# HP OpenView Select Identity

## Connector for Microsoft® Windows® Active Directory and Exchange

Connector Version: 3.7

## Installation and Configuration Guide

Document Release Date: March 2006
Software Release Date: March 2006

# Legal Notices

## Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

## Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

## Copyright Notices



This product includes software developed by the Apache Software Foundation (http://www.apache.org/). Portions Copyright (c) 1999-2003 The Apache Software Foundation. All rights reserved.

Select Identity uses software from the Apache Jakarta Project including:

- Commons-beanutils
- Commons-collections
- Commons-logging
- Commons-digester
- Commons-httpclient
- Element Construction Set (ecs)
- Jakarta-poi
- Jakarta-regexp
- Logging Services (log4j)

Additional third party software used by Select Identity includes:

- JasperReports developed by SourceForge
- iText (for JasperReports) developed by SourceForge
- BeanShell
- Xalan from the Apache XML Project
- Xerces from the Apache XML Project
- Java API for XML Processing from the Apache XML Project
- SOAP developed by the Apache Software Foundation
- JavaMail from SUN Reference Implementation
- Java Secure Socket Extension (JSSE) from SUN Reference Implementation
- Java Cryptography Extension (JCE) from SUN Reference Implementation
- JavaBeans Activation Framework (JAF) from SUN Reference Implementation

- OpenSPML Toolkit from OpenSPML.org
- JGraph developed by JGraph
- Hibernate from Hibernate.org
- BouncyCastle engine for keystore management, bouncycastle.org

This product includes software developed by Teodor Danciu http://jasperreports.sourceforge.net). Portions Copyright (C) 2001-2004 Teodor Danciu (teodord@users.sourceforge.net). All rights reserved.

Portions Copyright 1994-2004 Sun Microsystems, Inc. All Rights Reserved.

This product includes software developed by the Waveset Technologies, Inc. (www.waveset.com). Portions Copyright © 2003 Waveset Technologies, Inc. 6034 West Courtyard Drive, Suite 210, Austin, Texas 78730. All rights reserved.

Portions Copyright (c) 2001-2004, Gaudenz Alder. All rights reserved.

### Trademark Notices

AMD and the AMD logo are trademarks of Advanced Micro Devices, Inc.

Intel and Pentium are trademarks or registered trademarks of Intel Corporation in the United States, other countries, or both.

JAVA™ is a US trademark of Sun Microsystems, Inc.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

Oracle® is a registered US trademark of Oracle Corporation, Redwood City, California

UNIX® is a registered trademark of The Open Group.

## Support

Please visit the HP OpenView support web site at:

**http://www.hp.com/managementsoftware/support**

This web site provides contact information and details about the products, services, and support that HP OpenView offers.

HP OpenView online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit enhancement requests online
- Download software patches
- Submit and track progress on support cases
- Manage a support contract
- Look up HP support contacts
- Review information about available services
- Enter discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and log in. Many also require a support contract.

To find more information about access levels, go to:

**http://www.hp.com/managementsoftware/access_level**

To register for an HP Passport ID, go to:

**http://www.managementsoftware.hp.com/passport-registration.html**

# Contents

# 1 Installing the Connector

The Windows Active Directory connector enables HP OpenView Select Identity to provision users on Windows Active Directory systems. Because Microsoft Exchange relies on Active Directory for storing user data, you can also use this connector to provision user mailboxes in Exchange.

Due to a known Active Directory limitation, events are not generated when some attributes are modified on Active Directory 2003. See Operations Supported by the Connector on page 7 for the list of attributes for which events are generated.

The Windows Active Directory connector is a two-way connector and pushes user changes made in the Select Identity database to the target Windows Active Directory server. It also enables the Select Identity agent on the Active Directory Domain Controller to provision users in Select Identity based on changes made in Active Directory.

The Windows Active Directory connector is packaged in the following files, which are located in the `Active Directory & Exchange` folder on the Select Identity Connector CD:

- `ADConnector.rar` — contains the binaries for the connector.
- `ADSchema.jar` — contains the following mapping files, which control how Select Identity fields are mapped to Active Directory fields:
  - `aduser.properties` — maps the Select Identity user attributes to the Active Directory user attributes.
  - `adgroup.properties` — maps the Select Identity group attributes to Active Directory group attributes. Note that group provisioning is not currently supported, though this file must be extracted during installation.
  - `adcomputer.properties` — maps the Select Identity computer attributes to the Active Directory attributes. Note that computer provisioning is not currently supported, though this file must be extracted during installation.
  - `activedirectory.xsl` — maps attributes on the Windows server to attributes on the Select Identity server. This file is used by the agent during reverse synchronization.
- `ADSetup.zip` — contains the installation executable for the Active Directory agent.

## Operations Supported by the Connector

The Windows Active Directory connector enables Select Identity to perform the following provisioning tasks on Windows Active Directory systems:

- Add, update, and remove users
- Retrieve user attributes
- Enable and disable users

- Verify a user's existence
- Change user passwords
- Reset user passwords
- Expire user passwords
- Retrieve all entitlements
- Retrieve a list of supported user attributes
- Assign and unassign entitlements to and from users
- Provision user mailboxes in Exchange 2000 and Exchange 2003

When the connector adds a user to the Active Directory resource, the user is assigned to a default group called "Domain User." Do not use this group as an entitlement; you cannot remove this group from the user.

The Select Identity agent can also send changes made on the Windows system to Select Identity. This is called **reverse synchronization**. Additional configuration steps are required to enable reverse synchronization, as described later in this chapter and guide.

The updates made to Select Identity data depend on whether the Windows system is an authoritative or non-authoritative resource:

| Operation | If the Resource is Authoritative | If the Resource is Non-authoritative |
|---|---|---|
| User is added on the resource. | The user is added to the respective Service. | User is not added. However, if the user exists, the entitlements are modified (not the user attributes). |
| User attributes are modified on the resource. | The user attributes are updated in Select Identity. | The user attributes are not updated in Select Identity. |
| User entitlements are modified on the resource. | The entitlements are modified in Select Identity. | The entitlements are modified in Select Identity. |
| User is deleted on the resource. | The user's Service membership is deleted in Select Identity. | The user is not deleted. in Select Identity, though the entitlements for the resource are deleted. |
| Password is changed on the resource. | The user's password is reset in all Services for which the user is registered. | The user's password is reset in all Services for which the user is registered. |
| Move user to a different OU | The value of the UserSuffix attribute is updated in Select Identity to the new OU. | The value of the UserSuffix attribute is not updated in Select Identity. |

# System Requirements

This section lists the requirements for the system where the connector and agents will be installed.

## For the Connector

The Windows Active Directory connector is supported in the following environment:

| Select Identity Version | Application Server | Database |
|---|---|---|
| 3.0.2 | WebLogic 8.1.2 on Windows 2003 | SQL Server 2000 |
| | WebLogic 8.1.2 on Solaris 9 | Oracle 9i |
| | WebLogic 8.1.2 on HP-UX 11i | Oracle 9i |
| | WebSphere 5.1.1 on Solaris 9 | DB2 8.2 (or DB2 8.1 Service Pack 7) |
| 3.3 | WebLogic 8.1.4 on Windows 2003 | SQL Server 2000 |
| | WebLogic 8.1.4 on Solaris 9 | Oracle 9i |
| | WebLogic 8.1.4 on Red Hat Enterprise Linux 3.0 | SQL Server 2000 |
| 3.3.1 | WebLogic 8.1.4 on Windows 2003 | SQL Server 2000 |
| | WebSphere 5.1.1 on HP-UX 11i | Oracle 9i |
| 4.0 | The Active Directory connector is supported on all the platform configurations of Select Identity 4.0. | |

This connector supports Active Directory on Windows 2000 and 2003. It can also provision user mailboxes in Exchange 2000 and 2003.

Also, if the server and resource machines communicate across a firewall, they must allow bidirectional TCP flow on port 5001 (this can be configured on any other port, as well).

## For the Agent

The agent provided with this connector is supported in the following environment:

| | |
|---|---|
| **Operating system** | Microsoft Windows 2000 Server, Service Pack 4 or later, and Windows 2003 Server (see note below). |
| **ADSI version** | Version 5,0,00,0. See page 12 for information about determining the version of ADSI. The agent uses ADSI to query Active Directory during the reverse synchronization process. |
| **Browser version** | Internet Explorer 5.5 or later (supporting MSXML 2.0 or later). |
| **Winsock version** | Version 2.0 or later. |

The agent must be installed on a Primary Domain Controller (PDC), which is actually a PDC emulator for Active Directory 2000 and 2003. You must install on a PDC because the the agent uses system event logs to monitor changes.

In a multi-domain or multi-master environment, the agent should also be installed on additional domain controllers if changes done on those systems must be reconciled with Select Identity. If the agent is installed only on the PDC, changes made on the other domain controllers are replicated to the PDC but security logs are not generated and changes are not captured by the agent.

## Ports

The following are the required ports for configuring the connector and agent:

| Port | Protocol | Direction | Description |
|---|---|---|---|
| 389 or 636 (SSL) | MS-AD APIs | Agent → AD | AD listening port |
| 8090 (configurable) | Custom | Connector → Agent | Allows communication from the connector to the agent on the AD server |
| 7001 (or other port on which the Select Identity server is listening) | HTTP or HTTPS | Agent → Select Identity server | Reverse synchronization |

# Deploying on the Application Server

To install the Active Directory connector on Select Identity, you must deploy the connector on the application server. To deploy the connector on a application server, complete the following tasks:

1. Create a subdirectory in the Select Identity home directory where you can store the connector's Resource Adapter Archive (.rar) file.
   For example, you can create `<OVSI_HOME_DIR>/connectors`
   where `<OVSI_HOME_DIR> = /opt/Select_Identity` in Linux
   and `<OVSI_HOME_DIR> = C:\Select_Identity` in Windows (A connector subdirectory may already exist.)
2. Copy the `ADConnector.rar` file from the Select Identity Connector CD to the connector subdirectory.
3. Perform the following steps to deploy the connector on WebLogic. If deploying on WebSphere, skip to step 4 on page 11.
   a. Create a schema subdirectory in the Select Identity home directory where you can store the connector's mapping files.
      For example, you can create `<OVSI_HOME_DIR>/Schema`

where <OVSI_HOME_DIR> = /opt/Select_Identity in Linux and <OVSI_HOME_DIR> = C:\Select_Identity in Windows (A schema subdirectory may already exist.)

b Extract the contents of the ADSchema.jar file (on the Select Identity Connector CD) to the schema subdirectory. Ensure that the CLASSPATH environment variable in the WebLogic server startup script refers to the schema subdirectory.

c Start the application server if it is not currently running, and log on to the WebLogic Server Console.

d In the left pane, expand Deployments folder, and then right click on Connector Modules, and select **Deploy a New Connector Module**.
Alternatively, at the right-hand panel of the Server Console homepage, click on **Connector Modules** link, which is under Your Deployed Resources column of Domain Configurations section. Resource Connectors page appears. Click on **Deploy a New Connector Module** link on this page.

e Click the link in the Location field, locate, and select the ADConnector.rar file from the list. It is stored in the connector subdirectory.

f Click **Target Module**.

g If only one server is configured, skip to next step. If more than one server is configured, the next page prompts you to select the servers on which you want to deploy the connector. Select the server instance (for instance, My Server), and then click **Continue**.

h Review the settings. Keep all the default settings and click **Deploy**. The Status of Last Action column should display Success.

4 If you want to deploy the connector on WebSphere, perform the following steps:

a Stop the application server.

b Extract the contents of the ADSchema.jar file (on the Select Identity Connector CD) to WebSphere\AppServer\lib\ext, and then start the application server.

c Start the application server.

d Log on to the WebSphere Application Server Console.

e Navigate to **Resources** → **Resource Adapters**.

f Click **Install RAR**.

g In the Server path field, enter the path to the ADConnector.rar file. It is stored in the subdirectory created step 1.

h Click **Next**.

i In the Name field, enter a name for the connector, and then click **OK**.

j Click the **Save** link (at the top of the page).

k On the Save to Master Configuration dialog, click **Save**.

l Click **Resources** → **Resource Adapters**.

m Click the new connector.

n Click **J2C Connection Factories** in the Additional Properties table.

o Click **New**.

p In the Name field, enter the name of the factory for the connector. For this connector, enter **eis/AD**.

q Click **OK**.

r Click the **Save** link.

s On the Save to Master Configuraton dialog, click **Save**.

a Restart WebSphere.

To configure reverse synchronization on the server, extract the `activedirectory.xsl` file from the `ADSchema.jar` file to the Select Identity home directory. This file maps user attributes on the Windows server to attributes in Select Identity.
Because the attributes in the `activedirectory.xsl` file are based on those in the `aduser.properties` and `adgroup.properties` files, you must modify the `activedirectory.xsl` file to reflect changes made to these files..

After installing the connector, refer to Configuring the Connector on page 29 for information about registering and configuring this connector in Select Identity.

# Installing the Agent on the Windows Server

After you install the Windows Active Directory connector on the Select Identity server, you can install the agent on the Windows system. The agent is a suite of Services and support DLLs deployed on the resource.

You MUST obtain the administrative user name and password to log on to the system during the installation.

## Determining the Version of ADSI

To determine the version of ADSI, review the following key in the registry:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Active Setup\
Installed Components\{E92B03AB-B707-11d2-9CBD-0000F87A369E}

The following table describes ADSI versions with the values that may be found in this registry key:

| Version | Value |
|---|---|
| Earlier than 2.5 | N.A. |
| 2.5 | 2,5,00,0 |
| Windows 2000 | 5,0,00,0 |
| DSClient | 5,0,00,0 |

Versions earlier than ADSI 2.5 do not create this registry key. If this key is not present, look then for the following key:

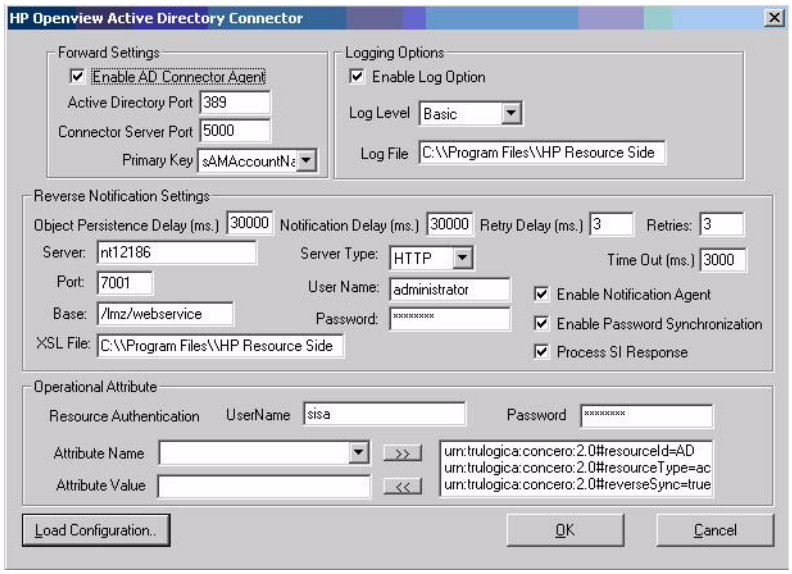
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Ads

If this key is present, ADSI 2.0 is installed. If this key is not present, it may be an improper installation, or no ADSI is installed at all.

## Installing the Agent

Perform the following to install the agent:

1. Copy the `ADSetup.zip` file from the Select Identity Connector CD to a folder on the Windows Active Directory server.
2. Extract the `ADSetup.zip` file.
3. Double-click `SETUP.exe` to start the installation program.
4. Click **Next** to proceed through the installation.
5. If needed, provide administrative logon information when prompted.
6. Configure the Windows Active Directory agent options. The configuration is defined on the HP Openview Active Directory Connector dialog.




   a. Select the **Enable AD Connector Agent** check box. This starts the connector, enabling it to receive provisioning requests from Select Identity.

   b. In the Active Directory Port field, enter the number of the Active Directory listening port, such as 389.

   c. Enter a port number in the Connector Server Port field. The connector uses this port to communicate with the agent. The default is 5001.

   d. There are three primary keys. This key is used to search the user on the resource. Choose only the default one, which is `sAMAccountName`. `sAMAccountName` works for all the functionalities, in forward and reverse provisioning. The other two keys, `distinguishedName` and `userPrincipalName`, do not support `resetPassword` and `deleteuser` operations on reverse synchronization.

   e. Select the **Enable Log Option** check box to enable logging for the agent. Then, configure the following logging options:

      – Select the depth of logging from the Log Level drop-down list. The levels include Basic, Intermediate, Advanced, and Developer, where Developer is the most verbose level.

– Specify where the log file will reside in the Log File field. The default value is *install_dir*\Logs.

7 Configure the following settings for reverse synchronization. Perform these steps if you want to synchronize changes made to users on the Windows server with Select Identify.

a Select the **Enable Notification Agent** option.

b If you want to synchronize the Windows server password with Select Identity, select **Enable Password Synchronization**. This is used by the agent to synchronize user account password changes with Select Identity. The information is sent back to Select Identity in the form of an SPML extendedRequest over SOAP/HTTP or HTTPS.

c In the Object Persistence Delay field, enter the length of time (in milliseconds) that an object is persisted in Active Directory before the user details are retrieved by the agent. During reverse synchronization when a user attribute is modified, there is a delay before the change is persisted in Active Directory. However, the event is generated before this. The Object Persistence Delay controls the agent and forces a pause before it retrieves the details related to the corresponding event. The optimum value is 1000 milliseconds.

d In the Notification Delay field, enter the number of milliseconds between requests sent to Select Identity. The optimal value is 5000 milliseconds.

e In the Server field, enter the IP address or fully-qualified name of the server running Select Identity.

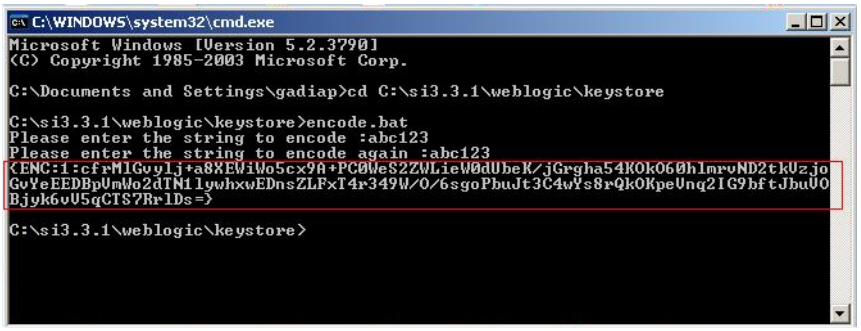
f In the Port field, enter the port on which Select Identity listens for reverse synchronization requests.

g Enter the base URL for the Select Identity Web Service in the Base field.

h Select **HTTP** or **HTTPS** from the Server Type drop-down list. This defines the protocol for transfer of data back to Select Identity.

i Enter the name of a user that has administrative privileges on the Windows server in the User Name field. This specifies the user under which the agent will run. Specify the Select Identity administrator or another administrative user on the sytem.

j Enter the password in the Password field. To encrypt the password, run `encode.bat` (on Windows) or `encode.sh` (on UNIX), which is provided in the `weblogic/keystore` subdirectory in the Select Identity home directory. This utility prompts you for the password to encrypt and will generate the encrypted password. Be sure to copy the entire encrypted password in the field, as shown here:




k Keep the Time Out, Retries, and Retry Delay settings:

– The Time Out field specifies the number of milliseconds after which the request times out.

- The Retries field specifies the number of retries that the agent will attempt to send the SPML request (which is useful, for example, if Select Identity was unavailable during the original request; the agent retries the request according to these settings after Select Identity is available again).
- The Retry Delay setting is used to specify the delay between multiple retries.

l In the UserName field (in the Operational Attribute section), enter the name of the administrator account in Select Identity.

m Enter the password of the administrative account in Select Identity in the Password field.

n Edit the following operational attributes. This builds the operational attributes that are sent in SPML requests back to Select Identity for synchronization. Click the **<<** button after each attribute and edit the value to specify the desired value, then click the **>>** button.

— Attribute Name: **urn:trulogica:concero:2.0#resourceId**

Attribute Value: ***resource_name***

This is the name of the resource that you add in Select Identity for this Active Directory server. For example, if you specify **AD_Exchange** here, then specify **AD_Exchange** as the resource name in Select Identity.

— Attribute Name: **urn:trulogica:concero:2.0#reverseSync**

Attribute Value: **true**

— Attribute Name: **urn:trulogica:concero:2.0#resourceType**

Attribute Value: **activedirectory**

This is the name of the XSL file (without the .xsl extension), which provides reverse mappings for the agent to send data back to Select Identity.

o Click the **Load Configuration** button to load all values in the console from a properties file (instead of entering the values in the console). For example, you could enter the values for the attributes in a .properties file in the following format:

```
PSNotify_EIS_Port=389
PSLog_Enabled=1
PSLog_Level=2
PSPassFilt_Enabled=1
PSSync_Server_Name=ps0111
PSSync_Server_Port=80
PSSync_Timeout=400
PSSync_Retries=8
PSSync_Retries_Delay=3000
PSSync_Server_Secure=0
PSSync_Server_Username=Administrator
PSSync_Server_Password=Trulogica
PSSync_Request_Delay=5000
PSConnector_Port=5000
PSSync_Server_BaseURL=/lmz/webservice
PSSync_Res_Username=sisa
PSSync_Res_Password=abc123
PSMap_Path=C:\\Program Files\\HP Openview\\HP Openview
ADConnector\\Map\\agent.properties
PSLog_Path=C:\\Program Files\\HP Openview\\HP Openview
ADConnector\\Logs
```

8 After defining all of your settings, click **OK**.

9 After the installation is complete, click **Finish**.

10 Restart the Windows server.

The installation process performed the following:

- Created the target folder with the binaries and support files in the appropriate folders. Placed `TLPassfilt.dll` and `TLUtils.dll` in the Windows System folder, $WinSysPath$ (`c:\winnt\system32`). The following folder structure was created:
  - `<TARGETDIR>` — The parent folder
  - `<TARGETDIR>\Bin` — Program binaries, including the `TLSuiteConfig.exe` file
  - `<TARGETDIR>\Lib` — Library files, including the `ADConnector.dll`, `TLConsole.dll`, and `UninstAD.dll` files
  - `<TARGETDIR>\Logs` — Connector log folder
  - `<TARGETDIR>\Map` — Mapping of operational attributes, including the `agent.properties`, TLADNotify.exe, and TLServer.exe files
  - `<TARGETDIR>\Servers` — Server binaries
- Created and configured corresponding services.
- Created a Program group and shortcuts for the connector configuration console and the uninstallation script.
- Set up the registry for program parameters.

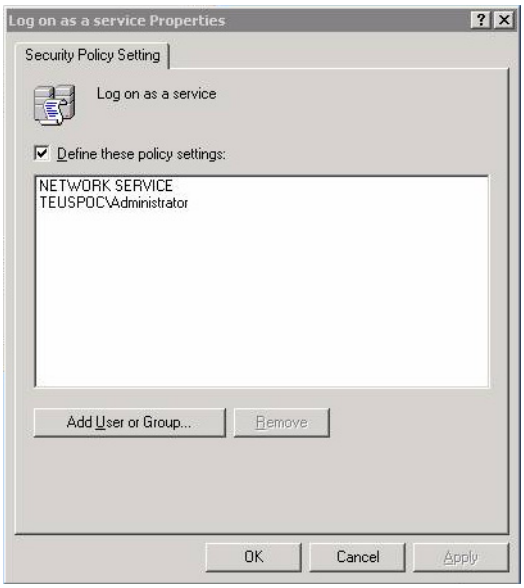
## Configuring a User for the Agent's Service

By default, the agent logs on as the Local System account on the Active Directory server. However, if the server reboots, the agent's service is not automatically started; the Local System account does not have permission to restart the agent's service. To ensure that the agent is automatically restarted after reboot, you can create a user for the agent and configure that user to automatically restart the service. Complete the following steps to do so, and refer to Windows documentation or your system administrator for details on each step:

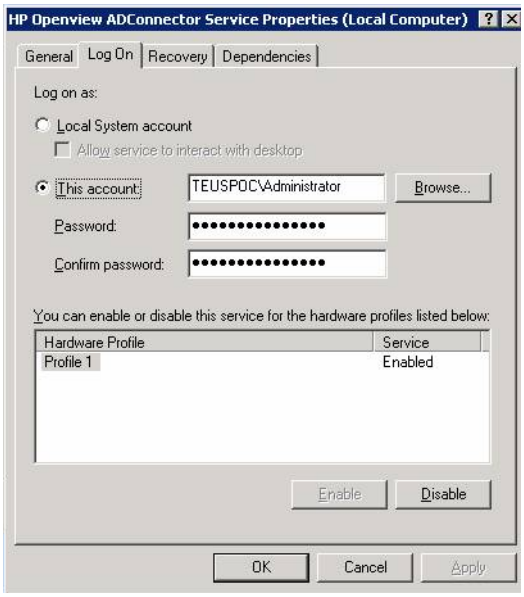
1 Go to Active Directory Users and Computers Console window from **Start** → **Programs** → **Administrative Tools** → **Active Directory Users and Computers** and create or identify a user on the Active Directory server that can be assigned as the Log On As user for the agent. You must have administrative permissions on the system to create a user.

2 Update the local security policy to allow the new user to run as a service. Set this policy from the Default Domain Controller Security Settings window through **Start**→**Programs** → **Administrative Tools** → **Default Domain Controller Security** → **Security Settings** → **Local Policies** → **User Rights Assignment** → **Log on as a service**. The following snapshot illustrates that

the TEUSPOC\Administrator user, which is the user created for the agent, is granted permission to log on as a service:



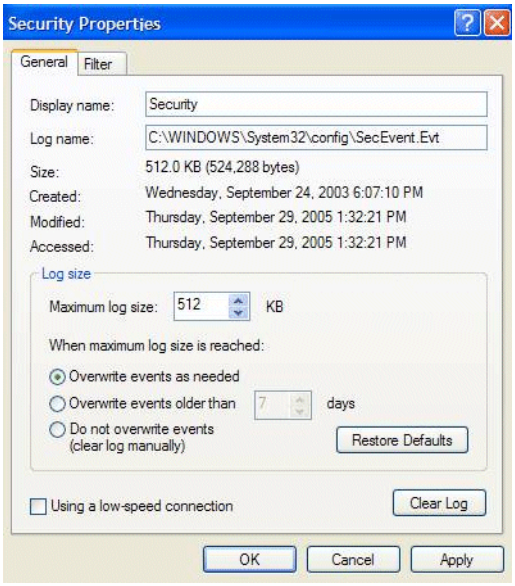

3 Configure the HP Openview ADConnector Service and ADNotification Service, which are installed with the agent, to use the newly created user as its Log On As user. Perform this step from the Services windows, which is accessible from the Administrative Tools window. In the following snapshot, the TEUSPOC\Administrator user is assigned to the HP Openview ADConnector Service:




# Configuring Active Directory to Support Reverse Synchronization

If you configured the agent to support reverse synchronization in step 7 on page 14, complete the following to enable Active Directory to support reverse synchronization:

- Verify that the **When maximum log size is reached: Overwrite Events as needed** option is enabled in the Security Log properties on the Windows system. To view this configuration, select **Start** → **Settings** → **Control Panel**, double-click **Administrative Tools**, then double-click **Event Viewer**. Right-click **Security Log** and select **Properties**.




- If you installed the agent on a Windows 2000 Server (Primary Domain Controller or Backup Domain Controller), you must enable strong password enforcement. To do so, select **Start** → **Settings** → **Control Panel** → **Administrative Tools** → **Domain Controller Security Policy** → **Security Settings**. Expand the Account Policies folder and double click **Passwords must meet complexity requirements**. Select the **Enable** option and click **OK**.
- Set the Audit Account Management and Audit Directory Service Access policies to Success in the Audit Policy. (This must be done for the Default Domain Controllers Group Policy.) To do so, launch the Control Panel, double-click **Administrative Tools**, then double-click **Domain Controller Security Policy**. Select **Local Policies** → **Audit Policy**. Right-click **Audit account management**, select **Security...**, and select the **Success** check box.

  Repeat this step for **Audit directory service access**.

  After configuring the policies, enter the following command in a Command Prompt window:

  **`gpupdate /force`**

- Configure the Access Control Lists (ACLs) to enable auditing by completing these steps:

  These steps are needed for Active Directory 2003 only. You can skip these steps on Active Directory 2000.

  a. Select **Start** → **Settings** → **Control Panel**, double-click **Administrative Tools**, then double-click **Active Directory Users and Computers**.

  b. Select **View** → **Advanced Features** .

  c. Right-click the Active Directory object that you want to audit, then click **Properties**.

  d. Click the **Security** tab, then click **Advanced**.

  e. Click the **Auditing** tab, then click **Add**.

  f. Enter the name of the user or group whose access you want to audit in the Enter the object name to select field, then click **OK**. (You can enter **`Everyone`**.)

g In the list of names, double-click the user or group whose access you want to audit.

h Select the **Successful** check box or the **Failed** check box for the actions that you want to audit. You can also select **Success** or **Failure** for the Write all properties option. Then, click **OK**.

i Click **OK**, then click **OK**.

# 2 Configuring the Connector

After you deploy the connector on the application server, you must configure Select Identity to use the connector by deploying it in the Select Identity client.

## Connector Deployment

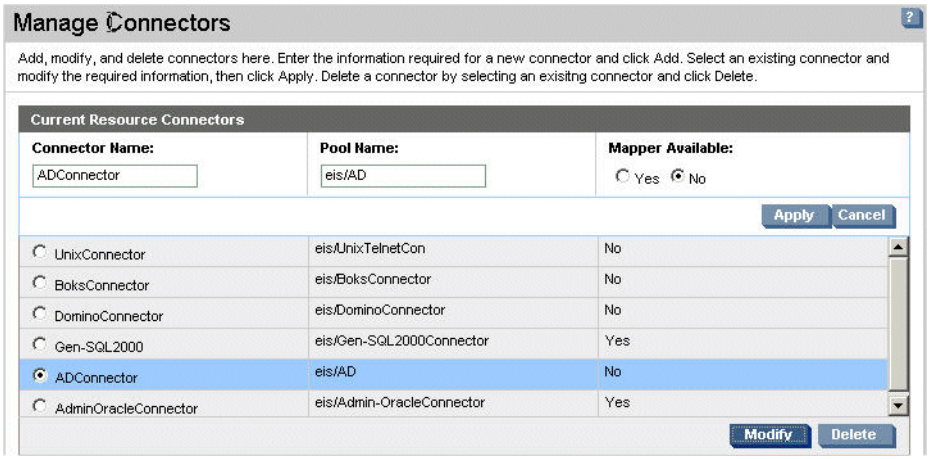
Complete the following steps to deploy and configure the connector with Select Identity:

- Register a new connector with Select Identity.
- Add a resource to Select Identity.
- Link the resource attributes to the Select Identity attributes.
- Create a service that uses the newly created resource

1 To register the connector with Select Identity, perform the following steps:

a On Select Identity home page, click **Service Studio** → **Resources**. The Resources page appears.

b Click **Manage Connectors** on Resources page and add the connector. Refer to *Service Studio* chapter of *HP OpenView Select Identity Administrator Guide* for more information. Select Identity displays the connector properties in the following format:




2 To deploy a resource that uses the newly created connector, perform the following steps:

a Click **Service Studio** → **Resources.** Resources page appears.

b Click **Add New Resource**. When configuring the resource, refer to the following table for parameters specific to this connector:

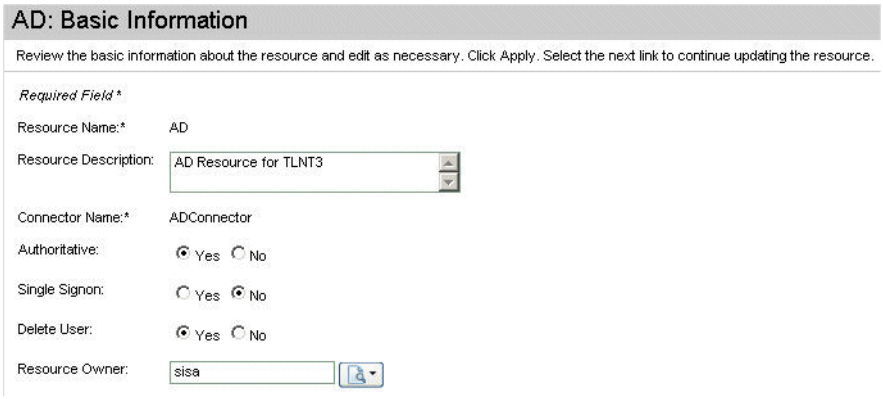
| Field Name | Sample Values | Description |
|---|---|---|
| Resource Name | AD_Exchange | Name given to the resource. If you enabled reverse synchronization, this must be the same as the value provided for the urn:trulog-ica:concero:2.0#resourceId attribute on the agent console. |
| Resource Type | ADConnector | The connector that was deployed in step 1 on page 21. |
| Authoritative Source* | No | Whether this resource is a system that is considered to be the authoritative source for user data in your environment. Specify **No** if the connector is not enabled for reverse synchronization. Specify **Yes** if you want to add users through reverse synchronization. If the resource is not authoritative, the resource can only modify user entitlements during reverse synchronization. |
| Delete User | No | Specifies whether the user should be deleted from the resource when a DeleteServiceMembership operation is performed for the user in Select Identity. |
| Reconciliation Workflow | ReconciliationDefault Process | Specifies the workflow to be used during reverse synchronization. |
| Resource Owner | sisa | Specifies the user who is the resource owner. |
| Associate to Group | Selected | Whether the system uses the concept of groups. For the Windows Active Directory connector, select this option. |
| Username | Administrator | Administrative account on the target Windows resource. |
| Password | Password123 | Password corresponding to the UserName account. |
| Server Name | server | The NETBIOS name or IP address of the Windows system running Active Directory. If you specify a server name, specify the name without the domain. For Active Directory 2003, specify the NETBIOS name without the domain extension and do not specify an IP address. |

| Field Name | Sample Values | Description |
|---|---|---|
| AD Port | 389 | Active Directory port on the Windows resource. |
| Agent Port | 5000 | Forward connector server port, as configured on the resource agent. |

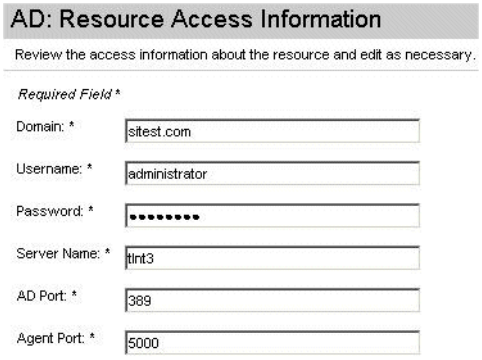
* Instead of creating an authoritative resource, you can create authoritative attributes (in the next step) for the attributes that will be synchronized. Entitlements are authoritative by default in a non-authoritative resource but other attributes are not.

If you modify the mapping file after creating the resource, the resource must be modified. However, the resource cannot be modified if any of the modified or deleted attributes are already mapped to a Select Identity attribute. Therefore, the attributes must be unmapped, the resource must be modified, then the attributes must be mapped again. Refer to the "Deleting an Attribute" section of the *HP OpenView Select Identity Administrator Guide* for more information.

Refer to *Service Studio* chapter of *HP OpenView Select Identity Administrator Guide* for information on managing resource. After deploying the resource for the connector, Select Identity displays the Basic Information page in the following format




The Access Info page looks like this:




3 Link the resource attributes to the Select Identity attributes.. To do this, perform the following steps.

a Click **Service Studio** → **Resources**. The Resources page appears.

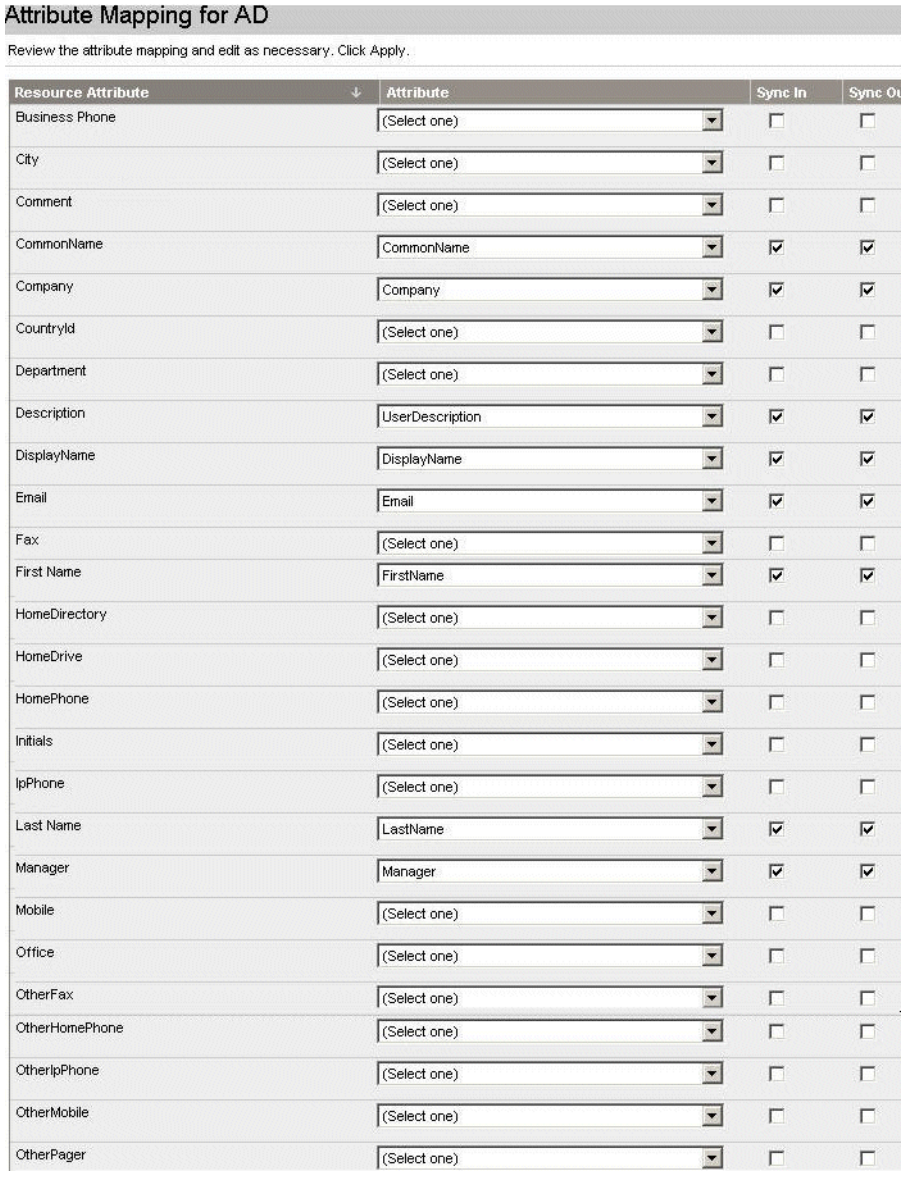
b Select the newly created resource from the Resource List, and then click **Modify**. Basic Information page appears.

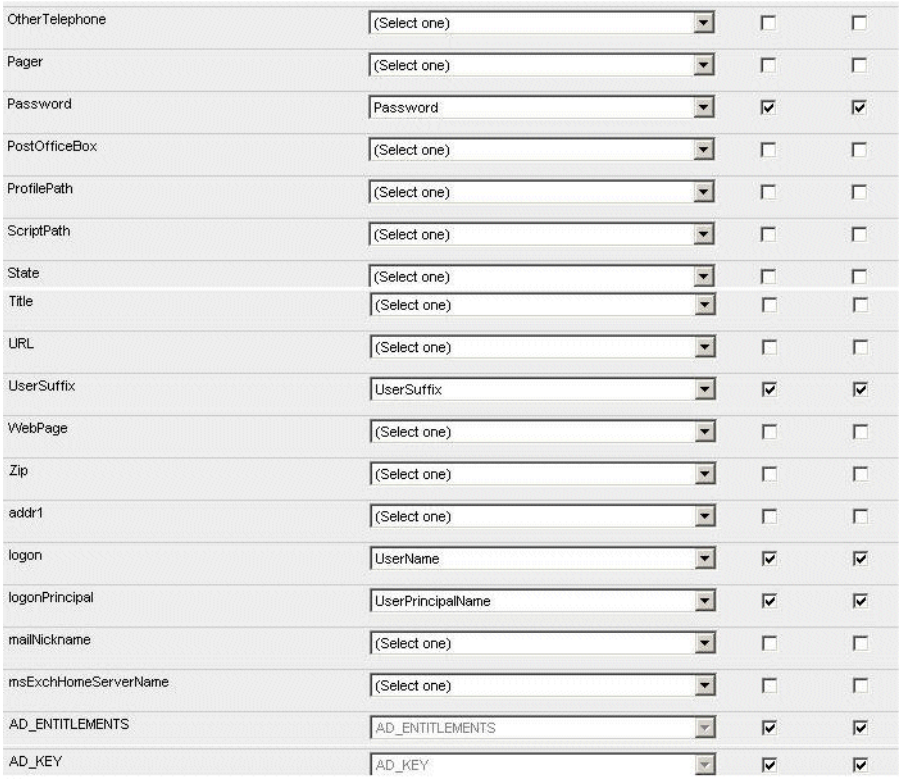
c Click **Resource Attribute Mapping** link on left pane. Resource Attribute Mapping page appears.

d Map each resource attribute to Select Identity Attribute using drop-down list.

Refer to the *Service Studio* chapter of *HP OpenView Select Identity Administrator Guide* for more information. After creating the attributes for the connector, Select Identity displays the View Attributes page for the resource in the following format:

After you create the attributes for the Windows Active Directory connector, the View Attributes page for the resource looks like this:






The Windows Active Directory connector can create a Microsoft Exchange Server account for the users created on Active Directory. Prerequisites for this feature are described in User Attributes for Exchange on page 46.

4 To create a Service that uses the newly created resource, perform the following steps.

a click **Service Studio** → **Services**. The Service List page appears.

b Click **Add New Service**.
Refer to *Service Studio* chapter in *HP OpenView Select Identity Administrator Guide* for more information. Reference the new resource created in step 2 while creating this service.

If you enable reverse synchronization, configure the Service as follows:

— When selecting the Business Relationship, choose the ReconciliationDefaultProcess workflow for the RECONCILIATION:Add Service and RECONCILIATION:Delete Service Membership request events. For RECONCILIATION:Add Service, use the user addition view.

— In the user addition view, specify mandatory attributes that are guaranteed to be passed by the reverse synchronization request when adding a user. If you specify a mandatory attribute that is not passed by the resource, the user can be created in Select Identity but reverse synchronization will not succeed.

— When specifying the context, obtain the value from the add request issued by the resource. For example, if the context is Country and the value is US, the <addRequest> element in the reverse synchroniation request should have an

attribute called country and a value of US. If the context attribute is not present in the add user request, the user can be created in Select Identity but will not be assigned to a Service.

If the attributes in the resource do not match with the attributes defined in the xml mapping file of the connector (available in .jar file), you need to modify the mapping file as needed. See Understanding the Mapping Files on page 39.

To configure reverse synchronization on the server, extract the `activedirectory.xsl` file from the `ADSchema.jar` file to the Select Identity home directory. This file maps user attributes on the Windows server to attributes in Select Identity.
Because the attributes in the `activedirectory.xsl` file are based on those in the `aduser.properties` and `adgroup.properties` files, you must modify the `activedirectory.xsl` file to reflect changes made to these files..

# Configuring Connector on Non-English Platforms

If you install the connector on non-English platform, you will have the following limitations while configuring the connector:

- When entering user attributes to provision (in the Select Identity client), you can enter local language characters except for the following attributes
  - UserName
  - Password
  - Email
- The attribute names on the resource cannot contain non-English characters. Thus, you cannot include non-English characters in the mapping file. Refer to page 25 to view Resource Attribute Mapping page for Active Directory.
- Non-English entitlements are not supported by the connector.
- All configuration and property file names must be in English.
- The exception messages from the resource are in English only.
- The log messages are in English only.
- The Select Identity resource name, which is included in the reverse synchronization configuration of the agent, must be in English.

Reverse synchronization of local language characters is supported. When provisioning users on the LDAP resource, you can enter local language characters as input data. These characters are reconciled with Select Identity through SPML communication. However, the following user attributes must contain English characters:

- UserName
- Password
- Email

# 3 Understanding the Mapping Files

The Windows Active Directory connector is deployed with the following mapping files:

- `aduser.properties`
- `adgroup.properties`
- `adcomputer.properties`

These files contain the attributes required by the resource and are used to map user account additions and modifications from Select Identity to the system resource. When you deploy a resource through the Resources pages on the Select Identity client, you can review this file.

The `adgroup.properties` and `adcomputer.properties` files are installed with the Windows Active Directory connector and must be present on the system, but group and computer provisioning is not supported at this time.

In addition, the Windows Active Directory connector provides the `activedirectory.xsl` file, which maps attributes in Active Directory to those in Select Identity (reverse mapping). Configure this file if you want to support reserve synchronization.

You can edit the Select Identity resource attributes using the Attributes pages on the Select Identity client. You can then use these attributes to associate Select Identity user accounts with system resources by mapping the attributes in the mapping file described in this chapter. The physical resource attributes are literal attributes of user accounts on Active Directory. These attributes cannot be changed. This process becomes necessary because, for example, a single attribute “username” can have a different definition on three different resources, such as “login” for UNIX, “UID” for a database, and “userID” on a Windows server.

You do not need to edit the `aduser.properties` file unless you want to map additional attributes to the Active Directory resource. If attributes and values are not defined in this mapping file, they cannot be saved to the resource through Select Identity.

You must edit the `aduser.properties` mapping file if you want to provision user mailboxes in Exchange 2000. By default, the mapping file is configured for Active Directory only.

## User Attributes for Active Directory

The `aduser.properties` file is a text file that maps each Select Identity attribute to an attribute on the resource; the attribute  s are delimited by |. Consider this excerpt:

```
User Name|UserId
```

The Select Identity user attribute is named User Name and it is mapped to the UserId attribute on the Active Directory resource.

Attributes can be concatenated. The attribute names and the separators must not contain the | delimiter. For concatenation, the format is as follows:

```
[<SI Attribute>]<separator>[<SI Attribute>]|<Resource Attribute>
```

as in this example:

```
[First Name] [Last Name]|DisplayName
```

where First Name and Last Name are attributes in Select Identity. They are concatenated to form the value of the DisplayName attribute in Active Directory. A space is used as a separator between the two Select Identity attributes.

The `aduser.properties` file provides the mandatory mappings that must be configured for Select Identity to provision users in Active Directory. The primary key is UserId; this Active Directory attribute must be mapped to a Select Identity attribute in order for user information to be stored on the Active Directory server. It should be the first entry in `aduser.properties`, and Password must be the second mapping in the file.

The following table provides a list of all Active Directory attributes that you can map if you wish to provision users with this information. Here is a description of the columns provided in the table:

- **Select Identity Resource Attribute**— The attribute used by the Windows Active Directory connector, as defined in the mapping file.
- **Active Directory User Attribute —** The name of the attribute on the Windows server.
- **Label on Active Directory UI** — The name of the property on the  UI that corresponds to the attribute on the Windows server.
- **Description** — A description of the attribute and any noteworthy information needed when assigning values to the attribute.

The mandatory attributes that are mapped by default are noted.

| **Select Identity Resource Attribute** | **Active Directory User Attribute** | **Label on Active Directory UI** | **Description** |
|---|---|---|---|
| sAMAccountName | UserId | sAMAccountName | Primary key for the Active Directory user. *This attribute is mandatory and must be mapped.* |
| Password | Password | Password (on the Account tab) | User's password. *This attribute is mandatory and must be mapped.* |
| Common Name | cn | RDN portion of distinguished name or cn | *This attribute is mandatory and must be mapped else the value passed to the UserId will be set to this field.* |
| userPrincipal-Name | userPrincipal-Name | userPrincipal-Name | |

| Select Identity Resource Attribute | Active Directory User Attribute | Label on Active Directory UI | Description |
|---|---|---|---|
| DisplayName | displayName | Display Name | Name displayed in the address book. |
| countryName | C | Country/Region (on the Address tab) | Two-character abbreviation of the country or region, per the ISO 3166-1 format. |
| Comment | Info | Notes (on the Telephone tab) | Notes about the user. |
| ScriptPath | ScriptPath | Logon Script (on the Profile tab) | The path of the user's logon script, which can be a .CMD, .EXE, or .BAT file. The string can be null. |
| UserSuffix | UserSuffix | | The organization unit or the container in the distinguished name. For example, if the distinguished name is CN=Userid, OU=subdept1, OU=dept1,DC=tru, DC=hp,DC=com, then the value of this field is OU=subdept1, OU=dept1. |
| HomeDirectory | HomeDirectory | Home Folder: Local path or Home Folder: To (on the Profile tab, field dependent on homeDrive) | A path to a home share or a local directory path, but not both. |
| (not mapped by default) | GivenName | First Name (on the General tab) | First (given) name. |
| (not mapped by default) | sn | Last Name (on the General tab) | Last name (surname). |
| (not mapped by default) | Initials | Initials (on the General tab) | Single-valued property containing the initials of the user's full name. This may be used as the middle initial in the Windows Address Book. |

| Select Identity Resource Attribute | Active Directory User Attribute | Label on Active Directory UI | Description |
|---|---|---|---|
| (not mapped by default) | Description | Description (on the General tab) | Description of the user. |
| (not mapped by default) | physical Delivery OfficeName | Office (on the General tab) | The office location in the user's place of business. |
| (not mapped by default) | Telephone Number | Telephone Number (on the General tab) | Primary telephone number. |
| (not mapped by default) | Other Telephone | Telephone: Other (on the General tab) | Alternate telephone number. |
| (not mapped by default) | Mail | E-Mail (on the General tab) | Email address. |
| (not mapped by default) | wwwHomePage | Web Page (on the General tab) | URL of the user's primary web page. |
| (not mapped by default) | url | Web Page: Other (on the General tab) | Alternate web page address. |
| (not mapped by default) | StreetAddress | Street (on the Address tab) | Street address. |
| (not mapped by default) | PostOfficeBox | P.O.Box (on the Address tab) | Post Office box. |
| (not mapped by default) | L | City (on the Address tab) | Single-valued property containing the locality, such as the town or city, in the user's address. |
| (not mapped by default) | St | State/Province (on the Address tab) | State or province. |
| (not mapped by default) | PostalCode | Zip/Postal Code (on the Address tab) | Postal (zip) code. |
| (not mapped by default) | HomePhone | Home (on the Telephone tab) | User's home phone number. |
| (not mapped by default) | OtherHome Phone | Home: Other (on the Telephone tab) | Alternate home phone number. |
| (not mapped by default) | Pager | Pager (on the Telephone tab) | User's pager number. |
| (not mapped by default) | OtherPager | Pager: Other (on the Telephone tab) | Alternate pager number. |
| (not mapped by default) | Mobile | Mobile (on the Telephone tab) | Primary mobile telephone number. |

| Select Identity Resource Attribute | Active Directory User Attribute | Label on Active Directory UI | Description |
|---|---|---|---|
| (not mapped by default) | OtherMobile | Mobile: Other (on the Telephone tab) | Alternate mobile number. |
| (not mapped by default) | facsimile Telephone Number | Fax (on the Telephone tab) | Telephone number of the user's business fax machine. |
| (not mapped by default) | other Facsimile Telephone Number | Fax: Other (on the Telephone tab) | Alternate fax number. |
| (not mapped by default) | IpPhone | IP phone (on the Telephone tab) | Telephony phone number. |
| (not mapped by default) | OtherIpPhone | IP phone: Other (on the Telephone tab) | Alternate telephony number. |
| (not mapped by default) | ProfilePath | Profile Path (on the Profile tab) | A path to the user's profile. This value can be a null string, a local absolute path, or a UNC path. |
| (not mapped by default) | HomeDrive | Home Folder: Connect (on the Profile tab) | If a valid drive letter is specified, the HomeDirectory attribute becomes a share path; otherwise, it is considered a local directory path. |
| (not mapped by default) | Department | Department (on the Organization tab) | User’s department. |
| (not mapped by default) | Title | Title (on the Organization tab) | User's formal job title or designation, such as “Senior manager.” |

| Select Identity Resource Attribute | Active Directory User Attribute | Label on Active Directory UI | Description |
|---|---|---|---|
| (not mapped by default) | Company | Company (on the Organization tab) | Company for which the user works. |
| (not mapped by default) | Manager | Manager: Name (on the Organization tab) | The fully qualified, distinguished name of the manager. The manager's user object contains a directReports property that contains references to all user objects that have their manager properties set to the manager's user object. |

# User Attributes for Exchange

If you wish to configure the connector to provision user mailboxes in Exchange 2000, you *must* add the following Exchange 2000 attributes in the `aduser.properties` file:

```
<SI Attribute>|mailNickname
<SI Attribute>|msExchHomeServerName
```

where the SI attributes are attributes configured on the Select Identity server.

The mailNickname attribute on the Exchange 2000 server is the name portion of the Email address. For example, if the email address is vlee@mydomain.com, the mailNickname attribute is assigned the vlee portion of the email address.

The msExchHomeServerName attribute is a concatenation of several server values. Here is the syntax:

/o=*exOrg*/ou=First Administrative Group/cn=Configuration/cn=Servers/cn=*mailStorage*

where

- *exOrg* is the Exchange organization name. An example is **`First Organization`**.
- *mailStorage* is the Exchange mailbox name. An example is **`MYSTORAGE`**.

In addition, you can map a Select Identity attribute to the HomeMDB attribute on the Exchange 2000 server. (On the Exchange 2000 interface, this attribute maps to the Mailbox store property on the General tab for Active Directory User.) The HomeMDB attribute represents the URL of the user's mailbox. This property is read-only and is set when the mailbox is created.

# Reverse Synchronization

The agent can send changes made to user attributes on the Active Directory server to the Select Identity server. The agent sends an SPML request to the Select Identity server that contains the attribute changes. The names of the attributes in the SPML request are defined by Active Directory. To transform the attribute names to Select Identity attribute names, the request is parsed by Select Identity using the `activedirectory.xsl` file.

The `aduser.properties` file contains generic Active Directory attributes that are typically used when a user is created. As described above, you can configure this file to include or exclude attributes. Any addition or deletion of attributes in `aduser.properties` must also be made in `activedirectory.xsl`. Each block in `activedirectory.xsl` corresponds with each attribute entry in `aduser.properties`.

If the following mapping is added to `aduser.properties`:

*SI_RESOURCE_ATTRIBUTE*|*ACTIVEDIRECTORY_ATTRIBUTE*

You must add the following block to `activedirectory.xsl`:

```
<xsl:when test="$ATTRNAME =' ACTIVEDIRECTORY_ATTRIBUTE' ">
  <xsl:call-template name="AttributeBuilder">
    <xsl:with-param name="DSMLELEMENT" select="$DSMLELEMENT"/>
    <xsl:with-param name="ATTRNAME" select="'
     SI_RESOURCE_ATTRIBUTE' "/>
    <xsl:with-param name="ATTRVALUE" select="$ATTRVALUE"/>
    <xsl:with-param name="MODIFYFLAG" select="$MODIFYFLAG"/>
  </xsl:call-template>
</xsl:when>
```

where *ACTIVEDIRECTORY_ATTRIBUTE* represents the attribute passed from the Active Directory server and *SI_RESOURCE_ATTRIBUTE* represents the attribute defined by Select Identity and displayed in the resource attributes list.

Note that the XSL file is case sensitive; attributes must be specified exactly as they exist in Select Identity and on the resource. For example, if the mail attribute is defined in Active Directory, you must specify **`mail`**, not **`Mail`** or **`MAIL`**, and so on.

The following is an example. The mail attribute is added to `aduser.properties`, as follows:

**`Email`**|**`mail`**

Then, the following block is added to `activedirectory.xsl`:

```
<xsl:when test="$ATTRNAME = 'mail'">
  <xsl:call-template name="AttributeBuilder">
    <xsl:with-param name="DSMLELEMENT" select="$DSMLELEMENT"/>
    <xsl:with-param name="ATTRNAME" select="'Email'"/>
    <xsl:with-param name="ATTRVALUE" select="$ATTRVALUE"/>
    <xsl:with-param name="MODIFYFLAG" select="$MODIFYFLAG"/>
  </xsl:call-template>
</xsl:when>
```

where **`mail`** represents the attribute passed from the Active Directory server and **`Email`** represents the attribute in Select Identity.

For composite attributes defined in the `aduser.properties` file, such as [First Name] [Last Name], you must provide two attribute name-value pairs in the `activedirectory.xsl` file. For example, for the following entry in `aduser.properties`:

```
[First Name] [Last Name]|displayname
```

The XSL file must contain the following:

```
<xsl:when test="$ATTRNAME = 'displayname'">
  <xsl:choose>
    <xsl:when test="contains($ATTRVALUE, ' ')">
      <!-- First Name is before space char -->
      <xsl:call-template name="AttributeBuilder">
        <xsl:with-param name="DSMLELEMENT" select="$DSMLELEMENT"/>
        <xsl:with-param name="ATTRNAME" select="'First Name'"/>
        <xsl:with-param name="ATTRVALUE"
         select="substring-before($ATTRVALUE, ' ')"/>
        <xsl:with-param name="MODIFYFLAG" select="$MODIFYFLAG"/>
      </xsl:call-template>
      <!-- Last Name is after space char -->
      <xsl:call-template name="AttributeBuilder">
        <xsl:with-param name="DSMLELEMENT" select="$DSMLELEMENT"/>
        <xsl:with-param name="ATTRNAME" select="'Last Name'"/>
        <xsl:with-param name="ATTRVALUE"
         select="substring-after($ATTRVALUE, ' ')"/>
        <xsl:with-param name="MODIFYFLAG" select="$MODIFYFLAG"/>
      </xsl:call-template>
    </xsl:when>
    <xsl:otherwise>
      <!-- If no space, take the whole string as First Name -->
      <xsl:call-template name="AttributeBuilder">
        <xsl:with-param name="DSMLELEMENT" select="$DSMLELEMENT"/>
        <xsl:with-param name="ATTRNAME" select="'First Name'"/>
        <xsl:with-param name="ATTRVALUE" select="$ATTRVALUE"/>
        <xsl:with-param name="MODIFYFLAG" select="$MODIFYFLAG"/>
      </xsl:call-template>
    </xsl:otherwise>
  </xsl:choose>
</xsl:when>
```

# 4 Uninstalling the Connector

If you need to uninstall a connector from Select Identity, make sure that the following are performed:

- All resource dependencies are removed.
- The connector is deleted through the Connectors home page on the Select Identity client.

## Uninstalling the Connector from WebLogic

Perform the following to delete the AD Exchange connector from WebLogic:

1. Log on to the WebLogic Server Console.
2. Expand the Deployments folder on the left pane, and then double click on **Connector Modules**

   Alternatively, at the right-hand panel of the Server Console homepage, click on **Connector Modules** link, which is under Your Deployed Resources column of Domain Configurations section.
3. The right hand pane of the console displays a table showing all the deployed connectors. Click the delete icon next to the connector that you want to uninstall.
4. Click **Yes** to confirm the deletion.
5. Click **Continue**.

## Uninstalling the Connector from WebSphere

Complete the following steps to uninstall the connector on WebSphere:

1. Log on to the WebSphere Application Server Console.
2. Navigate to **Resources** → **Resource Adapters**.
3. Select the connector to uninstall.
4. Click **Delete**.
5. Click the **Save** link (at the top of the page).
1. On the Save to Master Configuraton dialog, click the **Save** button.

# Uninstalling the Agent

Perform the following steps to delete the agent on the Windows server:

1. From the Start menu, select **Programs** → **HP OpenView AD Connector** → **Uninstall Agent**.
2. Complete the installation as prompted by the wizard.

# 5 Frequently Asked Questions (FAQ)

This appendix provides questions to frequently asked questions about the Windows Active Directory connector and its agent.

## General

### FAQ 1: How does the connector interact with Active Directory to accomplish its purpose of provisioning users and entitlements?

The connector relies on an agent, which is installed on the Domain Controller, to provision users and entitlements in Active Directory. The agent also initiates reverse synchronization, to push changes made in Active Directory back to the Select Identity database. The agent uses the ADSI API to interact with Active Directory.

### Task 2: How does the connector interact with other servers?

The connector interacts with the agent, and the agent must reside on the Domain Controller.

### Task 3: Does the connector have internal administrative roles and security?

Yes, the agent uses 128-bit PC1 encryption for JCA to communicate with the agent.

### Task 4: How are queries defined and by whom?

The Select Identity server controls the connector.

### Task 5: How are OUs targeted for account creation and placement?

The target OU is a connection parameter that is defined when the connector is created (deployed) in Select Identity.

### Task 6: What objects can be deleted by the connector?

The Active Directory connect can remove users, groups, and computers from Active Directory.

### Task 7: When I attempt to provision in Active Directory, the connection fails. How can I debug this problem?

There are several actions you can perform to find the root of this problem:

- Stop the Active Directory services (HP Openview ADConnector Service and HP Openview ADNotification Service) and restart the server and services.
- Make sure that you can connect to the Active Directory resource using another tool and specifying the same parameters that you supplied for the attempted resource creation or modification.
- Ensure that the `aduser.properties`, `adgroups.properties`, and `adcomputer.properties` files are in the application server's class path.

- Verify the port specified in the agent console and ensure the same port is specified on the resource creation/modification page.
- On the agent console, be sure to supply the same resource name (in the operation attributes) that you specified in the resource creation/modification page.
- If necessary, remove the connector from the application server then redeploy it. In Select Identity, simply modify the connector to update it.

Task 8: Do we need to install agent on all the domain controllers under the domain forest?

Yes, it needs to be installed on all the domain controllers, whichever needs the reconciliation of user attributes and password to Select Identity.

# Permissions, Privileges, and Rights

Task 9: If a service account is given full control of an OU and the account does not have Domain Admin or BUILTIN\Administrator privileges, permissions, or rights, can the connector perform its functions?

Yes, the connector can provision (forward provisioning) users and entitlements on the Active Directory or Exchange resource. The agent is responsible for sending changes made in Active Directory to the connector for reconciliation with the Select Identity database. The following are required for forward provisioning and reverse synchronization:

- Any user account that is a member of the Account Operators group
- Access to the HKEY_LOCAL_MACHINE\SOFTWARE\HPOpenview\ ADConnector registry key is enabled for non-administrative accounts Refer to **http://support.microsoft.com/?kbid=245031** for information about changing registry access.

The Account Operators group has the following privileges:

- Domain Controller Security Policy → Local Policies → User Rights → Log on as Service right
- Domain Controller Security Policy → Local Policies → User Rights → Backup Files and Directory rights
- Domain Controller Security Policy → Local Policies → User Rights → Manage Auditing and Security Log rights
- Domain Controller Security Policy → Local Policies → User Rights → Access this computer from the network
- Read/Write/Execute Permission on the installation folder of the agent and its subfolders

Task 10: If the connector cannot function with lesser privileges, permissions, and rights, what specific areas or functions of Active Directory require administrative status?

For forward provisioning, membership in the Account Operators group is mandatory. The rest of the restrictions arise due to configuration storage (registry access) and logging (directory access) requirements.

Task 11: What facets of Select Identity and the Windows Active Directory connector require full control of all Domain Controllers, all facets of Active Directory, and every Windows-based computer in the Enterprise?

See FAQ 9 on page 40.

### Task 12: What account is used to run Select Identity services?

See FAQ 9 on page 40.

## Agent

### Task 13: How is the agent installed on the Domain Controller?

The agent is a Windows service that is installed by an EXE file. A console application is provided for configuring the agent.

### Task 14: Is the agent configurable? If so, what tools are used and where are those tools located?

A console application is provided for configuring the agent. This console application is also installed by the installer.

### Task 15: What services are used by the agent?

For forward provisioning, the agent uses the TLServer service. For reverse synchronization, the agent uses the TLADNotify service.

### Task 16: What does the agent access on the Domain Controller?

The agent uses ADSI to access users and groups in Active Directory on the Domain Controller for provisioning. It uses the registry for configuration storage (password filters cannot access files). It also uses the file system for logging.

For reverse synchronization, the agent uses the Windows Security Event Log to detect changes. It uses the registry for configuration storage (password filters cannot access files), and it uses the file system for logging.

### Task 17: Can the agent alone be used to view or change anything on the Domain Controller or within Active Directory? If so, what can be controlled?

No, the connector must be used to control the agent. However, stubs can be written to emulate the connector.

### Task 18: When the agent is installed, what changes are made to the registry, the boot and root partitions, share security, and NTFS security on the Domain Controller?

During installation, the following changes are made to the Domain Controller's system:

- Two services are created with the account specified
- Registry entries are created for the configuration
- The folder structure is created at the selected location
- Security policies related to Auditing Security Event Log are configured to enable auditing of account management and directory access
- LSA Security policy is edited to enable password synchronization

### Task 19: How is configuration information stored on the agent's host system?

The configuration data for the agent is stored in the registry.

Task 20: Does the agent have log files? If so, what is logged? Where are the logs stored and in what format?

The agent uses a custom logging utility to create its log files. The utility enables you to configure levels of logging, using the agent's console application. The amount of available disk space restricts the size of the log files.

Task 21: Who can access and delete logs created by the agent?

Any user with write permissions can acess and delete the logs.

Task 22: What configuration files are created for the agent? Where are they stored and in what format? Who has access and can make changes?

The agent stores its configuration in the registry. See FAQ 9 on page 40 for specific information on this registry key.

Task 23: Are local service accounts required by the agent? What privileges, permissions and rights do the accounts require? Who needs to know the passwords for these accounts?

See FAQ 9 on page 40.

Task 24: What Domain-level service accounts are required?

See FAQ 9 on page 40.

Task 25: To what service or function does each account relate?

See FAQ 9 on page 40.

Task 26: Do any of these accounts impact other servers? If so, what servers, how do they connect, what