 16, 2007 at 12:00 a.m. Because the Microsoft Management Console doesn't allow for a value of time to be set, the default is 12:00 a.m.

The driver uses the most restrictive settings. You can add an additional day to the expiration time in Microsoft depending upon what your requirements are.

Retaining eDirectory Objects When You Restore Active Directory Objects

Any Active Directory objects that are restored through the Active Directory tools delete the associated eDirectory object when the objects are synchronized. The Active Directory driver looks for a change in the isDeleted attribute on the Active Directory object. When the driver detects a change in this attribute, a Delete event is issued through the driver for the object associated with the Active Directory object.

If you don't want eDirectory objects deleted, you must add an additional policy to the Active Directory driver. Identity Manager 3.6.1 and later comes with a predefined rule that changes all Delete events into Remove Association events. For more information, see "Command Transformation - Publisher Delete to Disable " in the *NetIQ Identity Manager - Using Designer to Create Policies* guide.

3 Installing the Driver Files

There are several locations where you can install the driver files, as discussed in "Where to Install the Active Directory Driver" on page 20.

By default, the Active Directory driver files are installed on the Identity Manager server at the same time as the Identity Manager engine. The installation program extends the Identity Vault's schema and installs the driver shim, the driver configuration file, and a utility to help with the configuration of the driver. It does not create the driver in the Identity Vault (see Chapter 4, "Creating a New Driver," on page 33) or upgrade an existing driver's configuration (see Chapter 5, "Upgrading an Existing Driver," on page 43).

The following sections explain what to do if the Active Directory driver files are not on the server you want and how to install the Active Directory Discovery tool (used to gather configuration information) on the appropriate Active Directory server:

Installing the Driver Files

If you performed a custom installation and did not install the Active Directory driver on the Identity Manager server, you have two options:

- Install the files on the Identity Manager server.
- Install the Remote Loader (required to run the driver on a non-Identity Manager server) and the driver files on a non-Identity Manager server where you want to run the driver. This is the method you should use if you do not want to install eDirectory and Identity Manager on the server that has Active Directory installed on it.

If you decide to use the Remote Loader and install it on a member server, you must configure the driver to use an SSL connection between the Remote Loader and the Identity Manager server.

Installing the Active Directory Discovery Tool

The Active Directory Discovery tool helps gather information required to configure the driver. To install the Active Directory Discovery tool:

1. On the workstation that you use to configure Active Directory, launch the Identity Manager installation.
2. Select the language you want to use and click **OK**.
3. In the Welcome dialog box, click **Next**, accept the license agreement, then click **Next** to display the Select Components page.
4. Select both the **Utilities** option and the **Customize the selected components** options, clearing all other components, then click **Next**.

5 Deselect all components except for **Active Directory Discovery Tool**, then click **Next**.

6 Specify the installation path (the default is sufficient), then click **Next**.

7 Review the selected options, then click **Install**.

8 Click **Done** when finished.

4 Creating a New Driver

After the Active Directory driver files are installed on the server where you want to run the driver (see Chapter 3, “Installing the Driver Files,” on page 31), you can create the driver in the Identity Vault. You do so by installing the driver packages and then modifying the driver configuration to suit your environment. The following sections provide instructions:

Gathering Configuration Information

The Active Directory Discovery Tool gathers the information needed to configure the Active Directory driver (see “Installing the Active Directory Discovery Tool” on page 31 for information about installing the tool). The tool gathers a list of the domain controllers and Microsoft Exchange private message stores available in the domain and optionally creates an account in Active Directory suitable for the driver.

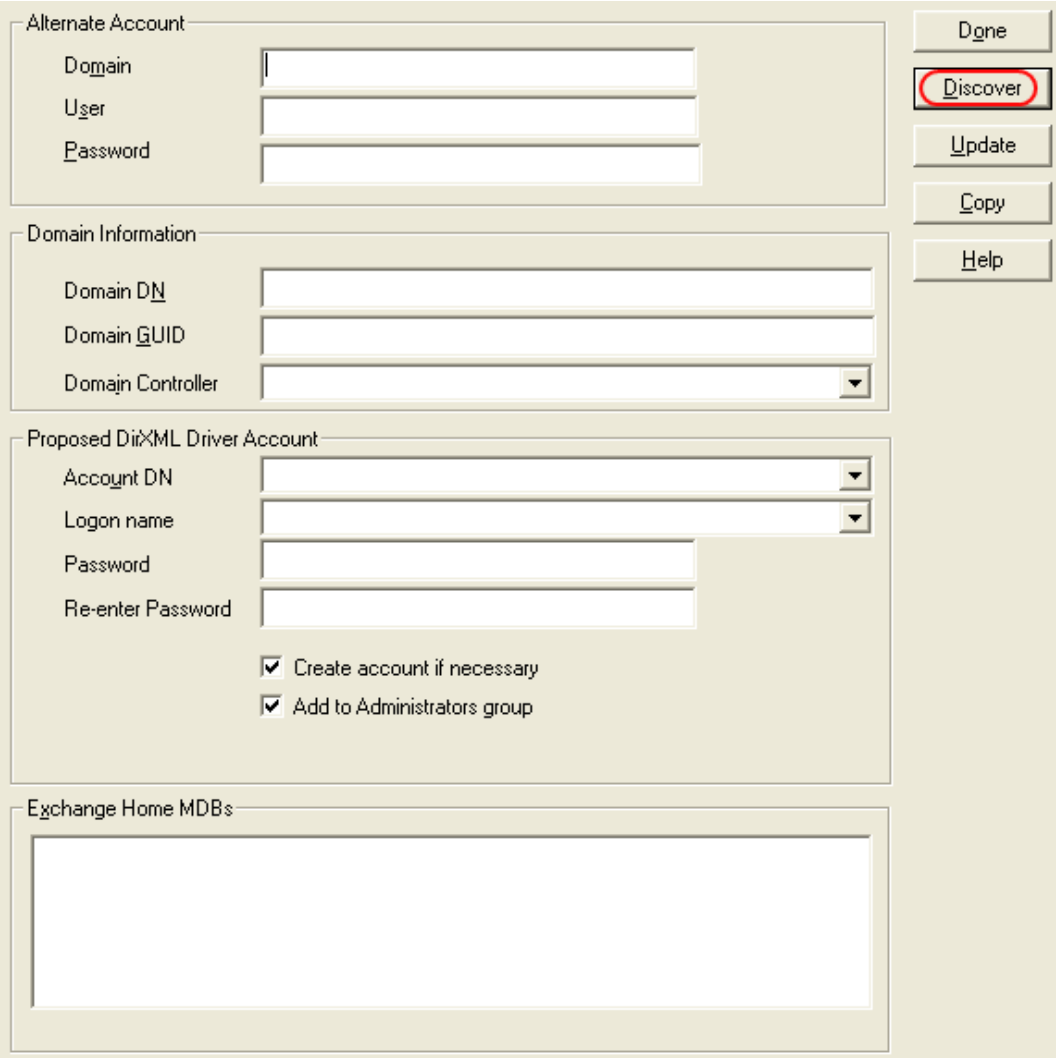
To run the tool:

1. On the workstation where you installed the tool, double-click the following file: `C:\Novell\NDS\DirXMLUtilities\ad_disc\ADManager`.

   This is the default installation location for the file.
2. Click **Discover** to populate the tool with your domain information.

   The tool lists the following information:

   - Domain DN
   - Domain GUID
   - Domain Controller name
   - Proposed driver account name and password
   - Exchange Home MDB attribute



3 If you want to see information for another domain, specify the domain name, a user with sufficient rights to look up domain information, and that user's password, then click **Discover**.

4 When finished, click **Done**.

Creating the Driver in Designer

You create the Active Directory driver by installing the driver packages and then modifying the configuration to suit your environment. After you create and configure the driver, you need to deploy it to the Identity Vault and start it.

NOTE: Drivers are created with packages, and iManager does not support packages. In order to create drivers with the current version of Identity Manager, you must use Designer.

- "Importing the Current Driver Packages" on page 35
- "Installing the Driver Packages" on page 35
- "Configuring the Driver" on page 39

- "Deploying the Driver" on page 40
- "Starting the Driver" on page 41

Importing the Current Driver Packages

Driver packages can be updated at any time and are stored in the Package Catalog. Packages are initially imported into the Package Catalog when you create a project, import a project, or convert a project. It is important to verify you have the latest packages imported into the Package Catalog before you install the driver.

To verify you have the latest packages imported into the Package Catalog:

1. Open Designer.
2. In the toolbar click **Help** > **Check for Package Updates**.
3. Click **OK** if there are no package update

   or

   Click **OK** to import the package updates. If prompted to restart Designer, click **Yes** and save your project, then wait until Designer restarts.
4. In the Outline view, right-click the Package Catalog.
5. Click **Import Package**.
6. Select the Active Directory packages

   or

   Click **Select All** to import all of the packages displayed, then click **OK**.

   By default, only the base packages are displayed. Deselect **Show Base Packages Only** to display all packages.
7. Click **OK** to import the selected packages, then click **OK** in the successfully imported packages message.
8. After the current packages are imported, continue with "Installing the Driver Packages" on page 35.

Installing the Driver Packages

After you have imported the current driver packages into the Package Catalog, you can install the driver packages to create a new driver.

1. In Designer, open your project.
2. In the Modeler, right-click the driver set where you want to create the driver, then select **New** > **Driver**.
3. Select **Active Directory Base** from the list of base packages, then click **Next**.
4. Select the optional features to install for the Active Directory driver. All options are selected by default. The options are:
   - **Default Configuration:** This package contains the default configuration information for the Active Directory driver. Always leave this option selected.

- **Password Synchronization:** This packages contains the policies that enable the Active Directory driver to synchronize passwords. If you want to synchronize passwords, verify that this option is selected. For more information, see the *NetIQ Identity Manager Password Management Guide*.
- **Data Collection:** This package contains the policies that enable the driver to collect data for reports. If you are using the Identity Reporting Module, verify that this option is selected. For more information, see the *Administrator Guide to NetIQ Identity Reporting*.
- **Account Tracking:** This package contains the policies that enable you to track accounts for reports. If you are using the Identity Reporting Module, verify that this option is selected. For more information, see the *Administrator Guide to NetIQ Identity Reporting*.

5 Click **Next**.

6 (Conditional) If there are package dependencies for the packages you selected to install, you must install them to install the selected package. Click **OK** to install the package dependencies listed.

7 (Conditional) If more than one type of package dependency must be installed, you are presented with these packages separately. Continue to click **OK** to install any additional package dependencies.

8 (Conditional) If not already configured, fill in the following fields on the Common Settings page, then click **Next**:

- **User Container:** Select the Identity Vault container where Active Directory users will be added if they don't already exist in the vault. This value becomes the default for all drivers in the driver set.

  If you want a unique location for this driver, set the value for all drivers on this page. After the driver is created, change the value on the driver's Global Configuration Values page.
- **Group Container:** Select the Identity Vault container where Active Directory groups will be added if they don't already exist in the vault. This value becomes the default for all drivers in the driver set.

  If you want a unique location for this driver, set the value for all drivers on this page. After the driver is created, change the value on the driver's Global Configuration Values page.

NOTE: The Common Settings page is only displayed if the Common Settings package is a dependency.

---

9 (Conditional) If not already configured, fill in the following fields on the Common Settings Advanced Edition page, then click **Next**:

- **User Application Provisioning Services URL:** Specify the User Application Identity Manager Provisioning URL.
- **User Application Provisioning Services Administrator:** Specify the DN of the User Application Administrator user. This user should have the rights for creating and assigning resources. For more information, see "Setting Up Administrative User Accounts" in the *NetIQ Identity Manager Driver Administration Guide*.

NOTE: This page is only displayed if you installed the Common Settings Advanced Edition package.

---

10 On the Driver Information page, specify a name for the driver, then click **Next**.

11 On the Authentication Parameters page, fill in the following fields to authenticate to Active Directory and click **Next**:

- **Authentication ID:** Specify an Active Directory account with administrative privileges to be used by Identity Manager. The form of the name used depends on the selected authentication mechanism.

  For the **Negotiate** authentication method, provide the name form required by your Active Directory authentication mechanism. For example:

  - Administrator: Active Directory Logon Name
  - Domain/Administrator: Domain qualified Active Directory Logon Name
- **Password:** Provide the password for the specified Active Directory account.
- **Authentication Context:** Specify the name of the Active Directory domain controller to use for synchronization.

  For example, for the **Negotiate** method, use the DNS name (for example, `mycontroller.domain.com`). For the **Simple** authentication method, you can use the IP address of your server (for example, `10.10.128.23` or the DNS name).

  If no value is specified, `localhost` is used.

12 On the Remote Loader page, fill in the following fields to configure the driver to connect using the Remote Loader, then click **Next**:

- **Connect to Remote Loader:** By default, the driver is configured to connect using the Remote Loader. If you want to run the driver locally, select **no**, then click **Next**. Otherwise, fill in the remaining fields to configure the driver to connect by using the Remote Loader.
- **Host Name:** Specify the hostname or IP address of the server where the driver's Remote Loader service is running.
- **Port:** Specify the port number where the Remote Loader is installed and is running for this driver. The default port number is 8090.
- **KMO:** Specify the Key Name of the Key Material Object (KMO) that contains the keys and certificate the Remote Loader uses for an SSL connection. This parameter is only used when you use SSL for connections between the Remote Loader and the Identity Manager engine.
- **Other parameters:** Specify any other parameters required to connect to the Remote Loader. Any parameters specified must use a key-value pair format, as follows: `paraName1=paraValue1 paraName2=paraValue2`
- **Remote Password:** Specify the Remote Loader's password as defined on the Remote Loader. The Identity Manager server (or Remote Loader shim) requires this password to authenticate to the Remote Loader
- **Driver Password:** Specify the driver object password that is defined in the Remote Loader service. The Remote Loader requires this password to authenticate to the Identity Manager server.

13 On the Synchronization page, fill in the following fields to configure the driver's synchronization settings, then click **Next**:

- **Domain DNS Name:** Specify the DNS name of the Active Directory domain managed by this driver.

- **Subscriber Channel Placement Type:** Select the desired form of placement for the Subscriber channel. This option determines the Subscriber channel Placement policies.
  - **mirrored:** Places objects hierarchically within the base container
  - **flat:** Places objects only in the base container
- **Active Directory User Container:** Specify the container where user objects reside in Active Directory.
- **Publisher Channel Placement Type:** Select the desired form of placement for the Publisher channel. This option determines the Publisher channel Placement policies.
  - **mirrored:** Places object hierarchically within the base container
  - **flat:** Places objects only in the base container

14 (Conditional) On the Account Tracking page, fill in the following field, then click **Next**:

Realm: Specify the name of the realm, security domain, or namespace where the account name is unique.

NOTE: This page is only displayed if you installed the Account Tracking package.

---

15 Click **Next**.

16 (Conditional) On the General Information page, fill in the following fields to define your Active Directory system, then click **Next**:

- **Name:** Specify a descriptive name for this Active Directory system. The name is displayed in reports.
- **Description:** Specify a brief description for this Active Directory system. The description is displayed in reports.
- **Location:** Specify the physical location of this Active Directory system. The location is displayed in reports.
- **Vendor:** Leave Microsoft as the vendor of Active Directory. This information is displayed in reports.
- **Version:** Specify the version of this Active Directory system. The version is displayed in the reports.

NOTE: This page is only displayed if you installed the Managed System package.

---

17 (Conditional) On the System Ownership page, fill in the following fields to define the ownership of the Active Directory system, then click **Next**:

- **Business Owner:** Select a user object in the Identity Vault that is the business owner of the Active Directory system. This can only be a user object, not a role, group, or container.
- **Application Owner:** Select a user object in the Identity Vault that is the application owner of the Active Directory system. This can only be a user object, not a role, group, or container.

NOTE: This page is only displayed if you installed the Managed System package.

---

18 (Conditional) On the System Classification page, fill in the following fields to define the classification of the Active Directory system, then click **Next**:

- **Classification:** Select the classification of the Active Directory system. This information is displayed in the reports. The available options are:
  - Mission-Critical
  - Vital
  - Not-Critical
  - Other

    If you select **Other**, you must specify a custom classification for the Active Directory system.
- **Environment:** Select the type of environment the Active Directory system provides. The available options are:
  - Development
  - Test
  - Staging
  - Production
  - Other

    If you select **Other**, you must specify a custom classification for the Active Directory system.

NOTE: This page is only displayed if you installed the Managed System package.

---

19 Review the summary of tasks that will be completed to create the driver, then click **Finish**.

The driver is now created. You can modify the configuration settings, by continuing with the next section, "Configuring the Driver" on page 39. If you don't need to configure the driver, continue to "Configuring the Driver" on page 39.

Configuring the Driver

There are many settings that can help you customize and optimize the driver. The settings are divided into categories such as Driver Configuration, Engine Control Values, and Global Configuration Values (GCVs). Although it is important for you to understand all of the settings, your first priority should be to review the Driver Parameters located on the Driver Configuration page and the Global Configuration Values. These settings must be configured properly for the driver to start and function correctly.

To access the Driver Properties page:

1 Open your project.

2 In the Modeler, right-click the driver icon  or the driver line, then select **Properties**.

3 Click **Apply**.

4 Modify any other settings as necessary

In addition to the driver settings, you should review the set of default policies and rules provided by the basic driver configuration. Although these policies and rules are suitable for synchronizing with Active Directory, your synchronization requirements for the driver might

differ from the default policies. If this is the case, you need to change them to carry out the policies you want. The default policies and rules are discussed in "Default Driver Configuration" on page 14.

5 Click **OK** when finished.

6 Continue with "Deploying the Driver" on page 40.

Deploying the Driver

After a driver is created in Designer, it must be deployed into the Identity Vault.

1 In Designer, open your project.

2 In the Modeler, right-click the driver icon or the driver line, then select **Live** > **Deploy**.

3 If you are authenticated to the Identity Vault, skip to Step 5; otherwise, specify the following information:

Host: Specify the IP address or DNS name of the server hosting the Identity Vault.

Username: Specify the DN of the user object used to authenticate to the Identity Vault.

Password: Specify the user's password.

4 Click **OK**.

5 Read through the deployment summary, then click **Deploy**.

6 Read the success message, then click **OK**.

7 Click **Define Security Equivalence** to assign rights to the driver.

The driver requires rights to objects within the Identity Vault. The Admin user object is most often used to supply these rights. However, you might want to create a user account called `DriversUser`, for example, and assign security equivalence to that user. Whatever rights that the driver needs to have on the server, the `DriversUser` object must have the same security rights.

7a Click **Add**, then browse to and select the object with the correct rights.

7b Click **OK** twice.

8 Click **Exclude Administrative Roles** to exclude users that should not be synchronized.

You should exclude any administrative User objects (for example, Admin and DriversUser) from synchronization.

8a Click **Add**, then browse to and select the user object you want to exclude.

8b Click **OK**.

8c Repeat Step 8a and Step 8b for each object you want to exclude.

8d Click **OK**.

9 Click **OK**.

Starting the Driver

When a driver is created, it is stopped by default. To make the driver work, you must start the driver and cause events to occur. Identity Manager is an event-driven system, so after the driver is started, it won't do anything until an event occurs.

To start the driver:

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

For information about management tasks for the driver, see Chapter 7, "Managing Active Directory Groups and Exchange Mailboxes," on page 65.

Activating the Driver

The Identity Manager driver for Active Directory does not need a separate activation. If you create the driver in a driver set where you have already activated the Identity Manager server and service drivers, the driver inherits the activation from the driver set.

If you create the driver in a driver set that has not been previously activated, the driver will run in the evaluation mode for 90 days. You must activate the driver during the evaluation period; otherwise, the driver will be disabled. If you try to run the driver, `ndstrace` displays an error message indicating that you need to reactivate the driver to use it. For information on activation, refer to *NetIQ Identity Manager Setup Guide for Windows*.

Adding Packages to an Existing Driver

You can add new functionality to an existing driver by adding new packages to it.

1. Right-click the driver, then click **Properties**.
2. Click **Packages**, then upgrade the already installed Active Directory Base package.
   - **2a** Select the package from the list of packages, then click the **Select Operation** cell.
   - **2b** Click **Upgrade** from the drop-down list, then click **Apply**.
   - **2c** Click **OK** to close the Package Management page.

     You can upgrade the Password Synchronization package in a similar way.
3. Click the **Add Packages** icon .
4. Select the packages to install.
5. (Optional) If you want to see all available packages for the driver, clear the **Show only applicable package versions** option, if you want to see all available packages for the driver, then click **OK**.

   This option is only displayed on drivers. By default, only the packages that can be installed on the selected driver are displayed.
6. Click **Apply** to install all of the packages listed with the Install operation.
7. (Conditional) Fill in the fields with appropriate information to install the package you selected for the driver, then click **Next**.
8. Read the summary of the installation, then click **Finish**.

9 Click **OK** to close the Package Management page after you have reviewed the installed packages.

10 Modify the driver configuration settings. See “Configuring the Driver” on page 39.

11 Deploy the driver. See “Deploying the Driver” on page 40.

12 Start the driver. See “Starting the Driver” on page 41.

13 Repeat Step 1 through Step 9 for each driver where you want to add the new packages.

5 Upgrading an Existing Driver

If you are running the driver on the Identity Manager server, the driver shim files are updated when you update the server unless they were not selected during a custom installation. If you are running the driver on another server, the driver shim files are updated when you update the Remote Loader on the server.

This version of the driver shim supports drivers created by using any 3.*x* version of the driver configuration file. You can continue to use these driver configurations until you are prepared to start using packages.

The following sections provide information to help you upgrade an existing driver:

What's New

What's New in Version 4.1.1.0

This version of the driver does not provide any new features.

What's New in Version 4.1.0.0

The driver supports Subscriber Service channel. This channel enables you to separately process the out-of-band queries without interrupting the normal flow of cached events. For example, the Subscriber Service channel can separately process code map refresh, data collection, and queries triggered from dxcmd. This helps to improve the performance of the driver. For more information, see Improving Driver Performance Using Subscriber Service Channel in the *NetIQ Identity Manager Driver Administration Guide*.

Upgrading the Driver

The driver upgrade process involves upgrading the installed driver packages and updating the driver files. The driver patch file contains the software to update the driver files.

Upgrading the Installed Packages

1 Download the latest available packages.

To configure Designer to automatically read the package updates when a new version of a package is available, click **Windows** > **Preferences** > **NetIQ** > **Package Manager** > **Online Updates** in Designer. For more information, see *NetIQ Designer for Identity Manager Administration Guide*.

2 Upgrade the installed packages.

2a Open the project containing the driver.

2b Right-click the driver for which you want to upgrade an installed package, then click **Driver > Properties**.

2c Click **Packages**.

If there is a newer version of a package, there is check mark displayed in the Upgrades column.

2d Click **Select Operation** for the package that indicates there is an upgrade available.

2e From the drop-down list, click **Upgrade**.

2f Select the version that you want to upgrade to, then click **OK**.

NOTE: Designer lists all versions available for upgrade.

2g Click **Apply**.

2h (Conditional) Fill in the fields with appropriate information to upgrade the package, then click **Next**.

Depending on which package you selected to upgrade, you must fill in the required information to upgrade the package.

2i Read the summary of the packages that will be installed, then click **Finish**.

2j Review the upgraded package, then click **OK** to close the Package Management page.

For detailed information, see the Upgrading Installed Packages in the *NetIQ Designer for Identity Manager Administration Guide*.

Applying the Driver Patch

The driver patch updates the driver files.

This section provides a general procedure for updating the driver files. For instructions about updating the driver to a specific version, search for the driver patch that you want to upgrade to in the Patch Finder Download Page and follow the instructions from the Readme file that accompanies the driver patch release.

Prerequisites

Before installing the patch, complete the following steps:

1 Take a back-up of the current driver configuration.

2 (Conditional) If the driver is running locally, stop the driver instance and the Identity Vault.

3 (Conditional) If the driver is running with a Remote Loader instance, stop the driver and the Remote Loader instance.

4 In a browser, navigate to the NetIQ Patch Finder Download Page.

5 Under **Patches**, click **Search Patches**.

6 Specify **Identity Manager *nn* Active Directory driver nn** in the search box.

7 Download and unzip the contents of the patch file to a temporary location on your server.

For example, IDM45_ADDriver_4020.zip.

Applying the Patch

1 Update the driver files:

1a Navigate to the <*Driver Patch File Temporary Location*>\windows directory.

1b Copy the addriver.dll file to <IdentityManager installation>\Novell\NDS or <IdentityManager installation>\RemoteLoader\Novell\RemoteLoader folder.

1c Copy the adutil.jar file to <IdentityManager installation>\NDS\lib or <IdentityManager installation>\RemoteLoader\<architecture>\lib folder.

1d Navigate to the <Driver Patch File Temporary Location>\<architecture>nls folder and copy the PassSyncConfig.cpl file to <drive>\Windows\System32\nls folder.

- On a 32-bit operating system, copy the PassSyncConfig.cpl from the <Driver Patch File Temporary Location>\x86 folder to C:\Windows\System32 folder.
- On a 64-bit operating system, copy the PassSyncConfig.cpl from the <Driver Patch File Temporary Location>\x64 folder to C:\Windows\System32 folder.

2 Update the Password Sync Filter.

NOTE: You must reboot each Domain Controller for the changes to take effect. Therefore, check your current pwfilter.dll file version before starting the update. If the current version and the version shipped with the driver patch file are same, skip this step.

2a Verify the current version of your Password Sync Filter (pwfilter.dll).

2a1 On all Domain Controllers, browse to the <Drive>:\Windows\System32 folder.

2a2 Right-click the pwfilter.dll file.

2a3 Click Properties.

2a4 Click the **Details** tab and check the version of the file.

2b Update the Password Sync Filter files.

2b1 On each Domain Controller, rename the pwfilter.dll file to pwfilter.old.

2b2 Navigate to the <Driver Patch File Temporary Location>\<architecture> folder and copy the pwfilter.dll to \Windows\System32 folder.

Alternatively, run the Control Panel applet and check the filter status. Any old password sync filters should show as outdated and can be updated using that utility. A reboot of the Domain Controller is still needed because pwfilter.dll is loaded by the LSA process and that is only run at the startup of a server.

3 (Conditional) Copy the Exchange Service files.

This step is only required if you enabled the driver to synchronize Exchange data or if you want to use Active Directory PowerShell.

Your Exchange Service files must match the Microsoft Exchange version you are using.

Exchange Service Version	Microsoft Exchange Version
`IDM_PowerShell_Service`	Exchange 2016
`IDM_PowerShell_Service`	Exchange 2013

3a Stop the currently running Exchange service.

For example, `IDM_PowerShell_Service`.

3b Remove the Exchange service.

3c Copy the Exchange Service files.

3c1 Navigate to the `<Driver Patch File Temporary Location>\<architecture>` folder.

3c2 (Conditional) If the driver is running with the Remote Loader, copy `IDMEx<version>ManagementServer.dll` and `IDMEx<version>Service.exe` files to `<drive>\Novell\RemoteLoader` folder.

3c3 (Conditional) If the driver is running with the Identity Manager engine, copy `IDMEx<version>ManagementServer.dll` and `IDMEx<version>Service.exe` files to `<drive>\Novell\NDS` folder.

3d Install the Identity Manager Exchange service. For more information, see Appendix C, “Provisioning Exchange Accounts,” on page 113.

3e Start the Exchange Service.

3f Reboot each Domain Controller to apply the Password Sync Filter changes.

4 (Conditional) If the driver is running locally, start the Identity Vault and the driver instance.

5 (Conditional) If the driver is running with a Remote Loader instance, start the Remote Loader instance and the driver instance.

6 Synchronizing Passwords

To set up password synchronization among Active Directory, the Identity Vault, and connected systems, you need to complete the tasks in the "Password Management Checklist" in the *NetIQ Identity Manager Password Management Guide*. The information in the following sections supplements the information in that guide.

For information on troubleshooting password synchronization, see "Tips on Password Synchronization" on page 79.

Securing Driver Connections

For the driver to set a password in Active Directory (Subscriber channel), it must have a secure connection provided by one of the following conditions:

- **The remote loader runs on a domain controller:** Use the Negotiate authentication method to create the connection. The driver does not require connection security between the remote loader and Active Directory. The driver supports bi-directional password synchronization.
- **The remote loader runs on a member server:** Use the Negotiate authentication method to create the connection. The driver requires connection security between the remote loader and Active Directory, using either SSL or signing and sealing. The driver supports bi-directional password synchronization.
- **The remote loader runs on a server outside the domain:** Use the Simple authentication method to create the connection. The driver requires connection security using SSL between the remote loader and Active Directory. The driver supports password synchronization only on the Subscriber channel.

Configure the authentication method and enable SSL or signing and sealing in the driver parameters. For more information, see "Driver Parameters" on page 95.

Setting Up Password Synchronization Filters

The Active Directory driver must be configured to run on only one Windows server. However, for password synchronization to occur, you must install a password filter (`pwFilter.dll`) on each domain controller and configure the registry to capture passwords to send to the Identity Vault.

The password filter is automatically started when the domain controller is started. The filter captures password changes that users make by using Windows clients, encrypts the changes, and sends them to the driver to update the Identity Vault.

NOTE: ◆You do not need to install a password filter on a read-only domain controller.

- The Active Directory driver can detect whether a user account password is modified by an administrator or by the user themselves. Based on this information, the Identity Manager engine sets the password during synchronization using the administrator account or user account, as appropriate.
- Password filter allows you to specify multiple hosts.

To simplify installation and administration of password filters, an Identity Manager PassSync utility is added to the Control Panel when the driver is installed. This utility gives you two choices for setting up the password filters, depending on whether you want to allow remote access to the registry on your domain controllers:

- "Allowing Remote Access to the Registry" on page 48
- "Not Allowing Remote Access to the Registry" on page 52
- "Updating Password Sync Filter" on page 55

Allowing Remote Access to the Registry

If you allow remote access to the registry of each domain controller from the machine where you are running the driver, use the procedure in this section to configure the password filter. It allows the Identity Manager PassSync utility to configure each domain controller from one machine.

If you configure all the domain controllers from one machine, the Identity Manager PassSync utility provides the following features to help you during setup:

- Lets you specify which domain you want to participate in password synchronization.
- Automatically discovers all the domain controllers for the domain.
- Lets you remotely install the `pwFilter.dll` on each domain controller.
- Automatically updates the registry on the machine where the driver is running and on each domain controller.
- Lets you view the status of the filter on each domain controller.
- Lets you reboot a domain controller remotely.

  Rebooting the domain controller is necessary when you first add a domain for password synchronization, because the filter that captures password changes is a DLL file that starts when the domain controller is started.

Because setting up the filter requires rebooting the domain controller, you might want to perform this procedure after hours, or reboot only one domain controller at a time. If the domain has more than one domain controller, keep in mind that each domain controller where you want Password Synchronization to function must have the filter installed and must be rebooted.

1 Confirm that port 135 (the RPC endpoint mapper) is accessible on the domain controllers and on the machine where the Active Directory driver is configured to run.

   If you are using NetBIOS over TCP, you also need these ports:

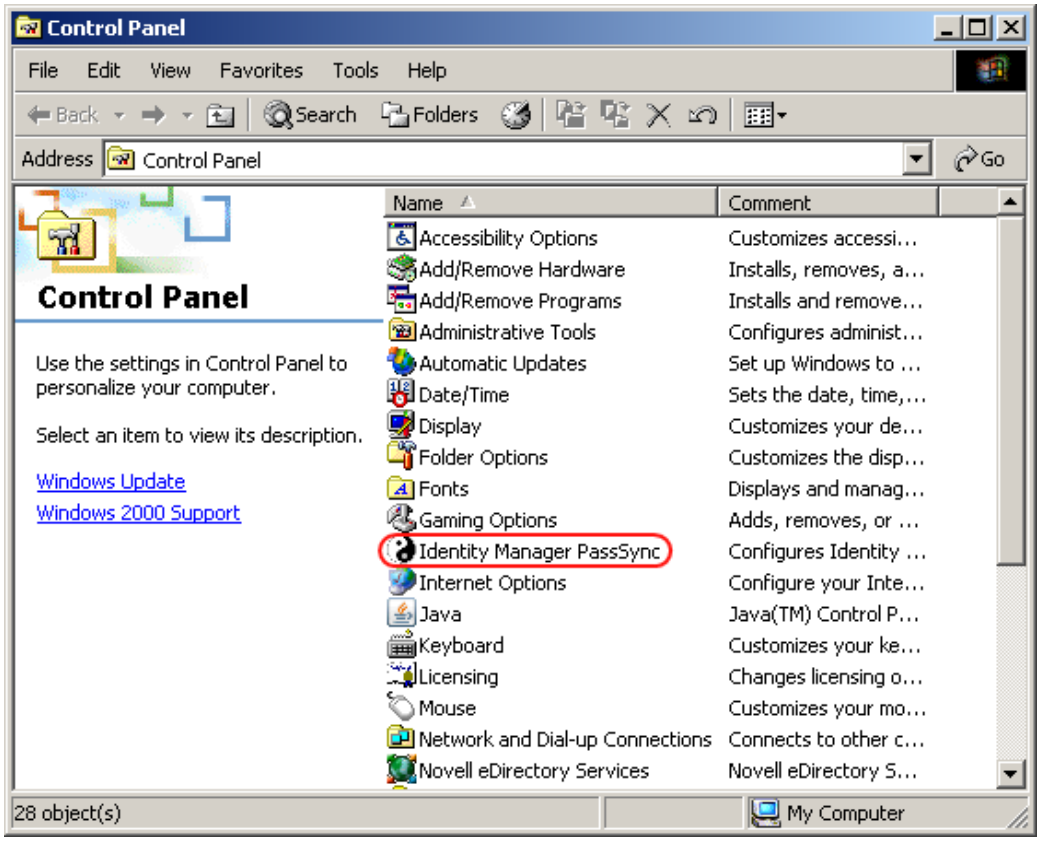
   - 137: NetBIOS name service
   - 138: NetBIOS datagram service
   - 139: NetBIOS session service

   A firewall could prevent the ports from being accessible remotely.

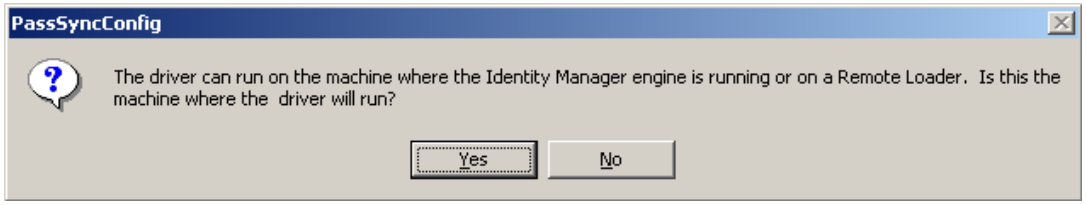
2 Log in with an administrator account on the computer where the driver is installed.

3 At the computer where the driver is installed, click **Start > Control Panel > Identity Manager PassSync.**

NOTE: Because there may be security policies in place that could block the PassSync utility from running, we recommend you run the utility using an account with Administrator privileges.



4 In the dialog box that is displayed, click **Yes** to specify that this is the machine where the driver is installed.

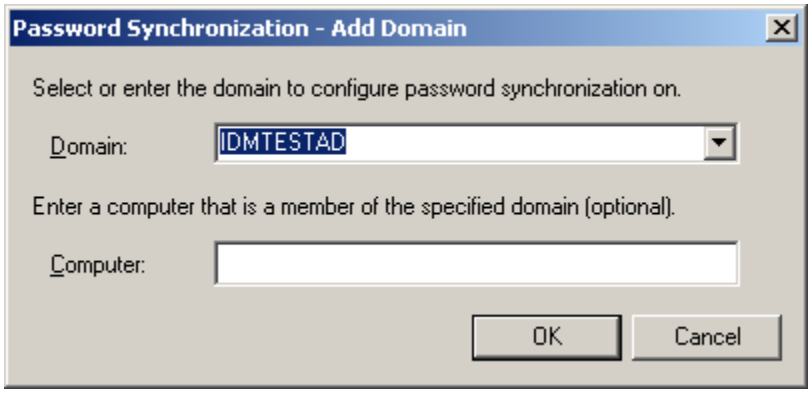


You only receive this prompt the first time you run the utility. After you complete the configuration, you are not shown this prompt again unless you remove this domain from the list.

5 Click **Add**, then browse to and select the domain that you want to participate in password synchronization.

The drop-down list displays known domains.

6 If no domains are listed, or if a 1208 error is displayed, you must manually type the domain name.



The Identity Manager PassSync utility discovers all the domain controllers for that domain, and installs `pwFilter.dll` on each domain controller. It also updates the registry on the computer where you are running the drivers, and on each domain controller. This might take a few minutes.

The `pwFilter.dll` doesn't capture password changes until the domain controller has been rebooted. The Identity Manager PassSync utility lets you see a list of all the domain controllers and the status of the filter on them. It also lets you reboot the domain controller from inside the utility.

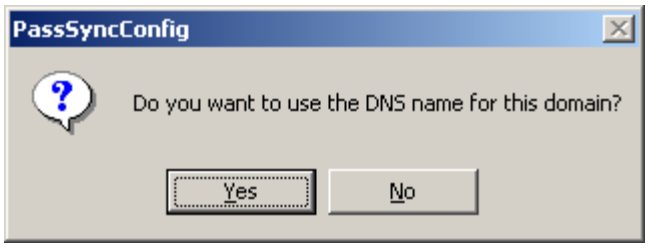
7 (Optional) Specify a computer in the domain, then click **OK**.

If you leave the **Computer** field blank, PassSync queries the local machine. Therefore, if you are running PassSync on a domain controller, you don't need to specify a name. PassSync queries the local machine (in this case, a domain controller) and gets (from the database) the list of all domain controllers in the domain.

If you aren't installing on a domain controller, specify the name of a computer that is in the domain and that can get to a domain controller.

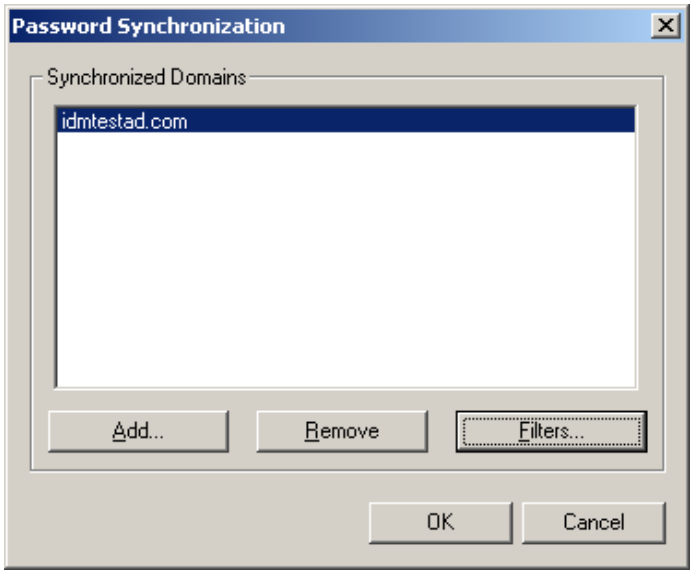
If you receive an error message indicating that PassSync can't locate a domain, specify a name.

8 Click **Yes** to use the domain's DNS name.

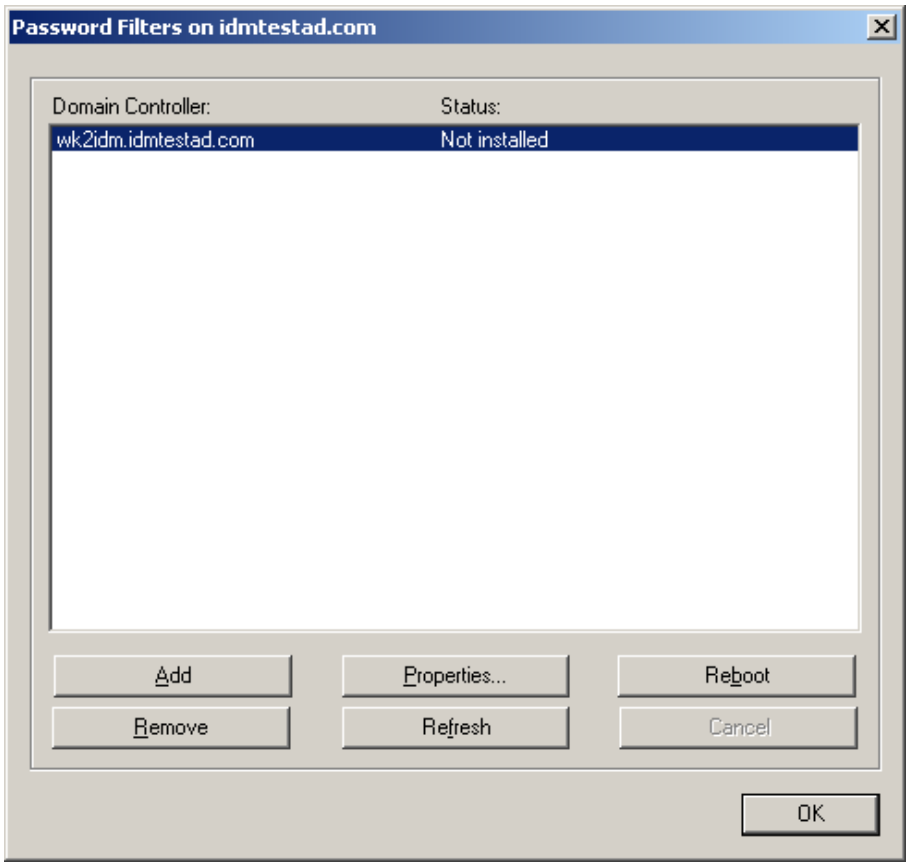


You can select **No**, but the DNS name provides more advanced authentication and the ability to more reliably discover domains in bigger installations. However, the choice depends on your environment.

9 Select the name of the domain you want to participate in password synchronization from the list, then click **Filters**.



The utility displays the names of all the domain controllers in the selected domain and the status of the filter.



The status for each domain controller should display the filter state as **Not installed**. However, it might take a few minutes for the utility to complete its automated task, and in the meantime the status might say **Unknown**.

10 To install the filter, click **Add**, then click **Reboot**.

You can choose to reboot the domain controllers at a time that makes sense for your environment. Just keep in mind that password synchronization won't be fully functional until every domain controller has been rebooted.

11 When the status for all domain controllers is **Running**, test password synchronization to confirm that it is working.

12 To add more domains, click **OK** to return to the list of domains, and repeat Step 5 through Step 11.

Not Allowing Remote Access to the Registry

If you do not want to allow remote access to the registry of each domain controller, you must set up the password filters on each domain controller separately. To do this, go to each domain controller, install the remote loader service so you have the Identity Manager PassSync utility, and use the utility on each machine to install the password filter and update the registry.

In the procedure in this section, you install the driver so that you have the Identity Manager PassSync utility. Then you use the utility to install the `pwFilter.dll` file, specify the port to use, and specify which host machine is running the Identity Manager Driver for Active Directory.

Because setting up the filter requires rebooting the domain controller, you might want to perform this procedure after hours, or reboot only one domain controller at a time. If a domain has more than one domain controller, keep in mind that each domain controller where you want Password Synchronization to function must have the filter installed and must be rebooted.

This procedure is for any domain controller that does not have the Active Directory driver installed on it.

1 Confirm that port 135 (the RPC endpoint mapper) is accessible on the domain controllers and on the machine where the Active Directory driver is configured to run:

   If you are using NetBIOS over TCP, you also need these ports:

   - 137: NetBIOS name service
   - 138: NetBIOS datagram service
   - 139: NetBIOS session service

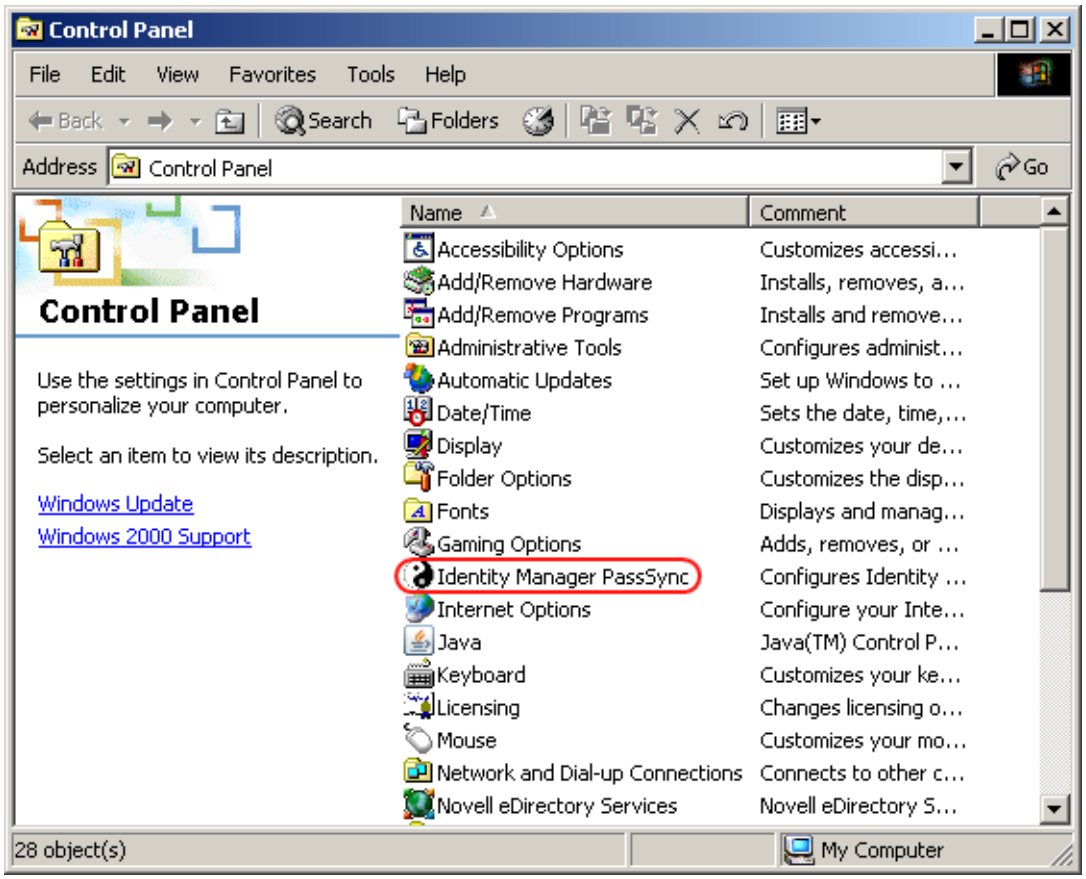
   A firewall could prevent the ports from being accessible remotely.

2 On the domain controller, install only the Active Directory driver. For more information, see "Planning Your Installation " in the *NetIQ Identity Manager Setup Guide for Windows*.

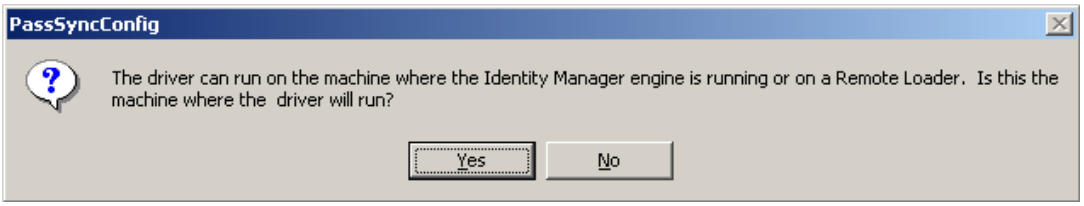
   Installing the driver installs the Identity Manager PassSync utility.

3 Click **Start > Settings > Control Panel > Identity Manager PassSync.**

NOTE: Because there may be security policies in place that could block the PassSync utility from running, we recommend you run the utility using an account with Administrator privileges.

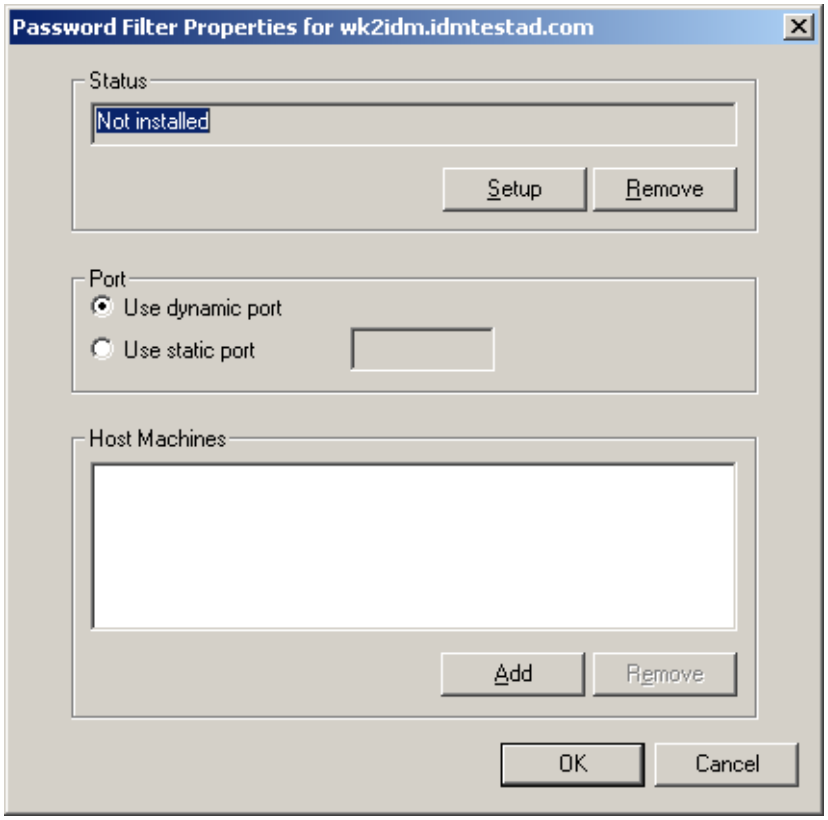


4 In the dialog box that displays, click **No** to specify that this machine is not running the Active Directory driver.



After you complete the configuration, you are not shown this prompt again unless you remove the password filter by using the **Remove** button in the Password Filter Properties dialog box.

After you click **No**, the Password Filter Properties dialog box appears, with a status message indicating that the password filter is not installed on this domain controller.

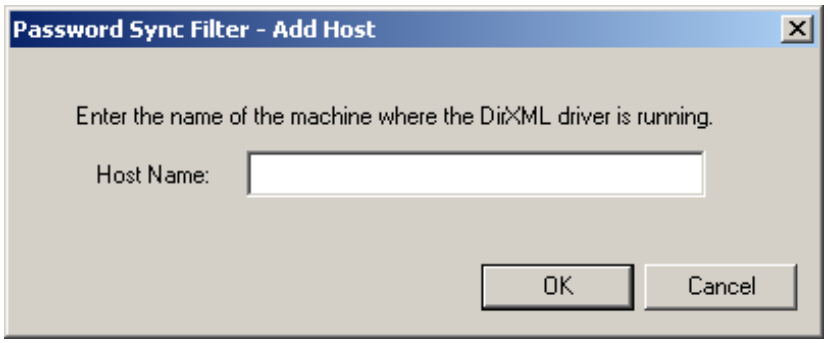


5 Click the **Setup** button to install the password filter, `pwFilter.dll`.

6 For the **Port** setting, specify whether to use dynamic port or static port.

Use the static port option only if you have decided to configure your remote procedure call (RPC) for the domain controller differently than the default.

7 Click **Add** to specify the hostname of the machine running the Identity Manager driver, then click **OK**.



This step is necessary so that the password filter knows where to send the password changes. The password filter captures password changes, and must send them to the Identity Manager driver to update the Identity Manager data store.

8 Verify that the information specified in Step 5 through Step 7 is correct, then click **OK**.

9 Reboot the domain controller to complete the installation of the password filter.

You can choose to reboot at a time that makes sense for your environment. Just keep in mind that password synchronization won't be fully functional until every domain controller has the password filter installed and has been rebooted.

After the installation is complete and the domain controller is rebooted, the password filter is loaded automatically whenever the domain controller starts.

10 Check the status for the password filter again by clicking **Start > Settings > Control Panel**, and double-clicking the Identity Manager PassSync utility.

Confirm that the status says Running.

11 Repeat Step 2 through Step 10 for each domain controller that you want to participate in password synchronization.

12 When the status says Running for all the domain controllers, test password synchronization to confirm that it is working by having a user change his or her password by using the Windows Client. This should initiate the synchronization process.

Updating Password Sync Filter

To update the Password Sync Filter files, perform the steps as shown below:

1 Copy the new `pwfilter.dll` from the extracted folder to `C:\Novell\IDM_PassSync\w64` and `C:\Windows\System32` folders on the server running remote loader.

2 Run the Control Panel applet and select domain from the list and click on Filter. In case the `pwfilter.dll` on the Domain Controller is older than the one present in `C:\Windows\System32` of server running Remote Loader then it will appear as outdated.

3 Remove the filter and reboot the Domain Controller.

4 After the Domain Controller restarts, add the new filter using the Control Panel applet on the Remote Loader Server.

5 Reboot the Domain Controller.

Retrying Synchronization after a Failure

The following sections explain the retry methods used after a synchronization failure:

Retrying after an Add or Modify Event

If a password change sent from Active Directory is not successfully completed in the Identity Vault, the driver caches the password. It is not retried again until an Add or Modify event occurs for the user that the password belongs to. (Previously, these saved passwords were retried at every polling interval.)

When the driver polls for changes in Active Directory, the driver receives Add or Modify events for users. For each user Add or Modify event, the driver checks to see if it has a password saved for this new user. If it does, the driver sends the password to the Identity Vault as a Modify user event.

If you have set up password synchronization to send e-mail messages to users when password synchronization fails, this enhancement minimizes the number of e-mails that a user might receive.

Password Expiration Time

The Password Expiration Time parameter lets you determine how long to save a particular user's password if synchronization is not successful on the first try. The driver saves a password until it is successfully changed in the Identity Vault, or until the Password Expiration Time elapses.

You are prompted to specify an expiration time when you import the sample driver configuration. If you don't specify a time, or if the interval field contains invalid characters, the default setting is 60 minutes. If the time specified is less than three times the polling interval specified, the driver changes the time to be three times the polling interval.

Set the value large enough to handle whatever temporary backlog of passwords exists. If you are doing bulk changes, set the timeout large enough to handle all the changes. The rule of thumb is to allow one second per password. For example, to synchronize 18,000 passwords, allow 300 minutes (18,000 passwords divided by 60 seconds).

A setting of -1 is indefinite. Although this setting can handle bulk changes, it can cause problems. For example, a password might never be synchronized because the account wasn't associated. Such a password would remain in the system forever. A number of similar situations could result in a large inventory of unsynchronized passwords held by the system.

Scenarios Relating to Password Expiration Time

On the Publisher channel, password synchronization might occur before the Add event. The driver retries immediately following the Add event.

- “Scenario: Never Meeting Requirements” on page 58
- “Scenario: E-Mail Notifications” on page 58

Scenario: No Effect

A new user with a password is created in Active Directory. The filter immediately sends the new password to the driver. However, the driver hasn’t yet received that user Add event because the event occurred between polling intervals. Because the driver has not yet created the user in the Identity Vault, the password synchronization is not successful on this first attempt. The driver caches the password.

At the next polling interval, the driver receives the Add user event for the new user. The driver also checks to see if it has a password cached for this new user. The driver sends the Add user event to the Identity Vault, and also sends a Modify user event to synchronize the password.

In this case, the password synchronization is delayed by only one polling interval.

The Password Expiration Time parameter doesn’t have an effect in this situation.

Scenario: Increasing the Expiration Time

A new user with a password is created in Active Directory. However, the user information doesn’t meet the requirements of the Create policy for the Active Directory driver.

For example, perhaps the Create rule requires a full name, and the required information is missing. Like the No Effect example, the filter immediately sends the password change to the driver. However, on the first try the password change is not successful in the Identity Vault because the user doesn’t exist yet. The driver caches the password.

In this case, however, even when the driver polls for changes in Active Directory and discovers the new user, the driver can’t create the new user because the user information doesn’t meet the Create policy’s requirements.

Creating the new user and synchronizing the password are delayed until all the user information is added in Active Directory to satisfy the Create policy. Then the driver adds the new user in the Identity Vault, checks to see if it has a password cached for this new user, and sends a Modify user event to synchronize the password.

The Password Expiration Time parameter affects this scenario only if the time interval elapses before the user information in Active Directory meets the requirements of the Create policy. If the Add event comes in after the password has expired and the driver doesn’t have the password cached for that user, synchronization can’t occur. Because the driver doesn’t have a cached password, the driver uses the default password in the password policy.

After the user changes the password in either Active Directory or the Identity Vault, that password is synchronized.

If Password Synchronization is set up for bidirectional flow of passwords, a password can also be synchronized from the Identity Vault to Active Directory when a password change is made in the Identity Vault.

If your Create policy is restrictive, and it generally takes longer than a day for a new user’s information to be completed in Active Directory, you might want to increase the Password Expiration Time parameter interval accordingly. The driver can then cache the passwords until the user is finally created in the Identity Vault.

Scenario: Never Meeting Requirements

A user with a password is created in Active Directory. However, this user never meets the criteria of the Create policy for the Active Directory driver.

For example, perhaps the new user in Active Directory has a Description that indicates the user is a contractor, and the Create policy blocks creation of User objects for contractors because the business policy is that contract employees are not intended to have a corresponding user account in the Identity Vault. Like the previous example, the filter immediately sends the password change, but the password synchronization isn't successful on the first attempt. The driver caches the password.

In this case, a corresponding user account is never created in the Identity Vault. Therefore, the driver never synchronizes the cached password. After the Password Expiration Time has passed, the driver removes the user password from its cache.

Scenario: E-Mail Notifications

Markus has an Active Directory account and a corresponding Identity Vault account. He changes his Active Directory password, which contains six characters. However, the password doesn't meet the eight-character minimum required by the Password policy that the administrator created in eDirectory. Password Synchronization is configured to reject passwords that do not meet the policy and to send a notification e-mail to Markus saying that password synchronization failed. The driver caches the password and retries it only if a change is made to the User object in Active Directory.

In this case, shortly after changing a password, Markus receives an e-mail stating that the password synchronization wasn't successful. Markus receives the same e-mail message each time the driver retries the password.

If Markus changes the password in Active Directory to one that complies with the Password policy, the driver synchronizes the new password to the Identity Vault successfully.

If Markus doesn't change to a compliant password, the password synchronization is never successful. When the Password Expiration Time elapses, the driver deletes the cached password and no longer retries it.

Disabling Password Synchronization on a Driver

You can disable password synchronization on a driver by setting the **Password Sync Timeout** parameter to 0. Sometimes there is a need to have two Active Directory drivers enabled for one domain, but you only want one driver handling the password synchronization. Make sure that the **Password Sync Timeout** parameter is set to 0 on the driver that does not synchronize passwords.

A use case for this is if one driver is synchronizing User objects and another driver is synchronizing Contacts. Contacts are displayed in the Exchange Global Address List (GAL), but they do not require an Active Directory license because they do not authenticate.

See "Password Sync Timeout (minutes):" on page 97 for more information about this parameter.

Diagnosing Password Synchronization Issues

Identity Manager provides the PassSync Troubleshooting Tool to diagnose issues encountered during password synchronization. This tool is a standalone executable that collects the following information to help you analyze synchronization issues:

- Domain Controller information
- Password filter details
- RPC connection details

Ensure you have the appropriate permissions to log in to this tool. For more information, see "Logging In with Right Permissions" on page 63.

You must launch this tool on the computer where Active Directory driver is installed. For more information, see "Verifying Remote Loader is Locally Available to PassSync Tool" on page 63.

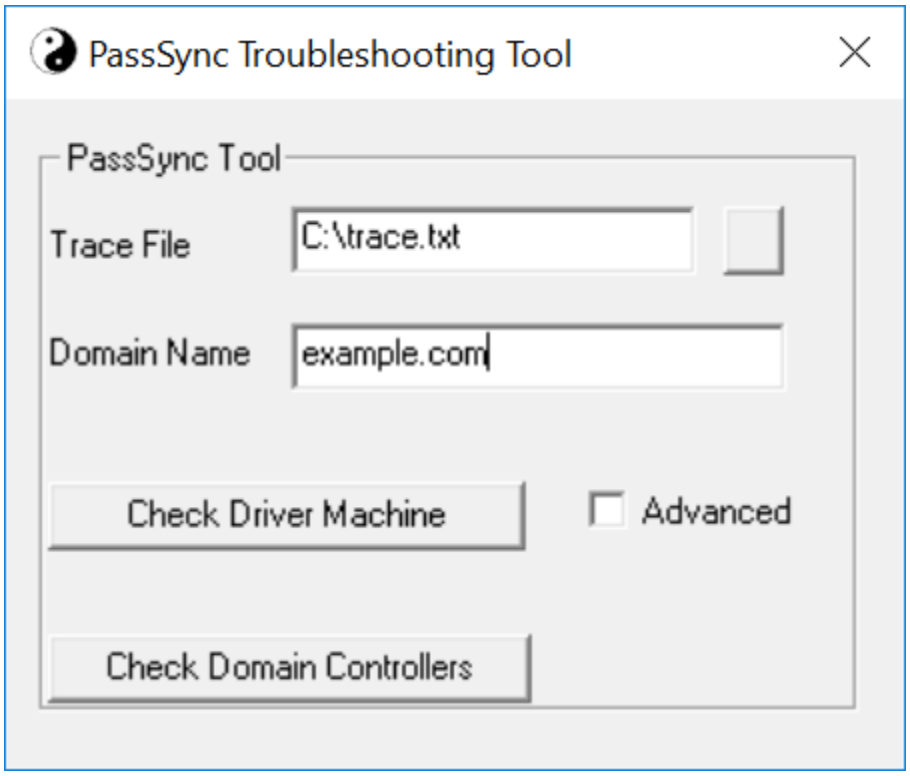
This tool is available in the Identity Manager utilities folder located at:

`\products\IDM\windows\setup\utilities\PassSyncTroubleshootingTool`

Using PassSync Troubleshooting Tool

1. Launch PassSync Troubleshooting Tool.
2. Specify the following details:

Field	Description
Trace File	Specifies the location of the trace file where you want to store the trace messages. If you do not specify a path, the file is created in the same directory from where you launched the tool.
Domain Name	Specifies the name of the Active Directory domain you are synchronizing passwords to and from.



3 Perform the following actions in any order:

3a Click **Check Driver Machine** and specify the credentials. For more information, see "Verifying the Driver Machine Information" on page 60.

3b Click **Check Domain Controllers** and specify the credentials. For more information, see "Verifying the Domain Controller Information" on page 61.

NOTE: If you do not log in with right permissions, it reports an error. For more information, see "Logging In with Right Permissions" on page 63.

When you click **Check Driver Machine** and **Check Domain Controllers**, the trace information is stored in the file specified in Step 2.

Verifying the Driver Machine Information

The **Check Driver Machine** option provides the following information about drivers that are installed on a particular domain:

- **RPC Service:** Establishes a remote connection with other computers. The RPC service status in the trace indicates whether the RPC service is running on the computer.

  You must have administrative access to start the RPC service. Perform the following actions to start the RPC service:

  1. Right-click on your **Start** button and click **Run**.
  2. Type `Services.msc` and click **OK**.
  3. Right-click **Remote Procedure Call (RPC)** and select **Start**.

- **Driver Instances:** Provides driver file path, connection details, and driver version. It also provides information about the driver instances running on the Remote Loader.
- **Registry Information:** Displays registry key values of the computer running the driver and domain.

The following is a sample trace output displaying these parameters for `example.com` domain:

```
Fri Aug 17 02:00:31 2018 : Starting Checks on Driver Machine .....

Fri Aug 17 02:00:34 2018: Logging as default user.

Fri Aug 17 02:00:34 2018 :
The List of all Domain Controllers -
1. WIN-LIDKNP4JGO5.example.com

Fri Aug 17 02:00:34 2018 : RPC Service is running
Fri Aug 17 02:00:34 2018 : Full DNS name of the driver machine is WIN-
LIDKNP4JGO5.example.com

Fri Aug 17 02:00:34 2018 : The version of the Operating System is :
Microsoft  (build 9200)
Fri Aug 17 02:00:34 2018 : An AD driver instance is found configured on
Remote Loader
Fri Aug 17 02:00:34 2018 : AD Driver which is configured with Connection
port 8090 and Command port 8000 is running

Fri Aug 17 02:00:34 2018 : List of local files related to Driver are :
        C:\novell\remoteloader\64bit\ADDriver.dll
        C:\novell\remoteloader\64bit\ad-driver-Config.txt
        C:\novell\remoteloader\64bit\ad-driverexample.com-Trace.log
Fri Aug 17 02:00:35 2018 : Driver version is "4.1.0.0">AD</
pr"20180125_120000"</cook
Fri Aug 17 02:00:35 2018 : Driver version is c1fe230"/> and Build ID is
"20180125_120000"</cook
Fri Aug 17 02:00:35 2018 : Driver version is "4.1.0.0">
Fri Aug 17 02:00:35 2018 : The 'Driver Machine' value in the registry
key[SOFTWARE\NOVELL\PASSSYNC] is : 1.

Fri Aug 17 02:00:35 2018 : Number of subkeys(passwords cached) under the
key[SOFTWARE\NOVELL\PASSSYNC\DATA\example.com]is 1


Fri Aug 17 02:00:35 2018 : Tests on this driver machine are done

Press any key to close this trace ...
```

Verifying the Domain Controller Information

The **Check Domain Controllers** option provides the following information about domain controller servers within a server domain:

- **Basic Diagnostic Checks:** Displays the password filter version on each domain controller server. It also displays the hostname of the domain controller server and the computer where the driver is running.

- **RPC Checks:** Displays information whether domain controller servers and drivers are able to connect to password filters via RPC tool.

The following is a sample trace output displaying these parameters for `example.com` domain:

```
Sun Aug 19 22:04:40 2018 : Starting Checks on All DCs .....

Sun Aug 19 22:04:41 2018: Logging as default user.

Sun Aug 19 22:04:41 2018 :
The List of all Domain Controllers -
1. WIN-LIDKNP4JGO5.example.com

Sun Aug 19 22:04:41 2018 : Checking the Domain Controller WIN-
LIDKNP4JGO5.example.com ....

Running Basic Diagnostic Checks.

Password filter files installed on this DC are
C:\Windows\System32\PWFILTER.DLL and C:\Windows\System32\PSEVENT.DLL

This 64 bit System has INCORRECT 32 bit PWFILTER dll version v3.0.0
(20180117) installed

The value of 'Host Names' '[WIN-LIDKNP4JGO5.example.com]' in DC[WIN-
LIDKNP4JGO5.example.com] is same as the name of driver machine[WIN-
LIDKNP4JGO5.example.com]

Opened key [SOFTWARE\NOVELL\PWFILTER\DATA].
No items to process.

Running RPC Checks.

Checking whether this tool can reach the filter through RPC
This tool can reach the filter through RPC

Checking if the filter can connect to the driver
pwFilter can connect to PassSync RPC server on driver machine  - 0

Sun Aug 19 22:04:42 2018 : Tests on all DCs are done

Press any key to close this trace ...
```

Troubleshooting Tips

Ensure the following conditions are met when driver is remotely installed:

- All Active Directory servers belong to the same domain that is hosting the Remote Loader server.
- RPC service is running and able to connect to PWfilter modules of that Active Directory server.

  To verify the status of RPC service and the number of driver instances running in your domain, see “Verifying the Driver Machine Information” on page 60.

Additionally, the following actions can help you troubleshoot the issues:

Specifying the Registered Domain Name

This tool can only analyze the domains that are registered to the driver computer. If you specify an unregistered domain, it displays the following error in the driver machine trace:

```
No Such Domain.
```

Therefore, always specify the registered domain name to this tool.

Verifying Remote Loader is Locally Available to PassSync Tool

The following error occurs if the Active Directory driver is configured with the Remote Loader and the PassSync tool is launched from a different computer:

```
Error occured while opening the registry key [SOFTWARE\NOVELL\RLCONSOLE].
```

Therefore, you must launch the PassSync tool on the Remote Loader computer where the Active Directory driver is running.

Using Out of Band Sync

Enable Out of Band Sync attribute for the password change event. This setting processes the password change event before other events in the queue. For more information, see Enabling Out of Band Sync in *NetIQ Identity Manager Driver Administration Guide*.

Enabling the Password Synchronizing Driver Instance to Use RPC Service

You can configure one or more Active Directory driver instances on one Remote Loader. An Active Directory driver instance that you want to synchronize the password require the RPC service to establish a remote connection with the domain controller servers. Therefore, it is recommended to set a delay at the startup for the remaining instances so that the required Active Directory driver instance can use the RPC service to synchronize the passwords in a registry key.

After making the changes to the key, restart the Windows server.

Logging In with Right Permissions

If you do not log in to the server with right permissions, it reports an access denied error. For example, if you log in without the domain administrator rights, it displays the following error when running the domain controller check:

```
Error occurred while opening the registry
key[SOFTWARE\NOVELL\PWFILTER\DATA]. Access is denied.
```

To resolve this issue:

1. Run **regedit** and right click the **HKLM\Software\Novell\PwFilter\Data** key.
2. Select **Permissions**.
3. Select **Advanced** and add **Administrators Group**.
4. Set the **Read** permission.
5. Verify that **Replace all child object permission entries with inheritable permission entries from this object** is selected.

7 Managing Active Directory Groups and Exchange Mailboxes

The following sections provide information to help you use the Active Directory driver to manage groups and Exchange mailboxes that reside in Active Directory:

Managing Groups

The Active Directory group class defines two types of groups and three scopes for membership in the group. Type and scope are controlled by the groupType attribute, which can be set via an Identity Manager policy when a group is created in Active Directory and changed by modifying the attribute.

A group holds a collection of object references. The Distribution Group type gives no special rights or privileges to its members and is commonly used as a distribution list for Exchange. The Security Group type is a security principal. Its members receive the rights and privileges of the group. Security Groups have a pre-Windows 2000 logon name (samAccountName) and a Security Identifier (SID) that can be used in Security Descriptor (SD) Access Control Lists (ACL) on other objects to grant or deny rights and privileges to its members.

Group scope controls whether an object from a foreign domain can be a member of the group and also whether the group itself can be a member of another group. The three scopes are Domain Local, Global, and Universal.

In general, Domain Local groups can hold references to objects anywhere in the forest but can be assigned permissions only within the domain. Global groups are the opposite. They can only hold references to objects within the domain but can be assigned permissions throughout the forest. Universal groups can hold references and can be assigned permissions throughout the forest. However, Universal groups come with their own restrictions and performance issues. Groups should be created and used in conformance with Microsoft recommendations.

The groupType attribute is a 32-bit integer whose bits define type and scope. Groups can have only a single scope at any given time.

Table 7-1 GroupType Attribute

GroupType Attribute	Scope	Bits That Define Type and Scope
GROUP_TYPE_GLOBAL_GROUP	Distribution	0x00000002
GROUP_TYPE_DOMAIN_LOCAL_GROUP	Distribution	0x00000004
GROUP_TYPE_UNIVERSAL_GROUP	Distribution	0x00000008
GROUP_TYPE_SECURITY_ENABLED	Security	0x80000000

Managing Microsoft Exchange Mailboxes

The Active Directory driver can be configured to create, move, and delete Microsoft Exchange mailboxes for users in Active Directory. Mailboxes are managed by setting and removing the value for the homeMDB attribute on the user object. This attribute holds the Distinguished Name of the Exchange Private Message Database (MDB) where the mailbox resides. The driver manages mailboxes on Exchange servers that are in the same domain as the driver only.

There are several different ways to manage Exchange mailboxes. The default configuration manages mailboxes through policy decisions made in the Subscriber Command Transformation policy. When a user meets the given conditions, a mailbox is created, moved, or removed. The import file gives you three choices for mailbox management:

- Policies
- Do not Manage Exchange Mailboxes

When you use the entitlement method for provisioning, a user is granted or denied a mailbox based on the entitlement set on the user in the Identity Vault. The entitlement holds the Distinguished Name of the MDB and a state value that tells the driver whether the entitlement is granted or revoked. The entitlement itself is managed by the User Application or the Role-Based Entitlements driver. In either case, the external tool grants (or revokes) the right to the mailbox, the Subscriber Command Transformation policy translates that right into an add-value or remove-value on the homeMDB attribute and the driver shim translates the change to homeMDB into the proper calls to the Exchange management system.

When you use the policy-based method for provisioning, the Subscriber Command Transformation policy uses information about the state of the user object in the Identity Vault to assign the MDB. The driver shim translates the change into the proper calls to the Exchange management system. The default policy uses a simple rule for assigning the mailbox. It assumes that there is only one MDB and that all users that have come this far through the policy chain should be assigned to that MDB.

Because the rules for assigning different MDBs vary widely from company to company, the default configuration does not attempt to establish a “right way” of doing it. You implement your own policies simply by changing the default assignment rules. You use DirXML Script if statements to define the conditions for mailbox assignments and the `do-set-dest-attribute` command for the homeMDB attribute to effect the change. You can get a list of Exchange MDBs by using the `ADManager.exe` tool or by your own means.

When it is not managing Exchange mailboxes, the driver synchronizes the user’s e-mail address and mail nickname.

There are other ways to manage the Exchange mailbox. For instance, you could extend the schema of the Identity Vault to hold the homeMDB information and use basic data synchronization to assign the mailbox to the user in Active Directory. In this case, you would use your own tool to make assignments in the Identity Vault.

The default policy works well for simple mailbox assignment to a single MDB. If you want the policy to reflect more complex rules demanded in your environment, the policy must be changed.

8 Managing the Driver

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

- Starting, stopping, and restarting the driver
- Viewing driver version information
- Using Named Passwords to securely store passwords associated with the driver
- Monitoring the driver's health status
- Backing up the driver
- Inspecting the driver's cache files
- Viewing the driver's statistics
- Using the DirXML Command Line utility to perform management tasks through scripts
- Securing the driver and its information
- Synchronizing objects
- Migrating and resynchronizing data
- Activating the driver
- Upgrading an existing driver

Because these tasks, as well as several others, are common to all Identity Manager drivers, they are included in one reference, the *NetIQ Identity Manager Driver Administration Guide*.

9 Security Best Practices

This following sections contains a description of the security parameters unique to the Active Directory driver.

NOTE: Exchange and Powershell service must be run with lease privilege required for exchange cmdlets and the configured AD powershell cmdlets.

Also only system admin and IDM administrator should be allowed to access the machine where AD driver and powershell service is running.

For additional information about securing your Identity Manager system, see the *NetIQ Identity Manager Security Guide*.

Default Configuration of the Security Parameters

The security parameters must be configured correctly for the driver to function properly. In most instances, the driver does not start if the parameters are not configured correctly.

To change these parameters in iManager:

1. Click **Identity Manager > Identity Manager Overview**, then click **Search** to search for the driver set that is associated with the driver.
2. Browse to the driver, then click the upper right corner of the driver icon.
3. Click **Edit Properties > Driver Configuration > Driver Parameters**.
4. Review the driver parameters in Table 9-1, and decide if you need to make any changes.

To change these parameters in Designer:

1. Open a project in the Modeler, then right-click the driver line and select **Properties > Driver Configuration.**
2. Click **Driver Parameters**.
3. Review the driver parameters in Table 9-1, and decide if you need to make any changes.

Table 9-1 *Security Parameters*

Security Parameter	Description
Authentication ID	The account the driver uses to access the domain data. The **Authentication ID** can be specified by using different formats: ◆ If the **Authentication method** is set to **negotiate**, the user name is specified with the domain name or the full qualified domain name. For example, `user` or `domain\user`. ◆ If the **Authentication method** is set to **simple**, the user name must be specified using an LDAP fully distinguished name. For example, `cn=IDMadmin,cn=Users,dc=domain,dc=com`.
Authentication context	The context used to access domain data. The **Authentication context** can be specified by using different formats: ◆ If the **Authentication method** is set to **negotiate**, use the DNS name of the Active Directory domain controller. For example, `mycontroller.mydomain.com`. ◆ If the **Authentication method** is set to **simple**, use the DNS name of the Active Directory domain controller or the IP address of the LDAP server. For example, `mycontroller.mydomain.com` or `10.0.0.1`.
Application password	The password for the **Authentication ID** account.
Authentication Method	The method of authentication to Active Directory. **Negotiate** uses Microsoft's security package to negotiate the logon type. Typically Kerberos or NTLM is selected. **Simple** uses LDAP style simple bind for logon. If you want to use Password Synchronization, select **Negotiate**.
Digitally sign communications	This setting enables signing on a Kerberos or NTLM v2 authenticated connection between the driver shim and the Active Directory database. Signing ensures that a malicious computer is not intercepting data. This does not hide the data from view on the network, but it reduces the chance of security attacks. Signing only works when you use the Negotiate authentication method and the underlying security provider selects NTLM v2 or Kerberos for its protocol. Select **Yes** to digitally sign the communication between the driver shim and Active Directory. Do not use this option with SSL. Select **No** if you do not want to sign communication between the driver shim and the Active Directory database.

Security Parameter	Description
Digitally sign and seal communications	This setting enables encryption on a Kerberos or NTLM v2 authenticated connection between the driver shim and the Active Directory database. Sealing encrypts the data so that it cannot be viewed by a network monitor. Sealing only works when you use the Negotiate authentication method and the underlying security provider selects NTLM v2 or Kerberos for its protocols. Select Yes to digitally encrypt communication between the driver shim and the Active Directory database. Do not use this option with SSL. Select No if you do not want to sign and seal communication between the driver shim and the Active Directory database.
Use SSL for encryption	Select Yes to digitally encrypt communication between the driver shim and the Active Directory database. This option can be used with Negotiate or Simple authentication methods. SSL requires that the Microsoft server running the driver shim imports the domain controller's server certificate. For more information, see "Microsoft Security Compliance Manager" (http://technet.microsoft.com/en-us/library/cc677002.aspx). By default, the parameter is set to No. If you set this value to Yes, the SSL pipe is encrypted for the entire conversation. An encrypted pipe is preferred because the driver typically synchronizes sensitive information. However, encryption slows the general performance of your servers.
Logon and impersonate	Select Yes to log on and impersonate the driver authentication account for CDOEXM (Collaboration Data Object for Exchange Management) and Password Set support. The driver performs a local logon. The authentication account must have the proper rights assignment. For more information, see "Creating an Administrative Account" on page 26. If No is selected, the driver performs a network logon only.

Recommended Security Configurations for the Remote Loader

If you are using the Remote Loader, the following table lists the recommended security configurations for the driver.

Table 9-2 *Recommended Security Configuration for the Remote Loader*

Parameter	Description and Recommended Setting
Authentication ID	The account the driver uses to access the domain data. Use the domain logon name, for example Administrator.

Parameter	Description and Recommended Setting
Authentication Context	The DNS name of the domain controller. If you don't want to run the driver on your Active Directory domain controller, use *hostname* for the Negotiate method but use *hostname* or the IP address for the **simple** method.
Application Password	The password used for the **Authentication ID**.
Remote Loader Password	The password for the Remote Loader service.
Authentication Method	Select **negotiate**.
Digitally sign communications	In most environments, we recommend you select **No** for this option and use SSL to secure communication between the Remote Loader and the domain controller. However, if the Remote Loader is installed on a member server, and you need to synchronize passwords, select **Yes** for this option. Do not use this option with SSL.
Digitally sign and seal communications	In most environments, we recommend you select **No** for this option and use SSL to secure communication between the Remote Loader and the domain controller. However, if the Remote Loader is installed on a member server, and you need to synchronize passwords, select **Yes** for this option. Do not use this option with SSL. **NOTE:** Sealing only works when you use the **Negotiate** authentication method and the underlying security provider selects NTLM v2 or Kerberos for its protocols.
Use SSL for encryption	Select **Yes** if Remote Loader is on a member server. If Remote Loader is on a domain controller, select **No**. SSL is required to perform a Subscriber password check, a Subscriber password set, and a Subscriber password modify operation when the driver shim is not running on the domain controller. SSL requires that the Microsoft server running the driver shim imports the domain controller's server certificate. For more information, see "Microsoft Security Compliance Manager" (http://technet.microsoft.com/en-us/library/cc677002.aspx). By default, the parameter is set to **No**. If you set this value to **Yes**, the SSL pipe is encrypted for the entire conversation. An encrypted pipe is preferred because the driver typically synchronizes sensitive information. However, encryption slows the general performance of your servers.

Recommended Security Configurations for the Simple Authentication Method

SSL is recommended if you have selected the Simple authentication mechanism because Simple authentication passes passwords in clear text.

Table 9-3 *Recommended Security Configuration for the Simple Authentication Method*

Parameter	Description and Recommended Setting
Authentication ID	The account the driver uses to access the domain data. Use LDAP format for the **Authentication ID**. For example, cn=IDMadmin,cn=Users,dc=domain,dc=com
Authentication Context	IP address of domain controller.
Password	The password for the specified **Authentication ID**.
Digitally sign communications	Select **No**.
Digitally sign and seal communications	Select **No**.
Use SSL for encryption	Select **Yes**. SSL requires that the Microsoft server running the driver shim imports the domain controller's server certificate imported. For more information, see "Microsoft Security Compliance Manager" (http://technet.microsoft.com/en-us/library/cc677002.aspx).

Recommended Security Configuration for Powershell Service

- Powershell service must be run by the same user who has been configured for driver authentication.
- Local login to the machine where the driver and Powershell services are running must be restricted to the user who has been configured for driver authentication.

10 Troubleshooting

Refer to the following sections if you are experiencing a problem with the Active Directory driver.

Changes Are Not Synchronizing from the Publisher or Subscriber

To synchronize changes in Active Directory, the account used by the Identity Manager driver must have the proper rights set up. For information on the necessary rights, see "Creating an Administrative Account" on page 26.

If you use the default policies, you must also meet the requirements for the Create, Match, and Placement policies.

The dirxml-uACLockout attribute is not synchronized on the Subscriber channel.

Using Characters Outside the Valid NT Logon Names

The default Subscriber creation policy generates an NT Logon Name (also known as the sAMAccountName and the Pre-Windows 2000 Logon Name) based on the relative distinguished name (RDN) of the account in the Identity Vault. The NT Logon name uses a subset of the ASCII character set. The default policy strips any character outside of the valid range before creating an object in Active Directory.

If the policy doesn't satisfy the business rules of your company, you can change the policy after import. Businesses that use Identity Vault account names outside of the traditional ASCII character set should pay particular attention to this policy.

Synchronizing c, co, and countryCode Attributes

When you use the Active Directory management console to select a country for a user, three attributes are set:

Table 10-1 *Attributes for Country*

Attribute	Description
c	Contains a two-character country code as defined by the ISO.
co	Contains a longer name for the country.
countryCode	Contains a numeric value (also defined by the ISO) that represents the country.

Because the ISO-defined numeric country codes are intended for use by applications that can't handle alphabetic characters, the default schema in the Identity Vault includes c and co but not countryCode.

Identity Manager is capable of mapping c and co. It can also map countryCode if you add a similar attribute to the eDirectory schema.

Active Directory's management console tries to keep all three of these attributes synchronized, so that when you set the country in the console, all three attributes have appropriate values. Some administrators might want a similar behavior when the attribute is set through Identity Manager. For example, you might want to configure the driver so that even though only c is in the Filter, co and countryCode are also set when a change for c is sent on the Subscriber channel.

Synchronizing Operational Attributes

Operational attributes are maintained by an LDAP server that contains special operational information. Operational attributes are read-only. They cannot be synchronized or changed.

For more information about operational attributes and attributes in general in Active Directory, see "Active Directory Operational Attributes" and "Attributes defined by Active Directory" in the Microsoft documentation.

Password Complexity on Windows Server

Passwords must meet criteria that the password policies specify.

Complexities and requirements in Windows password policies are different from complexities and requirements in eDirectory. If you plan to use password synchronization, create and use passwords that match the rules of complexity in both Active Directory and eDirectory. Otherwise, the passwords fail.

For information about password complexity requirements for the supported Windows platforms, see "Password must meet complexity requirements" (http://technet.microsoft.com/en-us/library/hh994562%28v=ws.10%29.aspx).

For information about managing passwords in eDirectory, see the *Password Management Administration Guide* (https://www.netiq.com/documentation/password_management33/pwm_administration/data/bookinfo.html).

TIP: Make the password policies for both systems as similar to each other as you can. In a lab environment, disable strong-password functionality on Windows Server servers before installing the Active Directory driver. After the Active Directory driver is working properly, make sure that passwords used in eDirectory and Active Directory satisfy the rules of complexity for both systems. Then re-enable strong-password functionality on the Windows server.

Tips on Password Synchronization

We recommend that you use a secure connection when you are synchronizing passwords. Vulnerable connections are between the following:

- The Identity Manager engine and the Remote Loader
- The Remote Loader and Active Directory

  This is true only when you run the Remote Loader remotely from the domain controller that you're connecting to.

- The Identity Manager engine and Active Directory when you aren't using the Remote Loader

  This is true only if the domain controller isn't local to this machine.

You can create a secure connection by doing one or more of the following:

- Configure SSL between the Identity Manager engine and the Remote Loader
- Run the Remote Loader on the domain controller
- Configure SSL between the driver shim and Active Directory

  This doesn't apply if you are running the driver on the domain controller that you're connecting to.

For password synchronization to work when the driver shim isn't running on the domain controller, you must have SSL configured.

Providing Initial Passwords

If you see an error about a password not complying when a user is initially created, you need to check your password policies.

For example, perhaps you want the Active Directory driver to provide the initial password for a user when the Active Directory driver creates a User object in the Identity Vault. When a user is created, the driver shim creates the user and then sets the password.

Because adding the user and setting the password are done separately, the new user in this example receives the default password, even if only momentarily. The password is soon updated because the Active Directory driver sends it immediately after adding the user.

If the default password doesn't comply with the eDirectory Password policy for the user, an error is displayed. For example, if a default password that was created by using the user's surname is too short to comply with the Password Policy, you might see a -216 error saying that 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:

- Change the policy on the Publisher channel that creates default passwords, so that default passwords conform to the Password policies (created by using the **Manage Password Policies** option in Password Management) that have been defined for your organization in the Identity Vault. When the initial password comes from the authoritative application, it replaces the default password.

  This option is preferable. We recommend that a default password policy exist in order to maintain a high level of security within the system.
- 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 only for a short time.

These measures are especially important if the initial password does not come with the Add event, but comes in a subsequent event. A user is added to eDirectory in two stages. The object is created in the initial Add event and then the password is set for this user. In the Create rule in the Subscriber channel, there is a suggested rule to veto if the `nspmDistributionPassword` operational attribute is not available. This causes the initial Add event to end with a veto, and the subsequent Modify event ends with only the `modify-attr attr-name="nspmDistributionPassword"` attribute, which turns the Modify event into a synthetic Add event. Because the initial Add event was vetoed, the password Modify event is converted into another Add event, but this time it can complete.

Where to Set the SSL Parameter

SSL is used for securing communication in two different ways:

- **For securing communication between the Remote Loader and the engine:** This is activated by specifying the string `kmo="<name of SSL Cert>"` in the Remote Loader connection parameters of the driver configuration. For more information, see "Creating a Secure Connection to the Identity Manager Engine" in the *NetIQ Identity Manager Driver Administration Guide*.
- **For securing communication between the driver shim and the domain controller:** If you select the **Use SSL** option, this setting is done in the driver configuration for securing communication between the Remote Loader and the domain controller when the driver shim is installed on a member server instead of a domain controller.

The SSL parameter in the driver configuration is for SSL connection between the Active Directory driver and Active Directory. It is not for SSL connection between the Identity Manager engine and the Remote Loader. See "Encryption Using SSL" on page 23.

Password Filter Synchronization State Definitions

The SyncState attribute provides information about the hosts to which passwords have been sent. Each bit in the SyncState value is set if the password has been successfully sent to the corresponding host in the Host Names list.

The SyncState value is located in the `HKEY_LOCAL_MACHINE\SOFTWARE\Novell\PwFilter\Data\` registry key.

Table 10-2 *Password Filter Synchronization State Values*

Host Name Value of Password Filter Key	Synchronization State
If there is one host name	• Sync State 00 - Password is not sent to any host.
If there are two host names	• Sync State 00 - Password is not sent to any host. • Sync State 01 - Password is sent only to the first host. • Sync State 02 - Password is sent only to the second host.

Host Name Value of Password Filter Key	Synchronization State
If there are three host names	◆ Sync State 00 - Password is not sent to any host. ◆ Sync State 01 - Password is sent only to the first host. ◆ Sync State 02 - Password is sent only to the second host. ◆ Sync State 03 - Password is sent only to the first and second hosts. ◆ Sync State 04 - Password is sent only to the third host. ◆ Sync State 05 - Password is sent only to the first and third hosts. ◆ Sync State 06 - Password is sent only to the second and third hosts.

You can see more than six synchronization states if there are four or more hosts in the Hosts Name list.

Unable to Retrieve Passwords When Google Password Synchronization is Installed

The Active Directory password filter retrieves blank passwords if Google Password Synchronization software is installed.

Workaround: Uninstall the Google Password Synchronization software and reboot the domain controller.

Passwords Are Not Synchronized from Active Directory to the Identity Vault with Service Account

Passwords are not synchronized from Active Directory to the Identity Vault if the Active Directory driver is run with Service Account instead of Domain Administrator. The driver reports `error 5 (PassSyncCache::StorePwdInfo() returned 0x00000005).`

The Remote Loader Trace level 5 shows the following error:

```
DirXML: [03/27/10 18:19:22.19]: ADDriver: [PWD 4396]
PassSyncCache::StorePwdInfo()
DirXML: [03/27/10 18:19:22.19]: ADDriver: [PWD 4396] StorePwdInfo() - open
the cache.
DirXML: [03/27/10 18:19:22.19]: ADDriver: [PWD 4396] StorePwdInfo() -
acquire the mutex.
DirXML: [03/27/10 18:19:22.19]: ADDriver: [PWD 4396] StorePwdInfo() - mutex
acquired.
DirXML: [03/27/10 18:19:22.19]: ADDriver: [PWD 4396] StorePwdInfo() -
enumindex 0.
DirXML: [03/27/10 18:19:22.19]: ADDriver: [PWD 4396] StorePwdInfo() -
create the entry MC8314.
DirXML: [03/27/10 18:19:22.19]: ADDriver: [PWD 4396] StorePwdInfo() -
an error occurred ... delete this entry.
DirXML: [03/27/10 18:19:22.19]: ADDriver: [PWD 4396] StorePwdInfo() -
release the mutex.
DirXML: [03/27/10 18:19:22.19]: ADDriver: [PWD 4396] StorePwdInfo() - mutex
released.
DirXML: [03/27/10 18:19:22.19]: ADDriver: [PWD 4396] StorePwdInfo() - close
the cache
DirXML: [03/27/10 18:19:22.19]: ADDriver: [PWD 4396]
PassSyncCache::StorePwdInfo() returned 0x00000005
DirXML: [03/27/10 18:19:22.19]: ADDriver: [PWD] PassSyncPassword()
returned 0x00000005
```

To workaround this issue, set the Active Directory Service Account read, write, delete, and inheritance rights to the HKEY_LOCAL_MACHINE\SOFTWARE\Novell\PassSync\Data registry key on the Remote Loader. This allows the Remote Loader to read the password changes from HKLM\SOFTWARE\Novell\PwFilter\Data\<*Username*> key for each user that has changed password.

Active Directory Account Is Disabled After a User Is Added on the Subscriber Channel

The default configuration maps the Identity Vault Logon Disabled attribute to the dirxml-uACAccountDisable bit of the userAccountControl attribute in Active Directory. A Subscriber Add operation might set Logon Disabled to False (account enabled), but the Publisher loopback of the Add operation reports that Logon Disabled is True (account disabled).

Additionally, inspecting the object in Active Directory might show that the account is disabled. This happens in part because of the way that the driver creates objects in Active Directory and in part because of a mismatch of policies between the driver and Active Directory itself.

If the account remains disabled in Active Directory after the provisioning cycle completes, you might have a mismatch between policies configured for the driver and policies enforced by Active Directory.

For example, consider a Password Required policy. If a user Add operation contains an invalid password (or no password at all), the account created in Active Directory should be disabled. But Active Directory might set the dirxml-uACPasswordNotRequired bit in userAccountControl without the driver's knowledge.

This causes the logon enable action of the Add operation to fail if the Add operation does not include a policy for dirxml-uACPasswordNotRequired. Therefore, the account stays disabled.

Later (perhaps almost immediately because of a Merge operation), the driver might attempt to enable the account again by setting Logon Disabled to False. If you want to override the Active Directory policy and ensure that accounts always require a password, you should set dirxml-uACPasswordNotRequired to False whenever Logon Disabled changes on the Subscriber channel.

Moving a Parent Mailbox to a Child Domain

If you move a parent mailbox to a mailbox store in a child domain by changing a user's homeMDB attribute, the driver fails the move. The error code returned is 0x80072030.

This error occurs on inter-domain moves. Moving an Exchange parent mailbox to a child domain isn't supported.

Restoring Active Directory

When you need to restore some or all of Active Directory, the driver might pick up interim events and perform unwanted actions on the Identity Vault. To restore safely, temporarily disable the driver during the restore operation and then bring the Identity Vault back into synchronization with Active Directory.

1. Disable the driver.
2. Delete the Dirxml-DriverStorage attribute on the driver object in the Identity Vault.
3. Restore Active Directory.
4. Set the Active Directory driver to Manual or Automatic startup, then select the **Do not automatically synchronize the driver** option.
5. Start the driver.
6. Re-migrate to find unassociated objects.

Moving the Driver to a Different Domain Controller

Perform the following steps to move the AD driver to a different Domain Controller.

1. Stop the driver.
2. Update the Authentication Context parameter.
3. Start the driver.

Migrating from Active Directory

When you migrate from Active Directory to the Identity Vault, you need to be concerned about object containment, DN references, and search limits on the Active Directory server. The general strategy for dealing with containment is to migrate containers first, objects that might be members of groups (including user objects) second, and groups last. If you have a moderately large number of objects to migrate, you need to adjust your strategy to handle the LDAP search constraints

configured on the Active Directory server. You can change the constraints on the LDAP server or adjust your migration to get only a subset of objects each time (for instance, migrating container by container or migrating objects starting with A, B, etc.).

Setting LDAP Server Search Constraints

This section contains an example terminal session showing you how to use `ntdsutil.exe` to change the LDAP search parameters on your domain controller. You should only change these settings on the domain controller being used for Identity Manager synchronization for the duration of the migration. Write down the current configuration values and run `ntdsutil.exe` after migration completes to restore the original values. `ntdsutil.exe` can be run on any member server.

1. At a command prompt, type `ntdsutil`.
2. Type `LDAP Policies`, then press Enter.
3. Type `Connections`, then press Enter.
4. Type `Connect to domain` *`domain_name`*, then press Enter.
5. Type `Connect to server` *`server_name`*, then press Enter.
6. Type `Quit`, then press Enter.
7. Type `Show Values`, then press Enter.

```
C:\>ntdsutil
ntdsutil: LDAP Policies
ldap policy: Connections
server connections: Connect to domain raptor
Binding to \\raptor1.raptor.lab ...
Connected to \\raptor1.raptor.lab using credentials of locally logged on
user.
server connections: Connect to server raptor1
Disconnecting from \\raptor1.raptor.lab...
Binding to raptor1 ...
Connected to raptor1 using credentials of locally logged on user.
server connections: Quit
ldap policy: Show Values

Policy                          Current(New)
MaxPoolThreads                  4
MaxDatagramRecv                 4096
MaxReceiveBuffer                10485760
InitRecvTimeout                 120
MaxConnections                  5000
```

```
MaxConnIdleTime                 900
MaxPageSize                     1000
MaxQueryDuration                120
MaxTempTableSize                10000
MaxResultSetSize                262144
MaxNotificationPerConn          5
MaxValRange                     1500
ldap policy: set MaxQueryDuration to 1200
ldap policy: set MaxResultSetSize to 6000000
ldap policy: Commit Changes
ldap policy: Quit
ntdsutil: Quit
Disconnecting from raptor1...
C:\>
```

Error Messages

The following sections contains a list of common error messages.

LDAP_SERVER_DOWN

Source: The status log or DSTrace screen.

Explanation: The driver can't open the LDAP port on the Active Directory domain controller configured for synchronization.

Possible Cause: The server named in the driver authentication context is incorrect.

Possible Cause: You are using an IP address for the authentication context, and you have disabled non-kerberos authentication to Active Directory. kerberos requires a DNS name for the authentication context.

Possible Cause: You have incorrectly configured the driver to use an SSL connection to Active Directory.

Action: The authentication context should hold the DNS name or the IP address of the domain controller you use for synchronization. If you leave the parameter empty, the driver attempts to connect to the machine that is running the driver shim (either the same server that is running Identity Manager, or the server hosting the Remote Loader).

Action: The driver shim can authenticate only by using the pre-Windows 2000 Logon method or simple bind. If you have disabled NTLM, NTLM2, and simple bind on your network, you might receive the LDAP_SERVER_DOWN message. Enable NTLM, NTML2, and simple bind on your network.

Action: Something is wrong with the certificate that was imported to the driver shim server, or no certificate was imported. Either import a certificate, or generate a new certificate and import it.

LDAP_AUTH_UNKNOWN

Source: The status log or DSTrace screen.

Explanation: The driver is unable to authenticate to the Active Directory database.

Action: Try to authenticate to the Active Directory database again.

Solution: Unhide the retry-ldap-auth-unknown driver parameter to allow the driver to retry the authentication when it fails:

1. Open the driver configuration file in the an XML editor.
2. Search for retry-ldap-auth-unknown.
3. Change hide="true" to hide="false".
4. Access the driver parameters. See "Driver Parameters" on page 95 for more information.
5. Select **Driver Settings > Access Options > Retry LDAP Auth unknown** error, then select **Yes**.
6. Click **OK**, then restart the driver.

An error was encountered while reading domain on the network 1208

Source: Password Sync Control Panel Applet on Windows server 2008

Action: The Computer Browser service must be started to get the list of computers on the network. By default, it is disabled. In the control panel, go to **Administrative tools** > **Services** and start the service.

Unable to locate language file NSL\ENU\ADManagerRes.dll

Source: Running the **ADManager** tool on Windows 2016 Domain Controller.

Explanation: Displays the following warning message:

```
Unable to locate language file NLS\ENU\ADManagerRes.dll
```

Action: Ignore this warning message. This does not cause any functionality loss.

Performance is Degraded if eDirectory is Installed

Performance is degraded if the Active Directory domain controller and file services are configured on the disk on which eDirectory is installed.

For better performance of Identity Manager on Windows, enable write caching on the disk as follows:

1. Right-click **My Computer > Properties > Hardware > Device Manager > Disk drives**.
2. Right-click the drive on which eDirectory/Identity Manager is installed and click **Properties > Policies**.
3. Select **Enable write caching on the disk**.

Modify Operations Fail on AD LDS Instances

If you want to use the Active Directory driver to modify objects in an AD LDS application directory partition, you must ensure you set the Default Naming Context for the LDS instance to point to the partition. If you do not correctly configure the Default Naming Context, when the driver attempts to modify an object in an application directory partition, the operation fails with an `LDAP_UNWILLING_TO_PERFORM` error.

For more information about configuring the Default Naming Context for an AD LDS instance, see "Setting the Default Naming Context for Your AD LDS/ADAM Instance" on page 108.

PowerShell Service Installation Fails for Active Directory Drivers on Windows 2012 Devices

Installation of the PowerShell service fails when Windows detects that it is downloaded from an untrusted source and the `System.IO.FileLoadException` exception is thrown.

To be able to run the PowerShell service successfully, right-click the `IDMPowerShellService.exe` file, and then select **Properties** > **Unblock**.

Setting a Password in Active Directory Driver Resets the eDirectory Password Expiration Date to the Current Date

Whenever you set a password in Active Directory driver, the password syncs to Identity Manager, as expected. However, this also resets password expiration date in eDirectory to the current date and time. Because of this, a user with a future password expiration date in eDirectory now has an expired password.

To workaround this issue, perform the following steps:

1. Click the upper-right corner of the Active Directory driver, then click **Edit** properties.
2. Navigate to **Driver Properties > Global Configuration Values**.
3. In Password synchronization policy tab, ensure that you `deselect` the **If password does not comply, enforce Password Policy on the connected system by resetting user's password** option.

This ensures that the eDirectory password expiration date is not reset whenever you set passwords in Active Directory.

PowerShell Service Does Not Automatically Start on Windows Server 2012 R2

When the Windows server is started, the Identity Manager PowerShell service does not automatically start. It fails with the following error:

```
The account name is invalid or does not exist, or the password is invalid
for the account name specified.
```

To resolve this issue, make the Identity Manager PowerShell service dependent on Active Directory Domain Services. For more information, see this Active Directory documentation page.

PowerShell Service Consumes Lot of Disk Space When Multiple PSSessions are Initialized

A remote PSSession is initialized when you start the AD driver. It is recommended to initialize only one remote PSSession at any point of time. Running multiple sessions consumes lot of memory and may cause disk space issues.

Working with TimeToLive(minute) Attribute

This section provides answers to questions for specific scenarios, that you might have while working with the TimeToLive(minute) attribute.

How are passwords managed when the value of the TimeToLive(minute) attribute is set to a default value of 0 (zero).

When the attribute is set to the default value, a password is prevented from being deleted from the Domain Controller registry without getting transferred to the driver registry. If the value is set to x, where x is greater than the default value, the password will be deleted from the registry after x minutes if the password is not successfully synchronized within that time.

How are passwords managed when you have PWFilter registry on 11 Domain Controllers containing old and new password changes.

The password will be added to the Domain Controller registry where the password has changed. If the old password for the same user is still in the registry when a new password is added, the old password is overwritten by the new password.

In what order are passwords added and removed from the registry.

The registry keeps the keys (user name) sorted at all times.

How is a new password added when the registry is removing the stored passwords.

The registry keeps the keys (user name) sorted at all times. After all the stored passwords are transferred to the driver registry, the passwords are removed from the registry of the Domain Controller.

Troubleshooting Driver Processes

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

Driver Loses An Event That Does Not Have class-Name

Issue: When you send an event to the Active Directory driver through the driver shim, the event is dropped without an error returned from the driver, the shim, or the engine. The lost event is not detected without checking explicitly. The following event is sent to the Active Directory shim:

```
<modify dest-dn="cn=FACSN Users,ou=FACSN
users,OU=Units,DC=its,DC=cads,DC=ORG" event-
id="testserver01#20171016161104#1#1:4f1bc979-33b2-43bb-84ea-79c91b4fb233">
  <modify-attr attr-name="member">
    <add-value>
      <value type="string">cn=testuser,ou=FACSN
users,OU=Units,DC=its,DC=cads,DC=ORG<
DirXML: [10/16/17 10:11:04.41]: /value>
    </add-value>
  </modify-attr>
</modify>
```

The Remote Loader trace shows the following text that indicates that there is an event, but the class-name is missing:

```
DirXML: [10/16/17 10:11:04.55]: ADDriver: parse command
  className
  destDN       cn=FACSN Users,ou=FACSN users,OU=Units,DC=its,DC=cads,DC=ORG
  eventId      testserver01#20171016161104#1#1:4f1bc979-33b2-43bb-84ea-
79c91b4fb233
  association
DirXML: [10/16/17 10:11:04.55]: ADDriver: parse modify class =
DirXML: [10/16/17 10:11:04.57]: Loader: subscriptionShim->execute()
returned:
DirXML: [10/16/17 10:11:04.57]: Loader: XML Document:
DirXML: [10/16/17 10:11:04.57]: <nds ndsversion="8.7" dtdversion="1.1">
        <source>
              <product version="4.0.2.1" asn1id="" build="20170106_120000"
instance="\ORG-IDV\ORG\services\ORG-VaultDriverSet\ORG-CADS">AD</product>
                <contact>NetIQ Corporation</contact>
        </source>
        <output>
                <add-association dest-dn="\ORG-
IDV\ORG\data\users\employees\testuser" dest-entry-id="342222" event-
id="testserver01#20171016161104#1#1:4f1bc979-33b2-43bb-84ea-
79c91b4fb233">52b7c854d68c2a439be0bbb8fa597332</add-association>
                <status level="success" event-
id="testserver01#20171016161104#1#1:4f1bc979-33b2-43bb-84ea-79c91b4fb233"/
>
                <status level="success" event-
id="testserver01#20171016161104#1#1:4f1bc979-33b2-43bb-84ea-79c91b4fb233"/
>
        </output>
</nds>
```

Workaround: Ensure that you add a value for class-name when the custom event in Identity Manager policy is synthesized.

Applying the Latest Driver Package Does Not Change the Default Setting of Enable Service Channel ECV

Issue: If you upgraded to Identity Manager 4.7 and updated the base packages for your driver, the package update process does not overwrite the default setting (`False`) of **Enable Service Channel** ECV.

This issue does not occur when you create a new driver.

Workaround: Manually change the ECV for the driver.

To change the ECV in Designer:

1. In Modeler, right-click the driver line.
2. Select **Properties > Engine Control Values**.
3. Click the tooltip icon to the right of **Engine Controls for Server**.

   If a server is associated with the Identity Vault, and if you are authenticated, the engine control values display in the large pane.
4. Change the value for **Enable Subscriber Service Channel**.
5. Click **OK**.
6. For the change to take effect, deploy the driver to the live Identity Vault.

To change the ECV in iManager:

1. Log in to the instance of iManager that manages your Identity Vault.
2. In the navigation frame, select **Identity Manager**.
3. Select **Identity Manager Overview**.
4. Use the search page to display the Identity Manager Overview for the driver set that contains your driver.
5. Click the round status indicator in the upper right corner of the driver icon.
6. Select **Edit Properties > Engine Control Values**.
7. Change the value for **Enable Subscriber Service Channel**.
8. Click **OK**, then click **Apply**.
9. For the change to take effect, restart the driver.

A Driver Properties

This section provides information about the Driver Configuration and Global Configuration Values properties for the Active Directory driver. These are the only unique properties for drivers. All other driver properties (Named Password, Engine Control Values, Log Level, and so forth) are common to all drivers. Refer to "Driver Properties" in the *NetIQ Identity Manager Driver Administration Guide* for information about the common properties.

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

Driver Configuration

In iManager:

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

   By default, the Driver Configuration page is displayed.

In Designer:

1. Open a project in the Modeler.
2. Right-click the driver line, then select click **Properties**.
3. Click **Driver Configuration**.

The Driver Configuration options are divided into the following sections:

- “ECMAScript (Designer Only)” on page 98
- “Global Configurations (Designer Only)” on page 98

Driver Module

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

Java: This option is not used with the Active Directory driver.

Native: Used to specify the name of the `.dll` file that is instantiated for the application shim component of the driver. If this option is selected, the driver is running locally.

The driver `.dll` is: `addriver.dll`

Connect to Remote Loader: Used when the driver is connecting remotely to the connected system. Designer includes one sub-option:

- **Remote Loader client configuration for documentation:** Includes information on the Remote Loader client configuration when Designer generates documentation for the driver.

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

Driver Object Password (iManager Only)

Driver object password: Use this option to set a password for the driver object. If you are using the Remote Loader, you must enter a password on this page or the remote driver does not run. This password is used by the Remote Loader to authenticate itself to the remote driver shim.

Authentication

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

Authentication ID: Specify a user application ID. This ID is used to pass Identity Vault subscription information to the application.

Example: `Administrator`

Authentication context/Connection Information: Specify the IP address or name of the server the application shim should communicate with.

If you are synchronizing Exchange mail boxes, you must specify the full qualified name of the domain controller.

Example: `myserver.company.com`

Remote Loader Connection Parameter: Used only if the driver is connecting to the application through the Remote Loader. Enter `hostname=xxx.xxx.xxx.xxx port=xxxx secureprotocol=TLS version enforceSuiteB=`*`true/false`* `kmo=certificatename.`

- `hostname` specifies the IP address of the Remote Loader server.

- `port` specifies the TCP/IP port on which the Remote Loader listens for connections from the remote interface shim. The default port for the Remote Loader is 8090.
- `secureprotocol` specifies the version of the TLS protocol that the Remote Loader uses to connect to the Identity Manager engine. Identity Manager supports TLSv1, TLS v1_1, and TLSv1_2 versions only. For example: `secureprotocol=TLSv1`
- `enforceSuiteB` specifies whether Remote Loader uses Suite B for communicating with the Identity Manager engine. To use Suite B, specify `enforceSuiteB=true`. This communication is supported only on TLS 1.2 protocol. When the connection uses non-Suite B authentication algorithms and this parameter is enabled, the communication cannot be established.
- The `kmo` entry is optional. Use it only when an SSL connection exists between the Remote Loader and the Identity Manager engine. You must specify this parameter when you enable Suite B communication. For example: `hostname=10.0.0.1 port=8090 kmo=IDMCertificate`

Driver cache limit: Specify the maximum event cache file size (in KB). If it is set to zero, the file size is unlimited. Click **Unlimited** to set the file size to unlimited in Designer.

Application password: Specify the password for the user object listed in the **Authentication ID** field.

Remote loader password: Used only if the driver is connecting to the application through the Remote Loader. The password is used to control access to the Remote Loader instance. It must be the same password specified during the configuration of the Remote Loader on the connected system.

Startup Option

The Startup Option section allows you to set the driver state when the Identity Manager server is started.

Auto start: The driver starts every time the Identity Manager server is started.

Manual: The driver does not start when the Identity Manager server is started. The driver must be started through Designer or iManager.

Disabled: The driver has a cache file that stores all of the events. When the driver is set to Disabled, this file is deleted and no new events are stored in the file until the driver state is changed to Manual or Auto Start.

Do not automatically synchronize the driver (Designer only): This option only applies if the driver is deployed and was previously disabled. If this is not selected, the driver re-synchronizes the next time it is started.

Driver Parameters

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

The parameters are divided into the following categories:

- "Authentication Options" on page 96
- "Exchange Options" on page 97

- "Access Options" on page 97
- "Advanced Options" on page 98

Authentication Options

Show authentication options: Enables you to see and change the authentication options for the driver. The options are **show** or **hide**.

Authentication Method: The method of authentication to Active Directory. Negotiate uses Microsoft's security package to negotiate the logon type. Typically kerberos or NTLM is selected. Simple uses LDAP style simple bind for logon.

If you want to use password synchronization, select **Negotiate**.

Digitally sign communications: Select **Yes** to digitally sign communication between the driver shim and Active Directory. The communication is in clear text, but signing ensures that the communication is not tampered with enroute to the destination. It reduces the chance of security attacks.

Signing only works when you use the Negotiate authentication method and the underlying security provider selects NTLM2 or kerberos for its protocol.

Do not use this option with SSL.

Select **No** to have communications not signed. You can use this option with the **Digitally sign and seal communications** option.

Digitally sign and seal communications: Select **Yes** to digitally encrypt communication between the driver shim and the Active Directory database.

Sealing only works when you the Negotiate authentication method and the underlying security provider selects NTLM2 or kerberos for its protocols.

Do not use this option with SSL.

Select **No** to not have communication between the driver shim and the Active Directory database signed and sealed. You can use this option with the **Digitally sign communications option**.

Use SSL for LDAP connection between Driver Shim and AD: Select **Yes** to digitally encrypt communication between the driver shim and the Active Directory database.

This option can be used with the Negotiate or Simple authentication methods. SSL requires that the Microsoft server running the driver shim imports the domain controller's server certificate. For more information, see "Microsoft Security Compliance Manager" (http://technet.microsoft.com/en-us/library/cc677002.aspx).

Logon and impersonate: Select **Yes** to log on and impersonate the driver authentication account for CDOEXM (Collaboration Data Object for Exchange Management) and Password Set support. The driver performs a local logon. The authentication account must have the proper rights assignment. For more information, see "Creating an Administrative Account" on page 26.

If **No** is selected, the driver performs a network logon only.

Exchange Options

Show Exchange Management Options: Select **show** to display the Microsoft Exchange options. These parameters control whether the driver shim uses the Microsoft CDOEXM Exchange management APIs and whether to interpret changes in the homeMDB attribute as a Move or a Delete of the mailbox.

Select **hide** if you are not synchronizing Exchange accounts.

Enable Exchange mailbox provisioning: Select **enable** to provision Exchange Mailbox accounts.

- **Allow Exchange mailbox move:** Select **Yes** to enable the driver to intercept modifications to the Active Directory homeMDB attribute and call into the CDOEXM subsystem to move the mailboxes to the new message data store.

  Select **No** if you do not want mailboxes moved when the Active Directory account is moved.
- **Allow Exchange mailbox delete:** Select **Yes** to enable the driver to intercept removals of the Active Directory homeMDB attribute and calls into the CDOEXM subsystem to delete the mailbox.

  Select **No** if you don't want to delete the mailbox account when the Active Directory account is deleted.
- **Exchange Management interface type:** Exchange mailboxes can be controlled by calls to the Microsoft Exchange management system instead of regular attribute synchronization. When this options is enabled, the driver intercepts changes to the Active Directory homeMDB attribute and calls into the desired interface for Exchange Management.
- **Exchange Server FQDN:** If you are configuring the Identity Manager Powershell service in a multiple exchange server domain, you can use this option to choose the preferred server to be connected by the Powershell service.

Access Options

Show access options: Select **show** to display the domain controller access options. These parameters control the scope of the Active Directory queries along with several Publisher polling and timeout parameters.

Select **hide** to hide the domain controller access options.

Driver Polling Interval: Specify the number of minutes to delay before querying the Active Directory data base for changes. A larger number reduces the load on the Active Directory database, but it also reduces the responsiveness of the driver.

The default value is 1 minute.

Publisher heartbeat interval: Allows the driver to send a periodic status message on the Publisher channel when there has been no Publisher channel traffic for the given number of minutes.

The default value is 1 minute.

Password Sync Timeout (minutes): Specify the number of minutes for the driver to attempt to synchronize a given password. The driver does not try to synchronize the password after this interval has been exceeded.

The recommended value is at least three times the value of the polling interval. For example, if the **Driver Polling Interval** is set to 10 minutes, set the **Password Sync Timeout** to 30 minutes.

If this value is set to 0, password synchronization is disabled for this driver.

If this value is set to -1, passwords never expire. It can reach a maximum value of 2147483647 minutes.

The default value is 5 minutes.

DC Passwords TimeToLive (minutes): Specify the time limit in minutes for the passwords to be stored in the Domain Controller registry.

This allows the passwords that are stored in the Domain Controller registry to time out if the password does not synchronize to the driver within the specified time.

If this value is set to -1, passwords will never be deleted from the registry.

The default value is -1.

Search domain scope: The driver reads information from other domains when objects in those domains are referenced. If the account you use for authentication has no rights in the other domain, the reads might fail. Select **Yes** to enable this option if you get access errors during regular operations.

Advanced Options

Show advanced options: Select **show** to display the advanced configuration options for the driver.

Enable Deletion of protected objects in Windows Server 2008: Select **Yes** to delete the protected objects that are created through MMC in Windows Server 2008. Select **No** for protecting these objects from accidental deletion.

Retry LDAP Auth unknown error: Ordinarily, the driver shim returns a fatal error when encountering an LDAP-AUTH_UNKNOWN error that causes the driver to shut down. If you want the driver to retry the LDAP bind request, select **Yes**.

Enable DirSync Incremental Values: The Publisher channel usually receives all the values of a multi-valued attribute. Enabling this option reports only the added or deleted values during the poll interval. This requires 2003 Forest functional mode or above. This option is hidden by default. It can be modified by selecting the **Edit XML** option in the Driver configuration tab.

ECMAScript (Designer Only)

Displays an ordered list of ECMAScript resource objects. The objects contain extension functions for the driver that Identity Manager loads when the driver starts. You can add additional ECMAScript objects, remove existing files, or change the order the objects are executed.

Global Configurations (Designer Only)

Displays an ordered list of Global Configuration objects. The objects contain extension GCV definitions for the driver that Identity Manager loads when the driver is started. You can add or remove the Global Configuration objects, and you can change the order in which the objects are executed.

Global Configuration Values

Global configuration values (GCVs) are values that can be used by the driver to control functionality. GCVs are defined on the driver or on the driver set. Driver set GCVs can be used by all drivers in the driver set. Driver GCVs can be used only by the driver on which they are defined.

The Active Directory driver includes several predefined GCVs. You can also add your own if you need additional ones as you implement policies in the driver.

To access the driver's GCVs in iManager:

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

   or

   To add a GCV to the driver set, click **Driver Set**, then click **Edit Driver Set properties**.

To access the driver's GCVs in Designer:

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

   or

   To add a GCV to the driver set, right-click the driver set icon  , then click **Properties > GCVs**.

The global configuration values are organized as follows:

Configuration

The following GCVs contain configuration information for the Active Directory driver. They are divided into the following categories:

Synchronization Settings

Domain DNS Name: Specify the DNS name of the Active Directory domain managed by this driver.

Subscriber Channel Placement Type: Specify the type of placement for the Subscriber Channel. Select **Flat** to strictly place objects within the base container. Select **Mirrored** to hierarchically place objects within the base container. This is used to determine the Subscriber Channel Placement policies.

Active Directory User Container: Specify the container where user objects reside in Active Directory.

Publisher Channel Placement Type: Specify the type of placement for the Publisher Channel. Select **Flat** to strictly place objects within the base container. Select **Mirrored** to hierarchically place objects within the base container. This is used to determine the Publisher Channel Placement policies.

Name Mapping Policy

Show name mapping policy: Select **show** to display the global configuration values for the name mapping policy. Select **hide** to not have the global configuration values displayed.

The following GCVs are used in the name mapping policy. If the policy does not meet your needs, you can modify it by editing the UserNameMap policies in the Subscriber and Publisher Command Transformation policies.

Full Name Mapping: Select **True** to synchronize the Identity Vault user's Full Name with the Active Directory object name and display name. This policy is useful when creating user accounts in Active Directory by using the Microsoft Management Console Users and Computers snap-in.

Logon Name Mapping: Select **True** to synchronize the Identity Vault user's object name with the Active Directory Pre-Windows 2000 Logon Name (also known as the NT Logon Name and the sAMAccountName).

User Principal Name Mapping: Allows you to choose a method for managing the Active Directory Logon Name (also known as the userPrincipalName). userPrincipalName takes the form of an e-mail address, such as usere@domain.com. Although the driver can place any value into userPrincipalName, it is not useful as a logon name unless the domain is configured to accept the domain name used with the name.

- **Follow Active Directory e-mail address:** Sets the userPrincipalName to the value of the Active Directory mail attribute. This option is useful when you want the user's e-mail address to be used for authentication and Active Directory is authoritative for e-mail addresses.
- **Follow Identity Vault e-mail address:** Sets the userPrincipalName to the value of the Identity Vault e-mail address attribute. This option is useful when you want the user's e-mail address to be used for authentication and the Identity Vault is authoritative for e-mail addresses.
- **Follow Identity Vault name:** This option is useful when you want to generate userPrincipalName from the user logon name plus a hard-coded string defined in the policy.
- **None:** This option is useful when you do not want to control userPrincipalName or when you want to implement your own policy.

Password Synchronization

These GCVs enable password synchronization between the Identity Vault and the Active Directory system.

In Designer, you must click the  icon next to a GCV to edit it. This displays the Password Synchronization dialog box for a better view of the relationship between the different GCVs.

In iManager, to edit the Password management options go to **Driver Properties > Global Configuration Values**, and then edit it in your Password synchronization policy tab.

For more information about how to use the Password Management GCVs, see "Configuring Password Flow" in the *NetIQ Identity Manager Password Management Guide*.

Connected System or Driver Name: Specify the name of the Active Directory system or the driver name. This value is used by the e-mail notification template to identify the source of the notification message.

Application accepts passwords from Identity Manager: If **True**, allows passwords to flow from the Identity Manager data store to the connected system.

Identity Manager accepts passwords from application: If **True**, allows passwords to flow from the connected system to Identity Manager.

Publish passwords to NDS password: Use the password from the connected system to set the non-reversible NDS password in eDirectory.

Publish passwords to Distribution Password: Use the password from the connected system to set the NMAS Distribution Password used for Identity Manager password synchronization.

Require password policy validation before publishing passwords: If True, applies NMAS password policies during publish password operations. The password is not written to the data store if it does not comply.

Reset user's external system password to the Identity Manager password on failure: If True, on a publish Distribution Password failure, attempt to reset the password in the connected system by using the Distribution Password from the Identity Manager data store.

Notify the user of password synchronization failure via e-mail: If True, notify the user by e-mail of any password synchronization failures.

Account Tracking

Account tracking is part of the Identity Reporting Module. For more information, see the *Administrator Guide to NetIQ Identity Reporting*.

Enable account tracking: Set this to **True** to enable account tracking policies. Set it to **False** if you do not want to execute account tracking policies.

Realm: Specify the name of the realm, security domain, or namespace in which the account name is unique.

Object Class: Add the object class to track. Class names must be in the application namespace.

Identifiers: Add the account identifier attributes. Attribute names must be in the application namespace.

Status attribute: Name of the attribute in the application namespace to represent the account status.

Status active value: Value of the status attribute that represents an active state.

Status inactive value: Value of the status attribute that represents an inactive state.

Subscription default status: Select the default status that the policies assume when an object is subscribed to the application and the status attribute is not set in the Identity Vault.

Publication default status: Select the default status that the policies assume when an object is published to the Identity Vault and the status attribute is not set in the application.

Managed System Information

These settings help the Identity Reporting Module function to generate reports. There are different sections in the **Managed System Information** tab.

- "General Information" on page 102
- "System Ownership" on page 102
- "System Classification" on page 103
- "Connection and Miscellaneous Information" on page 103

General Information

Name: Specify a descriptive name for this Active Directory system. This name is displayed in the reports.

Description: Specify a brief description of this Active Directory system. This description is displayed in the reports.

Location: Specify the physical location of this Active Directory system. This location is displayed in the reports.

Vendor: Select Microsoft as the vendor of the Active Directory system. This information is displayed in the reports.

Version: Specify the version of this Active Directory system. This version information is displayed in the reports.

System Ownership

Business Owner: Browse to and select the business owner in the Identity Vault for this Active Directory system. You must select a user object, not a role, group, or container.

Application Owner: Browse to and select the application owner in the Identity Vault for this Active Directory system. You must select a user object, not a role, group, or container.

System Classification

Classification: Select the classification of the Active Directory system. This information is displayed in the reports. The options are:

- Mission-Critical
- Vital
- Not-Critical
- Other

  If you select **Other**, you must specify a custom classification for the Active Directory system.

Environment: Select the type of environment the Active Directory system provides. The options are:

- Development
- Test
- Staging
- Production
- Other

  If you select **Other**, you must specify a custom classification for the Active Directory system.

Connection and Miscellaneous Information

Connection and miscellaneous information: This options is always set to **hide**, so that you don't make changes to these options. These options are system options that are necessary for reporting to work. If you make any changes, reporting stops working.

B Configuring the Driver for Use with an AD LDS/ADAM Instance

The Active Directory driver can be configured for use with an Active Directory Lightweight Directory Services instance (AD LDS).

You import a configuration file to create a driver to connect to the AD LDS/ADAM instance.

There are multiple ways to configure your environment to synchronize the information. For example, NetIQ recommends setting up your own certificate authority (CA) in order to issue certificates that can be used for SSL connections to AD LDS/ADAM. If you already have server certificates, or if you have access to another CA that can issue valid certificates, you can ignore the steps that describe how to set up your own CA. Likewise, if you don't want to configure SSL (required if you want to set passwords on the Subscriber channel) then you can skip the section about configuring Certificate Services.

Any discussion of setting passwords is referring to the Subscriber channel from Identity Manager to AD LDS/ADAM. Password synchronization on the Publisher channel from AD LDS/ADAM to Identity Manager is not currently possible, unless a regular user attribute (not the userPassword attribute) is used in AD LDS/ADAM to store the password.

Prerequisites

To achieve synchronization with an AD LDS/ADAM instance, you need the following items installed on one or more computers running the supported Windows server platforms:

- An Identity Manager server or Remote Loader where the Active Directory driver is configured.
- Internet Information Services (IIS) (must be installed before Certificate Services)
- Certificate Services
- A certification authority (can be your own standalone CA configured when you install Certificate Services)
- An AD LDS/ADAM instance (this example in this section uses a standalone instance)

Installation Tasks

The following installation tasks must be completed in the order that they are listed. If a step is not necessary for your setup, you can skip it.

- “Installing Internet Information Services” on page 106
- “Installing Certificate Services” on page 106
- “Installing AD LDS/ADAM” on page 106
- “Requesting and Installing the Server Certificate” on page 107

Installing Internet Information Services

If you want to set up your own CA in order to configure SSL on AD LDS/ADAM, you need to install Internet Information Services (IIS).

1. On your Windows Server computer, access the Control Panel, then click **Add or Remove Programs**.
2. In the left pane, select **Add/Remove Windows Components**.
3. Select **Application Server**, then click **Details**.
4. Select **Internet Information Services (IIS)**, then click **Details**.
5. Verify that at least the following are selected:
   - **Common Files**
   - **Internet Information Services Manager**
   - **World Wide Web Service**
6. Click **OK** twice, then click **Next** to complete the installation.

   You might be prompted to insert your original installation media for Windows Server.

Installing Certificate Services

1. On your Windows Server machine, access the Control Panel, then click **Add or Remove Programs**.
2. In the left pane, select **Add/Remove Windows Components**.
3. Select to **Certificate Services**, then click **Next** to complete the installation.

Installing AD LDS/ADAM

- “Installing AD LDS” on page 106
- “Installing ADAM” on page 107

Installing AD LDS

If you are installing AD LDS, use the steps given at the Microsoft TechNet Web site (http://technet.microsoft.com/en-us/library/cc754486(WS.10).aspx), then follow the installation instructions from Step 6 to Step 16 from the “Installing ADAM” section.

Installing ADAM

1. On your Windows Server machine, access the Control Panel, then click **Add or Remove Programs**.
2. In the left pane, select **Add/Remove Windows Components**.
3. Select **Active Directory Services**, then click **Details**.
4. Select **Active Directory Application Mode (ADAM)**, then click **OK**.
5. Click **Next** to complete the installation.
6. Click **Next**.
7. Select **Yes** to create an application directory partition, unless you plan on doing it later.
8. Specify the DN of the location where you'd like to synchronize users. For example, `CN=People,DC=adamtest1,DC=COM`.
9. Click **Next**.

   NOTE: If you specify a port number for the ADAM instance other than the default, ensure that you configure the Identity Manager ADAM driver to use the non-default port, as well. For more information about configuring the ADAM driver, see "Creating the AD LDS/ADAM Driver in iManager" on page 109.

   ---

10. Leave the default locations for data files and data recovery files, then click **Next**.
11. Select an account for the ADAM service, then click **Next**.

    If you are installing ADAM on a server that is not already part of a domain, you might get a warning at this point. This is usually not a problem with ADAM, and you should continue with the installation.
12. Click **Next** to assign the current user (the one you are logged in as) rights to administrate ADAM.
13. Select **Import the selected LDIF files for this instance of ADAM**.
14. Select **MS-User.LDF**, then click **Add**.
15. Click **Next**.
16. Review the installation summary, then click **Next**.

Requesting and Installing the Server Certificate

1. On the server where you installed IIS and Certificate Services, specify the following address in a Web browser: `http://localhost/certsrv`.
2. You should see a welcome message from Certificate Services. If you do not, go back and make sure you have IIS and Certificate Services both installed.
3. The steps for requesting and installing a certificate are found at "[.NET] Using SSL with ADAM (AD LDS)" (http://erlend.oftedal.no/blog/?blogid=7).
4. On your AD LDS/ADAM server, make sure you have the certificate installed in the following location in MMC: Certificates - Service (adaminstance) on `Local Computer\ADAM_adaminstance\Personal` directory.

5 On the Identity Manager server (or the Remote Loader computer) where the driver is running, make sure that you have only the CA certificate and make sure it is in `Certificates - Current User\Trusted Root Certificates` directory.

See "Active Directory Application Mode Technical Reference" (http://technet.microsoft.com/en-us/library/cc787075(v=ws.10)) for additional resources.

Configuration Tasks

- "Setting the Default Naming Context for Your AD LDS/ADAM Instance" on page 108
- "Creating a User in AD LDS/ADAM with Sufficient Rights" on page 109
- "Creating the AD LDS/ADAM Driver" on page 109

Setting the Default Naming Context for Your AD LDS/ADAM Instance

1 Start the ADSI Edit application by selecting **Start > All Programs > Administrative Tools> ADAM ADSI Edit**.

2 In the tree view, select the root item called **ADAM ADSI Edit**.

3 Under the **Action** menu, select **Connect to**.

4 In the **Connection name** field, type `Configuration`.

5 Select **Well-known naming context**. Make sure the value in the drop-down list is set to **Configuration**.

6 Set the other authentication credentials as appropriate, then click **OK**.

7 In the tree view, expand the **Configuration** item and those items underneath it until you can select the following entry:

`CN=NTDS Settings,CN=`*`ServerName`*`$`*`InstanceName`*`,CN=Servers,`
`CN=Default-First-Site-Name, CN=Sites,CN=Configuration,CN={`*`GUID`*`}`

Keep in mind that in the above DN, you should replace ServerName, InstanceName, and GUID with those values you specified when you installed your AD LDS/ADAM instance in Step 8 of "Installing AD LDS/ADAM" on page 106.

8 Under the **Action** menu, select **Properties**.

9 Select the **msDS-DefaultNamingContext** attribute, then click **Edit**.

10 Specify the same value you used in Step 8 of "Installing AD LDS/ADAM" on page 106.

NOTE: If you do not point the Default Naming Context of your AD LDS instance to the application directory partition specified during installation, the driver will not be able to successfully perform Modify operations.

---

11 Click **OK** twice.

12 Restart your AD LDS/ADAM instance so the new default naming context takes effect.

Creating a User in AD LDS/ADAM with Sufficient Rights

For the driver to work properly, it is best to create a user object specifically for the driver to use. This user should only have the rights to do the work that is required. For more information, see "Creating an Administrative Account" on page 26.

Creating the AD LDS/ADAM Driver

You can create the AD LDS/ADAM driver through Designer or iManager. The AD LDS/ADAM driver cannot use packages. You must use the driver configuration file to create the driver.

- "Creating the AD LDS/ADAM Driver in Designer" on page 109
- "Creating the AD LDS/ADAM Driver in iManager" on page 109

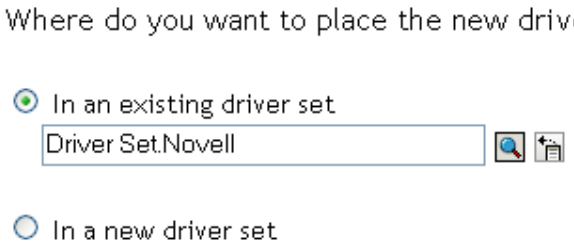
Creating the AD LDS/ADAM Driver in Designer

1. Open a project in Designer. In the Modeler, right-click the driver set and select **New > Driver**.
2. Click **Import Driver Configuration**.
3. From the drop-down list, select **ADAM,** then click **Run**.

   The ADAM driver is not listed alphabetically, so you might have to scroll to find it in the list.
4. Configure the driver by filling in the fields. Specify information for your environment. For information on the settings, see Table B-1 on page 110.
5. After specifying parameters, click **Finish** to import the driver.
6. After the driver is imported, customize and test the driver.
7. After the driver is fully tested, deploy the driver into the Identity Vault. See "Deploying a Driver to an Identity Vault" in the *NetIQ Designer for Identity Manager Administration Guide*.

Creating the AD LDS/ADAM Driver in iManager

1. In iManager, select **Identity Manager Utilities** > **Import Configuration**.
2. Select a driver set, 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. Import a configuration into the driver set by selecting a configuration from the server (.XML file):
   - All configurations
   - Identity Manager 3.0 configurations

- Identity Manager 3.5 configurations
- Identity Manager 3.6 configurations
- Identity Manager 4.0 configurations
- Configurations not associated with an Identity Manager version

4 Select the ADAM driver, then click **Next**.

5 Configure the driver by filling in the configuration parameters, then click **Next**. For information on the settings, see Table B-1 on page 110.

6 Specify the Remote Loader host name or IP address and port, as well as the Remote Loader authentication information, then click **Next**.

7 Define security equivalences, using a user object that has the rights that the driver needs to have on the server, then click **OK**.

Use the user created in "Creating a User in AD LDS/ADAM with Sufficient Rights" on page 109.

8 Identify all objects that represent administrative roles and exclude them from synchronization, then click **OK**.

Exclude the security-equivalence object (for example, DriversUser) that you specified in Step 7. If you delete the security-equivalence object, you have removed the rights from the driver, and the driver can't make changes to Identity Manager.

9 Click **ADAM** to specify additional configuration settings.

10 Under **Driver Parameters**, specify the authentication and access options you want to use for the ADAM driver. In the **LDAP server port** field, ensure that you specify the ADAM LDAP port number configured in ADAM.

11 Click **OK**.

12 Click **Finish**.

NOTE: The parameters are presented on multiple screens. Some parameters are only displayed if the answer to a previous prompt requires more information to properly configure the policy.

Table B-1 *Configuration Parameters for the AD LDS/ADAM Driver*

Parameter	Description
Driver name	Specify the name of the driver object.
Connected System or Driver Name	Specify the name of the connected system, application, or Identity Manager driver. This value is used by the e-mail notification templates to identify the source of notification messages.
Domain DNS Name	Specify the DNS name of the AD LDS/ADAM instance managed by this driver.
ADAM User Container	Specify the container where the objects reside in AD LDS/ADAM.

Parameter	Description
Driver is Local/Remote	Configure the driver for use with the Remote Loader service by selecting **Remote**, or select **Local** to configure the driver for local use.
Authentication ID	Specify the name of the user object created in "Creating a User in AD LDS/ADAM with Sufficient Rights" on page 109. The name needs to be specified as a full LDAP DN. For example, `CN=IDM,CN=Users,DC=domain,DC=com`
Authentication Password	Specify the password of the user object with sufficient rights.
Authentication Context	Specify the DNS name or IP address of the AD LDS/ADAM instance server.

C Provisioning Exchange Accounts

The Active Directory driver can be configured to provision Active Directory accounts as well as Exchange accounts.

The driver can synchronize Exchange Server 2016 accounts. It cannot synchronize all types of Exchange accounts at the same time. If you have multiple types of Exchange accounts, you must set up a separate driver to synchronize each type of Exchange accounts.

Provisioning Exchange Server 2019 and Exchange Server 2016 Accounts

The Active Directory driver includes support for the Exchange Server 2019 and Exchange Server 2016 server.

In order to provision Exchange Server 2019 and Exchange Server 2016 mailboxes, the Active Directory driver uses Windows PowerShell in the form of the IDM PowerShell service.

NOTE: The Active Directory driver only supports provisioning accounts on Exchange Server 2019 servers with Windows Server 2019 (64-bit), Exchange Server 2019 and Exchange Server 2016 servers with Windows Server 2016, Windows Server 2012, or Windows Server 2012 R2 installed.

The IDM PowerShell service is installed on the server that is running the Active Directory driver. If you decide to run the driver locally, the driver is installed on the Identity Manager server. If you decide to run the driver remotely, the driver is installed on the same server as the Remote Loader service.

The service listens on a default port of 8099. This is set when the service is installed. It is stored in the registry key `IDM_PowerShell_Service`, located in either `HKEY_LOCAL_MACHINE\SOFTWARE\Novell` or `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Novell`, depending on your Windows installation. The value can be edited if necessary. If you edit the registry key, both the service and the driver must be restarted.

The Active Directory driver creates, moves, and disables Exchange Server 2019 and Exchange Server 2019 and Exchange Server 2016 mailboxes. The cmdlets supported by the Active Directory driver to create, move, and disable mailboxes in Exchange Server 2019 and Exchange Server 2016 are Enable-Mailbox, New-MoveRequest, and Disable-Mailbox. The cmdlets use the following parameters in the Active Directory driver:

- **Enable-Mailbox:** -Identity, -Alias, -Database, -DomainController
- **Disable-Mailbox:** Identity, -DomainController, -Confirm
- **New-MoveRequest:** -Identity, -TargetDatabase, -DomainController, -Confirm

For more functionality support, use the Scripting driver or the native PowerShell support feature. For more information on the Scripting driver, see the *Identity Manager Driver for Scripting Implementation Guide*. For more information on PowerShell support in Identity Manager, see Appendix D, "Configuring PowerShell Support," on page 117.

To provision Exchange Server 2019 and Exchange Server 2016 mailboxes, you must complete the following steps:

Meeting the Prerequisites

On the server where the driver will run, whether as a Remote Loader service or if the driver is installed locally, the following items must be installed:

- ❒ Microsoft .NET Framework version 4.5.2 or later
- ❒ Windows Management Framework 4.0 or later

Installing the Service

To install the service, you must use the .NET Framework `InstallUtil.exe` utility. The version folder is the current version of the .NET Framework that is installed.

The default location for a 64-bit server is `C:\Windows\Microsoft.NET\Framework64\`*`version`*`\InstallUtil.exe`.

To use `InstallUtil.exe`:

1. Install the latest available patches and updates on your Identity Manager components and drivers.
2. On the driver server, open a .NET command prompt.
3. Issue the command `InstallUtil IDMPowerShellService.exe` to register the service and create the correct registry entries.

   The default location of the service is `C:\novell\remoteloader\`*`Version`*`\IDMPowerShellService.exe`, where *`Version`* is either the 32-bit folder or the 64-bit folder.
4. To start the service, go to the Settings view and click **Control Panel**.
5. Click **System and Security > Administrative Tools > Services**.
6. Right-click the service **IDM_PowerShell_Service** and select **Start**.
7. Run the IDM PowerShell service as a user and ensure that the user is a member of Recipient Management and View-Only Organization Management.

NOTE: To uninstall the service, open a .NET command prompt and issue the `InstallUtil /u IDMPowerShellService.exe` command.

Configuring the Driver

You need to modify the existing driver object to enable provisioning with Exchange Server 2019 and Exchange Server 2016.

Modifying an Existing Driver in Designer

1. Right-click the Active Directory driver in the Modeler, then select **Properties**.
2. Select **Driver Configuration > Driver Parameters**.
3. Click **Show Exchange Management Options** and select **show**.
4. Click **Exchange Management interface type** and select **IDM_Powershell_Service**.
5. In the **Exchange Server FQDN** box, specify the preferred server to be connected by the Powershell service in a multiple exchange server environment.

   Otherwise, IDM Powershell service will auto discover an exchange server and use it.
6. Click **OK**.

Modifying an Existing Driver in iManager

1. In iManager, click **Identity Manager Administration**.
2. Select **Administration > Identity Manager Overview**.
3. Select the driver set where the Active Directory driver is stored.
4. Click the upper right corner of the Active Directory driver, then click **Edit properties**.
5. In the **Driver Configuration** tab, click **Exchange Management interface type** and select **IDM Powershell Service**.
6. Click **OK**.
7. Click **Close**.

Configuring the Driver to Support Exchange Server 2019 and Exchange Server 2016 Database Load Balancing

The Active Directory driver supports the database load balancing feature included in Exchange Server 2019 and Exchange Server 2016. You can use the Active Directory driver to auto-provision Exchange Server 2019 and Exchange Server 2016 accounts and enable Exchange to load balance accounts across the databases in your Exchange environment.

To enable load balancing, use either Designer or iManager to set the value of the **HomeMDB** parameter to `defer`.

Configuring an Existing Driver in Designer

1. Right-click the Active Directory driver in the Modeler, then select **Driver** > **Properties**.
2. Select **GCVs**.
3. Select the **Entitlements** tab.
4. Click **Exchange Mailbox Provisioning** and select **Use Policy**.
5. Set the value of the **Exchange HomeMDB** parameter to `defer`.
6. Click **OK**.

Configuring an Existing Driver in iManager

1. In iManager, click **Identity Manager Administration**.
2. Select **Administration > Identity Manager Overview**.
3. Select the driver set where the Active Directory driver is stored.
4. Click the upper right corner of the Active Directory driver, then click **Edit properties**.
5. In the **Global Config Values** tab, click **Exchange Mailbox Provisioning** and select **Use Policy**.
6. Set the value of the **Exchange HomeMDB** parameter to `defer`.
7. Click **OK**.
8. Click **Close**.

Support for Multiple Exchange Server in the Environment

IDM Powershell service supports Exchange Server 2019 and Exchange Server 2016. It also works in an environment where the exchange servers co-exists. In such mixed environment, you must provide the exchange server FQDN to the service to connect to the desired exchange server. It works with only one exchange server at a time. You can reconfigure the driver to work with any exchange server.

D Configuring PowerShell Support

Identity Manager provides support for managing Active Directory and Microsoft Exchange using Windows PowerShell cmdlets.

Overview of PowerShell Functionality

PowerShell is a shell-based automation framework created by Microsoft that allows users to manage the internal functions of other Microsoft products, including Active Directory and Exchange. PowerShell uses special .NET classes called cmdlets to perform various processing actions on objects in your Active Directory or Exchange environments. Identity Manager can use PowerShell cmdlets to perform post-processing on events by sending cmdlets to the Active Directory driver using one or more policies.

NOTE: Only policies from the Subscriber channel can run PowerShell cmdlets. You can only use cmdlets to modify objects in Active Directory or Exchange, not in the Identity Vault.

For more information about PowerShell, see the following resources:

- "Getting Started with Windows PowerShell" (https://docs.microsoft.com/en-us/powershell/scripting/getting-started/getting-started-with-windows-powershell?view=powershell-6)
- "Windows PowerShell Owner's Manual" (http://technet.microsoft.com/library/ee221100.aspx)
- "A Task-Based Guide to Windows PowerShell Cmdlets" (http://technet.microsoft.com/en-us/scriptcenter/dd772285.aspx)

System Requirements

To use PowerShell cmdlets, the following must be installed and running on the Active Directory driver computer:

- Identity Manager AD Exchange service

  NOTE: The following are the service files are available in the supported Exchange Server environments:

  - Exchange Server 2019: The name of the service file is `IDMPowerShellService.exe`.
  - Exchange Server 2016: The name of the service file is `IDMPowerShellService.exe`.

For information about installing the Identity Manager AD Exchange service, see Installing the Exchange Service 2016 and 2019 (page 114) depending on the version of Exchange installed in your environment.

NOTE: Identity Manager only supports using PowerShell cmdlets for Active Directory and Microsoft Exchange with Windows PowerShell 2.0 or later. However, Active Directory driver supports out of the box Exchange Mailbox provisioning functionality using previous versions of PowerShell.

Implementing PowerShell Cmdlets in the Active Directory Driver

To call cmdlets, create a rule which adds the PSE`xecute` containing the PowerShell command string.

The Active Directory driver looks for the PSE`xecute` attribute in the input XDS code, reads any cmdlets embedded in a <`value/`> tag, and sends those commands to the Exchange service running on the Active Directory server. The Active Directory server executes the commands sequentially using a programmatic PowerShell interface.

NOTE: When including the PSE`xecute` attribute in an Add or Modify event policy, you must adhere to the XDS format, or the driver ignores the embedded cmdlets.

Sample Active Directory Policy Rule with Cmdlets

The following is a sample rule created in an Active Directory driver policy that allows an administrator to disable a newly-created user account in Active Directory using the Disable-ADAccount cmdlet.

```
<rule>
  <description>Adding PSExecute to Disable New User Account</description>
  <conditions>
   <and>
    <if-operation mode="regex" op="not-equal">query|status</if-operation>
   </and>
  </conditions>
  <actions>
   <do-set-local-variable name="identityname" scope="policy">
    <arg-string>
     <token-xpath expression='./add-attr[@attr-name="sAMAccountName"]/
value/text()'/>
    </arg-string>
   </do-set-local-variable>
   <do-set-dest-attr-value name="PSExecute">
    <arg-value type="string">
     <token-text xml:space="preserve">Disable-ADAccount -Identity </token-
text>
     <token-local-variable name="identityname"/>
    </arg-value>
   </do-set-dest-attr-value>
  </actions>
 </rule>
```

Note that the example rule above is used for including a PowerShell cmdlet in an Add event. You can also create rules to include PowerShell cmdlets in other types of events. However, if you include a cmdlet in a Modify event, ensure that you use the XDS format for constructing that type of event and including the PSExecute attribute.

Specifically, the PSExecute attribute must include XDS code similar to the following example, where the <value> tag includes the PowerShell command string:

```
   <modify-attr attr-name="PSExecute">
    <add-value>
     <value type="string">New-ADUser -SamAccountName rbigby
-Name "Robin Bigby" -AccountPassword (ConvertTo-SecureString -AsPlainText
"NetIQ1234" -Force) -Enabled $true -Path
'ou=active,ou=workforce,dc=w2008r2vm,dc=com'</value>
    </add-value>
   </modify-attr>
```

Available Active Directory and Exchange Cmdlets

PowerShell includes a wide variety of cmdlets and functions. However, the Active Directory driver only supports Active Directory and Exchange PowerShell modules and cmdlets. For information about using Active Directory PowerShell cmdlets, see "Active Directory Cmdlets in Windows PowerShell" (http://technet.microsoft.com/en-us/library/ee617195.aspx). For information about using Exchange cmdlets, see "Cmdlets" (http://technet.microsoft.com/en-us/library/aa996589.aspx).

NOTE: You can use the PSExecute attribute only with Active Directory and Exchange Cmdlets. It cannot be used for scripting.

Creating Active Directory Policies with Cmdlets

To use cmdlets in Identity Manager, first use Designer to create a new policy in the Active Directory driver. For more information about creating policies in Designer, see *Policies in Designer* (http://www.netiq.com/documentation/idm45/policy_designer/data/bookinfo.html) and *Understanding Policies for Identity Manager* (http://www.netiq.com/documentation/idm45/policy/data/bookinfo.html).

After you create a new policy, add a rule to the policy that includes an `add destination attribute value` action to create the PSE`xecute` attribute, which calls one or more PowerShell cmdlets. You can include several cmdlet strings in multiple `value` tags for a single PSE`xecute` attribute, as necessary.

To configure the rule using the Policy Builder, complete the following steps:

1. In Designer, right-click the policy in the Outline view and select **Edit**.
2. In the Policy Builder, select the location where you want to create the PSE`xecute` attribute.
3. In the toolbar, click **Rule** and select **Action** > **Insert Action After**.
4. In the Do field under **Define new action below**, select **add destination attribute value**.
5. Specify PSE`xecute` as the attribute name.
6. In the **Select mode** field, select **add to current operation**.
7. In the **Select object** field, select **Current object**.
8. In the **Specify value type** field, select **string**.
9. In the **Enter string** field, specify the PowerShell command string you want to use, enclosed in quotation marks.
10. Click **OK**.
11. Save the policy.

When specifying the PowerShell command string, you can include other variables configured in separate actions within the rule, as necessary.

For example, for the sample policy provided in "Sample Active Directory Policy Rule with Cmdlets" on page 118, you first add a rule to define the variable `identityname` as the name of the user account you want to disable using a PowerShell cmdlet, and then you specify the following string for the PSE`xecute` variable, which uses the new `identityname` variable in the PowerShell command string:

```
"Disable-ADAccount -Identity"+Local Variable("identityname")
```

NOTE: You can also configure a policy to execute a specified cmdlet by modifying the XML directly, in the XML Source tab of the Policy Builder.

Verifying Active Directory Cmdlet Execution

When a PowerShell cmdlet runs successfully, the Active Directory returns a specific `success` event in the output XML, with the type `powershell`. After you run a cmdlet, check the output XML file for the following event:

```
<status level="success" event-id="linux-djs#20120510164317#1#2:facdcbbb-
d440-4340-1b85-bbcbcdfa40d4" type="powershell"/>
```

If the PowerShell cmdlet does not run successfully, the driver instead logs an `error` event in the output XML. The error event is similar to the following, including the reason for the failure:

```
<status level="error" type="powershell" text1="Exchange" event-id="linux-
djs#20120528111905#99#1:a6ff6b29-950a-4141-8093-296bffa60a95">Exchange
Exception. code:0x000000b8 Error completing exchange command. ERROR: Cannot
process command because of one or more missing mandatory parameters:
Identity.</status>
```

NOTE: If you execute multiple cmdlets in a single rule and one of the cmdlets does not run successfully, the driver does not execute any subsequent cmdlets in the rule and only logs the error event for the failed cmdlet. The driver does not log error events for the subsequent cmdlets, even though they did not run successfully, because the driver does not run those cmdlets after the failure occurs.

E Trace Levels

The driver supports the following trace levels:

Table E-1 *Supported Trace Levels*

Level	Description
0	No trace messages are displayed or logged
1	Basic trace messages are displayed and logged
2	Level 1 messages and the contents of XML documents that are used during event processing are displayed and logged
3	Trace Level 2 messages and extensive rule processing messages are displayed and logged, plus template instantiations

NOTE: If the driver is installed locally on the Identity Manager server, the driver logs all trace messages together on the local server. However, if the driver uses the Remote Loader, the driver logs only driver shim trace messages on the Remote Loader, while the Identity Manager server logs engine trace messages.

F Microsoft Windows Events

The driver logs the following events to the Password Sync event log in Microsoft Windows environments:

Table F-1 *Logged Password Sync Events*

Event Description	Type	Explanation
The password filter has been fully initialized. Domain Name = *DomainName*, Computer Name = *ComputerName*, Host Name = *HostName*	Information	Identity Manager successfully initialized the PassSync utility.
The password filter could not initialize its registry values.	Error	Identity Manager could not open the registry key `/HKLM/SOFTWARE/NOVELL/PWFILTER`.
The password synchronization notification for user *UserName* failed	Error	Pwfilter could not send a password notification to the PassSync utility for this user account.
The password for user *UserName* could not be changed.	Error	Identity Manager could not change the password for the specified user account.
The password filter RPC server failed to load.	Error	The PassSync remote procedure call (RPC) server could not initialize. Check that RPC services are running on the server.
The password for user *UserName* in directory *DirectoryName* was not synchronized because the password change timed out.	Error	Identity Manager could not synchronize passwords for the specified user because the key exceeded its time to live as set in the driver.
The Cryptographic Service Provider has defaulted to *CSPProvider*. Encryption will be downgraded to the standards of this provider. Execution of the password synchronization server will not be affected. If higher encryption standards are required, please contact your network administrator.	Warning	Identity Manager has defaulted to using the base Cryptographic Service Provider (CSP) specified in the event description.
A request to allocate *RequestedSize* bytes of memory failed. Tag value = *TagValue*.	Error	Identity Manager could not allocate the requested memory.

Event Description	Type	Explanation
Driver NOT synchronizing passwords with the domain controller	Error	Identity Manager and the Active Directory driver are not synchronizing passwords with this domain controller (DC). This may be due to pwfilter not being installed or being installed incorrectly.
Driver is synchronizing passwords with the domain controller	Information	Identity Manager and the Active Directory driver are successfully synchronizing passwords with this DC