IBM Security Verify Governance Identity Manager

Active Directory Adapter with 64-bit Support Installation and Configuration Guide

Contents

Figures

Tables

Chapter 1. Overview

An adapter is an interface between a managed resource and the Identity server.

Adapters can be installed on the managed resource. The Identity server manages access to the resource by using the security system. Adapters function as trusted virtual administrators on the target operating system. The adapter creates, suspends, restores user accounts, and other functions that administrators run manually. The adapter runs as a service, independently of whether you are logged on to the Identity server.

Features of the adapter

You can use the Active Directory Adapter to automate administrative tasks.

- Active Directory
  - Creating an Active Directory account

    Use the adapter to create an Active Directory account on Windows domain servers. For supported versions of Windows domain servers, see the release notes for the adapter.
  - Managing an Active Directory account

    Use the adapter to manage an Active Directory account on Windows domain servers. For supported versions of Windows domain servers, see the release notes for the adapter.
  - Managing an Exchange Mailbox

    For the list of supported versions of Exchange, see the release notes for the adapter.
  - Creating home directories

    Use the adapter to create home directories.
  - Move user in hierarchy

    A user can be moved in different containers managed by the Active Directory Adapter by changing the container of the user from IBM® Security Verify Governance Identity Manager.
  - Managing an Active Directory group

    Use the adapter to add, modify, and delete an Active Directory group.

  The Active Directory Adapter does not create or manage local system accounts. Use the Windows Local Account Adapter for this purpose.

  The Active Directory Adapter requires administrator authority. IBM Security Verify Governance Identity Manager requests might fail if the adapter is not given sufficient authority to perform the requested task.

  The Active Directory Adapter can be installed within the managed domain or in a different domain. If the adapter is installed in a different domain, trusts must be configured on both the domain that is managed and the domain where the adapter is installed. For more information about configuring trusts for domains, see the Microsoft documentation that corresponds to your operating system.

  Configure the Active Directory Adapter to support both subdomains and multiple domains through the Base Point feature on the adapter service form.
- Skype for Business Server

  Running under an account with sufficient authority, the adapter supports Skype for Business. Skype for Business is communications software for instant messaging, conferencing, and telephony solutions.

  The adapter uses a remote PowerShell to interface with the Skype for Business Server and set the Skype for Business attributes.

To manage Skype for Business settings, the CA certificate of the Skype for Business server must be imported in the system trust store.

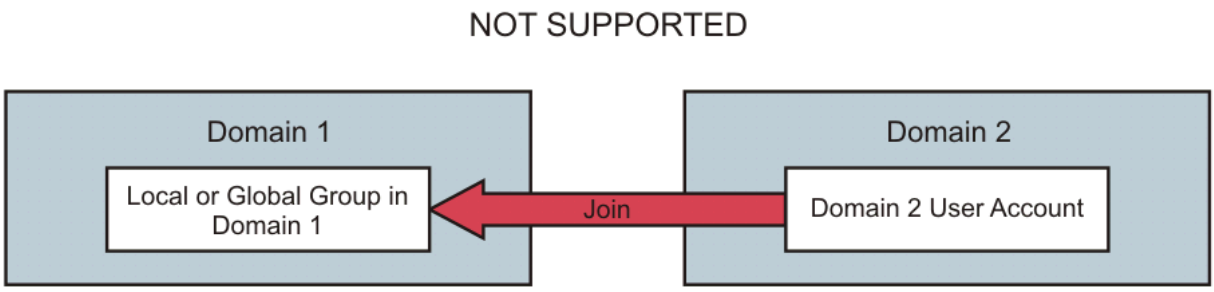
It is not necessary to install the Skype for Business management tools on the machine that runs the adapter. However, it does require that the machine running the adapter has PowerShell 2.0 and that the Execution Policy allows local scripts to be run (RemoteSigned or Unrestricted).

Limitations of the adapter

Running under an account with sufficient authority, the adapter is able to manage user accounts and Exchange mailboxes for all domains within a single forest. Some limitations and configuration issues exist.

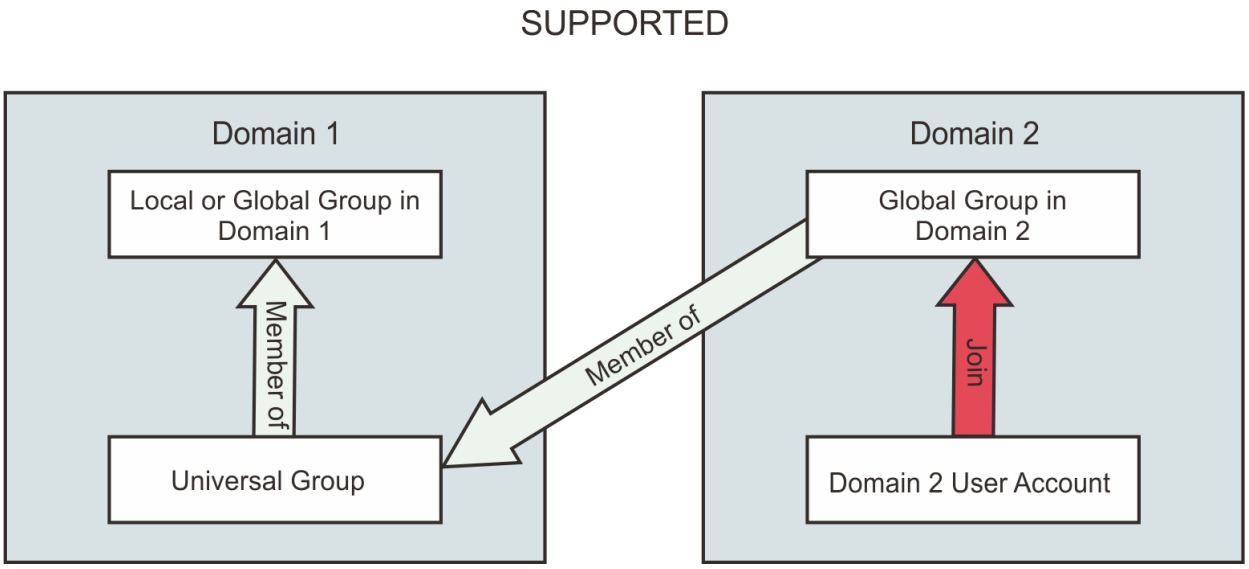
- The Exchange interface now uses a remote PowerShell session with the Exchange server to manage Exchange attributes. This means that it is no longer necessary to install the Exchange management tools on the machine that is running the adapter. However, it does require that the machine running the adapter has PowerShell version 2.0. The Execution Policy must also allow local scripts to be run (RemoteSigned or Unrestricted).
- The adapter cannot manage domains or Exchange servers that are in a different forest.
- The supporting data returned from a reconciliation only includes groups from the domain being reconciled. Local groups from other domains are not returned. Although you can join local groups in other domains, you cannot specify groups in other domains when sending requests to the adapter.

  In this illustration, the user account in Domain 2 is joined directly to a group in Domain 1. While this is possible to do in Active Directory, the adapter does not support it.



You can use Active Directory to create a universal group and make it a member of the group you wish to join. Do not add users directly to the universal group. Instead use Global groups that you can find in the directory. Global groups can be members of universal groups. Add users to a global group and then make them members of the universal group. See your *Microsoft Active Directory documentation* for more information. This configuration is supported by the adapter.

With this configuration, you join Domain 2 users to the global group in Domain 2 and by association they are members of the cross domain group in Domain 1.



- Because you can create multiple service instances on the Identity server that point to the same adapter, ensure that you do not specify base points that overlap. If you use a base point for one service instance that contains the base point of another service instance, only the users in the contained base point are returned as duplicates of the parent base point.

Account limitations for Skype for Business

Running under an account with sufficient authority, the adapter is able to manage Skype for Business accounts. Some limitations exist.

- To enable a user to use Skype for Business, you must provide a Session Initiation Protocol (SIP) address and a Registrar Pool.
- A SIP address specified on account form must be in Session Initiation Protocol format. For example: `sip:abc@test.com`
- When the Telephony field of the user is set to **`Remote call control`** or **`Remote call control only`** on the account, then both **`Line URI`** and **`Line server URI`** fields must also be set.

  The URI values must conform to RFC 3966. For example, a telephone number must start with `tel:` such as `tel:+9122222`.
- Moving a user to another registrar pool changes the Skype for Business Server account location of the user. It does not move the Active Directory account or the user to a new organizational unit (OU) or location.

Overview of SSL and digital certificates

In an enterprise network deployment, you must provide secure communication between the Identity server and the software products and components with which the server communicates.

SSL protocol uses signed digital certificates from a certificate authority (CA) for authentication. SSL secures communication in a configuration. SSL provides encryption of the data that is exchanged between the applications. Encryption makes data that is transmitted over the network intelligible only to the intended recipient.

Signed digital certificates enable two applications that connect in a network to authenticate their identity. An application that acts as an SSL server presents its credentials to verify to an SSL client. The SSL client then verifies that the application is the entity it claims to be. You can configure an application that acts as an SSL server so that it requires the application that acts as an SSL client to present its credentials in a certificate. In this way, the two-way exchange of certificates is completed. A third-party certificate authority issues signed certificates for a fee. Some utilities, such as those provided by OpenSSL, can also provide signed certificates.

You must install a certificate authority certificate (CA certificate) to verify the origin of a signed digital certificate. When an application receives a signed certificate from another application, it uses a CA certificate to verify the certificate originator. A certificate authority can be:

- Well-known and widely used by other organizations.
- Local to a specific region or a company.

Many applications, such as web browsers, use the CA certificates of well-known certificate authorities. Using a well-known CA eliminates or reduces the task of distributing CA certificates throughout the security zones in a network.

The use of SSL authentication

When you start the adapter, it loads the available connection protocols.

The DAML protocol is the only available protocol that supports SSL authentication. You can specify DAML SSL implementation.

The DAML SSL implementation uses a certificate registry to store private keys and certificates. The certTool key and certificate management tool manages the location of the certificate registry. You do not have to specify the location of the registry when you do certificate management tasks.

Private keys, public keys, and digital certificates

Keys, digital certificates, and trusted certificate authorities establish and verify the identities of applications.

SSL uses public key encryption technology for authentication. In public key encryption, a public key and a private key are generated for an application. The data encrypted with the public key can be decrypted only with corresponding private key. Similarly, the data encrypted with the private key can be decrypted only by using the corresponding public key. The private key is password-protected in a key database file. Only the owner can access the private key to decrypt messages that are encrypted with the corresponding public key.

A signed digital certificate is an industry-standard method of verifying the authenticity of an entity, such as a server, a client, or an application. To ensure maximum security, a third-party certificate authority provides a certificate. A certificate contains the following information to verify the identity of an entity:

Organizational information
This certificate section contains information that uniquely identifies the owner of the certificate, such as organizational name and address. You supply this information when you generate a certificate with a certificate management utility.

Public key
The receiver of the certificate uses the public key to decipher encrypted text that is sent by the certificate owner to verify its identity. A public key has a corresponding private key that encrypts the text.

Certificate authority's distinguished name
The issuer of the certificate identifies itself with this information.

Digital signature
The issuer of the certificate signs it with a digital signature to verify its authenticity. The corresponding CA certificate compares the signature to verify that the certificate is originated from a trusted certificate authority.

Web browsers, servers, and other SSL-enabled applications accept as genuine any digital certificate that is signed by a trusted certificate authority and is otherwise valid. For example, a digital certificate can be invalidated for the following reasons:

- The digital certificate expired.
- The CA certificate that is used to verify that it is expired.

- The distinguished name in the digital certificate of the server does not match with the distinguished name specified by the client.

Self-signed certificates

You can use self-signed certificates to test an SSL configuration before you create and install a signed certificate that is provided by a certificate authority.

A self-signed certificate contains a public key, information about the certificate owner, and the owner signature. It has an associated private key; however, it does not verify the origin of the certificate through a third-party certificate authority. After you generate a self-signed certificate on an SSL server application, you must:

1. Extract it.
2. Add it to the certificate registry of the SSL client application.

This procedure is equivalent to installing a CA certificate that corresponds to a server certificate. However, you do not include the private key in the file when you extract a self-signed certificate to use as the equivalent of a CA certificate.

Use a key management utility to:

- Generate a self-signed certificate.
- Generate a private key.
- Extract a self-signed certificate.
- Add a self-signed certificate.

Usage of self-signed certificates depends on your security requirements. To obtain the highest level of authentication between critical software components, do not use self-signed certificates or use them selectively. You can authenticate applications that protect server data with signed digital certificates. You can use self-signed certificates to authenticate web browsers or adapters.

If you are using self-signed certificates, you can substitute a self-signed certificate for a certificate and CA certificate pair.

Certificate and key formats

Certificates and keys are stored in the files in several formats.

.pem format
A privacy-enhanced mail (.pem) format file begins and ends with the following lines:

```
-----BEGIN CERTIFICATE-----
-----END CERTIFICATE-----
```

A `.pem` file format supports multiple digital certificates, including a certificate chain. If your organization uses certificate chaining, use this format to create CA certificates.

.arm format
An `.arm` file contains a base-64 encoded ASCII representation of a certificate, including its public key, not a private key. The `.arm` file format is generated and used by the IBM Key Management utility.

.der format
A `.der` file contains binary data. You can use a `.der` file for a single certificate, unlike a `.pem` file, which can contain multiple certificates.

.pfx format (PKCS12)
A PKCS12 file is a portable file that contains a certificate and a corresponding private key. Use this format to convert from one type of SSL implementation to another. For example, you can create and export a PKCS12 file with the IBM Key Management utility. You can then import the file to another workstation with the certTool utility.

Chapter 2. Planning

Installing and configuring the adapter involves several steps that you must complete in a specific sequence. Follow the roadmap for the main tasks.

Roadmap for Adapter Development Kit based adapters, using `Setup.exe`

Follow this section when using the guide to install, configure, troubleshoot, or uninstall the adapter.

Pre-installation

Complete these tasks.

1. Verify that your environment meets the software and hardware requirements for the adapter. See *Prerequisites*.
2. Obtain the installation software. See *Software downloads*.
3. Obtain the necessary information for the installation and configuration. See *Installation worksheet*.

Installation

Complete these tasks.

1. Install the adapter binary.
2. Install 3rd party client libraries.
3. Set up the adapter environment.
4. Import the adapter profile.
5. Restart the adapter service.
6. Create an adapter service/target.
7. Install the adapter language package.
8. Verify that the adapter is working correctly.

Upgrade

You can do an upgrade or do a full installation. Review the *Release Notes*® for the specific adapter before you proceed.

Configuration

Complete these tasks.

1. Configure secure communication between the Identity server and the adapter.
    a. Configure 1-way authentication.
    b. Configure 2-way authentication.
2. Configure secure communication between the adapter and the managed target.
    a. Configure 1-way authentication.
    b. Configure 2-way authentication.
3. Configure the adapter.
4. Modify the adapter profiles.
5. Customize the adapter.

Troubleshooting

See the following topics.

- Techniques for troubleshooting problems
- Configure debugging
- Logs
- Error messages and problem solving

Uninstallation

Complete these tasks.

1. Stop the adapter service.
2. Uninstall the adapter binary
3. Remove 3rd party client libraries.
4. Delete the adapter service/target.
5. Delete the adapter profile.

Reference

See the following topics.

- Adapter attributes and object classes
- Adapter attributes by operations
- Special attributes

Prerequisites

Verify that your environment meets the software and hardware requirements for the adapter.

Use the Preinstallation roadmap to prepare the environment.

Table 1. Prerequisites to install the adapter

Prerequisite	Description
System	• A 64-bit x86-based microprocessor. • A minimum of 256 MB of memory. • At least 300 MB of free disk space.
Operating system	See the Release Notes for the supported software versions.
Network connectivity	• Internet Protocol network • For security purposes, the adapter must be installed on a Windows NT File System (NTFS).
System administrator authority	The person that performs the Active Directory Adapter installation procedure must have system administrator authority to complete the steps in this chapter.
Identity server	See the Release Notes for the supported software versions.
Optionally, Skype for Business Server	See the Release Notes for the supported software versions.

Table 1. Prerequisites to install the adapter (continued)	
Prerequisite	**Description**
Optionally, Exchange Server	See the Release Notes for the supported software versions.

Software downloads

Download the software through your account at the IBM Passport Advantage® website.

Go to IBM Passport Advantage.

See the corresponding *IBM Security Verify Governance Identity Manager Download Document* for instructions.

Note:

You can also obtain additional adapter information from IBM Support.

Chapter 3. Installing

Installing the adapter mainly involves importing the adapter profile and creating an adapter service. Depending on the adapter, several other tasks can be involved to completely install it.

Installing the adapter binaries and libraries

Use the Active Directory Adapter installer to manually install the adapter.

About this task

The Active Directory Adapter for IBM Security Verify Governance Identity Manager installation program is available for download from the IBM Web site. Contact your IBM account representative for the Web address and download instructions.

To manually install the adapter, complete these steps.

Note: All directory paths apply to Windows operating systems. Change the directory paths as needed for UNIX operating systems.

If you are updating a previous installation, the adapter you want to update must already exist. If it does not exist, the software generates the following message:

```
Adapter is not found at specified location.
Can not perform Update Installation. Please correct
the path of installed adapter or select Full Installation.
```

Procedure

1. If you downloaded the installation software from Passport Advantage, perform the following steps:
   a) Create a temporary directory on the computer on which you want to install the software.
   b) Extract the contents of the compressed file into the temporary directory.
2. Start the installation program with the *SetupAD64.exe* file in the temporary directory.
3. Select the language and click **OK** to display the Introduction window.
4. On the Introduction window, click **Next**.
5. Select either **Full installation** or **Update installation** and click **Next** to display the Choose Install Folder window. Remember that the adapter must already exist if you want to perform an updated installation.
6. Specify where you want to install the adapter in the Directory Name field. Do one of the following.
   - Click **Next** to accept the default location.
   - Click **Browse** and navigate to a different directory and click **Next**.
7. Review the installation settings at the Pre-Installation Summary window and do one of the following:
   - Click **Previous** and return to a previous window to change any of these settings.
   - Click **Install** when you are ready to begin the installation.
8. Click **Done** on the Install Complete window.

Verifying the adapter installation

To determine whether the adapter is installed correctly, verify that required components exist.

bin
The following components exist in the `bin` directory:

- ADAgent.exe

- agentCfg.exe
- CertTool.exe
- Exchg2010.dll
- fipenable.exe
- IsamTool.exe
- regis.exe
- LyncLib.dll

data
Initially, the `data` directory is empty.

license
The `license` directory contains files that provide license information in supported languages.

log
The `log` directory contains the adapter log files. After the adapter installation is complete, the adapter creates `WinADAgent.log` file.

Uninstall IBM Windows AD Adapter for ITIM (64 Bit)
The directory contains the *Uninstall IBM Windows AD Adapter for ITIM (64 Bit).exe* file. You can uninstall the adapter from agent server workstation by using the `uninstaller.exe` file.

After the adapter installation completes, ensure that windows service for Tivoli Active Directory Agent is created and its status is *Started*. To view the windows service status:

1. Click **Start** > **Programs** > **Administrative Tools** > **Services** to display the Services page.
2. Search for the service that is named ISIM Active Directory Adapter.

The adapter copies the following files to the `system32` directory:

- AdkApi.dll
- ErmApi.dll
- ErmApiDaml.dll
- icudt57.dll
- icuuc57.dll
- libcrypto-1_1-x64.dll
- libssl-1_1-x64.dll

Review the installer log file `IBM_Windows_AD_Adapter_for_ITIM_(64_Bit)_InstallLog.log` located in the installation directory for any errors.

Set up an adapter environment

Set up your adapter environment for use with Identity server.

Installing the CA certificate of the Skype for Business Server

If the adapter manages Skype for Business Server, install the CA certificate of the Skype for Business Server on the computer where the adapter is running.

About this task

After you install the adapter, you must to install the CA certificate of the Skype for Business Server in the truststore for the adapter service account.

Procedure

1. Run the **`mmc.exe`** command from the **Start** menu or a command prompt.

2. Add the certificate snap-in.
3. Select **Service Account** and click **Next**.
4. Select **Local computer** and click **Next**.
5. Select **ISIM Active Directory Adapter** and click **Next**.
6. Right-click **Trusted Root Certification Authorities** and select **All Tasks\Import...**
7. Select the CA certificate file and import the file to the truststore.
8. Restart the adapter service.

Importing the adapter profile

You can import a profile definition file, which creates a profile in Identity server. Use this option for importing adapter profiles.

Before you begin

- The Identity server is installed and running.
- You have administrator authority on the Identity server.
- The file to be imported must be a Java™ archive (JAR) file. The `<Adapter>Profile.jar` file includes all the files that are required to define the adapter schema, account form, service/target form, and profile properties. If necessary, you can extract the files from the JAR file, modify the files, and repackage the JAR file with the updated files.

About this task

Target definition files are also called adapter profile files. The profile definition files are provided with the various IBM Security Verify Adapter. The adapter profile must be imported because it defines the types of resources that the Verify Governance Identity Manager server can manage.

The adapter profile definition file is used to create a target profile on the Verify Governance Identity Manager server and to establish communication with the adapter. If the adapter profile is not imported, you cannot create a connector for that adapter type.

An upload error might occur when no file is selected, or when the file is empty, or due to any upload operation error, such as a timeout or connection error. If the adapter profile is not installed correctly, the adapter cannot function correctly. You cannot create a connector with the adapter profile or open and account on the service. You must import the adapter profile again.

This task can be completed from the Enterprise Connectors module in the Administration Console. To import an adapter target profile, complete these steps:

Procedure

1. Log in to the Verify Governance Identity Manager Administration Console.
2. From the Administration Console, select **Enterprise Connectors**.
3. Select **Manage** > **Profiles**.
4. Optional: Click **Filter** to toggle the filter on to refine your search results, or click **Hide Filter** to toggle the filter off. When the filter is visible, you can specify search criteria for your requests, and then click **Search**.
5. Click **Actions** > **Import**.
6. On the **Import** page, complete these steps:
   a) Select **Profile**.
   b) Click **Browse** to locate the JAR file that you want to import.
   c) Click **Upload file**.
   A message indicates that you successfully imported a profile.

7. Click **Close**.
   The new profile is displayed in the list of profiles.

Results

The upload is synchronous but has a timeout. The progress bar on the **Import** page accurately indicates the upload status. However, when a timeout is reached, the following message occurs: "The import is still in progress and will complete shortly. Close this window to proceed." If you see that message, allow a few minutes for the upload to complete and for the profile to be available.

What to do next

After the target profile is imported successfully, complete these tasks.

- Import the attribute mapping file. See Importing attribute mapping file.
- Create a connector that uses the target profile. See “Adding a connector” on page 14.

Adding a connector

After you import the adapter profile on the Verify Governance Identity Manager server, add a connector so that Verify Governance Identity Manager server can communicate with the managed resource.

Before you begin

Complete Importing the adapter profile.

Note: If you migrated from Verify Governance Identity Manager V5.2.2 or V5.2.2.1 and want to add or configure a connector, see *Adding and configuring a connector for each target* in the IBM Security Verify Governance Identity Manager product documentation.

About this task

The connectors consolidate, extract, and reconcile user identities, organization units, permissions, and user entitlements with the most common enterprise applications. Configure a connector to keep the Access Governance Core repository synchronized with the target system.

This task can be completed from the Enterprise Connectors module in the Administration Console.

Procedure

To add a connector, complete these steps.

1. Log in to the Verify Governance Identity Manager Administration Console.
2. From the Administration Console, select **Enterprise Connectors**.
3. Select **Manage** > **Connectors**.
   A list of connectors is displayed on the **Connectors** tab.
4. Click **Filter** to toggle the filter on to refine your search results, or click **Hide Filter** to toggle the filter off. When the filter is visible, you can specify search criteria for your requests, and then click **Search**.
5. To view all of the columns in the list of connectors, expand the **Connectors** pane.
6. Click **Actions** > **Add**.
   The **Connector Details** pane is enabled for your input.
7. On the **Connector Details** tab, complete these steps:
   a) Assign a name and description for the connector.
   b) Select the target profile type as `Identity Brokerage` and its corresponding target profile.
   c) Select the entity, such as **Account** or **User**.
      Depending on the connector type, this field might be preselected.
   d) Optional: Select **Trace ON** and the corresponding **Trace Level** to enable trace logs.

The available trace levels are DEBUG, INFO, and ERROR.

e) Optional: Select **History ON** to save and track the connector usage.

f) Click **Save**.

The fields for enabling the channels for sending and receiving data are now visible.

g) Select and set the connector properties in the **Global Config** accordion pane.

For information about the global configuration properties, see Global Config accordion pane.

h) Click **Save**. The fields for enabling the channels for sending and receiving data are now visible.

Results

The connector is saved and added to the list of connectors in the **Connectors** pane.

If you cannot create a connector with the target profile or open an account on an existing connector, the target profile was not installed correctly during the import. You must import the target profile again.

What to do next

Enable the channel modes to synchronize the data between the target systems and Verify Governance Identity Manager. For more information, see "Enabling connectors" on page 15.

Enabling connectors

After you create a connector, by default it is in a disabled state. You must enable a connector to use it.

Before you begin

Table 2. Prerequisites for enabling a connector

Prerequisite	Find more information
A connector must exist in Verify Governance Identity Manager.	"Adding a connector" on page 14.
Ensure that you enabled the appropriate channel modes for the connector.	"Reviewing and setting channel modes for each new connector" on page 16.

Procedure

To enable a connector, complete these steps:

1. Log in to the Verify Governance Identity Manager Administration Console.
2. From the Administration Console, select **Enterprise Connectors**.
3. Select **Manage** > **Connectors**.

   A list of connectors is displayed on the **Connectors** tab.
4. Optional: Click **Filter** to toggle the filter on to refine your search results, or click **Hide Filter** to toggle the filter off. When the filter is visible, you can specify search criteria for your requests, and then click **Search**.
5. Optional: To view all of the columns in the list of connectors, expand the **Connectors** pane.
6. Select the connector that you want to enable.
7. On the **Connector Details** tab, complete these steps:

   a) Select the channel modes that you want to enable, and then click **Save**. Depending on the channels that you enable, the corresponding **Channel** tabs are displayed.

   Enable write-to channel
   Propagates every change in the Access Governance Core repository into the target system.

For connectors that are not HR feed, the check boxes for enabling the read-from channel and the write-to channel are available.

Enable read-from channel
Reads the INPUT EVENTS and USER DATA from the target system. Imports data from the target system to the Access Governance Core repository.

For HR feed connectors, only the check box for enabling the read-from channel is available.

Enable reconciliation
Synchronizes the modified data between the Access Governance Core repository and the target system.

Results
The connector is enabled

What to do next

Enable the channel modes to synchronize the data between the target systems and Verify Governance Identity Manager.

Reviewing and setting channel modes for each new connector

Use this procedure to set up the read-from and write-to channels and to set the synchronization schedule for each new connector.

About this task

Note: Legacy Verify Governance Identity Manager Enterprise connectors use `Reconciliation` channel, whereas Identity Brokerage Enterprise connectors use `Read From Channel` and `Change Log Sync`.

For more information about any of tasks in the following steps, see the IBM® Security Identity Governance and Intelligence product documentation.

Procedure

To enable the read-from and write-to channels, and to set the change log synchronization schedule for each new connector, complete these steps in Verify Governance Identity Manager V5.2.3:

1. Log in to the Verify Governance Identity Manager Administration Console.
2. From the Administration Console, select **Enterprise Connectors**.
3. Select **Manage** > **Connectors**.
   A list of connectors is displayed on the **Connectors** tab.
4. Optional: Click **Filter** to toggle the filter on to refine your search results, or click **Hide Filter** to toggle the filter off. When the filter is visible, you can specify search criteria for your requests, and then click **Search**.
5. Optional: To view all of the columns in the list of connectors, expand the **Connectors** pane.
6. Select the connector that you want to enable.
7. On the **Connector Details** tab, complete these steps:
   a) Select the channel modes that you want to enable, and then click **Save**. Depending on the channels that you enable, the corresponding **Channel** tabs are displayed, in which you can do more configuration, such as mapping attributes and setting up rules.

      Enable write-to channel
      Propagates every change in the Access Governance Core repository into the target system.

      Enable read-from channel
      Reads the INPUT EVENTS and USER DATA from the target system. Imports data from the target system to the Access Governance Core repository.

Enable reconciliation
Synchronizes the modified data between the Access Governance Core repository and the target system.

8. Select **Monitor** > **Change Log Sync Status**.
   A list of connectors is displayed.
9. On the **Change Log Sync Status** tab, complete these steps:
   a) Optional: Click **Filter** to toggle the filter on to refine your search results, or click **Hide Filter** to toggle the filter off. When the filter is visible, you can specify search criteria for your requests, and then click **Search**.
   b) Select a connector, and click **Actions** > **Sync Now**.
      The synchronization process begins.
   c) Optional: To view the status of the synchronization request, select **Sync History** in the right pane.
      Information about the synchronization is displayed in the **Sync History** tab.
10. Set the change log synchronization schedule for each new connector that you migrated.
11. When the connector configuration is complete, enable the connector by completing these steps:
    a) Select **Manage** > **Connectors**.
    b) Select the connector that you want to enable, and then select the **Enable** check box in the **Connector Details** tab.
    c) Click **Save**.
       For more information, see "Enabling connectors" on page 15.

       For Identity Brokerage connectors that are not HR feed, the check boxes for enabling the read-from channel and the write-to channel are available.

       For Identity Brokerage HR feed connectors, only the check box for enabling the read-from channel is available.
12. Start the connector by selecting **Monitor** > **Connector Status**. Select the connector that you want to start, and then select **Actions** > **Start**.

Service/Target form details

Complete the service/target form fields.

On the General Information tab:

Service Name
Specify a name that defines this adapter service on the Identity server.

Description
Optional: Specify a description for this service.

URL
Specify the location and port number of the adapter. The port number is defined in the protocol configuration by using the **agentCfg** program. For more information, see "Modifying protocol configuration settings" on page 29. URL is a required field.

If https is specified as part of the URL, the adapter must be configured to use SSL authentication. If the adapter is not configured to use SSL authentication, specify `http` for the URL. For more information, see "Configuring SSL authentication" on page 44.

User Id
Specify the DAML protocol user name. The user name is defined in the protocol configuration by using the **agentCfg** program. For more information, see "Modifying protocol configuration settings" on page 29.

Password
Specify the password for the DAML protocol user name. This password is defined in the protocol configuration by using the **agentCfg** program. For more information, see "Modifying protocol configuration settings" on page 29.

Owner
Optional: Specify the service owner, if any.

Service Prerequisite
Optional: Specify an existing service that is a prerequisite for the adapter service.

On the Status and information tab
This page contains read only information about the adapter and managed resource. These fields are examples. The actual fields vary depending on the type of adapter and how the service form is configured. The adapter must be running to obtain the information. Click **Test Connection** to populate the fields.

Last status update: Date
Specifies the most recent date when the Status and information tab was updated.

Last status update: Time
Specifies the most recent time of the date when the Status and information tab was updated.

Managed resource status
Specifies the status of the managed resource that the adapter is connected to.

Adapter version
Specifies the version of the adapter that the service uses to provision request to the managed resource.

Profile version
Specifies the version of the profile that is installed in the Identity server.

ADK version
Specifies the version of the ADK that the adapter uses.

Installation platform
Specifies summary information about the operating system where the adapter is installed.

Adapter account
Specifies the account that running the adapter binary file.

Adapter up time: Date
Specifies the date when the adapter started.

Adapter up time: Time
Specifies the time of the date when the adapter started.

Adapter memory usage
Specifies the memory usage for running the adapter.

If the connection fails, follow the instructions in the error message. Also

- Verify the adapter log to ensure that the test request was successfully sent to the adapter.
- Verify the adapter configuration information.
- Verify service parameters for the adapter profile. For example, verify the work station name or the IP address of the managed resource and the port.

Verifying that the adapter is working correctly

After you install and configure the adapter, verify that the installation and configuration are correct.

Procedure

1. Test the connection for the service that you created on the Identity server.
2. Run a full reconciliation from the Identity server.
3. Run all supported operations such as add, modify, and delete on one user account.

4. Verify the `ibmdi.log` file after each operation to ensure that no errors are reported.
5. Verify the `trace.log` file to ensure that no errors are reported when you run an adapter operation.

Installing and uninstalling in silent mode

You can install and uninstall the Active Directory Adapter with 64-bit support by using the silent mode.

Silent installation suppresses the wizard and the Launcher User Interfaces (UIs) that do not display any information or require interaction. You can use the –i silent option to install or uninstall the adapter in silent mode.

Note: If you install the adapter in silent mode, the uninstaller runs in silent mode irrespective of whether you use the –i silent option or not.

Adapter installation in silent mode

You can install the adapter by using the silent mode.

Installing the adapter with default options

Run the following command from command line to install the Active Directory Adapter with 64-bit support by using the –i silent option:

```
SetupAD64.exe –i silent -DLICENSE_ACCEPTED=TRUE
```

When you install the adapter by using the specified command, the adapter is installed with these default values.

Table 3. Default values	
Installation directory	$PROGRAMS_DIR_64$\IBM\ISIM\Agents\ADAgent
Adapter name	ADAgent
Installation option	Full installation

Installing the adapter with command line options

You can specify the listed installation options from the command line when you install the adapter by using the silent mode. For example, if you want to override the default installation directory path, run the following command:

```
SetupAD64.exe –i silent  -DLICENSE_ACCEPTED=TRUE -DUSER_INSTALL_
DIR="c:\ISIM\MyFolder"
```

Note:

- The `-D` option is followed by a variable and a value pair without any space after the `-D` option.
- You must wrap arguments with quotation marks when the arguments contain spaces.

Table 4. Installation options	
Option	**Value**
-DUSER_INSTALL_DIR=Value	Value overrides the default installation directory path. For example, `D:\ISIM\MyFolder`.
-DLICENSE_ACCEPTED=Value	Accept the IBM license for the adapter, the value must be TRUE. When you do not specify this option, the default value is FALSE.

Installing the adapter by using the response file

Generating the response file

You can use response file to provide inputs during silent installation. Generate the response file by running the following command, which runs the installer in interactive mode and installs the adapter.

```
SetupAD64.exe –r "Full path of response file"
```

For example:

```
SetupAD64.exe –r "c:\temp\WinAD64Response.txt"
```

Note: If you run this command to only generate the response file, you must uninstall the adapter by using the uninstaller.

Creating the response file manually

You can also manually create the response file with the following content:

```
#Start of Response file
#Choose Install Folder
#---------------------
USER_INSTALL_DIR=c:\\ISIM\\agents\\ADAgent
#Has the license been accepted
#-----------------------------
LICENSE_ACCEPTED=TRUE
#End of Response file
```

After you create the response file, you can use it as:

```
SetupAD64.exe –i silent -f "Full path of response file"
```

Installing the adapter on Windows Server Core

To install the Active Directory Adapter with 64-bit support on Windows Server Core, run the installer from a command line with the `–i console` option.

Adapter uninstallation in silent mode

Run the following command from the command line to uninstall the Active Directory Adapter with 64-bit support by using the –i silent option.

Specify the full path when you are not running the command from the `Uninstall_IBM Windows AD Adapter for ITIM (64 Bit)` directory in the installation directory of the adapter.

```
"Uninstall IBM Windows AD Adapter for ITIM (64 Bit).exe" -i silent
```

For example, `"C:\Program Files\IBM\ISIM\Agents\ADAgent\Uninstall_IBM Windows AD Adapter for ITIM (64 Bit)\Uninstall IBM Windows AD Adapter for ITIM (64 Bit).exe" -i silent`.

Note: Restart the workstation after you install or uninstall the adapter.

Chapter 4. Upgrading

You can either update the Active Directory Adapter or the Adapter Development Kit (ADK).

The ADK is the base component of the adapter. While all adapters have the same ADK, the remaining adapter functionality is specific to the managed resource.

Updating an adapter is only supported for the current IBM Security Verify Identity release. If your current adapter is version 5.0.x or 5.1.x, you must uninstall the adapter first.

If only a code fix has been made to the ADK, instead of upgrading the entire adapter, you can upgrade just the ADK to the newer version. See Upgrading the ADK.

Upgrading the Active Directory Adapter

You can update the Active Directory Adapter.

About this task

For adapter versions 6.x and later, use the adapter update option if you want to keep the adapter configuration (registry keys and certificates) unchanged.

If the update installation option is selected, the path of the existing installed adapter is required. The installer replaces the binary files and the DLLs of the adapter and the ADK. The installer does not prompt for any configuration information during an update installation.

Note: Adapter-related registry keys are not modified. The update installation does not create an additional service for the adapter.

When you update to a higher version of the adapter, first install the new version of the adapter before you uninstall the old version. Installing the update in this sequence maintains all of your current configuration settings, the certificate, and private key. When you install the adapter, specify the same installation directory where the previous adapter was installed. For more information about installing the adapter, see Chapter 3, “Installing,” on page 11.

To update an existing adapter, complete the following steps:

Procedure

1. Stop the Active Directory Adapter service.
2. Install the new version of the adapter.

   When the upgraded adapter starts for the first time, new log files are created, replacing the old files.

   The adapter installer allows an update installation of the adapter, for adapters versions 6.0 or later.

Upgrading the Windows Active Directory in graphical user interface mode

Use the adapter update option, if you want to keep the adapter configuration (registry keys and certificates) unchanged.

About this task

If the update installation option is selected, the installer detects the path of the existing installed adapter. If no prior installation of the adapter is found on the system, the installer displays an error message. The installer replaces the binary files and the DLLs of the adapter and the ADK. The installer does not prompt for any configuration information during an update installation.

Note: Adapter-related registry keys are not modified. The update installation does not create a service for the adapter.

To maintain your current configuration settings, and the certificate and private key during an update, do not uninstall the old version of the adapter. For more information about installing the adapter, see "Installing the adapter binaries and libraries" on page 11.

Procedure

1. Downloaded the installation software from Passport Advantage.
   a) Create a temporary directory on the computer on which you want to install the software.
   b) Extract the contents of the compressed file into the temporary directory.
2. Run the `SetupAD64.exe` file in the temporary directory to start the installation program.
3. Select the language and click **OK** to display the **Introduction** window.
4. On the **Introduction** window click **Next**.
5. Select **Update installation** option and click **Next**.

   Note: The adapter must exist, if you want to perform an update installation. If it does not exist, the software generates the following message: `Update not supported when the adapter is not previously installed. Cannot perform Update Installation. IBM Tivoli Windows Active Directory Adapter (64 Bit) is not installed on this machine. Please select Full Installation.`

   The adapter displays the path of the adapter installation that is to be updated.
6. Click **OK** to view the pre-Installation **Summary** window.
7. Review the installation settings on the pre-Installation **Summary** window and click **Install**.
8. Click **Done** on the **Install Complete** window.

Upgrading Windows Active Directory in silent mode by using command-line parameters

You can use the **-i** silent option to update the adapter in silent mode.

About this task

Note: If you install adapter in silent mode, the uninstaller runs in silent mode irrespective of whether you are using -i silent option.

The installer refers to the adapter registry keys to detect if the adapter is installed on the system where you are running the command. The installer updates the adapter only if it successfully detects a prior installation of the adapter on the system. If no prior installation is found on the system, the installation ends. A log file `IBM_Tivoli_Windows_Active_Directory_Adapter_(64_Bit)_InstallLog` is generated with this information in the Desktop.

Note: When performing an update installation the **-DUSER_INSTALL_DIR** parameter must not be used.

Procedure

Issue one of the following commands on a single line:

- ```
  SetupAD64.exe -i silent -DLICENSE_ACCEPTED=TRUE
  -DUSER_INPUT_INSTALL_TYPE_1= -DUSER_INPUT_INSTALL_TYPE_2=\"Update Installation\"
  -DUSER_INPUT_INSTALL_TYPE_BOOLEAN_1=0
  -DUSER_INPUT_INSTALL_TYPE_BOOLEAN_2=1
  ```
- ```
  SetupAD64.exe -i silent -DLICENSE_ACCEPTED=TRUE
  -DUSER_INPUT_INSTALL_TYPE_BOOLEAN_1=0
  -DUSER_INPUT_INSTALL_TYPE_BOOLEAN_2=1
  ```

Upgrading Windows Active Directory in silent mode by using a response file

You can use response file to provide inputs during silent installation.

Procedure

1. Use one of these actions to create a response file.
   - Generate a response file by issuing the command:

     ```
     SetupAD64.exe -r "Full path of response file"
     ```

     This command runs the installer in interactive mode and installs the adapter. After the installation completes, the file specified as "*Full path of response file*" is created. The file contains the required parameters.

     Note: If you are running this command to generate only the response file, you must uninstall the adapter by using the uninstaller.
   - Manually create a response file:

     Use a text editor to create a text file. For example create a file `WinAD64InstallParameters.txt`, with the following content:

     ```
     #Has the license been accepted
     #-----------------------------
     LICENSE_ACCEPTED=TRUE

     #Select Install Type
     #-------------------
     USER_INPUT_INSTALL_TYPE=\"\",\"Update Installation\"
     USER_INPUT_INSTALL_TYPE_1=
     USER_INPUT_INSTALL_TYPE_2=Update Installation
     USER_INPUT_INSTALL_TYPE_BOOLEAN_1=0
     USER_INPUT_INSTALL_TYPE_BOOLEAN_2=1
     ```
2. Issue the command:

   ```
   SetupAD64.exe -i silent -f "Full path of response file"
   ```

   For example:

   ```
   SetupAD64.exe -i silent -f "C:\WinAD64InstallParameters.txt"
   ```
3. Restart the workstation.

Chapter 5. Configuring

After you install the adapter, configure it to function correctly. Configuration is based on your requirements or preference.

Communication between the Active Directory Adapter and the Identity server

The adapter communicates with the Identity server with the Directory Access Markup Language (DAML) protocol. You can configure SSL authentication for the adapter.

Data transfer to the adapter

The Active Directory Adapter is an individual IBM Security Verify Governance Identity Manager software program on a domain controller or a non-domain controller workstation.

Data is transferred between the Active Directory Adapter and the Identity server using the Directory Access Markup Language (DAML) protocol. DAML uses Secure Sockets Layer (SSL) to send XML-formatted messages between the adapter and IBM Security Verify Governance Identity Manager.

IBM Security Verify Governance Identity Manager communicates with the Active Directory Adapter in order to administer user accounts. When the Identity server issues a request to the Active Directory Adapter, the server opens a TCP/IP connection. This connection stays open until the agent completes the request and responds back to the server with an acknowledgment message. After the Identity server receives the anticipated response, it drops the connection to the adapter.

Basic configuration for server-to-adapter SSL communication

The following information pertains to IBM Security Verify Governance Identity Manager deployment on either the WebSphere or the WebLogic application server.

In this configuration, the Identity server initiates communication with the adapter (server-to-adapter) by using one-way authentication over SSL. The version of the SSL protocol that is used is either RSA or Open SSL.

Basic configuration for adapter-to-Active Directory SSL communication

The Active Directory Adapter can be on a domain controller or non-domain controller workstation.

Communication between Active Directory Adapter and Active Directory is not secure. Data sent over the network is in plain text. The Active Directory Adapter uses secure authentication method (no SSL) to identify itself to the active directory. For this, provision is made on the Active Directory service form to accept a user ID and password to authenticate to the Active Directory.

Active Directory uses Kerberos, and possibly NTLM, to authenticate the Active Directory Adapter. When the user name and password are NULL, ADSI binds to the object using the security context of the calling thread, which is either the security context of the user account under which the application is running or the context of the client user account that the calling thread represents.

When SSL communication is set up between the adapter and Active Directory, it allows data transfer over the network in encrypted form.

SSL communication between the adapter and Active Directory

To use SSL-based encryption while communicating with Active Directory:

- Active Directory must have enabled Public Key Infrastructure (PKI). PKI requires that enterprise certificate authority (CA) is installed on one of the domain controller workstations in the domain. Setting

up an enterprise certificate authority causes an Active Directory server to get a server certificate that can then be used to do SSL-based encryption.

- The certificate must be installed on the workstation on which Active Directory Adapter is running.

Installing Enterprise CA in one of the domain controllers in a domain

To install Enterprise CA in one of the domain controllers in a domain, take these steps:

About this task

Note: Internet Information Services must be stopped before installing the certificate.

Procedure

1. Go to **Control Panel > Add Remove Programs > Windows Components**. Click **Components**.
2. Select **Certificate Services** and click **Next**.
3. A dialog box is displayed. Click **Yes** to continue.
4. Select **Remote Administration mode**. Click Next.
5. Select **Enterprise root CA**. Click **Next** .
6. Specify the information to identify this CA. Click **Next**.
7. Accept the default location or specify a different location to store data related to the certificate server. Click **Next**.
8. If Internet Information Services is running, a dialog box is displayed. Click **OK** to stop the service and continue with the certificate installation.
9. Click **Finish** to complete the installation.

   Note: A restart of the server is not required for SSL communication.

Installing the certificate on the workstation where Active Directory Adapter is running

To install the certificate on the workstation where Active Directory Adapter is running, perform these steps:

Procedure

1. Get the trusted root certificate from certificate server. Usually the certificate is present in the `c:\winnt\system32\certsrv\certEnroll` folder. For example, a certificate name might be `ps0721.agents2.com_PS0721CA(1).crt`
2. Copy the certificate on the workstation where Active Directory Adapter is installed.
3. Double click the certificate.
4. Click **Install Certificate**.
5. Click **Next**.
6. Select **Place all certificates in the following store** and click **Browse**.
7. Select **Show Physical stores** and from the tree view select the folder **Local Computer**.
8. Click **OK**.
9. Click **Next**.
10. Click **Finish** to complete the installation of the certificate.

Configuring the adapter for IBM Security Verify Governance Identity Manager

After you install the adapter, configure the adapter to function correctly.

About this task

Note: The screens displayed in these tasks are examples, the actual screens displayed might differ.

To configure the adapter, perform the following steps:

Procedure

1. Start the adapter service. Use the Windows Services tool.
2. Configure the Directory Access Markup Language (DAML) protocol for the adapter to establish communication with the Identity server. See “Modifying protocol configuration settings” on page 29.
3. Configure the adapter for event notification.
   See Configuring event notification.
4. Install a certificate on the workstation where the adapter is installed and also on the Identity server to establish secure communication between them.
   See “Configuring SSL authentication” on page 44.
5. Import the adapter profile on the Identity server.
6. Configure the adapter service.
7. Use the adapter configuration program, **agentCfg**, to view or modify the adapter parameters.
   See “Starting the adapter configuration tool” on page 27.
8. Configure the adapter account form. See the product documentation.
9. Restart the adapter service after you modify the adapter configuration settings.

Starting the adapter configuration tool

Start the **agentCfg** tool to access the configuration menu, where you can modify the different adapter parameters.

About this task

Note: The screens displayed in the tasks topics are examples. Information in the actual screens might be different.

Procedure

1. Browse to the Windows Command Prompt.
2. In the command prompt, change to the read/write `/bin` subdirectory of the adapter.If the adapter is installed in the default location for the read/write directory, run the following command.

   ```
   cd C:\Program Files\IBM\ISIM\Agents\adapter_name\bin\
   ```

3. Run the following command

   ```
   agentCfg -agent adapterAGNT
   ```

4. At the **Enter configuration key for Agent '*adapterAGNT*'**, type the configuration key for the adapter.

   The default configuration key is `agent`.

   Note: To prevent unauthorized access to the configuration of the adapter, you must modify the configuration key after the adapter installation completes..

The **Agent Main Configuration Menu** is displayed.

```
Agent Main Configuration Menu
-----------------------------------------
A. Configuration Settings.
B. Protocol Configuration.
C. Event Notification.
D. Change Configuration Key.
E. Activity Logging.
F. Registry Settings.
G. Advanced Settings.
H. Statistics.
I. Codepage Support.

X. Done.

Select menu option:
```

The following table lists the different options available in the **Agent Main Configuration Menu**.

Table 5. Options for the main configuration menu

Option	Configuration task
A	Viewing configuration settings
B	Changing protocol configuration settings
C	Configuring event notification
D	Changing the configuration key
E	Changing activity logging settings
F	Changing registry settings
G	Changing advanced settings
H	Viewing statistics
I	Changing code page settings

Related tasks

Accessing help and other options

“Modifying protocol configuration settings” on page 29

The adapter uses the DAML protocol to communicate with the Identity server.

Viewing configuration settings

View the adapter configuration settings for information about the adapter, including version, ADK version, and adapter log file name.

Procedure

1. Access the **Agent Main Configuration** menu.
2. Type A to display the configuration settings for the adapter.

```
Configuration Settings
----------------------------------------
Name            : adapter_nameAgent
Version         : 6.0.4.1200
ADK Version     : 6.0.1017
ERM Version     : 6.0.4.1200
Adapter Events  : FALSE
License         : NONE
Asynchronous ADD Requests  : TRUE (Max.Threads:3)
Asynchronous MOD Requests  : TRUE(Max.Threads:3)
Asynchronous DEL Requests  : TRUE (Max.Threads:3)
Asynchronous SEA Requests  : TRUE (Max.Threads:3)
Available Protocols        : DAML
Configured Protocols       : DAML
Logging Enabled            : TRUE
Logging Directory          : C:\Program Files\IBM\ISIM\Agents\adapter_name\log
Log File Name              : adapter_name.log
Max. log files             : 3
Max.log file size (Mbytes) : 1
Debug Logging Enabled      : TRUE
Detail Logging Enabled     : FALSE
Thread Logging Enabled     : FALSE

Press any key to continue
```

3. Press any key to return to the **Main** menu.

Related tasks

"Starting the adapter configuration tool" on page 27
Start the **agentCfg** tool to access the configuration menu, where you can modify the different adapter parameters.

Modifying protocol configuration settings

The adapter uses the DAML protocol to communicate with the Identity server.

About this task

By default, when the adapter is installed, the DAML protocol is configured for a nonsecure environment. To configure a secure environment, use Secure Socket Layer (SSL) and install a certificate.

The DAML protocol is the only supported protocol that you can use. Do not add or remove a protocol.

Procedure

1. Access the Agent Main Configuration menu.
2. Type B. The DAML protocol is configured and available by default for the adapter.

```
Agent Protocol Configuration Menu
-----------------------------------
Available Protocols: DAML
Configured Protocols: DAML
A. Add Protocol.
B. Remove Protocol.
C. Configure Protocol.

X. Done

Select menu option
```

3. At the Agent Protocol Configuration menu, type C to display the Configure Protocol Menu.

```
Configure Protocol Menu
---------------------------------
A. DAML

X. Done

Select menu option:
```

4. Type a letter to display the Protocol Properties menu for the configured protocol with protocol properties.

   The following screen is an example of the DAML protocol properties.

```
DAML Protocol Properties
-------------------------------------------------------

A. USERNAME           ****** ;Authorized user name.
B. PASSWORD           ****** ;Authorized user password.
C. MAX_CONNECTIONS    100    ;Max Connections.
D. PORTNUMBER         45580  ;Protocol Server port number.
E. USE_SSL            FALSE  ;Use SSL secure connection.
F. SRV_NODENAME       -----  ;Event Notif. Server name.
G. SRV_PORTNUMBER     9443   ;Event Notif. Server port number.
H. HOSTADDR           ANY    ;Listen on address < or "ANY" >
I. VALIDATE_CLIENT_CE FALSE  ;Require client certificate.
J. REQUIRE_CERT_REG   FALSE  ;Require registered certificate.
K. READ_TIMEOUT       0      ;Socked read timeout (seconds)
L. MIN_TLS_LEVEL      1.0    ;Minimum TLS level (0 for none)
X. Done
Select menu option:
```

5. Follow these steps to change a protocol value:
   - Type the letter of the menu option for the protocol property to configure. The following table describes each property.
   - Take one of the following actions:
     – Change the property value and press **Enter** to display the Protocol Properties menu with the new value.
     – If you do not want to change the value, press **Enter**.

Table 6. Options for the DAML protocol menu

Option	Configuration task
A	Displays the following prompt: `Modify Property 'USERNAME':` Type a user ID, for example, `agent`. The Identity server uses this value to connect to the adapter. The default user ID is `agent`.
B	Displays the following prompt: `Modify Property 'PASSWORD':` Type a password, for example, `agent`. The Identity server uses this value to connect to the adapter. The default password is `agent`.
C	Displays the following prompt: `Modify Property 'MAX_CONNECTIONS':` Enter the maximum number of concurrent open connections that the adapter supports. The default number is `100`.
D	Displays the following prompt: `Modify Property 'PORTNUMBER':` Type a different port number. This value is the port number that the Identity server uses to connect to the adapter. The default port number is `45580`.

Table 6. Options for the DAML protocol menu (continued)

Option	Configuration task
E	Displays the following prompt: `Modify Property 'USE_SSL':` TRUE specifies to use a secure SSL connection to connect the adapter. If you set USE_SSL to TRUE, you must install a certificate. FALSE, the default value, specifies not to use a secure SSL connection. **Note:** By default event notification requires USE_SSL set to TRUE. To use event notification, you must set USE_SSL to TRUE and add a certificate and key from the PKCS12 file in the adapter.
F	Displays the following prompt: `Modify Property 'SRV_NODENAME':` Type a server name or an IP address of the workstation where you installed the Identity server. This value is the DNS name or the IP address of the Identity server that is used for event notification and asynchronous request processing. **Note:** If your operating system supports Internet Protocol version 6 (IPv6) connections, you can specify an IPv6 server.
G	Displays the following prompt: `Modify Property 'SRV_PORTNUMBER':` Type a different port number to access the Identity server. The adapter uses this port number to connect to the Identity server. The default port number is 9443.
H	The HOSTADDR option is useful when the system where the adapter is running has more than one network adapter. You can select which IP address the adapter must listen to. The default value is ANY.
I	Displays the following prompt: `Modify Property 'VALIDATE_CLIENT_CE':` Specify TRUE for the Identity server to send a certificate when it communicates with the adapter. When you set this option to TRUE, you must configure options D through I. Specify FALSE, the default value to enable the Identity server to communicate with the adapter without a certificate. **Note:** – The property name is VALIDATE_CLIENT_CERT; however, it is truncated by the `agentCfg` to fit in the screen. – You must use certTool to install the appropriate CA certificates and optionally register the Identity server certificate.

Table 6. Options for the DAML protocol menu (continued)	
Option	**Configuration task**
J	Displays the following prompt: `Modify Property 'REQUIRE_CERT_REG':` This value applies when option I is set to TRUE. Type TRUE to register the adapter with the client certificate from the Identity server before it accepts an SSL connection. Type FALSE to verify the client certificate against the list of CA certificates. The default value is FALSE.
K	Displays the following prompt: `Modify Property 'READ_TIMEOUT':` Type the timeout value in seconds for IBM Security Verify Governance Identity Manager and the adapter connection. This option applies to setups that have a firewall between IBM Security Verify Governance Identity Manager and the adapter. This firewall has a timeout value that is less than the maximum connection age DAML property on IBM Security Verify Governance Identity Manager. When your transactions run longer than the firewall timeout, the firewall terminates the connection. The sudden termination of connections might leave the adapter with incorrect connection threads causing the adapter to crash. When the adapter halts randomly because of the specified setup, change the value for the READ_TIMEOUT. The value must be in seconds and less than the timeout value of the firewall.
L	This option controls the minimum TSL level that is used when SSL is enabled. The setting supersedes the values DISABLE_SSLV3 and DISABLE_TLS10. The valid settings for this value are: – `0`: No restrictions. This setting allows SSLV3 connections which are known to have vulnerabilities. – `1.0`: TLS 1.0 and higher are supported. – `1.1`: TLS 1.1 and higher are supported. – `1.2`: TLS 1.2 and higher are supported. – `1.3`: TLS 1.3 and higher are supported. For backward compatibility, if MIN_TLS_LEVEL is not set, it will be set at startup based on the settings of DISABLE_SSLV3 and DISABLE_TLS10.

6. Follow these steps at the prompt:
   - Change the property value and press **Enter** to display the Protocol Properties menu with the new value.
   - If you do not want to change the value, press **Enter**.
7. Repeat step 5 to configure the other protocol properties.
8. At the Protocol Properties menu, type X to exit.

Related concepts

“SSL certificate management with certTool” on page 47
Use the certTool utility to manage private keys and certificates.

“Configuring SSL authentication” on page 44

You can provide SSL authentication, certificates, and enable SSL authentication with the certTool utility.

Related tasks

"Starting the adapter configuration tool" on page 27
Start the **agentCfg** tool to access the configuration menu, where you can modify the different adapter parameters.

"Installing the certificate" on page 50
After you receive your certificate from your trusted CA, install it in the registry of the adapter.

Changing the configuration key

Use the configuration key as a password to access the configuration tool for the adapter.

Procedure

1. Access the **Agent Main Configuration Menu**.
2. At the **Main Menu** prompt, type D.
3. Do one of the following actions:
   - Change the value of the configuration key and press Enter. The default configuration key is **agent**. Ensure that your password is complex.
   - Press **Enter** to return to the **Main Configuration Menu** without changing the configuration key.

Results

The following message is displayed:

```
Configuration key is successfully changed.
```

The configuration program returns to the **Main Menu** prompt.

Related tasks

"Starting the adapter configuration tool" on page 27
Start the **agentCfg** tool to access the configuration menu, where you can modify the different adapter parameters.

Changing `activity logging` settings

When you enable logging, the adapter maintains a log file of all transactions, *adapter_name*`Agent.log`.

About this task

By default, the log file is in the `\log` directory.

To change the adapter **`activity logging`** settings, take the following steps:

Procedure

1. Access the Agent Main Configuration menu.
2. At the **Main Menu** prompt, type E to display the Agent Activity Logging menu. The following screen displays the default **`activity logging`** settings.

```
Agent Activity Logging Menu
-----------------------------------
A. Activity Logging (Enabled).
B. Logging Directory (current: C:\Program Files\IBM\ISIM\Agents\adapter_nameAgent\log).
C. Activity Log File Name (current: adapter_nameAgent.log).
D. Activity Logging Max. File Size ( 1 mbytes)
E. Activity Logging Max. Files ( 3 )
F. Debug Logging (Enabled).
G. Detail Logging (Disabled).
H. Base Logging (Disabled).
I. Thread Logging (Disabled).
X. Done
Select menu option:
```

3. Perform one of the following steps:
   - Type the value for menu option B, C, D, or E and press **Enter**. The other options are changed automatically when you type the corresponding letter of the menu option. The following table describes each option.
   - Press **Enter** to return to the Agent Activity Logging menu without changing the value.

   Note: Ensure that Option A is enabled for the values of other options to take effect.

*Table 7. Options for the **activity logging** menu*

Option	Configuration task
A	Set this option to enabled to have the adapter maintain a dated log file of all transactions. When the option is set to: • Disabled, pressing the A to key changes to enabled. • Enabled, pressing the A to key changes to disabled. Type A to toggle between the options.
B	Displays the following prompt: `Enter log file directory:` Type a different value for the logging directory, for example, C:\Log. When the logging option is enabled, details about each access request are stored in the logging file that is in this directory.
C	Displays the following prompt: `Enter log file name:` Type a different value for the log file name. When the logging option is enabled, details about each access request are stored in the logging file.
D	Displays the following prompt: `Enter maximum size of log files (mbytes):` Type a new value such as 10. The oldest data is archived when the log file reaches the maximum file size. File size is measured in megabytes. It is possible for the activity log file size to exceed disk capacity.
E	Displays the following prompt: `Enter maximum number of log files to retain:` Type a new value up to 99 such as 5. The adapter automatically deletes the oldest activity logs beyond the specified limit.

*Table 7. Options for the **activity logging** menu (continued)*	
Option	**Configuration task**
F	If this option is set to enabled, the adapter includes the debug statements in the log file of all transactions. When the option is set to: • Disabled, pressing the F key changes the value to enabled. • Enabled, pressing the F key changes the value to disabled. Type F to toggle between the options.
G	If this option is set to enabled, the adapter maintains a detailed log file of all transactions. The **detail logging** option must be used for diagnostic purposes only. Detailed logging enables more messages from the adapter and might increase the size of the logs. When the option is set to: • Disabled, pressing the G key changes the value to enabled. • Enabled, pressing the G key changes the value to disabled. Type G to toggle between the options.
H	If this option is set to enabled, the adapter maintains a log file of all transactions in the Adapter Development Kit (ADK) and library files. Base logging substantially increases the size of the logs. When the option is set to: • Disabled, pressing the H key changes the value to enabled. • Enabled, pressing the H key changes the value to disabled. Type H to toggle between the options.
I	If this option is enabled, the log file contains thread IDs, in addition to a date and timestamp on every line of the file. When the option is set to: • Disabled, pressing the I key changes the value to enabled. • Enabled, pressing the I key changes the value to disabled. Type I to toggle between the options.

Related tasks

"Starting the adapter configuration tool" on page 27
Start the **agentCfg** tool to access the configuration menu, where you can modify the different adapter parameters.

Modifying registry settings

Use the **Agent Registry Menu** to change the adapter registry settings.

Procedure

1. Type F (Registry Settings) at the main menu prompt to display the Registry menu:

```
adapter_name and version Agent Registry Menu
----------------------------------------
A. Modify Non-encrypted registry settings.
B. Modify encrypted registry settings.
C. Multi-instance settings.
X. Done
Select menu option:
```

2. See the following procedures for modifying registry settings.

Modifying non-encrypted registry settings

To modify the non-encrypted registry settings, complete the following steps:

Procedure

1. At the Agent Registry Menu, type A to display the Non-encrypted Registry Settings Menu.

```
Agent Registry Items
----------------------------------
01. CreateUNCHomeDirectories  'FALSE'
02. DeleteUNCHomeDirectories  'FALSE'
03. delRoamingProfileOnDeprov    'FALSE'
04. delUNCHomeDirOnDeprov        'FALSE'
05. ForceRASServerLookup      'FALSE'
06. ForceTerminalServerLookup 'FALSE'
07. ManageHomeDirectories     'FALSE'
08. NotifyIntervalSeconds     '300'
09. ReconHomeDirSecurity      'FALSE'
10. ReconPrimaryGroup         'TRUE'
----------------------------------
           Page 1 of 3

A. Add new attribute
B. Modify attribute value
C. Remove attribute

D. Next Page

X. Done

Select menu option:D
Agent Registry Items
----------------------------------
11. SearchPasswordSettings    'FALSE'
12. UnlockOnPasswordReset     'FALSE'
13. useDefaultDC                    'FALSE'
14. useSSL                                'FALSE'
15. WtsDisableSearch          'TRUE'
16. WtsEnabled                'FALSE'
----------------------------------
           Page 2 of 3

A.  Add new attribute
B.  Modify attribute value
C.  Remove attribute

E.  Prev Page

X.  Done

Select menu option:
```

2. Type the letter of the menu option for the action that you want to perform on an attribute.

Table 8. Attribute configuration option descriptions

Option	Configuration task
A	Add new attribute
B	Modify attribute value
C	Remove attribute

3. Type the registry item name, and press Enter.
4. If you selected option A or B, type the registry item value and press Enter.

   The non-encrypted registry settings menu reappears and displays your new settings.

Results

The following table describes the registry keys and their available settings:

Table 9. Registry key descriptions

Key	Description
CreateUNCHomeDirectories	If this key is set to **TRUE**, the key enables creation of the UNC home directory. The default value is **FALSE**.
DeleteUNCHomeDirectories	If this key is set to **TRUE**, the key enables deletion of the UNC home directory on delete. The default value is **FALSE**.
delRoamingProfileOnDeprovision	If this key is set to **TRUE**, the key enables user profile directory deletion when the user is de-provisioned. After successfully deleting the user from the Active Directory, the adapter deletes the user home directory, subdirectories, and files. If this key is set to **FALSE**, or if the key does not exist, the adapter does not delete the user home directory. The default value is **FALSE**.
delUNCHomeDirOnDeprovision	If this key is set to **TRUE**, the key enables UNC Home directory deletion when the user is de-provisioned. After successfully deleting the user from the Active Directory, the adapter deletes the user home directory, subdirectories, and files. If this key is set to **FALSE**, or if the key does not exist, the adapter does not delete the user home directory. The default value is **FALSE**.
ForceRASServerLookup	If this key is set to **TRUE**, the RASServer is always found from the domain information. If this key is set to **FALSE**, one of these conditions exist: • If the target server is specified in the base point, the target server is used as the RAS server. • If the target server is not specified in the base point, the RAS server is found from the domain information. The default value is **FALSE**.
ForceTerminalServerLookup	If this key is set to **TRUE**, the terminal server is always found from the domain information. If this key is set to **FALSE**, one of these conditions exist: • If the target server is specified in the base point, the target server is used as the terminal server. • If the target server is not specified in the base point, the terminal server is found from the domain information. The default value is **FALSE**.
ManageHomeDirectories	If this key is set to **TRUE**, the adapter performs Add and Delete operations for actual directories. If this key is set to **FALSE**, the adapter updates only the home directory information in the Active Directory. The default value is **FALSE**.

Table 9. Registry key descriptions (continued)

Key	Description
NotifyIntervalSeconds	This key specifies the interval (in seconds) after which the adapter enabled event notification process starts. It can be modified by using the agentCfg tool. The default value is 300 seconds.
ReconHomeDirSecurity	If this key is set to **TRUE**, the adapter brings the Home Security information (NTFS security, share name, and share security) during a reconciliation. The default value is **FALSE**. The reconciliation operation is fast when this key is set to **FALSE**.
ReconPrimaryGroup	The recon operation does not add the primary group to the group list. The **memberof** attribute in Active Directory stores the user's group membership, except the primary group. The primaryGroupID attribute in Active Directory stores the primary group of the user. As a result the primary group must be explicitly added to group list. If this key is set to **TRUE**, the primary group is added to the group list. If this key is set to **FALSE**, the primary group is not added to the group list. The default value is **FALSE**.
SearchPasswordSettings	Most of the password attributes are stored in the Active Directory and are directly retrieved. But some (for example, **Require Unique Password and User Cannot Change Password**) are not stored in the Active Directory. These attributes must be retrieved by using APIs. If this key is set to **TRUE**, the password attributes are retrieved by using the respective API. If this key is set to **FALSE**, the attributes are not retrieved. The default value is **FALSE**. When this key is set to **FALSE**, the password flag attributes are not retrieved and the reconciliation operation is fast.
UnlockOnPasswordReset	If this key is set to **TRUE**, the adapter activates the user on a password change request. The default value is **FALSE**.
useDefaultDC	This key provides failover capability for the adapter when the host specified in the base point is not available. If the adapter cannot connect to the host specified in the base point and the key is set to TRUE, the adapter connects to the base point without the host name. If this key is set to **TRUE**, the key affects RASServer and Terminal server lookup behavior. The default value is **FALSE**.
useSSL	This key enables SSL communication between the adapter and the Active Directory. If this key is set to **TRUE**, the adapter uses SSL to communicate with the Active Directory. If this key is set to **FALSE** or does not exist, the adapter does not use SSL. The default value is **FALSE**.
WtsDisableSearch	This key takes effect only if WtsEnabled is set to **TRUE**. If set to **FALSE**, this key enables a reconciliation of the WTS attributes. If set to **TRUE**, the reconciliation is faster. The default value is **FALSE**.
WtsEnabled	If this key is set to **TRUE**, the key enables processing of Windows Terminal Server (WTS) attributes. The default value is **FALSE**.

Table 9. Registry key descriptions (continued)

Key	Description
UseGroup	You can set this key to one of the following options: • **CN**: When you set this key to CN, the adapter performance for add, modify, and reconciliation is lesser compared to the DN option. This lessening of performance is because adapter must perform extra binds to the Active Directory. • **DN**: When you set this key to DN, the adapter performance for add, modify, and reconciliation is higher compared to the CN and GUID options. • **GUID**: When you set this key to GUID, the adapter performance for add, modify, and reconciliation lesser compared to DN, however, higher compared to CN. Depending on the key the adapter retrieves the value for group during the reconciliation operation and processes during the add and modify operation of the adapter. When you change the value of this key, you must modify the profile and import it again on IBM Security Verify Governance Identity Manager. The default value is **DN**.
ReconMailboxPermissions	When this key is set to **FALSE**, the adapter does not retrieve the Mailbox Permission information. The reconciliation operation is fast when this key is set to **FALSE**. The default value is **TRUE**.
UPNSearchEnabled	When the registry key **UPNSearchEnabled** is set to **FALSE**, the adapter does not perform a search on the **User Principal Name** for uniqueness. It creates the user account with the supplied or generated value of the **User Principal Name**. When the registry key **UPNSearchEnabled** is set to **TRUE**, the adapter performs a search on the User Principal Name to ensure the uniqueness. The default value is **TRUE**. **Note:** This key is used only for the user add operation.
UseITIMCNAttribute	When this key is set to TRUE, the adapter uses IBM Security Verify Governance Identity Manager common schema attribute cn. The adapter processes the cn attribute for add, modify, and reconciliation operations. When this key is set to FALSE, the adapter uses the erADFullName attribute for add, modify, and reconciliation operations. When you set this registry key to FALSE, you must customize the account form. For more information, see “Configuring the cn attribute” on page 68. The default value is TRUE.

Table 9. Registry key descriptions (continued)	
Key	**Description**
MailUserRenameDelay	When you rename a user account with mail status, the Active Directory might take time to reestablish the user account mail status. This behavior causes the adapter to fail the exchange attributes in the rename request with the error message *Error setting attribute name. User does not have a mailbox.* In this case, renaming means modifying the **Eruid** and the **User Principal Name** attribute. When you use this key, the adapter waits before it modifies the exchange attribute when a user account is renamed. For example, set this key is set to **10** seconds. Submit a user account rename request. The adapter waits for **10** seconds before modifying the exchange attributes that are in the request. The default value of the registry key is **0** seconds. **Note:** The adapter uses this key only when the **Eruid**, User Principal Name, and the exchange attributes are modified.
SearchTimeout	In some of the Active Directory setups, the adapter might not complete the reconciliation operation. This failure occurs when the Microsoft ADSI API GetNextRow halts indefinitely. The adapter monitors the reconciliation operation. Set this registry key to a non-zero value. The adapter process ends if there is no activity by the adapter in the reconciliation operation for the time in seconds specified in this key. When you set the value of this registry key to **0** and if the adapter halts during the reconciliation operation, the reconciliation operation does not complete and the operation is timed out on IBM Security Verify Governance Identity Manager. In this case, restart the adapter service. The default value of the registry key is **0** seconds.
LyncDisableSearch	If this key is set to **TRUE**, the key disables the Lync attributes. It excludes the Lync attributes, which are not stored as LDAP values and are retrieved with a powershell call, from search results. The Lync attributes can significantly affect the performance during a search. The default value is **FALSE**.

Note: The following registry keys are no longer used:

- AbortReconOnFailure
- OverrideX500Addresses

In addition to the listed adapter registry keys, you can add registry keys with a name as the value of **Users BasePoint DN** on the service form. You can also provide additional target servers for that service. Each target server must be separated by a |.

Example 1

When a **Users BasePoint DN** specified on service form is `OU=TestOU,DC=MyDomain,DC=com`, you can specify the list of target server(s) in the adapter registry by using agentCfg.exe as:

- Create the registry with name `OU=TestOU,DC=MyDomain,DC=com`.
- Specify the value for the key as `DC01|DC02|DC03`.

Example 2

When a Users BasePoint DN specified on service form is `DC01|DC02|DC03/DC=MyDomain,DC=com`, you can specify the list of additional target server(s) in the adapter registry by using agentCfg.exe as:

- Create the registry with name `DC=MyDomain,DC=com`.

- Specify the value for the key as DC04 | DC05 | DC06.

Note: When the base point or target server has Unicode characters, use the regedit to create registry keys under HKEY_LOCAL_MACHINE\ SOFTWARE\Access360\ADAgent\Specific. For more information, see "Users Base Point configuration for the adapter" on page 61.

Modifying encrypted registry settings

You can access registry settings.

Procedure

1. Type B (Modifying Encrypted Registry Settings) at the Registry menu prompt to display the Encrypted Registry settings menu.

```
Encrypted Registry Items
----------------------------------------
A. Add new attribute
B. Modify attribute value.
C. Remove attribute.
X. Done
Select menu option:
```

2. Type one of the following options:

```
A) Add new attribute
B) Modify attribute value
C) Remove attribute
X) Done
```

3. Type the registry item name, and press **Enter**.
4. Type the registry item value, if you selected option A or B, and press **Enter**.

   The encrypted registry settings menu reappears and displays your new settings.

Modifying advanced settings

You can change the adapter thread count settings.

About this task

You can change the thread count settings for the following types of requests:

- System Login Add
- System Login Change
- System Login Delete
- Reconciliation

These settings determine the maximum number of requests that the adapter processes concurrently. To change these settings, take the following steps:

Procedure

1. Access the Agent Main Configuration menu.
2. At the **Main Menu** prompt, type G to display the Advanced Settings menu.

   The following screen displays the default thread count settings.

```
adapter_name and version number Advanced settings menu
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

A. Single Thread Agent (current:FALSE)
B. ADD max. thread count. (current:3)
C. MODIFY max. thread count. (current:3)
D. DELETE max. thread count. (current:3)
E. SEARCH max. thread count. (current:3)
F. Allow User EXEC procedures (current:FALSE)
G. Archive Request Packets (current:FALSE)
H. UTF8 Conversion support (current:TRUE)
I. Pass search filter to agent (current:FALSE)
J. Thread Priority Level (1-10) (current:4)
X. Done
Select menu option:
```

Table 10. Options for advanced settings menu

Option	Description
A	Forces the adapter to allow only 1 request at a time. The default value is FALSE.
B	Limits the number of ADD requests that can run simultaneously. The default value is 3.
C	Limits the number of MODIFY requests that can run simultaneously. The default value is 3.
D	Limits the number of DELETE requests that can run simultaneously. The default value is 3.
E	Limits the number of SEARCH requests that can run simultaneously. The default value is 3.
F	Determines whether the adapter can do the pre-exec and post-exec functions. The default value is FALSE. **Note:** Enabling this option is a potential security risk.
G	This option is no longer supported.
H	This option is no longer supported.
I	Active Directory Adapter supports processing filters directly. If you enable this option by setting it to TRUE, the adapter filters the results instead of the ADK. By default, this option is set to FALSE and the ADK does the filtering.
J	Sets the thread priority level for the adapter. The default value is 4.

3. Type the letter of the menu option that you want to change.
4. Change the value and press Enter to display the Advanced Settings menu with new settings.

Related tasks

“Starting the adapter configuration tool” on page 27

Start the **agentCfg** tool to access the configuration menu, where you can modify the different adapter parameters.

Viewing statistics

You can view an event log for the adapter.

Procedure

1. Access the **Agent Main Configuration Menu**.
2. At the **Main Menu** prompt, type H to display the activity history for the adapter.

```
  Agent Request Statistics
------------------------------------------------------------------
Date       Add       Mod       Del       Ssp       Res       Rec


------------------------------------------------------------------

02/15/06  000001    000000    000000     000000    000000    000001

------------------------------------------------------------------

X. Done
```

3. Type X to return to the **Main Configuration Menu**.

Related tasks

“Starting the adapter configuration tool” on page 27
Start the **agentCfg** tool to access the configuration menu, where you can modify the different adapter parameters.

Modifying code page settings

You can change the code page settings for the adapter.

About this task

To list the supported code page information for the adapter, the adapter must be running. Run the following command to view the code page information:

```
agentCfg -agent [adapter_name] -codepages
```

Procedure

1. Access the Agent Main Configuration menu.
2. At the **Main Menu** prompt, type I to display the Code Page Support menu.

```
adapter_name and version number Codepage Support Menu
-----------------------------------------
* Configured codepage: US-ASCII
-----------------------------------------
*
*****************************************
* Restart Agent After Configuring Codepages
*****************************************
A. Codepage Configure.
X. Done
Select menu option:
```

3. Type A to configure a code page.

   Note: The code page uses Unicode, therefore this option is not applicable.
4. Type X to return to the Main Configuration menu.

Related tasks

"Starting the adapter configuration tool" on page 27
Start the **agentCfg** tool to access the configuration menu, where you can modify the different adapter parameters.

Configuring SSL authentication

You can provide SSL authentication, certificates, and enable SSL authentication with the certTool utility.

For secure connection between the adapter and the server, configure the adapter and the server to use the Secure Sockets Layer (SSL) authentication with the DAML default communication protocol. Typically, SSL is used to establish a secure connection that encrypts the data that is being exchanged. While it can assist in authentication, you must enable registered certificates in DAML to use SSL for authentication. By configuring the adapter for SSL, the server can verify the identity of the adapter before the server makes a secure connection.

You can configure SSL authentication for connections that originate from the Identity server or from the adapter. The Identity server initiates a connection to the adapter to set or retrieve the value of a managed attribute on the adapter. Depending on the security requirements of your environment, you might configure SSL authentication for connections that originate from the adapter. For example, adapter events can notify the Identity server of changes to attributes on the adapter. In this case, configure SSL authentication for web connections that originate from the adapter to the web server used by the Identity server.

In a production environment, you must enable SSL security. If an external application communicates with the adapter (for example, the Identity server) and uses server authentication, enable SSL on the adapter. Enabling SSL verifies the certificate that the application presents.

Related concepts

"Overview of SSL and digital certificates" on page 3
In an enterprise network deployment, you must provide secure communication between the Identity server and the software products and components with which the server communicates.

Configuring certificates for SSL authentication

You can configure the adapter for one-way or two-way SSL authentication with signed certificates.

About this task

Use the certTool utility for these tasks:

- "Configuring certificates for one-way SSL authentication" on page 44
- "Configuring certificates for two-way SSL authentication" on page 45
- "Configuring certificates when the adapter operates as an SSL client" on page 46

Configuring certificates for one-way SSL authentication

In this configuration, the Identity server and the adapter use SSL.

About this task

Client authentication is not set on either application. The Identity server operates as the SSL client and initiates the connection. The adapter operates as the SSL server and responds by sending its signed certificate to the Identity server. The Identity server uses the installed CA certificate to validate the certificate that is sent by the adapter.

In Figure 1 on page 45, Application A operates as the Identity server, and Application B operates as the adapter.

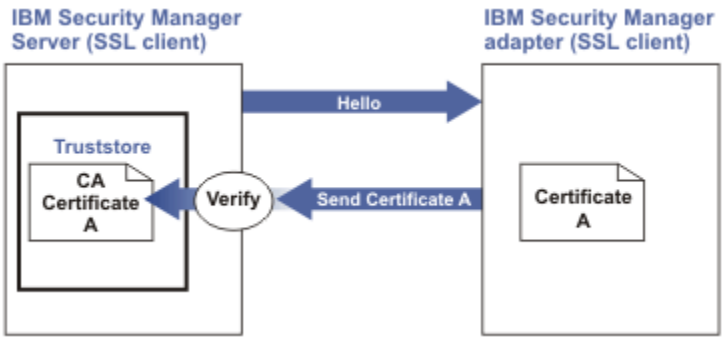


Figure 1. One-way SSL authentication (server authentication)

To configure one-way SSL, do the following tasks for each application:

Procedure

1. On the adapter, complete these steps:
   a. Start the certTool utility.
   b. To configure the SSL-server application with a signed certificate issued by a certificate authority:
      i) Create a certificate signing request (CSR) and private key. This step creates the certificate with an embedded public key and a separate private key and places the private key in the PENDING_KEY registry value.
      ii) Submit the CSR to the certificate authority by using the instructions that are supplied by the CA. When you submit the CSR, specify that you want the root CA certificate to be returned with the server certificate.
2. On the Identity server, do one of these steps:
   - If you used a signed certificate that is issued by a well-known CA:
     a. Ensure that the Identity server stored the root certificate of the CA (CA certificate) in its truststore.
     b. If the truststore does not contain the CA certificate, extract the CA certificate from the adapter and add it to the truststore of the server.
   - If you generated the self-signed certificate on the Identity server, the certificate is installed and requires no additional steps.
   - If you generated the self-signed certificate with the key management utility of another application:
     a. Extract the certificate from the keystore of that application.
     b. Add it to the truststore of the Identity server.

Related tasks

"Starting certTool" on page 47
To start the certificate configuration tool named certTool for the adapter, complete these steps:

Configuring certificates for two-way SSL authentication

In this configuration, the Identity server and adapter use SSL.

About this task

The adapter uses client authentication. After the adapter sends its certificate to the server, the adapter requests identity verification from the Identity server. The server sends its signed certificate to the adapter. Both applications are configured with signed certificates and corresponding CA certificates.

In the following figure, the Identity server operates as Application A and the adapter operates as Application B.

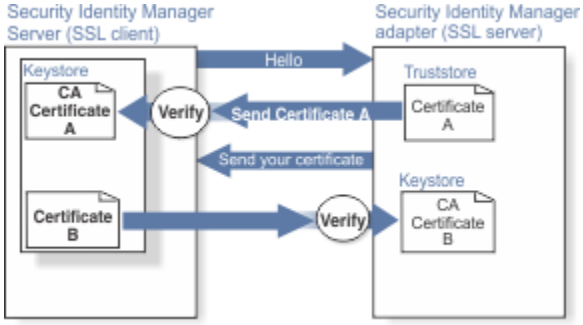


Figure 2. Two-way SSL authentication (client authentication)

Before you do the following procedure, configure the adapter and Identity server for one-way SSL authentication. If you use signed certificates from a CA:

- The CA provides a configured adapter with a private key and a signed certificate.
- The signed certificate of the adapter provides the CA certification for the Identity server.

To complete the certificate configuration for two-way SSL, do the following tasks:

Procedure

1. On the Identity server, create a CSR and private key. Next, obtain a certificate from a CA, install the CA certificate, install the newly signed certificate, and extract the CA certificate to a temporary file.
2. On the adapter, add the CA certificate that was extracted from the keystore of the Identity server to the adapter.

Results

After you configure the two-way certificate, each application has its own certificate and private key. Each application also has the certificate of the CA that issued the certificates.

Related tasks

"Configuring certificates for one-way SSL authentication" on page 44
In this configuration, the Identity server and the adapter use SSL.

Configuring certificates when the adapter operates as an SSL client

In this configuration, the adapter operates as both an SSL client and as an SSL server.

About this task

This configuration applies if the adapter initiates a connection to the web server (used by the Identity server) to send an event notification. For example, the adapter initiates the connection and the web server responds by presenting its certificate to the adapter.

Figure 3 on page 47 describes how the adapter operates as an SSL sever and an SSL client. When communicating with the Identity server, the adapter sends its certificate for authentication. When communicating with the web server, the adapter receives the certificate of the web server.

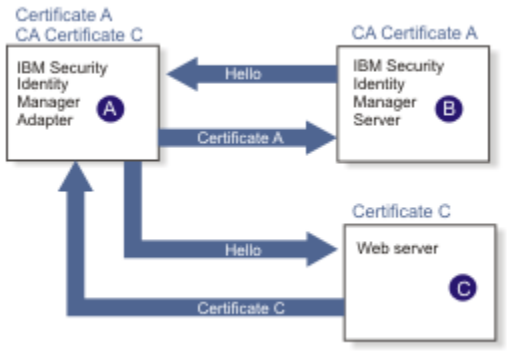


Figure 3. Adapter operating as an SSL server and an SSL client

If the web server is configured for two-way SSL authentication, it verifies the identity of the adapter. The adapter sends its signed certificate to the web server (not shown in the illustration). To enable two-way SSL authentication between the adapter and web server, perform the following process:

Procedure

1. Configure the web server to use client authentication.
2. Follow the procedure for creating and installing a signed certificate on the web server.
3. Install the CA certificate on the adapter with the certTool utility.
4. Add the CA certificate corresponding to the signed certificate of the adapter to the web server.

What to do next

If you want the software to send an event notification when the adapter initiates a connection to the web server (used by the Identity server), see the IBM Security Verify Governance Identity Manager product documentation.

SSL certificate management with certTool

Use the certTool utility to manage private keys and certificates.

Starting certTool

To start the certificate configuration tool named certTool for the adapter, complete these steps:

Procedure

1. Click **Start** > **Programs** > **Accessories** > **Command Prompt**.
2. At a DOS command prompt, change to the `bin` directory for the adapter.
   If the directory is in the default location, type the following command:

   ```
   cd C:\Program Files\IBM\ISIM\Agents\adapter_name\bin\
   ```

3. Type `CertTool -agent` *`agent_name`* at the prompt.

   For example, to display the main menu, type: `CertTool -agent NotesAgent`

```
Main menu - Configuring agent: agentnameAgent
 ------------------------------
A. Generate private key and certificate request
B. Install certificate from file
C. Install certificate and key from PKCS12 file
D. View current installed certificate

E. List CA certificates
F. Install a CA certificate
G. Delete a CA certificate

H. List registered certificates
I. Register certificate
J. Unregister a certificate

K. Export certificate and key to PKCS12 file

X. Quit

Choice:
```

Results

From the **Main** menu, you can generate a private key and certificate request, install and delete certificates, register and unregister certificates, and list certificates. The following sections summarize the purpose of each group of options.

By using the first set of options (A through D), you can generate a CSR and install the returned signed certificate on the adapter.

A. Generate private key and certificate request
Generate a CSR and the associated private key that is sent to the certificate authority.

B. Install certificate from file
Install a certificate from a file. This file must be the signed certificate that is returned by the CA in response to the CSR that is generated by option A.

C. Install certificate and key from a PKCS12 file
Install a certificate from a PKCS12 format file that includes both the public certificate and a private key. If options A and B are not used to obtain a certificate, the certificate that you use must be in PKCS12 format.

D. View current installed certificate
View the certificate that is installed on the workstation where the adapter is installed.

With the second set of options, you can install root CA certificates on the adapter. A CA certificate validates the corresponding certificate that is presented by a client, such as the Identity server.

E. List CA certificates
Show the installed CA certificates. The adapter communicates only with Identity server whose certificates are validated by one of the installed CA certificates.

F. Install a CA certificate
Install a new CA certificate so that certificates generated by this CA can be validated. The CA certificate file can either be in X.509 or PEM encoded formats.

G. Delete a CA certificate
Remove one of the installed CA certificates.

Options H through K apply to adapters that must authenticate the application to which the adapter is sending information. An example of an application is the Identity server or the web server. Use these options to register certificates on the adapter.

If you configure the adapter for event notification or enable client authentication in DAML, you must install the CA certificate. The CA certificate must correspond to the signed certificate of the Identity server. Use option F, **Install a CA certificate**.

H. List registered certificates
List all registered certificates that are accepted for communication.

I. Register a certificate
Register a new certificate. The certificate for registration must be in Base 64 encoded X.509 format or PEM.

J. Unregister a certificate
Unregister (remove) a certificate from the registered list.

K. Export certificate and key to PKCS12 file
Export a previously installed certificate and private key. You are prompted for the file name and a password for encryption.

Related concepts

"View of the installed certificate" on page 51
To list the certificate on your workstation, type D at the Main menu of certTool.

Related tasks

"Generating a private key and certificate request" on page 49
A certificate signing request (CSR) is an unsigned certificate that is a text file.

"Installing the certificate" on page 50
After you receive your certificate from your trusted CA, install it in the registry of the adapter.

"Installing the certificate and key from a PKCS12 file" on page 51
If the certTool utility did not generate a CSR to obtain a certificate, you must install both the certificate and private key.

"Installing a CA certificate" on page 51
If you use client authentication, you must install a CA certificate that is provided by a certificate authority vendor. You can install a CA certificate that was extracted in a temporary file.

"Deleting a CA certificate" on page 52
You can delete a CA certificate from the adapter directories.

"Viewing registered certificates" on page 52
The adapter accepts only the requests that present a registered certificate when client validation is enabled.

"Registering a certificate" on page 53
You can register a certificate for the adapter.

"Unregistering a certificate" on page 53
You can unregister a certificate for the adapter.

"Exporting a certificate and key to a PKCS12 file" on page 53
You can export a certificate and key to a PKCS12 file.

Generating a private key and certificate request

A certificate signing request (CSR) is an unsigned certificate that is a text file.

About this task

When you submit an unsigned certificate to a certificate authority, the CA signs the certificate with the private digital signature. The signature is included in their corresponding CA certificate. When the CSR is signed, it becomes a valid certificate. A CSR contains information about your organization, such as the organization name, country, and the public key for your web server.

Procedure

1. At the **Main Menu** of the certTool, type A. The following message and prompt are displayed:

```
Enter values for certificate request (press enter to skip value)
---------------------------------------------------------------------
```

2. At **Organization**, type your organization name and press **Enter**.

3. At **Organizational Unit**, type the organizational unit and press **Enter**.
4. At **Agent Name**, type the name of the adapter for which you are requesting a certificate and press **Enter**.
5. At **email**, type the email address of the contact person for this request and press **Enter**.
6. At **State**, type the state that the adapter is in and press **Enter**.
   For example, type TX if the adapter is in Texas. Some certificate authorities do not accept two letter abbreviations for states; type the full name of the state.
7. At **Country**, type the country that the adapter is in and press **Enter**.
8. At **Locality**, type the name of the city that the adapter is in and press **Enter**.
9. At **Accept these values**, take one of the following actions and press **Enter**:
   - Type Y to accept the displayed values.
   - Type N and specify different values.

   The private key and certificate request are generated after the values are accepted.
10. At **Enter name of file to store PEM cert request**, type the name of the file and press **Enter**. Specify the file that you want to use to store the values you specified in the previous steps.
11. Press **Enter** to continue. The certificate request and input values are written to the file that you specified. The file is copied to the adapter `bin` directory and the **Main** menu is displayed again.

Results

You can now request a certificate from a trusted CA by sending the `.pem` file that you generated to a certificate authority vendor.

Example of certificate signing request

Here is an example certificate signing request (CSR) file.

```
-----BEGIN CERTIFICATE REQUEST-----
MIIB1jCCAT8CAQAwgZUxEjAQBgNVBAoTCWFjY2VzczM2MDEUMBIGA1UECxMLZW5n
aW5lZXJpbmcxEDAOBgNVBAMTB250YWdlbnQxJDAiBgkqhkiG9w0BCQEWFW50YWdl
bnRAYWNjZXNzMzYwLmNvbTELMAkGA1UEBhMCVVMxEzARBgNVBAgTCkNhbGlmb3Ju
aWExDzANBgNVBAcTBklydmluZTCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEA
mR6AcPnwf6hLLc72BmUkAwaXcebtxCoCnnTH9uc8VuMHPbIMAgjuC4s91hPrilG7
UtlbOfy6X3R3kbeR8apRR9uLYrPIvQ1b4NK0whsytij6syCySaFQIB6V7RPBatFr
6XQ9hpsARdkGytZmGTgGTJ1hSS/jA6mbxpgmttz9HPECAwEAAaAAMA0GCSqGSIb3
DQEBAgUAA4GBADxA1cDkvXhgZntHkwT9tCTqUNV9sim8N/U15HgMRh177jVaHJqb
N1Er46vQSsOOOk4z2i/XwOmFkNNTXRVl9TLZZ/D+9mGZcDobcO+lbAKlePwyufxK
Xqdpu3d433H7xfJJSNYLYBFkrQJesITqKft0Q45gIjywIrbctVUCepL2
 -----END CERTIFICATE REQUEST-----
```

Installing the certificate

After you receive your certificate from your trusted CA, install it in the registry of the adapter.

Procedure

1. If you received the certificate as part of an email message, do the following actions.
   a. Copy the text of the certificate to a text file.
   b. Copy that file to the `bin` directory of the adapter.

   For Windows operating systems:

   ```
   C:\Program Files\IBM\ISIM\Agents\adapter_nameAgent\bin
   ```
2. At the **Main Menu** prompt of the certTool, type B. The following prompt is displayed:

   ```
   Enter name of certificate file:
   ------------------------------------------------------------------------
   ```
3. At **Enter name of certificate file**, type the full path to the certificate file and press **Enter**.

The certificate is installed in the registry for the adapter, and **Main Menu** is displayed again.

Installing the certificate and key from a PKCS12 file

If the certTool utility did not generate a CSR to obtain a certificate, you must install both the certificate and private key.

About this task

Store the certificate and private key in a PKCS12 file. The CA sends a PKCS12 file that has a `.pfx` extension. The file might be a password-protected file and it includes both the certificate and private key.

Procedure

1. Copy the PKCS12 file to the `bin` directory of the adapter.

   For Windows operating systems:

   ```
   C:\Program Files\IBM\ISIM\Agents\adapter_nameAgent\bin
   ```

2. At the **Main Menu** prompt for the certTool, type `C` to display the following prompt:

   ```
   Enter name of PKCS12 file:
   --------------------------------------------------------------------------
   ```

3. At **Enter name of PKCS12 file**, type the name of the PKCS12 file that has the certificate and private key information and press **Enter**. For example, `DamlSrvr.pfx`.
4. At **Enter password**, type the password to access the file and press **Enter**.

Results

After you install the certificate and private key in the adapter registry, the certTool displays **Main Menu**.

View of the installed certificate

To list the certificate on your workstation, type `D` at the Main menu of certTool.

The utility displays the installed certificate and the Main menu. The following example shows an installed certificate:

```
The following certificate is currently installed.
Subject: c=US,st=California,l=Irvine,o=DAML,cn=DAML Server
```

Installing a CA certificate

If you use client authentication, you must install a CA certificate that is provided by a certificate authority vendor. You can install a CA certificate that was extracted in a temporary file.

Procedure

1. At the **Main Menu** prompt, type F (Install a CA certificate).

   The following prompt is displayed:

   ```
   Enter name of certificate file:
   ```

2. At **Enter name of certificate file**, type the name of the certificate file, such as `DamlCACerts.pem` and press **Enter**.

   The certificate file opens and the following prompt is displayed:

   ```
   e=admin@ibm.com,c=US,st=California,l=Irvine,o=IBM,ou=Engineering,cn=Eng
   Install the CA? (Y/N)
   ```

3. At **Install the CA**, type Y to install the certificate and press **Enter**.

   The certificate file is installed in the `CACerts.pem` file.

Viewing CA certificates

Use the certTool utility to view a private key and certificate that are installed the adapter.

About this task

The certTool utility installs only one certificate and one private key.

Procedure

Type E at the **Main Menu** prompt.

Results

The certTool utility displays the installed CA certificates and the **Main** menu. The following example shows an installed CA certificate:

```
Subject: o=IBM,ou=SampleCACert,cn=TestCA
Valid To: Wed Jul 26 23:59:59 2006
```

Deleting a CA certificate

You can delete a CA certificate from the adapter directories.

Procedure

1. At the **Main Menu** prompt, type G to display a list of all CA certificates that are installed on the adapter.

   ```
   0 - e=admin@ibm.com,c=US,st=California,l=Irvine,o=IBM,ou=Engineering,cn=Eng
   1 - e=support@ibm.com,c=US,st=California,l=Irvine,o=IBM,ou=Support,cn=Support
   Enter number of CA certificate to remove:
   ```

2. At **Enter number of CA certificate to remove**, type the number of the CA certificate that you want to remove and press **Enter**.

Results

After the CA certificate is deleted from the `CACerts.pem` file, the certTool displays the Main menu.

Viewing registered certificates

The adapter accepts only the requests that present a registered certificate when client validation is enabled.

Procedure

To view a list of all registered certificates, type H on the **Main Menu** prompt.

The utility displays the registered certificates and the **Main** menu. The following example shows a list of the registered certificates:

```
0 - e=admin@ibm.com,c=US,st=California,l=Irvine,o=IBM,ou=Engineering,cn=Eng
1 - e=support@ibm.com,c=US,st=California,l=Irvine,o=IBM,ou=Support,cn=Support
```

Registering a certificate

You can register a certificate for the adapter.

Procedure

1. At the **Main Menu** prompt, type `I` to display the following prompt:

```
Enter name of certificate file:
```

2. At **Enter name of certificate file**, type the name of the certificate file that you want to register and press **Enter**.

   The subject of the certificate is displayed, and a prompt is displayed, for example:

```
e=admin@ibm.com,c=US,st=California,l=Irvine,o=IBM,ou=Engineering,cn=Eng
Register this CA? (Y/N)
```

3. At **Register this CA**, type Y to register the certificate, and press **Enter**.

Results

After you register the certificate to the adapter, the certTool displays the **Main** menu.

Unregistering a certificate

You can unregister a certificate for the adapter.

Procedure

1. At the **Main Menu**prompt, type `J` to display the registered certificates. The following example shows a list of lists registered certificates:

```
0 - e=admin@ibm.com,c=US,st=California,l=Irvine,o=IBM,ou=Engineering,cn=Eng
1 - e=support@ibm.com,c=US,st=California,l=Irvine,o=IBM,ou=Support,cn=Support
```

2. Type the number of the certificate file that you want to unregister and press **Enter**.
   For example:

```
e=admin@ibm.com,c=US,st=California,l=Irvine,o=IBM,ou=Engineering,cn=Eng
Unregister this CA? (Y/N)
```

3. At **Unregister this CA**, type Y to unregister the certificate and press **Enter**.

Results

After you remove the certificate from the list of registered certificate for the adapter, the certTool displays the **Main Menu**.

Exporting a certificate and key to a PKCS12 file

You can export a certificate and key to a PKCS12 file.

Procedure

1. At the **Main Menu** prompt, type K to display the following prompt:

```
Enter name of PKCS12 file:
```

2. At the **Enter name of PKCS12 file** prompt, type the name of the PKCS12 file for the installed certificate or private key and press **Enter**.
3. At the **Enter Password** prompt, type the password for the PKCS12 file and press **Enter**.

4. At the **Confirm Password** prompt, type the password again and press **Enter**.

Results

After the certificate or private key is exported to the PKCS12 file, the certTool displays the Main menu.

Running the adapter in SSL mode

You can run the adapter in Secure Socket Layer (SSL) mode.

About this task

Note: If you do not do these steps, the certificate is not installed completely and the SSL is not enabled. See http://en.wikipedia.org/wiki/User_Account_Control.

Procedure

1. Disable the User Account Control (UAC) security.
2. Install the required certificate.
3. (Optional) If required, enable the UAC security.

Related concepts

"SSL certificate management with certTool" on page 47
Use the certTool utility to manage private keys and certificates.

Customizing the Active Directory Adapter

Active Directory can support custom attributes for the user class. The Active Directory Adapter only supports standard Windows attributes by default. However, you can customize the adapter to support custom (extended) attributes.

Before you begin

Before customizing an adapter, you must have working knowledge of these concepts:

- LDAP schema management
- LDAP object classes and attributes
- Scripting language that is appropriate for the installation system
- XML document structure

Note: The Active Directory Adapter supports customization only with the use of pre-Exec and post-Exec scripting. IBM does not support the your individual customization, scripts, or other modifications. If you experience a customization problem, IBM Support can require problem demonstration on the released version of the adapter before opening a problem report. For more information, see the IBM Security Verify Identity Support website.

About this task

Complete the steps to customize the Active Directory Adapter to support the extended attributes in the Active Directory.

Procedure

1. Extend the Active Directory Adapter schema and add the custom attributes to the Active Directory Server.

   For more information on extending the schema, see "Extend the schema and add the extended attributes" on page 55.

For information on the files that you can modify to customize the Active Directory Adapter, see "Files" on page 102.

2. Copy the JAR file to a temporary directory and extract the files. For more information on extracting the files, see "Copy the ADprofile.jar file and extract the files" on page 56.
3. Add the extended attributes to the `exschema.txt` or `exschemagrp.txt` file. For more information on extending the attributes, see "Modify the schema file" on page 56.
4. Update the `schema.dsml` file on the Identity server. For more information on updating this file, see "Update the schema file" on page 57.
5. Update the `customlabels.properties` file on the Identity server. For more information on updating this file, see "Modify the CustomLabels.properties file" on page 59.
6. Install the new attributes on the Identity server. For more information on updating this file, see "Create a JAR file and install the new attributes" on page 60.
7. Modify the form for the account. For more information on updating the form, see "Optionally modify the adapter form" on page 60.

Prepare to customize an adapter

An adapter customization is a twofold task. First step is to complete tasks such as extending schema, extracting adapter JAR file, and add attributes to the schema file so that an adapter knows about it. Then, modify the adapter profile.

Extend the schema and add the extended attributes

Extend the Windows Active Directory schema and add the custom attributes to the Active Directory Server by using the tools provided by Windows.

Note: The adapter does not support every attribute of the Active Directory user object. If you want to extend the adapter to support an attribute that is not currently supported by the adapter, but is already an Active Directory user attribute, you do not need to extend the Active Directory schema.

For more information about adding new attributes to the Active Directory, refer to the Microsoft Windows Server documentation.

The Active Directory Adapter supports attributes with these syntaxes:

- DN
- CaseExactString
- CaseIgnoreString
- PrintableString
- IA5String
- Integer8
- NumericString
- Boolean Integer
- UTCTime
- GeneralizedTime
- DirectoryString
- DnWithBinary
- OctetString

Consider prefixing the attribute names with *erAD* in order to easily identify the attributes that are used with IBM Security Verify Governance Identity Manager.

Note:

- If IBM Security Directory Server is being used as the directory server application, the name of the attribute must be unique within the first 16 characters.
- The Active Directory Adapter supports a multiline value for extended attributes with string syntax.
- The extended attributes are supported only for the User account class.

Copy the ADprofile.jar file and extract the files

The profile JAR file, ADprofile.jar, is included in the Active Directory Adapter compressed file that you downloaded from the IBM website.

About this task

The ADprofile.jar file contains the following files:

- CustomLabels.properties
- erADAccount.xml
- erADDAMLService.xml
- erADGroup.xml
- resource.def
- schema.dsml
- targetprofile.json

You can modify these files to customize your environment.

To modify the ADprofile.jar file, complete the following steps:

Procedure

1. Log in to the system where the Active Directory Adapter is installed.
2. On the **Start** menu, click **Programs** > **Accessories** > **Command Prompt**.
3. Copy the ADprofile.jar file into a temporary directory.
4. Extract the contents of` the ADprofile.jar file into the temporary directory by running the following command:

   ```
   cd c:\temp
   jar -xvf ADprofile.jar
   ```

   The **jar** command will create the c:\temp\ADprofile directory.
5. Edit the appropriate file.
6. When you finish updating the profile JAR file, import the profile on the IBM Security Verify Governance Identity Manager.

Modify the schema file

The exschema.txt file lists all extended user attributes in the Active Directory Server while the exschemagrp.txt file lists extended group attributes

About this task

Modify this file to allow the Active Directory Adapter to recognize an extended attribute in the Windows Active Directory Server.

To extend the schema for either user or group objects, complete the following steps.

Procedure

1. Change to the \data directory for the adapter.

2. Choose either one or both the choices depending upon your requirement.
   - For user objects: Create or open the `exschema.txt` file in a text editor.
   - For group objects: Create or open the `exschemagrp.txt` file in a text editor.
3. Add the extended attributes to the file.

   Note:

   - List the attribute name as it appears in the Active Directory.
   - If the name used in `schema.dsml` for the adapter profile is different, add the name from the schema separated by a pipe (|).
   - List only 1 attribute per line. For example:

   ```
   String1|erADCustomAtt
   Integer|erADInteger
   Date|erADDate
   Boolean|erADBoolean
   MultiValueString|erADMultiValueString
   ```
4. Save the changes, and close the file.
5. Start the adapter again.

   Start the adapter by using the Windows Services Console.

Modify an adapter profile

After you complete prerequisites to customize an adapter, start modifying adapter profile.

Update the schema file

The Active Directory Adapter `schema.dsml` file identifies all of the standard Windows account attributes.

About this task

Modify this file to identify the new extended attributes in the Active Directory Server. For more information about the attributes in this file, see “schema.dsml file” on page 102.

To update the `schema.dsml` file, complete the following steps:

Procedure

1. Change to the `\ADprofile` directory, where the `schema.dsml` file has been created.
2. Edit the `schema.dsml` file to add an attribute definition for each extended attribute.

   The Object Identifier (OID) must be incremented by 1, based on the last entry in the file. For example, if the last attribute in the file uses the OID 1.3.6.1.4.1.6054.3.125.2.**67**, the first new attribute uses the OID 1.3.6.1.4.1.6054.3.125.2.**68**.

   Consider starting a new range of numbers for your custom attributes. For example, start custom attributes with OID 1.3.6.1.4.1.6054.3.125.2.100. This prevents duplicate OIDs if the adapter is upgraded to support new attributes that are standard for newer versions of Windows.
3. Add the new attributes to the account or group class as appropriate.

   For example, add the following attribute definition under the erADAccount section of the `schema.dsml` file:

   ```
   <attribute ref="erADDate" required="false"/>
   ```

Update the `targetprofile.json` file

The Active Directory `targetprofile.json` file identifies all of the supported Windows account attributes for the IBM Security Verify Governance Identity Manager server.

About this task

Modify the file to identify the new extended attributes.

Procedure

1. Change to the `\ADprofile` directory, where the `targetProfile.json` file has been created.
2. Open the `targetProfile.json` file in a text editor.
3. Find the section for `userExtension`.
   For example:

```
"userExtension": {
"schema": "urn:ibm:idbrokerage:params:scim:schemas:extension:ADAccount:2.0:User",
"definition": {
"id": "urn:ibm:idbrokerage:params:scim:schemas:extension:ADAccount:2.0:User",
"name": "CustomUserExtension",
"description": "Security adapter view of a user",
"attributes": [
```

   The `attributes` section contains an array of attribute definitions. Each definition is separated by a comma.
4. Add your extended attributes to this `attributes` section. An attribute object contains the following fields:

Field	Description
name	Attributes name.
type	data type (string integer, boolean, binary)
multiValued	True, if attribute can have multiple values.
required	true, if required attribute.
caseExact	true, if value is case-sensitive.
mutability	immutable, read, write, readwrite
returned	Use "default".
uniqueness	User "server".
specialFlags	User "none".
canonicalValues	Optional list of valid values for this attribute as a json array.

   The attribute object is enclosed in braces ({}). Each field has the name in quotes followed by a colon and the value. Each field is separated by a comma.

   See the following example from the Active Directory adapter:

```
{
"name": "eruid",
"type": "string",
"multiValued": false,
"description": "An identifier used to uniquely identify a user",
"required": true,
"caseExact": false,
"mutability": "immutable",
"returned": "default",
"uniqueness": "server",
```

```
"specialFlags": "none"
},
```

5. Add the new attributes to the account class.
   For example:

```
"userExtension": {
"schema": "urn:ibm:idbrokerage:params:scim:schemas:extension:ADAccount:2.0:User",
"definition": {
"id": "urn:ibm:idbrokerage:params:scim:schemas:extension:ADAccount:2.0:User",
"name": "CustomUserExtension",
"description": "Security adapter view of a user",
"attributes": [
{
"name": "eruid",
"type": "string",
"multiValued": false,
"description": "An identifier used to uniquely identify a user",
"required": true,
"caseExact": false,
"mutability": "immutable",
"returned": "default",
"uniqueness": "server",
"specialFlags": "none"
},
…
{
"name": "title",
"type": "string",
"multiValued": false,
"description": "title",
"required": false,
"caseExact": false,
"mutability": "readWrite",
"returned": "default",
"uniqueness": "none",
"specialFlags": "none"
},
{
"name": "shirtSize",
"type": "string",
"multiValued": true,
"description": "Shirt Size",
"required": false,
"caseExact": false,
"mutability": "readWrite",
"returned": "default",
"uniqueness": "none",
"specialFlags": "none",
"canonicalValues": [
"small”,
"medium",
"large
]
}
]
```

   Note: Ensure that you separate each attribute definition with a comma. After you update the file, it is suggested that you verify that the syntax is correct by using one of the freely available json lint sites.

Modify the `CustomLabels.properties` file

After you add the extended attributes to the `schema.dsml` file, the attributes are available for use on the Active Directory Adapter form.

About this task

The attributes appear in the attribute list by their directory server name. You can modify the attribute names that appear in the attribute list. For more information about the attributes that appear on the adapter form, see “CustomLabels.properties file” on page 105.

To add an attribute and its corresponding label to the `CustomLabels.properties` file, complete the following steps:

Procedure

1. Change to the ADprofile directory where the CustomLabels.properties file has been created.
2. Edit the CustomLabels.properties file to add the attribute and its corresponding label using the following format:

```
attribute=label
```

Note: The attribute name must be in lower case.

For example:

```
#
# ADAgent Labels definitions
#
eradstring1=ADString1
eradinteger=ADInteger
eraddate=ADDate
eradboolean=ADBoolean
eradmultivaluestring=ADMultiValueString
```

Create a JAR file and install the new attributes

After you modify the schema.dsml and CustomLabels.properties files, import these files and any other files in the profile that were modified for the adapter, into the Identity server to cause the changes to take effect.

About this task

To install the new attributes, complete the following steps:

Procedure

1. Create a new JAR file using the files in the \temp directory by running the following commands:

```
cd c:\temp
jar -cvf ADprofile.jar ADprofile
```

2. Import the ADprofile.jar file into the Identity server.
3. Stop and start the Identity server.

Note: If you are upgrading an existing adapter profile, the new adapter profile schema is not immediately used. Stop and start the Identity server to refresh the cache and the adapter schema. For more information on upgrading an existing adapter, see “Upgrading the Active Directory Adapter” on page 21.

Optionally modify the adapter form

After the changes are available in the Identity server, you can modify the Active Directory Adapter forms to use the new extended attributes.

The attributes do not need to be added to the Active Directory Adapter form unless you want them to be available. The attributes will be returned during reconciliations unless you explicitly exclude them.

For more information on how to modify the adapter form, see the IBM Security Verify Governance Identity Manager product documentation.

Managing passwords when you restore accounts

When a person's accounts are restored from being previously suspended, you are not prompted to supply a new password for the reinstated accounts. However, there are circumstances when you might want to circumvent this behavior.

About this task

The password requirement to restore an account on Active Directory Server falls into two categories: allowed and required. How each restore action interacts with its corresponding managed resource depends on either the managed resource, or the business processes that you implement. Certain resources will reject a password when a request is made to restore an account. In this case, you can configure IBM Security Verify Governance Identity Manager to forego the new password requirement. If your company has a business process in place that dictates that the account restoration process must be accompanied by resetting the password, you can set the Active Directory Adapter to require a new password when the account is restored.

In the `resource.def` file, you can define whether or not a password is required as a new protocol option. When you import the adapter profile, if an option is not specified, the adapter profile importer determines the correct restoration password behavior. Adapter profile components also enable remote services to find out if you discard a password that is entered by the user in a situation where multiple accounts on disparate resources are being restored. In this scenario, only some of the accounts being restored might require a password. Remote services will discard the password from the restore action for those managed resources that do not require them.

To configure the Active Directory Adapter to prompt for a new password when restoring accounts:

Procedure

1. Stop the Identity server.
2. Extract the files from the `ADprofile.jar` file.

   For more information on customizing the adapter profile file, see “Copy the ADprofile.jar file and extract the files” on page 56.
3. Change to the `\ADprofile` directory, where the `resource.def` file has been created.
4. Edit the `resource.def` file to add the new protocol options. For example:

   ```
   <Property Name  = "com.ibm.itim.remoteservices.ResourceProperties.
   PASSWORD_NOT_REQUIRED_ON_RESTORE"    Value = "FALSE"/>
   <Property Name  = "com.ibm.itim.remoteservices.ResourceProperties.
   PASSWORD_NOT_ALLOWED_ON_RESTORE" Value = "FALSE"/>
   ```

   Adding the two options in the example above ensures that you are prompted for a password when an account is restored.
5. Create a new `ADprofile.jar` file using the `resource.def` file and import the adapter profile file into the Identity server

   For more information, see “Create a JAR file and install the new attributes” on page 60.
6. Start the Identity server again.

   Note: If you upgrade an existing adapter profile, the new adapter profile schema is not immediately used. Stop and start the Identity server to refresh the cache and therefore the adapter schema. For more information on upgrading an existing adapter, see “Upgrading the Active Directory Adapter” on page 21.

Users Base Point configuration for the adapter

You can configure the Active Directory Adapter to support both sub-domains and multiple domains through the base point feature on the adapter service form.

For more information on configuring the service form, see the IBM Security Verify Governance Identity Manager product documentation.

The base point for the Active Directory Adapter is the point in the directory server that is used as the root for the adapter. This point can be an OU or DC point. Because the base point is an optional value, if a value is not specified, the adapter uses the default domain of the workstation on which it is installed.

The following definition is an example of a base point defined from the root of the directory server:

```
dc=irvine,dc=IBM,dc=com
```

The following definition is an example of a base point defined from an organizational unit level:

```
ou=engineering,dc=irvine,dc=IBM,dc=com
```

The syntax of the base point also allows for an optional workstation name to prefix the base point DN, for example `server1/dc=ibm,dc=com`. This causes the adapter to bind to a specific server instead of connecting to the first available server when responding to an active directory bind request.

You can specify more that one target server for the base point on the Active Directory Adapter service form on IBM Security Verify Governance Identity Manager and in the Active Directory Adapter registry. Each target server must be separated by | as a delimiter. For example,

Base Point DN on the service form with more than one target server:
DC01|DC02|DC03/OU=engineering,DC=irvine,DC=IBM,DC=com

Base Point DN on the service form with only one target server:
DC01/OU=engineering,DC=irvine,DC=IBM,DC=com

Base Point DN on the service form with no target server:
OU=engineering,DC=irvine,DC=IBM,DC=com

The adapter iterates through all the target servers specified in the base point on the service form. The adapter uses the first available target server.

Note:

- There is a limit of 240 characters for the Base Point DN attribute on the adapter service form.
- The adapter service form and registry can specify their own set of target servers. However, the target servers specified on the service form are considered a high priority.
- When you do not provide a base point on the service form, the adapter does not use the registry.
- Specify the target server by using the adapter registry because it is cached to improve the performance compared to specifying on the adapter service form. The target server list on the service form is not cached and is parsed in each request to find all target servers.
- Use the `agentCfg.exe` to create and modify adapter registry keys. Restart the adapter service after you add or modify the registry keys. When the base point or target server have Unicode characters, use regedit to create registry keys under `HKEY_LOCAL_MACHINE\SOFTWARE\Access360\ADAgent\Specific`.

Note: Do not create services that overlap in scope in the directory tree. This could result in duplicate account creation during reconciliation.

Configuring the source attribute of erGroup and erADGroupIsMemberOf

You can configure Group CN, Group DN, or Group GUID for the Source Attribute of erGroup and erADGroupIsMemberOf on the account form and group form to meet the requirements of your organization.

About this task

The source attribute identifies the name of the group class attribute for which the value is supplied in the request when groups are added or removed from the account form and group form. The default configuration is DN. The adapter uses the Group DN to bind to a group for adding or removing members. Also, the adapter adds the Group DN to the erGroup attribute of user object and erADGroupIsMemberOf attribute of group object during the reconciliation operation.

The CN value is not required to be unique.

The GUID value is not human readable.

It is possible to set up a test environment that uses the same DN values as the production environment. Any customization based on the DN works in both environments because the DNs are the same. However, if you use the same GUID in both environments, the values are different even if the DN values are the same.

To use the Group CN, Group DN, or Group GUID, perform the following steps.

Procedure

- Set the UseGroup registry key to one of the following options by using the **agentCfg**:
  - CN
  - DN
  - GUID
- Modify the profile files `erADAccount.xml`, `erADGroup.xml`, and `resource.def`.
  For information about profile file modifications, see Table 11 on page 64.
- Build the `ADprofile.jar` and import the new profile on IBM Security Verify Governance Identity Manager.
- Perform a full reconciliation operation.

  Note: If an event notification is enabled, delete the event notification database and perform a full reconciliation operation. When you do so, you are ensuring that a new database is created with correct values.
- Modify the profile files `erADAccount.xml`, `erADGroup.xml`, and `resource.def` as specified in the following table:

Table 11. Profile files

Value of the UseGroup registry key	Modifications required in	Expected modification
DN	erADAccount.xml	<formElement direction="inherit" label="$ergroup" name="data.ergroup"> <searchFilter multiple="true" type="select"> <filter>(objectclass=eradgroup)</filter> <base>contextual</base> <attribute>erADGroupSamAccountName</attribute> **<sourceAttribute>erADGroupDN</sourceAttribute>** <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADGroup</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>true</paginateResults> </searchFilter> </formElement>
	erADGroup.xml	<formElement direction="inherit" label="$eradgroupismemberof" name="data.eradgroupismemberof"> <searchFilter multiple="true" type="select"> <filter>(objectclass=eradgroup)</filter> <base>contextual</base> <attribute>erADGroupSamAccountName</attribute> **<sourceAttribute>erADGroupDN</sourceAttribute>** <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADGroup</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>true</paginateResults> </searchFilter> </formElement>
	resource.def	<ServiceGroups> <GroupDefinition ProfileName="ADGroupProfile" ClassName = "erADGroup" RdnAttribute = "erADGroupSamAccountName" AccountAttribute = "erGroup"> <AttributeMap> **<Attribute Name = "erGroupId" Value="erADGroupDN" />** <Attribute Name = "erGroupName" Value="erADGroupSamAccountName" /> <Attribute Name = "erGroupDescription" Value="erADGroupDescription" /> </AttributeMap> <BehaviorProperties> <Property Name = "Managed" Value = "true"/> </BehaviorProperties> </GroupDefinition> </ServiceGroups>

Table 11. Profile files (continued)

Value of the UseGroup registry key	Modifications required in	Expected modification
CN	erADAccount.xml	<formElement direction="inherit" label="$ergroup" name="data.ergroup"> <searchFilter multiple="true" type="select"> <filter>(objectclass=eradgroup)</filter> <base>contextual</base> <attribute>erADGroupSamAccountName</attribute> **<sourceAttribute>erADGroupCN</sourceAttribute>** <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADGroup</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>true</paginateResults> </searchFilter> </formElement>
	erADGroup.xml	<formElement direction="inherit" label= "$eradgroupismemberof" name="data.eradgroupismemberof"> <searchFilter multiple="true" type="select"> <filter>(objectclass=eradgroup)</filter> <base>contextual</base> <attribute>erADGroupSamAccountName</attribute> **<sourceAttribute>erADGroupCN</sourceAttribute>** <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADGroup</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>true</paginateResults> </searchFilter> </formElement>
	resource.def	<ServiceGroups> <GroupDefinition ProfileName="ADGroupProfile" ClassName = "erADGroup" RdnAttribute = "erADGroupSamAccountName" AccountAttribute = "erGroup"> <AttributeMap> **<Attribute Name = "erGroupId" Value="erADGroupCN" />** <Attribute Name = "erGroupName" Value="erADGroupSamAccountName" /> <Attribute Name = "erGroupDescription" Value="erADGroupDescription" /> </AttributeMap> <BehaviorProperties> <Property Name = "Managed" Value = "true"/> </BehaviorProperties> </GroupDefinition> </ServiceGroups>

Table 11. Profile files (continued)

Value of the UseGroup registry key	Modifications required in	Expected modification
GUID	erADAccount.xml	<formElement direction="inherit" label="$ergroup" name="data.ergroup"> <searchFilter multiple="true" type="select"> <filter>(objectclass=eradgroup)</filter> <base>contextual</base> <attribute>erADGroupSamAccountName</attribute> **<sourceAttribute>erADGroupGUID</sourceAttribute>** <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADGroup</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>true</paginateResults> </searchFilter> </formElement>
	erADGroup.xml	<formElement direction="inherit" label="$eradgroupismemberof" name="data.eradgroupismemberof"> <searchFilter multiple="true" type="select"> <filter>(objectclass=eradgroup)</filter> <base>contextual</base> <attribute>erADGroupSamAccountName</attribute> **<sourceAttribute>erADGroupGUID</sourceAttribute>** <delimiter></delimiter> <size></size> <width>300</width> <objectClass>erADGroup</objectClass> <showQueryUI>false</showQueryUI> <paginateResults>true</paginateResults> </searchFilter> </formElement>
	resource.def	<ServiceGroups> <GroupDefinition ProfileName="ADGroupProfile" ClassName = "erADGroup" RdnAttribute = "erADGroupSamAccountName" AccountAttribute = "erGroup"> <AttributeMap> **<Attribute Name = "erGroupId" Value="erADGroupGUID" />** <Attribute Name = "erGroupName" Value="erADGroupSamAccountName" /> <Attribute Name = "erGroupDescription" Value="erADGroupDescription" /> </AttributeMap> <BehaviorProperties> <Property Name = "Managed" Value = "true"/> </BehaviorProperties> </GroupDefinition> </ServiceGroups>

Configuring the Proxy Addresses attribute

You can modify the Proxy Addresses attribute of a user account. In this case, IBM Security Verify Governance Identity Manager sends the erADEProxyAddresses attribute in the modify operation with an attribute operation type of replace.

About this task

When the attribute operation type is replace, the adapter resets the proxy addresses for the user account on the Active Directory. You do not get the proxy addresses that are added to the user account by using the external application with an attribute operation type of replace when:

- You have additions to the Proxy Addresses attribute of a user account on the Active Directory by using an external application.
- The user accounts are not reconciled frequently.

To avoid a reset of the Proxy Addresses attribute on the Active Directory, modify the adapter profile for sending the erADEProxyAddresses attribute in the modify operation with an attribute operation type of Add or Delete. To handle the erADEProxyAddresses attribute with an attribute operation type of Add or Delete, modify the profile for Active Directory. The adapter profile (`ADprofile.jar`) is included in the JAR file for the adapter.

To modify the `ADprofile.jar` file for handling the erADEProxyAddresses attribute with an attribute operation type of Add or Delete, perform the following steps:

Procedure

1. Copy the `ADprofile.jar` file to a temporary directory, for example, `C:\Temp` directory.
2. Extract the contents of the `ADprofile.jar` file into the temporary directory by running the following command:

   ```
   cd C:\Temp
   jar -xvf ADprofile.jar
   ```

   The jar command creates the `C:\Temp\ADprofile` directory that has all the profile files.
3. From the extracted `ADprofile` directory, open the `resource.def` file in a text editor and search for this entry:

   ```
   <Parameter Name="erADEProxyAddresses" Source="account"
   ReplaceMultiValue="true" />
   ```
4. Delete all the occurrences of the above entry from the `resource.def` file and save the file.
5. Run the following command to create the new jar file:

   ```
   cd C:\Temp
   jar -cvf ADprofile.jar ADprofile
   ```
6. Import the new `ADprofile.jar` file on IBM Security Verify Governance Identity Manager.
7. After you import the adapter profile, restart IBM Security Verify Governance Identity Manager to reflect the updates.

Configuring the erGroup attribute

When you modify the Groups attribute of a user account, IBM Security Verify Governance Identity Manager sends the erGroup attribute in the modify operation with an attribute operation type of replace.

About this task

When the attribute operation type is replace, the adapter removes the membership of the user from the groups of which the user is a member on the Active Directory and that are not included in the modify

request. You do not get the membership of a user account to groups that are added to the user account by using the external application when:

- You modify the user account membership on the Active Directory by using an external application.
- The user accounts are not reconciled frequently.

When you modify the user account membership on the Active Directory, modify the profile for sending the erGroup attribute in the modify request with an attribute operation type of Add or Delete. To handle the erGroup attribute with attribute operation type as Add or Delete, modify the profile for Active Directory. The adapter profile (`ADprofile.jar`) is included in the JAR file for the adapter.

To modify the `ADprofile.jar` file for handling the erGroup attribute with an attribute operation type of Add or Delete, perform the following steps:

Procedure

1. Copy the `ADprofile.jar` file to a temporary directory, for example, `C:\Temp` directory.
2. Extract the contents of `ADprofile.jar` file into the temporary directory by running the following command:

   ```
   cd C:\Temp
   jar -xvf ADprofile.jar
   ```

   The jar command creates the `C:\Temp\ADprofile` directory that has all the profile files.
3. From the extracted `ADprofile` directory, open the `resource.def` file in a text editor and search for the entry `<Parameter Name="erGroup" Source="account" ReplaceMultiValue="true" />`.
4. Delete all the occurrences of the above entry from the `resource.def` file and save the file.
5. Run the following command to create a new jar file:

   ```
   cd C:\Temp
   jar -cvf ADprofile.jar ADprofile
   ```

6. Import the new `ADprofile.jar` file on IBM Security Verify Governance Identity Manager.
7. After you import the adapter profile, restart IBM Security Verify Governance Identity Manager to reflect the updates.

Configuring the cn attribute

You can configure the cn attribute.

About this task

The Active Directory Adapter user account class contains the following attributes for the attribute cn defined in the Active Directory:

- IBM Security Verify Governance Identity Manager common schema attribute cn
- erADFullName.

The cn attribute is used on the account form, by default. The adapter processes the cn attribute in the user add, modify, and reconciliation operations. The adapter uses the registry key UseITIMCNAttribute to use either the cn or the erADFullName attribute. When the compliance alerts on IBM Security Verify Governance Identity Manager are enabled, avoid using the cn attribute on the account form. To use the erADFullName attribute, you must customize the account form and set the registry key UseITIMCNattribute.

When you set the registry key UseITIMCNAttribute to FALSE, the adapter uses the erADFullName attribute for the cn attribute defined in the Active Directory for the user add, modify, and reconciliation operations.

To use the erADFullName attribute on the account form, perform the following steps:

Procedure

1. Add the erADFullName attribute to the user account form by customizing the Active Directory account form.

   For more information about customizing the user account form, see the IBM Security Verify Governance Identity Manager product documentation.
2. Set the registry key UseITIMCNAttribute to FALSE by using the **agentCfg** utility.

Verifying that the adapter is working correctly

After you install and configure the adapter, take these steps:

Procedure

1. Test the connection for the service that you have created on IBM Security Verify Governance Identity Manager.
2. Perform a full reconciliation from the Identity server.
3. Perform all supported operations (add, change and delete) on one account and examine the `WinADAgent.log` file after each operation to ensure that no errors were reported.

Chapter 6. Troubleshooting

Troubleshooting is a systematic approach to solving a problem. The goal of troubleshooting is to determine why something does not work as expected and how to resolve the problem. This topic provides information and techniques for identifying and resolving problems that are related to the adapter, including troubleshooting errors that might occur during the adapter installation.

Techniques for troubleshooting problems

Certain common techniques can help with the task of troubleshooting. The first step in the troubleshooting process is to describe the problem completely.

Problem descriptions help you and the IBM technical-support representative find the cause of the problem. This step includes asking yourself basic questions:

- What are the symptoms of the problem?
- Where does the problem occur?
- When does the problem occur?
- Under which conditions does the problem occur?
- Can the problem be reproduced?

The answers to these questions typically lead to a good description of the problem, which can then lead you to a problem resolution.

What are the symptoms of the problem?

When you start to describe a problem, the most obvious question is "What is the problem?" This question might seem straightforward; however, you can break it down into several more-focused questions that create a more descriptive picture of the problem. These questions can include:

- Who, or what, is reporting the problem?
- What are the error codes and messages?
- How does the system fail? For example, is it a loop, hang, crash, performance degradation, or incorrect result?

Where does the problem occur?

Determining where the problem originates is not always easy, but it is one of the most important steps in resolving a problem. Many layers of technology can exist between the reporting and failing components. Networks, disks, and drivers are only a few of the components to consider when you are investigating problems.

The following questions help you to focus on where the problem occurs to isolate the problem layer:

- Is the problem specific to one operating system, or is it common across multiple operating systems?
- Is the current environment and configuration supported?
- Do all users have the problem?
- (For multi-site installations.) Do all sites have the problem?

If one layer reports the problem, the problem does not necessarily originate in that layer. Part of identifying where a problem originates is understanding the environment in which it exists. Take some time to completely describe the problem environment, including the operating system and version, all corresponding software and versions, and hardware information. Confirm that you are running within an environment that is a supported configuration. Many problems can be traced back to incompatible levels of software that are not intended to run together or are not fully tested together.

When does the problem occur?

Develop a detailed timeline of events that lead up to a failure, especially for those cases that are one-time occurrences. You can most easily develop a timeline by working backward: Start at the time an error was reported (as precisely as possible, even down to the millisecond), and work backward through the available logs and information. Typically, you use the first suspicious event that you find in a diagnostic log.

To develop a detailed timeline of events, answer these questions:

- Does the problem happen only at a certain time of day or night?
- How often does the problem happen?
- What sequence of events leads up to the time that the problem is reported?
- Does the problem happen after an environment change, such as upgrading or installing software or hardware?

Responding to these types of questions can give you a frame of reference in which to investigate the problem.

Under which conditions does the problem occur?

Knowing which systems and applications are running at the time that a problem occurs is an important part of troubleshooting. These questions about your environment can help you to identify the root cause of the problem:

- Does the problem always occur when the same task is being done?
- Is a certain sequence of events required for the problem to occur?
- Do any other applications fail at the same time?

Answering these types of questions can help you explain the environment in which the problem occurs and correlate any dependencies. Remember that just because multiple problems might occur around the same time, the problems are not necessarily related.

Can the problem be reproduced?

From a troubleshooting standpoint, the ideal problem is one that can be reproduced. Typically, when a problem can be reproduced you have a larger set of tools or procedures at your disposal to help you investigate. Problems that you can reproduce are often easier to debug and solve.

However, problems that you can reproduce can have a disadvantage: If the problem is of significant business impact, you do not want it to recur. If possible, re-create the problem in a test or development environment, which typically offers you more flexibility and control during your investigation.

- Can the problem be re-created on a test system?
- Do multiple users or applications have the same type of problem?
- Can the problem be re-created by running a single command, a set of commands, or a particular application?

Error messages and problem solving

A warning or error message might be displayed in the user interface to provide information about the adapter or when an error occurs.

These errors might be displayed in the user interface when the adapter is installed on your system.

Table 12. Troubleshooting the Active Directory Adapter errors	
Error message	**Corrective action**
Unable to bind to base point	Ensure that: • The Users Base Point is correctly specified on the adapter service form. • The target servers are up and reachable when they are specified in the base point. • The user ID is correctly specified on the adapter service form. • The password is correctly specified on the adapter service form. • The Active Directory Server is reachable from the workstation where the adapter is installed.
Unable to bind to group base point.	Ensure that: • The Groups Base Point is correctly specified on the adapter service form. • The user ID is correctly specified on the adapter service form. • The password is correctly specified on the adapter service form. • The target servers are up and reachable when they are specified in the base point. • The Active Directory Server is reachable from the workstation where the adapter is installed.
Unable to determine default domain	This error occurs when the Active Directory Adapter fails to: • Bind to root DSE • Get the default naming context Ensure that: • The Users Base Point is correctly specified on the adapter service form. • The user ID is correctly specified on the adapter service form. • The password is correctly specified on the adapter service form. • The Active Directory Server is reachable from the workstation where the adapter is installed.
Error binding to DN: *DN String*	This error occurs when the Active Directory Adapter fails to bind to a user object of the Active Directory Server for processing. Ensure that the user processed in the Active Directory Server is not deleted by any other process simultaneously.

Table 12. Troubleshooting the Active Directory Adapter errors (continued)

Error message	Corrective action
Extended attribute *attribute name* has unsupported syntax	The Active Directory Adapter does not support the data type used for the extended attribute. Use one of the following data types: • Boolean • Integer • Case-sensitive string • Not case-sensitive string • Numerical string • Unicode string • Distinguished name • UTC coded time • OctetString • Integer8 For more information about customizing the adapter to use the extended attributes, see "Customizing the Active Directory Adapter" on page 54.
Extended attribute *attribute name* not found in Active Directory schema	The extended attribute specified in the `exschema.txt` file does not exist on the Active Directory Server. Either remove the attribute name from the `exschema.txt` file or add the attribute to the Active Directory Server.
Error binding to schema container *error code*. Loading of extended schema attribute *attribute name* failed.	These errors occur when the Active Directory Adapter fails to extract the schema of the extended attributes. • Ensure that the Active Directory Server is reachable from the workstation where the adapter is installed. • Verify that the extended attribute is correctly defined and added to the user class. When the adapter service is started, the adapter reads the exschema.txt file and binds to the domain in which the adapter is running. The adapter checks the syntax of the specified. Because checking the syntax of an extended attribute is a one-time process, it is done at startup. If the adapter fails to bind to the domain, it does not manage any of the extended attributes. Ensure that: • • At least 1 domain controller is accessible before starting the Active Directory Adapter service. • The user account under which the adapter service is running has permission to read the Active Directory schema.
Error getting parent of schema *error code*. Loading of extended schema attribute *attribute name* failed.	
Error binding to DN of schema *error code*. Loading of extended schema attribute *attribute name* failed.	
Unable to connect to default domain. Loading of extended schema attribute *attribute name* failed.	
Extended schema file not found. No extensions loaded.	This information message occurs when the Active Directory Adapter fails to find the extended schema file (`exschema.txt`) or fails to open the file.

Table 12. Troubleshooting the Active Directory Adapter errors (continued)

Error message	Corrective action
Unable to bind to user *user name*	This error occurs when the Active Directory Adapter fails to connect to a user object in the Active Directory Server for processing. Ensure that the user *user name* exists on the Active Directory Server.
Error determining RAS server name	Check the value of the registry key **ForceRASServerLookup**. If the value of the key is TRUE, the Active Directory Adapter determines the RAS server regardless of whether you specify the server name on the adapter service form. This error might be because the domain does not exist or the domain controller is not available for the specified domain. Ensure that the Active Directory Server is reachable from the workstation where the adapter is installed.
Unable to get domain name. Terminal and RAS servers cannot be determined.	This error occurs when the Active Directory Adapter fails to get the domain name from the specified base point or from the default domain. Ensure that a base point is specified with a correct domain name.
Invalid domain name syntax	Use one of the following formats to specify the domain name: • *Server name*/ou=*org1*,dc=*ibm*,dc=*com* • ou=*org1*,dc=*ibm*,dc=*com*
User not found	Ensure that the user exists on the Active Directory Server and is not directly deleted or modified on the Active Directory Server.
Group not found.	Ensure that the group exists on the Active Directory Server and is not directly deleted or modified on the Active Directory Server.
Error setting attributes *country*. Unknown country code.	The country code specified for the user is not valid. Specify a valid country code and submit the request again. For information about valid country codes, see the country and region codes section in the *Active Directory Adapter User Guide*.
Could not modify the attribute– msExchUserAccountControl	This warning occurs when the user mailbox is not disabled on suspending a user account.
Error removing membership from group *group name*	The Active Directory Adapter failed to remove the membership of a user or group from the group *group name*. Ensure that: • The user or group exists on the Active Directory Server. • The user or group is a member of the group *group name*. • The group specified exists on the Active Directory Server.

Table 12. Troubleshooting the Active Directory Adapter errors (continued)

Error message	Corrective action
Error adding membership to group *group name*	The Active Directory Adapter failed to add membership of the user or group to the group *group name*. Ensure that: • The user or group exists on the Active Directory Server. • The user or group is not already a member of the group *group name*. • The group specified exists on the Active Directory Server.
Unable to get info on share *share name*	This error occurs when the Active Directory Adapter fails to retrieve share information from the home directory of the user. Ensure that: • The user account under which the adapter is running has access to the home directory. • The share name exists on the workstation where the home directory is created.
Invalid home directory path *path name*	The Active Directory Adapter supports creation and deletion of only UNC home directories. Specify the UNC home directory path in the following format: *servername**sharename**foldername* **Note:** • NTFS security and Shares can be set only on the Home Directories that are a UNC path. • Share Access can be set only on the Home Directories that are a UNC path that have a share created.
Unable to delete home directory *home directory name*	The Active Directory Adapter is not able to delete the specified home directory. If the adapter is unable to delete the UNC home directory, ensure that: • The value of the registry key **DeleteUNCHomeDirectories** is TRUE. • The user account under which the adapter is running has permissions to delete the directory.
Home directory deletion is not enabled. Home directory will not be deleted.	To enable home directory deletion, set the values of **DeleteUNCHomeDirectories** and **ManageHomeDirectories** registry keys to TRUE. Resend the modify request from IBM Security Verify Governance Identity Manager.
Home directory creation not enabled. Directory will not be created.	To enable home directory creation, set the values of **CreateUNCHomeDirectories** and **ManageHomeDirectories** registry keys to TRUE. Resend the modify request from IBM Security Verify Governance Identity Manager.

Table 12. Troubleshooting the Active Directory Adapter errors (continued)	
Error message	**Corrective action**
Error creating home directory *home directory name*	The Active Directory Adapter is not able to create home directory. Ensure that: • A directory with the same name does not exist. • The user account has permissions to create home directory. • Intermediate directories exist. The adapter creates only the final directory in the specified path.
Unable to set Home Directory Drive. Failed to create Home Directory.	
Unable to set Home Directory NTFS security. Failed to create Home Directory.	
Unable to set Home Directory Share. Failed to create Home Directory.	
Unable to set Home Directory Share Access. Failed to create Home Directory.	
Error deleting share *share name*	The Active Directory Adapter is not able to delete the share when you clear the value of the share-related attributes from the Active Directory Server account form. Ensure that: • The user account has access to the specified share. • The specified share name exists. • The user account under which the adapter is running has permissions to create home directory.
Search failed. Unable to retrieve additional data after 3 retries.	The Active Directory Adapter retrieves data from the Active Directory Server in a paged manner. The adapter reconciles users, groups, and containers and attempts to retrieve data in a maximum of three attempts. If all three attempts fail, the adapter abandons the search. The adapter cannot retrieve data because of one of the following reasons: • The network response is slow. • The Active Directory Server is busy. • The Active Directory Adapter installed on the Active Directory Server server is overloading the server. For information about configuring the Active Directory Server, see http://support.microsoft.com.
User search failed	
Group search failed. Error code: *error code - error description*. Provider: *provider name*.	
Container search failed. Error code: *error code - error description*. Provider: *provider name*.	
Error performing User Lookup	
errorMessage="Unsupported filter"	The adapter does not support the attribute specified in the filter. For the list of supported attributes, see supported attributes in the *Active Directory Adapter User Guide*.
Error setting attribute eradprimarygroup. ADSI Result code: 0x80072035 - The server is unwilling to process the request.	Ensure that: • The user is a member of the specified group. • The specified group is either a universal security group or a global security group.

Table 12. Troubleshooting the Active Directory Adapter errors (continued)

Error message	Corrective action
ADSI Result code: 0x80072014 - The requested operation did not satisfy one or more constraints associated with the class of the object. ADSI Result code: 0x8007202f - A constraint violation occurred.	These errors occur when the specified value for the attribute violates any constraint associated with that attribute. For example, a constraint might be: • Minimum or maximum length of characters the attribute can store. • Minimum or maximum value the attribute can accept. Ensure that the specified value for the attribute does not violate these constraints. **Note:** If any one of the attribute specified in the request violates a constraint, the adapter gives the same error for all the subsequent attributes. This error is issued even though they do not violate any constraint. For example, the **Title** attribute on the Active Directory Server can store a description of maximum of 64 characters. If you specify a description of more than 64 characters, the adapter gives these errors for the **Title** attribute and for all the other attributes specified in the request.
Request for proxy email types should contain at least one primary SMTP address	Verify that the request for proxy email types contains a primary SMTP address.
Unable to load XML transformation buffer from '*adapter installation directory*\data \xforms.xml'	The Active Directory Adapter does not use the `xforms.xml` file. Therefore, you can safely ignore the xforms-related errors that are recorded in the `WinADAgent.log` file.
Unable to bind to group *group name*.	This error occurs when the Active Directory Adapter fails to connect to a group object in the Active Directory Server for processing. Ensure that the group *group name* exists on the Active Directory Server.
The specified User Principal Name (UPN) *UPN* values already exists in the enterprise. Specify a new one.	This error occurs when an attempt is made to create user request and the user account exists in the Active Directory Server with the same value for User Principal Name attribute. Ensure that: • The value specified for the User Principal Name attribute when you create a user account is not already used by an existing user account on the Active Directory Server. • You set the registry key UPNSearchEnabled to FALSE when you do not want the adapter to check the uniqueness of the User Principal Name attribute. For more information about usage of the registry key UPNSearchEnabled, see "User Principal Name of a user account" in the *Active Directory Adapter User Guide*.
Error while fetching the group interface for group DN.	This error occurs when the Active Directory Adapter fails to bind to a group object on the Active Directory Server for processing. Ensure that the group processed in the Active Directory Server is not deleted by any other process simultaneously.

Table 12. Troubleshooting the Active Directory Adapter errors (continued)

Error message	Corrective action
Unable to bind to the container object in move operation.	This error occurs when the Active Directory Adapter binds to the requested container when a user or group object is moved in the Active Directory Server hierarchy. Ensure that the container exists on the Active Directory Server.
Cannot set Fixed Callback without Callback number. Callback number not found in the request.	When you select Callback Settings as Fixed Callback, you must specify the Callback Number.
Error setting the RAS attribute *RAS attribute name*. Error reading RAS info.	Ensure that: • The user account under which the adapter is running has administrator rights to the Active Directory Server. • The RAS service is running on the Domain Controller.
Not a valid IPv4 address.	The IP address specified for the Static IPv4 Address is in an incorrect format. Specify the IP address in the IPv4 format.
Agent ADAgent is not installed.	This error occurs when an attempt is made to run the certTool utility by running the following command: `CertTool -agent ADAgent` Ensure that: • The user who runs the certTool utility has administrator permissions. • You disabled the User Account Control (UAC) security feature before you run the certTool utility on the workstation where the adapter is installed.
Home Directory will not be created. Home directory management is disabled. Cannot create share *share name*. Home directory management is disabled. Cannot set share access. Home directory management is disabled. Cannot set NTFS access. Home directory management is disabled.	Set the adapter registry keys CreateUNCHomeDirectories and ManageHomeDirectories to TRUE to: • Create a home directory • Create home directory share • Set share access • Set home directory NTFS access for a user account. For more information about creating the home directory and modifying the home directory attribute, see *Active Directory Adapter User Guide*.
Value specified is not in the proper format.	Ensure that the value format of extended attribute of type DNWithBinary is `B:char count:binary value:object DN`
Value specified for the attribute does not start with character 'B'.	Ensure that value specified for extended attribute of type DNWithBinary is start with the character ‘B’ only.

Table 12. Troubleshooting the Active Directory Adapter errors (continued)	
Error message	**Corrective action**
Value given after 'B:' is not correct. Expected value is the total number of Hexadecimal Digit count	For extended attribute of type DNWithBinary, verify that value given for the *char count* is the total number of Hexadecimal Digit count. Ensure that it does not contain any alphabetical characters or any special characters.
Hexadecimal value does not contain the number of characters specified in the character count.	For extended attribute of type DNWithBinary, verify that total hexadecimal digit count specified in the *char count* is equal to number of hexadecimal characters.
Wrong Digit in Hex String.	For extended attribute of type DNWithBinary, verify that value given in the *binary value* contains only hexadecimal character. Valid characters are numerals 0 through 9 and letters A through F. The value can be a combination of valid numerals and letters.
Value is not set on resource due to invalid constraint.	This error occurs when the specified value for the extended attribute of type DNWithBinary violates any constraint associated with that attribute. For example, some constraints might be: • The *object DN* in the value must be a distinguished name of existing user object. • The maximum or minimum number of bits in the hexadecimal value. Ensure that the specified value for the attribute does not violate any constraints.
Hexadecimal value should always contain even number of characters.	For extended attribute of type DNWithBinary, verify that value given in the *binary value* contains an even number of hexadecimal characters.
Attribute can be set only if Mailbox is enabled for Unified Messaging. To enable Unified Messaging both values UMMailbox Policy and UM Addresses(Extensions) are required.	Ensure that valid values of both UMMailbox Policy and UM Addresses(Extensions) are specified in the request to enable the user for Unified Messaging.
Attribute Operation Type is not supported.	Ensure that the value specified for UM Addresses (Extensions) is not of operation type, MODIFY.
Attribute cannot be set. Mailbox is Disabled for Unified Messaging.	Ensure that the request does not contain Unified Messaging attributes with operation ADD or MODIFY when the MailBox of the user is disabled for Unified Messaging.
Attribute cannot be set. Error occurred while trying to Disable MailBox for Unified Messaging.	This error occurs if disable Unified Messaging is failed and if request contains UM Addresses (Extensions) attribute with operation types ADD or MODIFY.
Attribute cannot be delete. Error occurred while trying to Disable MailBox for Unified Messaging.	This error occurs if disable Unified Messaging is failed and if the request contains UM Addresses (Extensions) attribute with operation type DELETE.

Known behaviors

The following behaviors and limitations are known to exist in the operation of the Active Directory Adapter.

Directory NTFS and share access

The agent returns the actual, effective permissions that are granted to a user and not the specific access that is assigned to the user account.

Expiration date

The Active Directory Users and Computers Microsoft Management Console (MMC) snap-in displays the account expiration date as one day earlier than the date contained in the **accountExpires** attribute. The Identity server displays the value that is contained in the **account expires** attribute.

Password properties

The password properties are specific to the account. However, these properties can be overridden by the security policies of the managed resource.

For example:

- Domain controller security policies
- Domain security policies
- Local security policies

Language preference settings for accounts

The languages attribute **eradlanguage** is an Exchange attribute. If you are using a configuration without Exchange, setting this attribute returns a warning.

Log message: Error More Data

The `Error_More_Data` message might be in the adapter log if a reconciliation is run while the Active Directory server is under load.

The Active Directory Adapter is designed to retry the query three times before terminating the Reconciliation. For more information, see the Microsoft Knowledge base.

Replication delay solutions for a mailbox addition on Microsoft Exchange

Requesting a user account on Active Directory with mail status on Exchange might generate the error `User does not exist.`

This behavior is caused by replication delay. Exchange might not find the user account on a domain controller, if the account is created on another domain

The solution here is to target both the following operations to the same Domain Controller:

- Create user account operation.
- The Exchange operation, to either mailbox enable or mail-enable the user account.

To specify a target server use the **Users Base Point DN** on the Active Directory profile service form. The Base Point must contain the name of the domain controller. For more information about how to specify Users Base Point DN, see “Users Base Point configuration for the adapter” on page 61

Example

```
Users Base Point DN: DC01/ou=Test,dc=MyDomain,dc=com.
```

Errors in Exchange mailbox permissions

The adapter might not set the mailbox permissions correctly, even though the request generates a SUCCESS message.

If multiple permissions are set in a modify request but only the last permission takes effect, you must modify the **SetMailboxPermissionDelay** registry key.

The **SetMailboxPermissionDelay** registry key cause the adapter to wait a specified time in seconds before processing the next permission. The default setting of this key is 0 or no delay. Typically setting this registry to 20 resolves the problem.

For information about setting registry keys, see "Modifying registry settings" on page 35.

No provisioning provider installed

This error is a known configuration issue with Exchange.

This error is misleading in that it is typically caused by a lack of permissions by the adapter logon account. Typically the error is not because a "provisioning provider" is installed.

To provision mailboxes to Exchange, the logon account must be a member of the appropriate security groups. Because of these variations, it is not possible to provide a definitive list of group memberships that are required by the adapter logon account:

- Active Directory installations can be on single domains, multiple domains, or sub domains.
- Groups can be customized, that is added to other groups.

Membership in the Domain Admins group is required to provision accounts.

Typically, membership in the following Exchange groups is sufficient for the adapter to provision mailboxes:

- Recipient Management
- Organization Management
- Exchange Windows Permissions

If the adapter logon account is a member of these groups and the error persists, add a membership to Enterprise Admins group. This action can determine if the problem is due to permissions. If adding this membership resolves the issue, see the Microsoft documentation about trial and error to determine which group memberships are needed.

Exchange connection issues

The adapter uses remote powershell sessions to manage Exchange servers. If the adapter has issues connecting to the servers, you can manually run the powershell cmdlets that the adapter uses to troubleshoot the connection errors.

New-PSSession

Use the following command to create a new session on the remote server. Replace *<hostAddr>* with the actual hostname or IP of the Exchange server.

```
PS>$mySession = New-PSSession -configurationname Microsoft.Exchange -connectionuri http://
<hostAddr>/Powershell -authentication Kerberos
```

import-pssession

Use the following command to import the remote session into your local session. If this command is successful, you should be able to run any Exchange cmdlets as if you are on the Exchange server.

```
PS>import-pssession $mySession
```

Issues with different Exchange Server versions

Different versions of Exchange Server have issues when you modify mailboxes on a server of one version from a powershell session on a server with a different version.

Preferred servers

Preferred servers let you specify which Exchange servers are used to execute requests.

There is no API for managing Exchange servers. Exchange servers are managed by using powershell cmdlets. The required cmdlets are only available on the Exchange servers. The adapter must use a remote powershell connection to one of the servers to execute the cmdlets to process a request.

The adapter uses the concept of preferred servers for both Exchange and Skype for Business.

When a request comes in, the adapter must connect to a remote server to execute the request. By default, it does an LDAP search into AD to find the servers, then tries to connect. It uses the first server that it can connect with. If preferred servers are specified, the adapter will try to connect with those servers first. Setting the exclusive flag to TRUE will force the adapter to only use the preferred servers.

Remember that the preferred servers are where the request is executed. This has nothing to do with where mailboxes are created. The account attribute `erMailboxStore` specifies the mail database which is not necessarily on the preferred server.

Chapter 7. Uninstalling

To remove an adapter from the Identity server for any reason, you must remove all the components that were added during installation. Uninstalling the adapter involves running the uninstaller and removing the adapter profile from the Identity server.

Before you remove the adapter, inform your users that the adapter is unavailable. If the server is taken offline, adapter requests that were completed might not be recovered when the server is back online.

Uninstalling the adapter from the target server

You can uninstall the adapter from the target server.

Procedure

1. Stop the adapter service.
2. Run the uninstaller. To run the uninstaller:
   a. Navigate to the adapter home directory. For example, `Tivoli\agents\`*`adaptername`*`\Uninstall_IBM Windows AD Adapter for ITIM (64 Bit)`
   b. Double click the *Uninstall IBM Windows AD Adapter for ITIM (64 Bit).exe* file.
   c. On the Uninstall IBM Windows AD Adapter for ITIM (64 BIT) window, click **Uninstall**.
   d. On the Uninstall Complete window, click **Done**.

Deleting the adapter profile

Remove the adapter service/target type from the Identity server. Before you delete the adapter profile, ensure that no objects exist on the Identity server that reference the adapter profile.

Objects on the Identity server that can reference the adapter profile:

- Adapter service instances
- Policies referencing an adapter instance or the profile
- Accounts

Note: The Dispatcher component must be installed on your system for adapters to function correctly in a Security Directory Integrator environment. When you delete the adapter profile, do not uninstall the Dispatcher.

For specific information about how to delete the adapter profile, see the IBM Security Verify Governance Identity Manager product documentation.

Chapter 8. Reference

Reference information is organized to help you locate particular facts quickly, such as adapter attributes, registry settings, and environment variables.

Adapter attributes and object classes

Adapter attributes and object classes are required for customization, creating provisioning rules, and understanding what service/target attributes are supported by the adapter. The Identity server communicates with the adapter by using attributes, which are included in transmission packets that are sent over a network.This topic is not applicable for this adapter.

The combination of attributes depends on the type of action that the Identity server server requests from the adapter.

The next table lists the account form attributes that the adapter uses.

Table 13. Attributes, descriptions, and corresponding data types

Adapter Attribute	Active Directory Attribute	Description	Syntax
cn erADFullName **Note:** erADFullName is used only if cn is not specified	cn	Specifies the full name of the user (given name and surname).	String
description	description	Specifies the description for the user.	String
erAccountStatus	userAccountControl		
erADAllowDialin	msNPAllowDialin	Specifies whether the user can Dial in to the network	Boolean
erADAllowEncryptedPassword	userAccountControl	Specifies whether encrypted passwords are allowed.	Boolean
erADBadLoginCount	badPwdCount	Specifies the number of invalid login attempts that are allowed since the last reset.	Long
erADCallbackNumber	msRADIUSCallbackNumber	Specifies the callback number for remote access services that is used when DialinCallBack is set to fixed.	String
erADCannotBeDelegated	userAccountControl	Specifies that this account cannot be assigned for delegation by another account.	Boolean
erADContainer	DN of container	Specifies the Relative Distinguished Name (RDN) of a container object in which to create the user account. The container is relative to the basepoint.	RN string
erADCountryCode	countryCode	Specifies the country where the user resides.	Integer

Table 13. Attributes, descriptions, and corresponding data types (continued)			
erADDialinCallback	msRADIUSServiceType	Sets the Dial-in Callback for the user. 1 - No Callback 2 - Fixed callback using erADCallbackNumber 3 - This option is not used 4 - User supplied callback	Integer
erADDisplayName	displayName	Specifies the Active Directory displayName attribute.	String
erADDistinguishedName	distinguishedName	Specifies the distinguished name of the account on the Active Directory.	String
erADEActiveSyncEnabled	msExchOmaAdminWirelessEnable	Specifies the distinguished name of the account on the Active Directory.	Boolean
erADEAddressBookPolicy	msExchAddressBookPolicyLink	Specified the DN of the Address Book Policy. From supporting data erADEAddrBookPlcy	String
erADEAlias	mailNickname	Specifies the alias for the Exchange Mailbox.	String
erADEAllowedAddressList	authOrig	Specifies a list of email IDs that the user accepts mail from.	String
erADEAllowPermTo1Level	msExchMailboxSecurityDescriptor	Specifies if permission is inherited	Boolean
erADEApplyOntoAllow	msExchMailboxSecurityDescriptor	Specifies a Allow permission	Boolean
erADEApplyOntoDeny	msExchMailboxSecurityDescriptor	Specifies Deny permission	Boolean
erADEAssociatedExtAcc	msExchMailboxSecurityDescriptor	Specifies whether the user has **associated external** **account** permission.	Boolean
erADEAutoGenEmailAddrs	msExchPoliciesExcluded	Specifies whether the recipient update services updates the email address.	Boolean
erADEChgPermissions	msExchMailboxSecurityDescriptor	Specifies whether to change the user's Mailbox permission.	Integer
erADEConnectToMailbox	Not mapped – write only	Specify DN of disconnected mailbox to connect to user. From supporting data erADEConnectToMailbox	String
erADEDaysBeforeGarbage	garbageCollPeriod	Specifies the number of days that deleted mail is retained before it is permanently deleted.	Integer
erADEDelegates	publicDelegates	Specifies the list of all users that have access to the Exchange Mailbox.	String
erADEDelMailboxStorage	msExchMailboxSecurityDescriptor	Specifies whether the user has delete Mailbox storage permission.	Integer
erADEDenyPermTo1Level	msExchMailboxSecurityDescriptor	Specifiis whether deny permission is inherited	Boolean

Table 13. Attributes, descriptions, and corresponding data types (continued)			
erADEEnableRetentionHold	msExchELCMailboxFlags	Specifies whether retention hold is enabled	Boolean
erADEEnableStoreDeflts	mDBUseDefaults	Specifies whether to use only default store values for storage limits, or to use other properties that pertain to the Mailbox.	Boolean
erADEEndRetentionHold	msExchELCExpirySuspensionEnd	Specifies whether to enable or disable Retention Hold.	Boolean
erADEExtension1	extensionAttribute1	Specifies a user-defined extension attribute.	String
erADEExtension10	extensionAttribute10	Specifies a user-defined extension attribute.	String
erADEExtension11	extensionAttribute11	Specifies a user-defined extension attribute.	String
erADEExtension12	extensionAttribute12	Specifies a user-defined extension attribute.	String
erADEExtension13	extensionAttribute13	Specifies a user-defined extension attribute.	String
erADEExtension14	extensionAttribute14	Specifies a user-defined extension attribute.	String
erADEExtension15	extensionAttribute15	Specifies a user-defined extension attribute.	String
erADEExtension2	extensionAttribute2	Specifies a user-defined extension attribute.	String
erADEExtension3	extensionAttribute3	Specifies a user-defined extension attribute.	String
erADEExtension4	extensionAttribute4	Specifies a user-defined extension attribute.	String
erADEExtension5	extensionAttribute5	Specifies a user-defined extension attribute.	String
erADEExtension6	extensionAttribute6	Specifies a user-defined extension attribute.	String
erADEExtension7	extensionAttribute7	Specifies a user-defined extension attribute.	String
erADEExtension8	extensionAttribute8	Specifies a user-defined extension attribute.	String
erADEExtension9	extensionAttribute9	Specifies a user-defined extension attribute.	String
erADEForwardingStyle	deliverAndRedirect	Specifies whether email is also delivered to an alternate email address.	String
erADEForwardTo	altRecipient	Specifies the URL where email is to be forwarded.	String
erADEFullMailboxAccess	msExchMailboxSecurityDescriptor	Specifies whether the user has full Mailbox access permission. 1=Allow 2=Deny 0 or no value=None	Integer

Table 13. Attributes, descriptions, and corresponding data types (continued)			
erADEGarbageAfterBckp	deletedItemFlags	Specifies whether deleted messages can be permanently deleted after the Mailbox is backed up.	Boolean
erADEHardLimit	mDBOverHardQuotaLimit	Specifies the maximum Mailbox size in KB when sending and receiving email is disabled.	Integer
erADEHideFromAddrsBk	msExchHideFromAddressLists	Specifies whether the address is displayed in the address book.	Boolean
erADEHomeMDB	homeMDB	Specifies the URL of the store for the recipient.	String
erADEIMAP4Enabled	protocolSettings	Specifies whether to enable or disable MAPI support.	Boolean
erADEIMAP4Format	protocolSettings	Specifies the IMAP4 format. 0=Text 1=HTML 2=HTML and alternative text 3=Enriched Text 4=Enriched Text and alterative text 5=Best body format 6=TNEF	Integer
erADEIMAP4FormatUseDefault	protocolSettings	Specifies whether to use the default IMAP4 format	Boolean
erADEIncomingLimit	delivContLength	Specifies the max incoming message size	Integer
erADELanguages	language	Specifies languages	String
erADEMailboxFolderPolicy	msExchMailboxTemplateLink	Specifies DN of Mailbox folder policy. From supporting data erADMBFldPolicy objects	String
erADEMailboxStore	homeMDB	Specifies DN of mailbox store of the mailbox	
erADEMailboxUMExtensions	null	Specifies a list of Unified Messaging extension numbers	String
erADEMailboxUMPolicy	msExchUMTemplateLink	Specifies the Unified Messaging policy. From supporting data erADMBUMPolicy objects	String
erADEMAPIBlockOutlookRpcHttp	protocolSettings	Specifies whether to block Outlook Rpc	Boolean
erADEMAPIEnabled	protocolSettings	Specifies whether MAPI is enabled	Boolean
erADEmployeeID	employeeID	Specifies the user's employee identifier.	String
erADEmployeeNumber	employeeNumber	Specifies the employee number	String
erADEMsOwaPolicy	msExchOWAPolicy	Specfies the OWA Policy	String
erADEOutgoingLimit	submissionContLength	Specifies the maximum size in KB of a message that is sent from the recipient.	Integer
erADEOutlookWebAccessEnabled	protocolSettings	Specifies whether to enable or disable **Outlook Web Access**.	Boolean

Table 13. Attributes, descriptions, and corresponding data types (continued)			
erADEOverQuotaLimit	mDBOverQuotaLimit	Specifies the maximum size of a Mailbox in KB before sending messages is suspended.	Integer
erADEOverrideGarbage	deletedItemFlags	Specifies whether the store is prevented from permanently deleting messages.	Boolean
erADEPOP3Enabled	protocolSettings	Specifies whether POP3 is enabled	Boolean
erADEPOP3FormatUseDefault	protocolSettings	Specifies whether the default POP3 format is used	Boolean
erADEPOP3Format	protocolSettings	Specifies the POP3 format 0=Text 1=HTML 2=HTML and alternative text 3=Enriched Text 4=Enriched Text and alterative text 5=Best body format 6=TNEF	Integer
erADEProxyAddresses	proxyAddresses	Specifies a list of proxy addresses for the recipient.	String
erADEReadPermissions	msExchMailboxSecurityDescriptor	Specifies whether the user has **read Mailbox** permission. 1=Allow 2=Deny 0 or no value=None	Integer
erADERecipientLimit	msExchRecipLimit	Specifies the maximum number of people to whom the recipient can send email.	Integer
erADERemoteAddress	targetAddress	Specifies the target address for a Mail enabled user. (Enable-MailUser)	String
erADERstrctAdrsFg	No longer used		
erADERstrctAdrsLs	authOrig/unauthOrig	Specifies a list of email addresses to reject mail from.	String
erADEServerName	*Null*	Specifies the name of the Microsoft Exchange Server.	String
erADEShowInAddrBook	showInAddressBook	Specifies the list of address books that the user is a member of.	String
erADESMTPEmail	mail	Specifies the primary SMTP address that is used for the recipient.	String
erADEStartRetentionHold	msExchELCExpirySuspensionStart	Specifies the date to start retention hold.	Date
erADEStoreQuota	mDBStorageQuota	Specifies a limit when the recipient receives a warning for exceeding their mail file storage allocation.	Integer

Table 13. Attributes, descriptions, and corresponding data types (continued)			
erADETakeOwnership	msExchMailboxSecurityDescriptor	Specifies whether the user has **take Mailbox ownership** permission.	Integer
erADETargetAddress	targetAddress	Specifies the external email address to be used by the user.	String
erADEX400Email	textEncodedORAddress	Specifies the primary X.400 address that is used for the recipient.	String
erADExDialin	msNPAllowDialin	Specifies whether to allow dialin	Boolean
erADExpirationDate	accountExpires	Specifies the date and time after which the user cannot log in.	Date
erADFailIntrLgonCnt	msDS-FailedInteractiveLogonCount	Specifies the failed interactive logon count. Read only	Integer
erADfax	facsimileTelephoneNumber	Specifies the fax numbers of the user.	String
erADFlIntrLgonCntAtLastSucLgon	msDS-FailedInteractiveLogonCountAtLastSuccessfulLogon	Specifies the failed interactive logon count at the last successful logon	Integer
erADHomeDir	homeDirectory	Specifies a null-terminated string that contains the path of the user's home directory. This string can specify a local path or a UNC path. For example: \\machine\share\path	String
erADHomeDirAccessShare	*File system*	Specifies the user access level on the share.	String
erADHomeDirDrive	homeDrive	Specifies the drive letter to assign to a UNC-based home directory.	String
erADHomeDirNtfsAccess	*File system*	Specifies the NTFS security level for the home directory of the user.	String
erADHomeDirShare	*File system*	Specifies the name of the share to create for home directory. Append a dollar sign ($) to create a hidden share.	String
erADHomePage	wWWHomePage	Specifies the URL for the home page of the user.	String
erADInitial	initials	Specifies the middle initials of the name of the user.	String
erADIsAccountLocked	lockoutTime	Specifies whether the account is locked because of intruder detection.	Boolean
erADLastFailedLogin	badPasswordTime	Specifies the date and time of the last failed network login.	Date
erADLastFailIntrLgonTime	msDS-LastFailedInteractiveLogonTime		

Table 13. Attributes, descriptions, and corresponding data types (continued)			
erADLastLogoff	lastLogoff	Specifies the date and time of the last network logoff.	Date
erADLastLogon	lastLogon	Specifies the date and time of the last successful network login.	Date
erADLastLogonTimeStamp	lastLogonTimestamp	Specifies the timestamp of the last logon	Date
erADLastSuccIntrLgonTime	msDS-LastSuccessfulInteractiveLogonTime	Specifies the time of the last successful interactive logon	Date
erADLoginScript	scriptPath	Specifies the login script path.	String
erADLoginWorkstations	userWorkstations	Specifies a comma-separated list of addresses or names of workstations from which the user can log in to.	String
erADManager	manager	Specifies the DN of the manager's Active Directory account.	String
erADLyncArchPolicy	Managed via Get-CSUser and Set-CSUser	Specifies the Lync Archive Policy. From supporting data erADLyncArchivingPolicy	String
erADLyncCVPolicy	Managed via Get-CSUser and Set-CSUser	Specifies the Lync Client Version Policy. From supporting data erADLyncClntVerPolicy	String
erADLyncClntPolicy	Managed via Get-CSUser and Set-CSUser	Specifies the Lync Client Policy. From supporting data erADLyncClntVerPolicy	String
erADLyncConfPolicy	Managed via Get-CSUser and Set-CSUser	Specifies the Lync Conferencing Policy. From supporting data erADLyncClntVerPolicy	String
erADLyncDialPolicy	Managed via Get-CSUser and Set-CSUser	Specifies the Lync Dial Plan Policy. From supporting data erADLyncDialPlanPolicy	String
erADLyncExAcPolicy	Managed via Get-CSUser and Set-CSUser	Specifies the Lync External Access Policy. From supporting data erADLyncExtAccPolicy	String
erADLyncLocPolicy	Managed via Get-CSUser and Set-CSUser	Specifies the Lync Location Policy. From supporting data erADLyncLocationPolicy	String
erADLyncMobilityPolicy	Managed via Get-CSUser and Set-CSUser	Specifies the Lync Mobility Policy. From supporting data erADLyncCMobilityPolicy	String
erADLyncPersistentChatPolicy	Managed via Get-CSUser and Set-CSUser	Specifies the Lync Persistent Chat Policy. From supporting data erADCLyncPersistentChatPolicy	String
erADLyncPnPolicy	Managed via Get-CSUser and Set-CSUser	Specifies the Lync PIN Policy. From supporting data erADLyncPinPolicy	String
erADLyncRegPool		Specifies the Lync Registrar Pool. From supporting data erADLyncPool	String
erADLyncSipAdr	msRTCSIP-PrimaryUserAddress	Specfies the primary SIP address	String

Table 13. Attributes, descriptions, and corresponding data types (continued)			
erADLyncTelephony	Managed via Get-CSUser and Set-CSUser	Specifies the Lync Telephony setting 0=PC to PC only 1=Audio/video disabled 2=Enterprise Voice 3=Remote call control 4=Remote call control only	Integer
erADLyncVoicePolicy	Managed via Get-CSUser and Set-CSUser	Specifies the Lync Voice Pool. From supporting data erADLyncVoicPolicy	String
erADLyncEnable	msRTCSIP-UserEnabled	Specifies whether the Lync account is enabled	Boolean
erADLyncLineURI	msRTCSIP-Line	Specifies Lync Line URI	String
erADLyncLineSerURI	msRTCSIP-LineServer	Specifies the Lync Line Server URI	String
erADManager		Specifies the DN of the manager's Active Directory account.	String
erADNamePrefix	personalTitle	Specifies the title of the user, for example Ms. or Mr.	String
erADNameSuffix	generationQualifier	Specifies the name suffix of the user, for example Jr., or III.	String
erADNoChangePassword	*Security descriptor on user object in AD*	Specifies whether the user can change their password.	String
erADOfficeLocations	physicalDeliveryOfficeName	Specifies the office location.	String
erADOtherName	middleName	Specifies an additional name, for example, the middle name, for the user.	String
erADPasswordForceChange	pwdLastSet	Specifies whether to force a password change on next login.	Boolean
erADPasswordLastChange	pwdLastSet	Specifies the last time that the password was changed.	Date
erADPasswordMinimumLength	*Null*	Specifies the minimum length of the password.	Integer
erADPasswordNeverExpires	userAccountControl	Specifies whether a password can never expire.	Boolean
erADPasswordRequired	userAccountControl	Specifies whether the password is required.	Boolean
erADPrimaryGroup	primaryGroupID	Specifies the primary group ID.	String
erADRadiusFramedIPv4Addr	msRASSavedFramedIPAddress		
erADRequireUniquePassword	*Null*	Specifies whether a new password must be different from those passwords in the password history.	Boolean
erADSmartCardRequired	userAccountControl	Specifies whether a smart card is required for login.	Boolean
erADTrustedForDelegation	userAccountControl	Specifies that the user can assign responsibility for management and administration of a portion of the domain namespace to another user, group, or organization.	Boolean

Table 13. Attributes, descriptions, and corresponding data types (continued)			
erADUPN	userPrincipalName	Specifies the principal name for the user account.	String
erADWTSAllowLogon		Specifies whether the user account is allowed to log on to a terminal server.	Boolean
erADWTSBrokenTimeout		Specifies what happens when the connection or idle timers expire or when a connection is lost due to a connection error.	Long
erADWTSCallbackNumber		*Citrix ICA clients* must specify a null-terminated string that contains the phone number to use for callback connections.	String
erADWTSCallbackSettings		*Citrix ICA clients* must specify a value that indicates the configuration for dialup connections in which the terminal server hangs up and then calls back the client to establish the connection. Valid values indicate: 1 - The server prompts the user to enter a phone number, and calls the user back at that phone number. You can use the WtsCallbackNumber value to specify a default phone number. 2 - The server automatically calls the user back at the phone number that is specified by the WtsCallbackNumber value.	Integer
erADWTSClientDefaultPrinter		*RDP 5.0 clients and Citrix ICA clients* must specify whether the client printer is the default printer.	Boolean
erADWTSClientDrives		*Citrix ICA clients* must specify whether the terminal server automatically establishes client drive mappings at login.	Boolean
erADWTSClientPrinters		*RDP 5.0 clients and Citrix ICA clients* must specify whether the terminal server automatically establishes client printer mappings at login.	Boolean

Table 13. Attributes, descriptions, and corresponding data types (continued)			
erADWTSHomeDir		Specifies a null-terminated string for the path of the home directory of the user for terminal server login. This string can specify a local path or a UNC path (\\machine\share\path).	String
erADWTSHomeDirAccessShare		Specifies the user access level to the share on the WTS home directory.	Integer
erADWTSHomeDirDrive		Specifies a null-terminated string for a drive letter to which the UNC path specified in the WtsHomeDir string is mapped	String
erADWTSHomeDirNtfsAccess		Specifies the NTFS access to the home directory.	String
erADWTSHomeDirShare		Specifies the name of a share to create the WTS home directory. Append a dollar sign ($) to create a hidden share.	String
erADWTSInheritInitialProg		Specifies whether the client can specify the initial program. If not set, WtsInitialProgram is the only program that the user can run. The terminal server logs off the user when the user exits that program.	Boolean
erADWTSInitialProgram		Specifies a null-terminated string for the path of the initial program that Terminal Services runs when the user logs in. If the WtsInheritInitialProgram value is 1, the initial program can be any program that is specified by the client.	String
erADWTSProfilePath		Specifies a null-terminated string for the path of the profile of the user for terminal server login.	String

Table 13. Attributes, descriptions, and corresponding data types (continued)			
erADWTSReconnectSettings		Specifies a value that indicates how a disconnected session for a user can be reconnected. Valid values indicate: 0 - The user can log in to any client computer to reconnect to a disconnected session. Sessions started at clients other than the system console cannot be connected to the system console. Sessions started at the system console cannot be disconnected. 1 - The user can reconnect to a disconnected session by logging on to the client computer used to establish the disconnected session. If the user logs on from a different client computer, the user gets a new login session.	Integer
erADWTSRemoteHomeDir		Specifies the home directory of the user on the Windows Server.	String
erADWTSServerName			
erADWTSShadowSettings		*RDP 5.0 clients and Citrix ICA clients* must specify a value that indicates whether the user session can be shadowed. Shadowing allows a user to remotely monitor the on-screen operations of another user.	String
erADWTSTimeoutConnections		Specifies a value that specifies the maximum connection duration, in milliseconds. One minute before the connection timeout interval expires, the user is notified of the pending disconnection. The user session is disconnected or terminated depending on the WtsBrokenTimeout value. Every time the user logs on, the timer is reset. A value of zero indicates that the connection timer is disabled.	String

Table 13. Attributes, descriptions, and corresponding data types (continued)			
erADWTSTimeoutDisconnections		Specifies the maximum duration, in milliseconds, that a WTS retains a disconnected session before the login is terminated. A value of zero indicates that the disconnection timer is disabled.	Integer
erADWTSTimeoutIdle		Specifies the maximum idle time, in milliseconds. If there is no keyboard or mouse activity for the specified interval, the user's session is disconnected or terminated depending on the WtsBrokenTimeout value. A value of zero indicates that the idle timer is disabled.	Integer
erADWTSWorkingDir		Specifies a null-terminated string for the path of the working directory for the initial program	String
erCompany	company	Specifies the name of the company that the user works for.	String
erDepartment	department	Specifies the department within the company to which the user belongs.	String
erDivision	division	Specifies the division within a company (organization) that the employee belongs to.	String
erGroup	memberOf	Specifies names of groups.	String
erLogonTimes	logonHours	Specifies the time periods for each day of the week during which logins are allowed for the user. Represented as a table of Boolean values for the week, each indicating whether that time slot is a valid login time.	Byte array Login time (LT)
erMaxStorage	maxStorage	Specifies the maximum amount of disk space, in KB, that the user can have.	Long
erPassword	*Null*	Specifies the password for the user account.	String
erProfile	profilePath	Specifies the path to the profile of the user.	String
eruid	sAMAccountName	Specifies the user ID.	String
givenName	givenName	Specifies the given name of the user.	String
homePhone	homePhone	Specifies the home telephone number of the user.	String
l	l	Specifies the user's city or location (shown as the lowercase letter 'l').	String

Table 13. Attributes, descriptions, and corresponding data types (continued)			
mail	mail	Specifies the email address of the user.	String
mobile	mobile	Specifies the mobile telephone number of the user.	String
pager	pager	Specifies the pager number of the user.	String
postalCode	postalCode	Specifies the user's postal code for their address	String
postOfficeBox	postOfficeBox	Specifies the user's Post Office Box	String
sn	sn	Specifies the surname of the user.	String
st	st	Specifies the state where the user resides.	String
street	streetAddress	Specifies the street address where the user resides.	String
telephoneNumber	telephoneNumber	Specifies the work telephone number of the user.	String
title	title	Specifies the title of the user.	String

Skype for Business account form attributes

The adapter uses Skype for Business account form attributes.

Table 14. Attributes, descriptions, and corresponding data types		
Directory server attribute	**Description**	**Data type**
erADLyncSipAdr	Specifies SIP address of the user.	String
erADLyncenable	Specifies whether the Skype for Business account is enabled or not for a user.	Boolean
erADLyncMobilityPolicy	Specifies the Mobility Policy of a user.	String
erADLyncPersistentChatPolicy	Specifies the Persistent Chat Policy of the user.	String
erADLyncRegpool	Specifies to which registrar pool user is assigned.	String
erADLyncTelephony	Sets the Telephony for the user. 1. PC to PC only 2. Audio/video disabled 3. Enterprise voice 4. Remote call control 5. Remote call control only	Integer
erADLyncLineUri	Specifies telephone number of the user.	String
erADLyncLineSerUri	Specifies line server URI of user.	String
erADLyncConfPolicy	Specifies the features and capabilities that can be used in a conference.	String
erADLyncCvPolicy	Specifies the policy name that contains information about which client version is able to connect to the Skype for Business Server and also do updates if it is a Skype for Business Client.	String

Table 14. Attributes, descriptions, and corresponding data types (continued)		
Directory server attribute	**Description**	**Data type**
erADLyncPnPolicy	Using this policy, the administrator can control PIN (Personal Identification Number) which can be used instead of user name and password when PIN authentication is enabled.	String
erADLyncExacPolicy	This policy allows an administrator to control if a specific user can communicate with federated organizations, Public IM providers, or access the Skype for Business infrastructure from an external source without VPN.	String
erADLyncArchpolicy	This policy allows the administrator to control the archiving perspective of the communications. The scope can be Internal, External or both to be stored on an SQL database.	String
erADLyncLocPolicy	A location policy contains the settings that define how E9-1-1 is implemented.	String
erADLyncClntPolicy	Specifies client-related settings.	String
erADLyncDialpPolicy	Specifies dial plan policy of user.	String
erADLyncVoicePolicy	Specifies calling features that can be enabled or disabled and public switched telephone network (PSTN) usage records.	String

Adapter attributes by operations

Adapter attributes by operations refer to adapter actions by their functional transaction group. They are required for customization, creating provisioning rules, and understanding what service/target attributes are supported by the adapter.

System Login Add

A System Login Add is a request to create a user account with the specified attributes.

Table 15. Add request attributes	
Required attribute	**Optional attribute**
erUid	All other supported attributes

System Login Change

A System Login Change is a request to change one or more attributes for the specified users.

Table 16. Change request attributes	
Required attribute	**Optional attribute**
erUid	All other supported attributes

System Login Delete

A System Login Delete is a request to remove the specified user from the directory.

Table 17. Delete request attributes

Required attribute	Optional attribute
erUid erEntProfileType erEntUserState erEntUserDN	All other supported attributes

System Login Suspend

A System Login Suspend is a request to disable a user account.

The user is neither removed nor are their attributes modified.

Table 18. Suspend request attributes

Required attribute	Optional attribute
erUid erEntProfileType	None

System Login Restore

A System Login Restore is a request to activate a user account that was previously suspended.

After an account is restored, the user can access the system using the same attributes as the ones before the Suspend function was called.

Table 19. Restore request attributes

Required attribute	Optional attribute
erUid erEntProfileType	None

Reconciliation

The Reconciliation function synchronizes user account information between IBM Security Verify Governance Identity Manager and the adapter.

Table 20. Reconciliation attributes

Attributes returned during reconciliation
All supported attributes

Special attributes

Certain attributes have special syntax and meaning that customers needs to be aware off. This information will be used to help the customer in how to supply the attribute value.This topic is not applicable for this adapter.

Files

You can configure several adapter-specific files.

This appendix includes information about the files that are associated with the Active Directory Adapter:

- "schema.dsml file" on page 102
- "CustomLabels.properties file" on page 105

schema.dsml file

The `schema.dsml` file contains all of the attributes that are common to all adapters.

This common file also contains Identity server attributes that can be used by any adapter. The `schema.dsml` file defines all of the classes used by the adapter. The classes are used to declare accounts, services, and supporting data.

The `schema.dsml` file defines the attributes and objects that the adapter supports and uses to communicate with the Identity server. All attributes must be unique; therefore, they are assigned an OID.

The OID is defined with the `<object-identifier>...</object-identifier>` tags.

The `schema.dsml` file has the following format:

```
SCHEMA.DSML File
<?xml version="1.0" encoding="UTF-8"?>

<!-- ********************************************************* -->
     <!-- Schema supported by the Windows adapter.      -->
<!-- ********************************************************* -->
<directory-schema>        ...
<!-- ********************************************************* -->
     <!-- eraADString1-->
<!-- ********************************************************* -->
<attribute-type single-value="true">
<name>erADString1</name>
<description/>
<object-identifier>1.3.6.1.4.1.6054.3.125.2.100</object-identifier>
<syntax>1.3.6.1.4.1.1466.115.121.1.15</syntax>
</attribute-type>
<!-- ********************************************************* -->
     <!-- erADInteger-->
<!-- ********************************************************* -->
<attribute-type single-value="true">
<name>erADInteger</name>
<description/>
<object-identifier>1.3.6.1.4.1.6054.3.125.2.101</object-identifier>
<syntax>1.3.6.1.4.1.1466.115.121.1.27</syntax>
</attribute-type>
<!-- ********************************************************* -->
     <!-- erADDate-->
<!-- ********************************************************* -->
```

```
<attribute-type single-value="true">
<name>erADDate</name>
<description/>
<object-identifier>1.3.6.1.4.1.6054.3.125.2.102</object-identifier>
<syntax>1.3.6.1.4.1.1466.115.121.1.24</syntax>
</attribute-type>
<!-- ********************************************************* -->
     <!-- erADBoolean-->
<!-- ********************************************************* -->
<attribute-type
single-value="true">
```

```
<name>erADBoolean</name>
<description/>
<object-identifier>1.3.6.1.4.1.6054.3.125.2.103</object-identifier>
<syntax>1.3.6.1.4.1.1466.115.121.1.7</syntax>
</attribute-type>
<!-- ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ -->
      <!-- erADMultiValueString-->
<!-- ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ -->
<attribute-type>
<name>erADMultiValueString</name>
<description>List of string values</description>
<object-identifier>1.3.6.1.4.1.6054.3.125.2.104</object-identifier>
<syntax>1.3.6.1.4.1.1466.115.121.1.15</syntax>
</attribute-type>       ...
<!-- ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ -->
     <!-- erADAccount Class           -->
<!-- ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ -->
<class superior="top">
<name>erADAccount</name>
<description>Windows account.</description>
<object-identifier>1.3.6.1.4.1.6054.3.125.1.1</object-identifier>   ...
<attribute ref="erADBoolean" required="false"/>
<attribute ref="erADDate" required="false"/>
<attribute ref="erADInteger" required="false"/>
<attribute ref="erADMultiValueString" required="false"/>
<attribute ref="erADString1" required="false"/>
</class>       ...
</directory-schema>
</dsml>
```

The sections of this schema file are described in the following sections.

Object identifier

The IBM Security Verify Identity server uses LDAP directory services to add, delete, modify, and search IBM Security Verify Identity data.

Each data item in an LDAP directory server must have a unique OID. Each attribute and class that is defined in the `schema.dsml` file in IBM Security Verify Identity has an OID.

OIDs have the following syntax:

```
enterprise ID.product ID.adapter ID.object ID.instance ID
```

The *enterprise ID* is always 1.3.6.1.4.1.6054 for IBM.

The *product ID* is always 3 because these `schema.dsml` files are used with adapters.

The *adapter ID* is 125 for the Active Directory Adapter.

The *object ID* is 2 . An attribute uses 2 as the object ID.

The *instance ID* is a sequential number of the object.

Attribute definition

Before you define unique attributes for the adapter, ensure that the attribute does not exist in the common `schema.dsml` file.

The following example defines an attribute:

```
<!-- ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ -->
<!-- erSampleHome                                        -->
<!-- ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ -->
    <attribute-type single-value = "true" >
        <name>erSampleHome</name>
        <description>User home directory</description>
        <object-identifier>1.3.6.1.4.1.6054.3.125.2.100</object-identifier>
        <syntax>1.3.6.1.4.1.1466.115.121.1.15</syntax>
    </attribute-type>
```

Comment lines are denoted by the `<!— ... —>` markers.

The attribute type is defined as single-value or multivalue. A single-value attribute is denoted by the line: `<attribute-type single-value ="true">`. To denote a multivalued attribute, change the `true` value to `false`.

The name of the attribute that is used by the Identity server is defined in the schema. To simplify the tracking of new Active Directory Adapter attributes, use *erAD* as the preface for all new attributes, so that they can be easily identified in your Windows Active Directory. When attributes have already been defined in the Windows Active Directory, and they do not conflict with existing attributes, they can be used without changing their names.

The description of the attribute is denoted by the `<description>...</description>` tags.

The OID is defined using the `<object-identifier>...</object-identifier>` tags. Because OIDs are already assigned to the existing, standard attributes, the OID can be copied from the last attribute in the list. Increment the last number by a value of one for each new attribute that you add to the `schema.dsml` file.

The data type is defined using the `<syntax>...</syntax>` tags. The following table lists various data types and the value that you specify in the syntax tags.

Table 21. Data types and values for syntax tags

Data Type	**Value**
Bit string	1.3.6.1.4.1.1466.115.121.1.6
Boolean	1.3.6.1.4.1.1466.115.121.1.7
Directory String	1.3.6.1.4.1.1466.115.121.1.15
UTC Coded Time	1.3.6.1.4.1.1466.115.121.1.24
Integer	1.3.6.1.4.1.1466.115.121.1.27

Classes

At least one account class and one service class must be defined in the `schema.dsml` file.

Each class requires at least one attribute to identify the class: a name attribute. Additional attributes might be required depending on the class defined.

The following syntax defines a class:

```
<class superior="top">
    <name> ... </name>
    <description> ... </description>
    <object-identifier> ... </object-identifier>
    <attribute ref = "..." required = "true" />
    <attribute ref = "..." required = "true" />
</class>
```

To make an attribute optional for a class, change `required = "true"` to `required = "false"` in the `<attribute ref>` tag.

An account class defines the attributes that are used to describe an account. An account class must be defined in the `schema.dsml` file.

The following example defines an account class:

```
<class superior="top" >
    <name>erSampleAccount</name>
    <description>Sample Account</description>
    <object-identifier>1.3.6.1.4.1.6054.3.125.1.101</object-identifier>
    <attribute ref = "eruid" required = "true" />
    <attribute ref = "erAccountStatus" required = "false" />
```

```
    <attribute ref = "erSampleGroups" required = "false" />
    <attribute ref = "erSampleHome" required = "false" />
    <attribute ref = "erSampleDesc" required = "false" />
    <attribute ref = "erPassword" required = "false" />
</class>
```

In this example, the class name is `erSampleAccount` and the only required attribute is `eruid`. However, note that `erAccountStatus` is a required attribute to suspend or restore accounts.

CustomLabels.properties file

The `CustomLabels.properties` file is a text file that defines the labels on the form for the adapter.

The syntax for the information in the file is:

```
attribute=text
```

where *attribute* is the same attribute defined in the `schema.dsml` file and *text* is the label that appears on the form in the IBM Security Verify Governance Identity Manager user interface for the account.

The *attribute* must be in lowercase. This requirement comes from the Identity server.

Index

D

E

F

G

I

K

L

M

N

O

P

R

S

T

U

V

W