HP OpenView Select Identity

Connector for Microsoft Windows Active Directory and Exchange

Installation and Configuration Guide

Connector Version: 3.3
Select Identity Version: 3.3

April 2005

Legal Notices

Warranty

Hewlett-Packard makes no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company
United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

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.

Trademark Notices

HP OpenView Select Identity is a trademark of Hewlett-Packard Development Company, L.P.

Microsoft, Windows, the Windows logo, and SQL Server are trademarks or registered trademarks of Microsoft Corporation.

Sun™ workstation, Solaris Operating Environment™ software, SPARCstation™ 20 system, Java technology, and Sun RPC are registered trademarks or trademarks of Sun Microsystems, Inc. JavaScript is a trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

This product includes the Sun Java Runtime. This product includes code licensed from RSA Security, Inc. Some portions licensed from IBM are available at http://oss.software.ibm.com/icu4j/.

IBM, DB2 Universal Database, DB2, WebSphere, and the IBM logo are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

This product includes software provided by the World Wide Web Consortium. This software includes xml-apis. Copyright © 1994-2000 World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. http://www.w3.org/Consortium/Legal/

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

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

BEA and WebLogic are registered trademarks of BEA Systems, Inc.

VeriSign is a registered trademark of VeriSign, Inc. Copyright © 2001 VeriSign, Inc. All rights reserved.

All other product names are the property of their respective trademark or service mark holders and are hereby acknowledged.

Support

Please visit the HP OpenView web site at:

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

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

You can also go directly to the support web site at:

http://support.openview.hp.com/

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 valuable support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- 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://support.openview.hp.com/access_level.jsp

To register for an HP Passport ID, go to:

https://passport.hp.com/hpp2/newuser.do

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 8 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 Windows server to provision users in Select Identity based on changes made on the Windows system.

The Windows Active Directory connector is packaged in the following files, which are located in the `Active Directory & Exchange 2000/Active Directory` 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

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**. 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.

Note that due to a known Active Directory limitation, events are not generated when some attributes are modified on Active Directory 2003. Events are generated when the following attributes are modified:

- Sam Account Name
- Display Name
- User Principal Name
- Home Directory
- Home Drive
- Script Path
- Profile Path
- User Workstations
- Password Last Set
- Account Expires
- Primary Group ID
- AllowedToDelegateTo
- User Account Control
- User Parameters
- Sid History
- Logon Hours
- Country
- description

If an attribute other than one in this list is modified in Active Directory 2003, an event is not generated. This means that a reverse synchronization request cannot be sent to Select Identity.

The following URL describes the Active Directory limitation fully:
http://www.windowsecurity.com/articles/Auditing-Users-Groups-Windows-Security-Log.html

Additional configuration steps are required to enable reverse synchronization.

System Requirements

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)

Select Identity Version	Application Server	Database
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

This connector supports Active Directory 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 5000 (this can be configured on any other port, as well).

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

Operating system	Microsoft Windows 2000 Server, Service Pack 4 or later. The system must also be a Primary or Backup domain controller.
ADSI version	Version 5,0,00,0. See page 14 for information about determining the version of ADSI.
Browser version	Internet Explorer 5.5 or later (supporting MSXML 2.0 or later).
Winsock version	Version 2.0 or later.

The agent uses ADSI to perform reverse synchronization.

Deploying on the Web Application Server

To install the Windows Active Directory connector on the Select Identity server, complete these steps:

1. Create a subdirectory in the Select Identity home directory where the connector's RAR file will reside. For example, you could create the `C:\Select_Identity\connectors` folder on Windows. (A connector subdirectory may already exist.)
2. Copy the `ADConnector.rar` file from the Select Identity Connector CD to the connector subdirectory.
3. If deploying the connector on WebLogic, complete the following steps. If deploying on WebSphere, skip to Step 4 on page 13.
   a. Create a schema subdirectory in the Select Identity home directory where the connector's mapping file(s) will reside. For example, you could create the `C:\Select_Identity\schema` folder. (This subdirectory may already exist.)
   b. Extract the contents of the `ADSchema.jar` file (on the Select Identity Connector CD) to the schema subdirectory.
   c. Ensure that the CLASSPATH environment variable in the WebLogic server startup script references the schema subdirectory.
   d. Start the application server if it is not currently running.
   e. Log on to the WebLogic Server Console.
   f. Navigate to ***My_domain*** → **Deployments** → **Connector Modules**.
   g. Click **Deploy a New Connector Module**.
   h. Locate and select the `ADConnector.rar` file from the list. It is stored in the connector subdirectory.
   i. Click **Target Module**.
   j. Select the **My Server** (your server instance) check box.
   k. Click **Continue**. Review your settings.
   l. Keep all default settings and click **Deploy**. The Status of Last Action column should display Success.

4 If deploying the connector on WebSphere, complete the following steps:

   a Stop the application server.

   b Extract the contents of the `ADSchema.jar` file (on the Select Identity Connector CD) to the `WebSphere\AppServer\lib\ext` directory.

   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 in Step 1.

   h Click **Next**.

   i In the Name field, enter a name for the connector.

   j Click **OK**.

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

   l On the Save to Master Configuraton dialog, click the **Save** button.

   m Click **Resources** → **Resource Adapters**.

   n Click the new connector.

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

   p Click **New**.

   q In the Name field, enter the name of the factory for the connector. For the SQL connector, enter **`eis/AD`**.

   r Click **OK**.

   s Click the **Save** link.

   t On the Save to Master Configuraton dialog, click the **Save** button.

   u Restart WebSphere.

5 Modify the mapping files, if necessary. These files are described in detail in Understanding the Mapping Files on page 21.

6 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 (Step 5).

After installing the connector, refer to Configuring the Connector on page 32 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 also need 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:

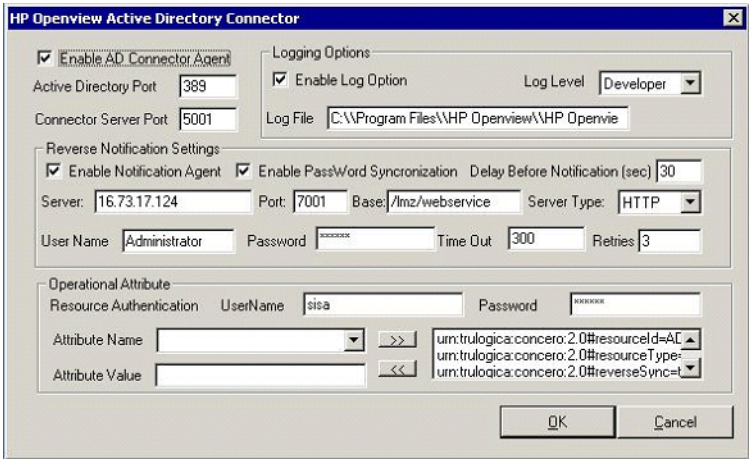
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Ads

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 5051.

d. 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 Delay Before Notification field, enter the number of seconds between requests sent to Select Identity.

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

   e. In the Port field, enter the port on which Select Identity listens for reverse synchronization requests. For example, on WebLogic, the default is 7001.

   f. Enter the base URL for the Select Identity Web Service in the Base field. The default value is /lmz/webservice/.

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

   h. Enter the name of a user that has administrative privileges on the Windows server in the User Name field.

   i. Enter the password in the Password field.

j Keep the **TimeOut** and **Retries** 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.

k In the UserName field (in the Operational Attribute section), enter the name of the administrator account in Select Identity. The default is sisa.

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

m Add 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 addition.

— 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.

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

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

10 If you configured reverse synchronization in Step 7 on page 16, 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**.

Also, 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**, double-click **Administrative Tools**, then double-click **Local Security Policy**. Expand the `Account Policies` folder and double-click **Passwords must meet complexity requirements**. Select the **Enable** option and click **OK**.

11 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
  - `<TARGETDIR>\Logs` — Connector log folder
  - `<TARGETDIR>\Map` — Mapping of operational attributes
  - `<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.

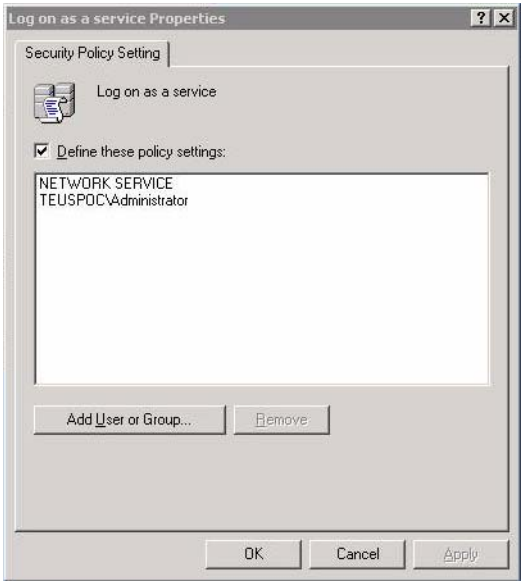
Configuring a User for the Agent's Service

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:

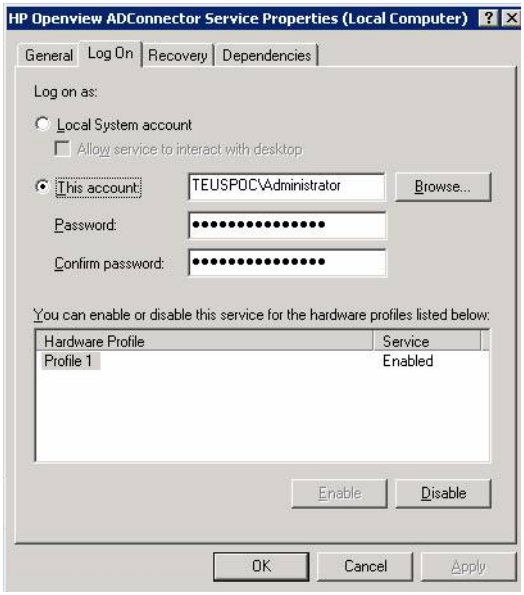
1. Create or identify a user on the Active Directory server that can be assigned as the Log On As user for the agent. Perform this step from the Computer Management window, which is displayed by right-clicking My Computer (on the desktop). You must have administrative permissions on the system to create 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. 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:



2

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.

Note that 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 wish 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 wish 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 attributes 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**
User Name	UserId	User Logon Name (on the Account tab)	Primary key for the Active Directory user. *This attribute is mandatory and must be mapped.* Same as sam-AccountName and UserPrincipalName.
Password	Password	Password (on the Account tab)	User's password. *This attribute is mandatory and must be mapped.*

Select Identity Resource Attribute	Active Directory User Attribute	Label on Active Directory UI	Description
[First Name] [Last Name]	DisplayName	Display Name (on the General tab)	Name displayed in the address book, usually comprising the user's first name, middle initial, and last name. *This attribute is mandatory and must be mapped.*
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.
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).

Select Identity Resource Attribute	Active Directory User Attribute	Label on Active Directory UI	Description
(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.
(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.

Select Identity Resource Attribute	Active Directory User Attribute	Label on Active Directory UI	Description
(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.
(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.

Select Identity Resource Attribute	Active Directory User Attribute	Label on Active Directory UI	Description
(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>
```

3

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. The following provides an overview of the procedures you must complete in order to deploy your connector. It also provides connector-specific information you must provide when configuring Select Identity to use the connector.

1 Register the connector with Select Identity by clicking the **Deploy New Connector** button on the Connectors home page. Complete this procedure as described in the "Connectors" chapter of the *HP OpenView Select Identity Administrator Guide*.

After you deploy the connector, the connector properties will look similar to this:

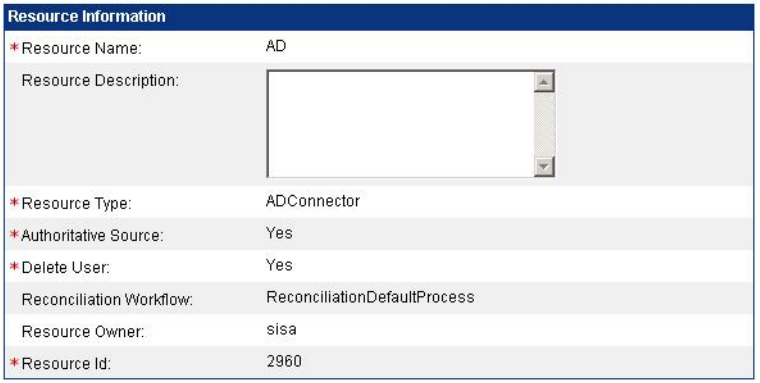
2 Deploy a resource that uses the newly created connector. On the Resources home page, click the **Deploy New Resource** button. Enter these values:

Field Name	Sample Values	Description
Resource Name	ad_server	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	AD Exchange	The connector that was deployed in Step 1 on page 32.
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.
Associate to Group	Selected	Whether the system uses the concept of groups. For the Windows Active Directory connector, select this option.
Domain	mydomain.com	Active Directory domain name.
Username	Administrator	Administrative account on the target Windows resource.
Password	Password123	Password corresponding to the UserName account.

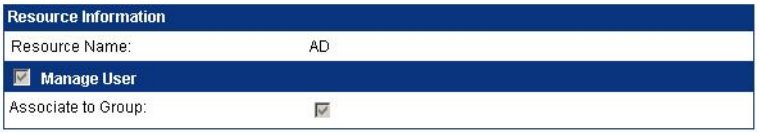
Field Name	Sample Values	Description
Server Name	server	The NETBIOS name or IP address of the Windows system running Active Directory. If you specify a server name, specify the name without the domain.
AD Port	389	Active Directory port on the Windows resource.
Agent Port	5051	Forward connector server port, as configured on the resource agent.
Container	OU=stsd,OU=hp	Name for provisioning the users. Users will be created in this OU on the Active Directory server.

* 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.

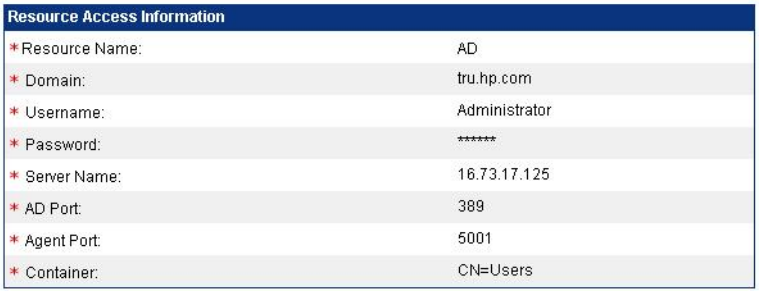
Complete the steps in this procedure as described in the “Resources” chapter of the *HP OpenView Select Identity Administrator Guide.* After you deploy the resource for the Active Directory connector, the Basic Info page of the resource properties will look similar to this:



The Additional Info page will look similar to this:



The Access Info page will look similar to this:



3 Create attributes that link Select Identity to the connector. For each mapping in the connector's mapping file, create an attribute using the Attributes capability on the Select Identity client. Refer to the "Attributes" chapter in the *HP OpenView Select Identity Administrator Guide* for more information.

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 28.

Active Directory supports a special Entitlement called EXCHANGE ACCOUNT, which is added to the list of entitlements retrived from Active Directory. You must assign this entitlement to the user to create a user account and mail box in Microsoft Exchange Server.

Note that removal of a mail box is not supported, therefore removing this entitlement will not remove the Exchange account for the user.

After you create the attributes for the Windows Active Directory connector, the View Attributes page for the resource will look similar to this:

(Resource Name=AD)

Page 1 of 1 Total Records:19

Name	Min Length	Max Length	Attribute Mapped To	Authorative
AD_ENTITLEMENTS	1	255	AD_ENTITLEMENTS	Y
AD_KEY	1	255	AD_KEY	Y
addr1	0	255	Addr1	N
Business Phone	0	255	PhBus	N
City	0	255	City	N
Company	0	255	Company	N
CountryId	0	255	Country	N
Description	0	255	Description	N
DisplayName	0	255	DisplayName	N
Email	0	255	Email	N
First Name	0	255	FirstName	N
HomeDirectory	0	255	HomeDirectory	N
Last Name	0	255	LastName	N
mailNickname	0	255		
msExchHomeServerName	0	255		
Office	0	255	OfficeCity	N
Password	0	255	Password	N
ScriptPath	0	255	ScriptPath	N
User Name	0	255	UserName	N

4 Create a Service that will use the newly created resource. To do so, click the **Deploy New Service** button on the Services home page. Complete this procedure as described in “Services” of the *HP OpenView Select Identity Administrator Guide*. You will reference your new resource created in Step 2 while creating this service.

If you are enabling 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 will 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 will be created in Select Identity but will not be assigned to a Service.

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 a connector:

1. Log on to the WebLogic Server Console.
2. Navigate to ***My_Domain*** → **Deployments** → **Connector Modules**.
3. 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).
6. 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.