HP Select Identity

Software Version: 4.21

Connector Deployment Guide

Document Release Date: January 2008
Software Release Date: January 2008

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/).

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 © 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 © 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

You can visit the HP software support web site at:

http://www.hp.com/go/hpsoftwaresupport

HP Software online support provides an efficient way to access interactive technical support tools. As a valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into 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 sign in. Many also require an active support contract.

To find more information about support access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

Contents

1 Documentation Map

This chapter describes the organization of HP Select Identity connector documentation and provides necessary information on how to use the documentation set to install and configure the connectors.

Figure 1 illustrates the documentation map for Select Identity connector. For a list of available product documentation, refer to the Table 1.

Figure 1 Documentation Map

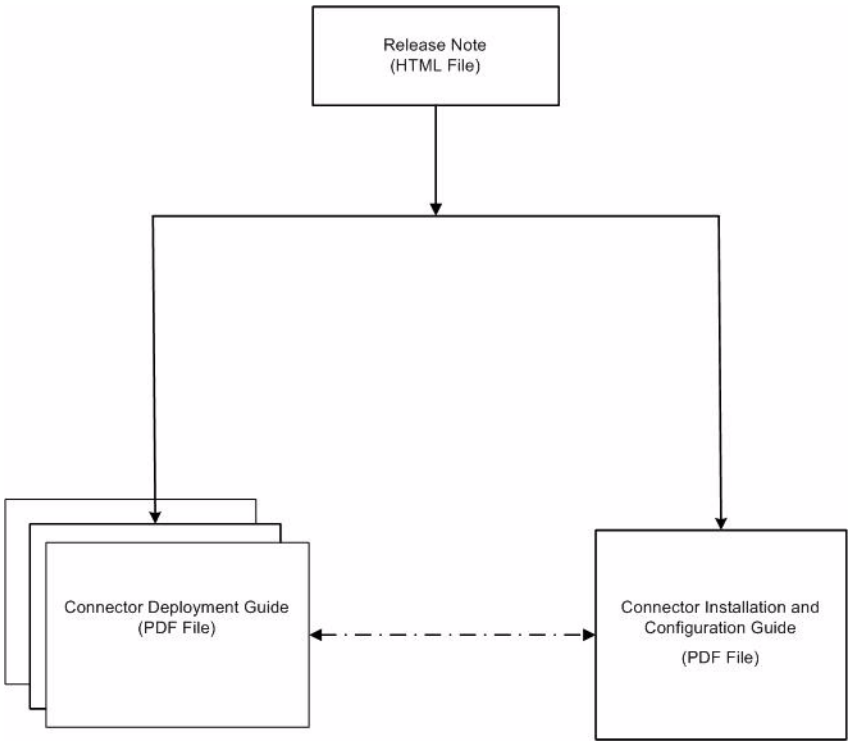


Table 1 Connector Documentation

Document Title and Filename	Contents	Location
Release Note	This file contains necessary information on new features of the connector, enhancements, known problems or limitations, and support information.	`/Docs/` subdirectory under connector directory.
Connector Deployment Guide (for Select Identity 4.20/4.21) `connector_deploy_SI4.21.pdf`	Connector deployment guides provide detailed information on: • Deploying a connector on an application server. • Configuring a connector with Select Identity. Refer to these guides when you need detailed and generic information on connector installation.	`/Docs/` root directory under the product's release folder.
Connector Deployment Guide (for Select Identity 4.10-4.13) `connector_deploy_SI4.13.pdf`		
Connector Deployment Guide (for Select Identity 4.0-4.01) `connector_deploy_SI4.pdf`		
Connector Deployment Guide (for Select Identity 3.3.1) `connector_deploy_SI3.3.1.pdf`		
Connector Installation and Configuration Guide <*connector_name*>`_guide.pdf`	Connector installation and configuration guide provides installation instructions for a specific connector.	`/Docs/` subdirectory under connector directory.

2 Introduction

This chapter gives an overview of the HP Select Identity connector. An HP Select Identity connector enables you to provision users and manage identities on an enterprise information system. At the end of this chapter, you will be able to know about:

- The benefits of HP Select Identity.
- The role of a connector.

About this Guide

The *HP Select Identity Connector Deployment Guide* gives you an overview of generic installation and configuration tasks to be performed to install a connector on the Select Identity server. The guide elaborates the following instructions:

- Instructions to deploy a connector on an application server.
- Instructions to configure the connector on Select Identity.

The instructions explained in this guide are common for all the connectors. For additional connector specific or resource specific installation instruction, refer to the specific connector's Installation and Configuration Guide.

About HP Select Identity

HP Select Identity provides a new approach to identity management. It helps you manage the entire identity lifecycle of an enterprise application. By using Select Identity, you can automate the process of provisioning and managing user accounts and access privileges across platforms, applications, and corporate boundaries. Select Identity communicates with the enterprise information system through connectors, and automates the tasks of identity management. The enterprise information system, which is also referred to as **resource**, can be a database, a directory service, or an ERP package, among many others.

About Connectors

You can establish a connection between a resource and Select Identity by using a connector. A connector is resource specific. It is installed on the system where Select Identity is installed. The combination of Select Identity and connector helps you perform a set of tasks on the resource to manage identity. A connector can be **unidirectional** or **bidirectional**. A unidirectional connector helps you manage identities from Select Identity, but if any change

takes place in resource, it cannot communicate that back to Select Identity. On the other hand, a bidirectional connector can reflect the changes made on resource back to Select Identity. This property of bidirectional connectors is known as **reverse synchronization**.

Features and Capabilities

A connector enables Select Identity to access a resource to manage users, groups, and entitlements. Select Identity can typically perform the following tasks by using a connector.

- Add, update, and remove users
- Retrieve user attributes
- Enable and disable users
- Verify a user's existence
- Change user passwords
- Reset user passwords

The set of tasks, which can be performed by the connector on resource, varies from connector to connector.

A connector usually consists of:

- A **Resource Adapter Archive** (**RAR**) file
  - A RAR file with Java 2 Security enabled (`ConnectorName_420.rar` for Select Identity 4.20; `ConnectorName.rar` for later versions) — this file contains connector binaries with Java 2 security enabled. This file is for WebSphere.
  - A RAR file without Java 2 Security enabled (`ConnectorName_420WL9.rar` for Select Identity 4.20; `ConnectorName_WL9.rar` for later versions) — this file contains connector binaries without Java 2 security enabled. This file is for WebLogic 9. Current connectors do not support WebLogic 9 with Java 2 security enabled.
- A **Schema JAR/ZIP** file — this contains the mapping file for the connector. A mapping file contains resource attribute information of the connector, which must be linked to Select Identity attributes. The mapping file could be an XML file or a simple .properties file. Many connectors are supplied with a default mapping file that is usable in most cases. For some connectors you can modify a mapping file using the Attribute Mapping Utility. Refer to the *HP Select Identity Administration Online Help* for information on using this utility.

In addition to above two files, there could be other files packaged with the connector, such as an agent file, a script file, and so on.

About Deploying a Connector

In order to use a connector with Select Identity, you must deploy it on an application server, and then configure it with Select Identity. The RAR file of the connector, which contains the binaries, must be deployed on an application server. You must perform the following tasks to deploy and configure a connector.

1 Extracting Contents of the Schema JAR/ZIP File

2 Deploying the Connector on Application Server

3 Extracting Contents of the Schema JAR/ZIP File

Many connectors contain at least one mapping file. This file contains the resource attribute information for the connector.

You must extract contents of the schema JAR or ZIP file to a location on the Select Identity server. Perform one of the procedures explained below depending on the application server (WebLogic, WebSphere, or JBoss) on which the connector will be deployed.

WebLogic

1. Create a subdirectory in Select Identity home directory where you can store the connector's mapping files and XSL files.
   For example, you can create <*SI_HOME_DIR*>`/Schema` where <*SI_HOME_DIR*> = `/opt/si420/weblogic/` for Select Identity installed on UNIX and <*SI_HOME_DIR*> = `C:\si420\weblogic\` for Select Identity installed on Windows.
2. Extract contents of the schema JAR or ZIP file to the `Schema` directory. Some connectors may contain more than one schema file. Refer to the connector's Installation and Configuration Guide to find out the right schema file to be used.
3. To ensure that the `CLASSPATH` environment variable in WebLogic startup script references the `Schema` directory created above, perform the following steps:
   a. Open the `myStartWL.cmd/.sh` file from the location <*SI_HOME_DIR*>`/weblogic/scripts/weblogic` with a text editor.
   b. Add the directory path of the `Schema` directory to the `CLASSPATH` variable in the script.

If you install more than one connector, you can extract the `Schema` JAR or ZIP file of all the connectors to the same location.

- The XML files should be placed in the path `com\trulogica\truaccess\connector\schema\spml` under the `Schema` directory.
- The XSL files must be available directly under the `Schema` directory.
- For the connectors which do not have an XML schema mapping file but have `properties` file, you must place the `properties` file directly under the `Schema` directory.

WebSphere

On WebSphere, <*WebSphere_Install_Dir*>/AppServer/lib/ext folder is present in WAS CLASSPATH by default. Extract contents of the Schema JAR or ZIP file to the location <*WebSphere_Install_Dir*>/AppServer/lib/ext.

- The XML files should be placed in the path com\trulogica\truaccess\connector\schema\spml under the ext directory.
- The XSL files must be available directly under the ext directory.
- For the connectors which do not have an XML schema mapping file but have properties file, you must place the properties file directly under the ext directory.

JBoss

1. Create a subdirectory in Select Identity home directory where you can store the connector's mapping files and XSL files.
   For example, you can create <*SI_HOME_DIR*>/Schema where <*SI_HOME_DIR*> = /opt/si413/jboss/ for Select Identity installed on UNIX and <*SI_HOME_DIR*> = C:\si413\jboss\ for Select Identity installed on Windows.
2. Extract contents of the schema JAR or ZIP file to the Schema directory. Some connectors may contain more than one schema file. Refer to the connector's *Installation and Configuration Guide* to find out the right schema file to be used.
3. To ensure that the CLASSPATH environment variable in JBoss startup script references the Schema directory created above, perform the following steps:
   a. Open the SIStartJBoss.conf file from the location <*SI_HOME_DIR*>/scripts with a text editor.
   b. Add the directory path of the Schema directory to the JBoss CLASSPATH variable in the script.

- The XML files should be placed in the path com\trulogica\truaccess\connector\schema\spml under the Schema directory.
- The XSL files must be available directly under the Schema directory.
- For the connectors which do not have an XML schema mapping file but have properties file, you must place the properties file directly under the Schema directory.

4 Deploying the Connector on Application Server

To install the connector on Select Identity, you must deploy the connector on the application server.

WebLogic/WebSphere

Perform the following steps to deploy the connector on a WebLogic or WebShpere application server:

1. Create a subdirectory in Select Identity home directory where you can store the connector's Resource Adapter Archive (RAR) file.
   For example, you can create `<SI_HOME_DIR>/connectors`
   where `<SI_HOME_DIR>` = `/opt/Select_Identity` on UNIX
   and `<SI_HOME_DIR>` = `C:\Select_Identity` on Windows (A connector subdirectory may already exist.)
2. Copy the 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 16.
   c. Start the application server in the domain for Select Identity, if it is not currently running, and log on to the WebLogic Server Console.
   d. In the left pane, click **Lock & Edit** button in Change Center panel, then click **Deployments** in Domain Structure panel, the deployed applications are displayed in the right pane.
   e. Click **Install** button right pane, then click the link in the Location field, locate, and select the RAR file without Java2 Security (`ConnectorName_WL9.rar`) from the list. It is stored in the connector subdirectory.
   f. Click **Next**.
   g. Select Install this deployment as an application option, then click **Next**.

      If more than one server is configured, the next page prompts you to select the server on which you want to deploy the connector. Select the server instance, then click **Continue**.
   h. Provide a unique name for the connector in the Name field. Select Source Accessibility option. Click **Next**.
   i. Select No, I will review the configuration later. option, then click **Finish**.
   j. Click **Active Changes**.
   k. In the right pane, locate the connector you just installed, select the connector. Click **Start**, then select **Servicing all requests** from the popup menu. In the next page displayed, click **Yes**. The status of State column should be Active.

l 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, and then click **Continue**.

m 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 Start the application server, if necessary.

b Log on to the WebSphere Application Server Console.

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

d Click **Install RAR**. The Install RAR File page appears.

e If it is a cluster setup, select a WebSphere node from the Node drop-down box.

f In the Server path field, enter the path to the connector's RAR file with Java2 Security (`ConnectorName_420.rar`). It is stored in the subdirectory created in the beginning.

g Click **Next**.

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

i Click **OK**.

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

k Repeat step f to step j for all the available nodes (for cluster setup).

l Click the new connector.

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

n Click **New**.

o In the Name field, enter the name of the factory for the connector. In the JNDI Name field, enter the JNDI name of the connector. This is the pool name of the connector. Refer to respective connector's Installation and Configuration Guide to find out the specific pool name.

p Click **OK**.

q Click the **Save** link.

r Repeat step l to step q for all available nodes (for cluster setup).

s Restart WebSphere.

JBoss

To deploy the connector on a JBoss application server, perform the following steps:

1 Copy the connector's Resource Adapter Archive (RAR) file from the Select Identity Connector CD to the <*JBoss_Install_Dir*>`/server/profile/deploy` directory.

2 Create a xml file for the connector and deploy the xml file into <*JBoss_Install_Dir*>/`server/profile/deploy` directory.

If you deploy more than one connector, create a xml file for each.

Below is a sample xml file for Oracle 11i connector for your reference:

```
<?xml version="1.0" encoding="UTF-8"?>


<connection-factories>
    <no-tx-connection-factory>
        <jndi-name>eis/ORAERP</jndi-name>
        <use-java-context>false</use-java-context>
        <rar-name>oraerp.rar</rar-name>
        <connection-definition>com.trulogica.truaccess.connector.TAConnect
        orFactory</connection-definition>
    </no-tx-connection-factory>
</connection-factories>
```

Replaceable attributes:

- **eis/*ORAERP*** – the JNDI name.
- ***oraerp.rar*** – the connector's RAR file name.
- ***com.trulogica.truaccess.connector.TAConnectorFactory*** – value of `connection-definition` attribute, which is also the value of `connectionfactory-Interface` attribute in `ra.xml` file (present in META-INF folder of the connector RAR file).

> Some connectors need special configurations, see BiDirectional LDAP Connectors-Specific Deployment Configurations on page 18 and Oracle Connector-Specific Deployment Configuration on page 19 for more information.

3. Change `Isolated` attribute value in <*JBoss_Install_Dir*>`/server/profile/deploy/ear-deployer.xml` to true. The default value is false.
4. Restart the JBoss application server.

After restarting the application server, some errors similar to the below might be logged into `SIStartJBoss.log` file:

```
=========================================================================
10:39:55,597 ERROR [URLDeploymentScanner] Incomplete Deployment listing:


--- Packages waiting for a deployer ---
org.jboss.deployment.DeploymentInfo@654eaeca { url=file:/opt/jboss-4.0.5.GA/
server/default/tmp/deploy/tmp18060ActiveDirConnector.rar-contents/
_connectorModule.jar }
  deployer: null
  status: Starting
  state: START_SUBDEPLOYMENTS
  watch: file:/opt/jboss-4.0.5.GA/server/default/tmp/deploy/
tmp18060ActiveDirConnector.rar-contents/_connectorModule.jar
  altDD: null
  lastDeployed: 1177814322337
```

```
lastModified: 1177814321000
mbeans:
```

No function break is noticed in Select Identity JBoss testing so far (on JBoss 4.0.5 GA), so you can ignore the error.

BiDirectional LDAP Connectors-Specific Deployment Configurations

This section contains connector-specific deployment configurations. You can also find this information in each connector-specific Installation and Configuration Guide.

For all BiDirectional LDAP connectors, modify <*Connector_Name*-ds.xml> file for each connector (present in <*JBoss_Install_Dir*>/server/profile/deploy/ folder) to set JNDI name by adding two config-property attributes, one for conectorName and the other for jndiName.

After modification, the xml file should look similar to the sample below:

```
<?xml version="1.0" encoding="UTF-8"?>
<connection-factories>
    <no-tx-connection-factory>
        <jndi-name>eis/ActiveDirConnector</jndi-name>
        <use-java-context>false</use-java-context>
        <rar-name>ActiveDirConnector.rar</rar-name>
        <connection-definition>com.trulogica.truaccess.connector.TAConnectorFactory</
        connection-definition>
        <config-property name="connectorName"
        type="java.lang.String">ActiveDir</config-property>
        <config-property name="jndiName" type="java.lang.String">eis/
        ActiveDirConnector</config-property>
    </no-tx-connection-factory>
</connection-factories>
```

See Table 2 below for a complete list of all bidirectional LDAP connectors:

Table 2 BiDirectional LDAP Connectors

Connector Short Name	Connector Folder Name
Active Directory Bidirectional LDAP connector	Bidirectional LDAP Connector - Active Directory
Sun ONE Bidirectional LDAP connector	Bidirectional LDAP Connector - SunOne Directory
eDirectory Bidirectional LDAP connector	Bidirectional LDAP Connector - Novell E-Directory
eTrust Bidirectional LDAP connector	Bidirectional LDAP Connector - Etrust
Oracle Internet Directory Bidirectional LDAP connector	Bidirectional LDAP Connector - Oracle Internet Directory
IBM Tivoli Directory Bidirectional LDAP connector	Bidirectional LDAP Connector - IBM Directory

Table 2 BiDirectional LDAP Connectors (cont'd)

Connector Short Name	Connector Folder Name
TopSecret connector	Bidirectional LDAP Connector - TopSecret
RACF connector	Bidirectional LDAP Connector - RACF
ACF2 connector	Bidirectional LDAP Connector - ACF2
OpenLDAP connector	LDAP OpenLDAP
DirX Bidirectional LDAP connector	Bidirectional LDAP Connector - DirX Directory (not supported for JBoss yet)

Oracle Connector-Specific Deployment Configuration

If Select Identity is deployed on JBOSS with Oracle as the backend database, there is no need to copy `ojdbc14.jar` file.

5 Configuring the Connector with Select Identity

After you deploy the connector on the application server, you must configure the connector with Select Identity to be able to use it.

Perform the following steps to deploy and configure the connector on Select Identity:

1 Add a New Connector

2 Add a New Resource

Add a New Connector

To add the connector with Select Identity, perform the following steps:

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

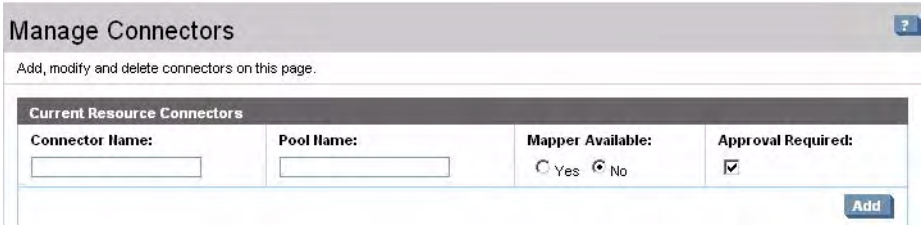
2 Click **Manage Connectors** on Resources List page. Manage Connectors page appears.

3 In Manage Connectors page, do the following:

- In the Connector Name text box, specify a name for the connector.
- In the Pool Name text box, enter the pool name of the connector. Exact pool name of the connector is given in the connector Installation and Configuration Guide.
- Under Mapper Available section, select **Yes** if the connector is supported by attribute mapper utility of Select Identity. Otherwise, select **No**.

➤ Select the Approval Required check box if configuration management is enabled for the connector and approvals are required for any changes.

Select Identity displays the Manage Connectors page in the following format:



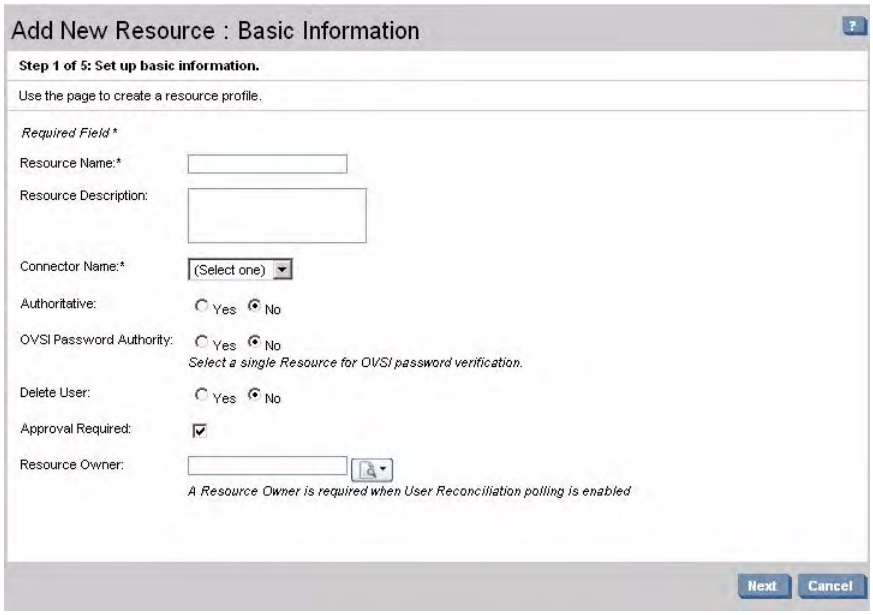
Refer to *Connectors* chapter of *HP Select Identity Administration Online Help* for more information on managing connectors.

Add a New Resource

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

1 Click **Service Studio** → **Resources.** Resources List page appears.

2 Click **Add New Resource**.
Add New Resource: Basic Information page appears.



Select the Approval Required check box if configuration management is enabled and approvals are required for any changes.

3 Specify a Mutual Authentication Policy

From the Add Resource: Mutual Authentication Policy page, you can specify a mutual authentication policy by specifying the inbound and outbound security settings. The inbound security settings apply to incoming web service requests. The outbound security settings apply to outgoing connections to the connectors.

Specifying a mutual authentication policy by defining these security parameters is optional; however, these parameters are required for mutual authentication.

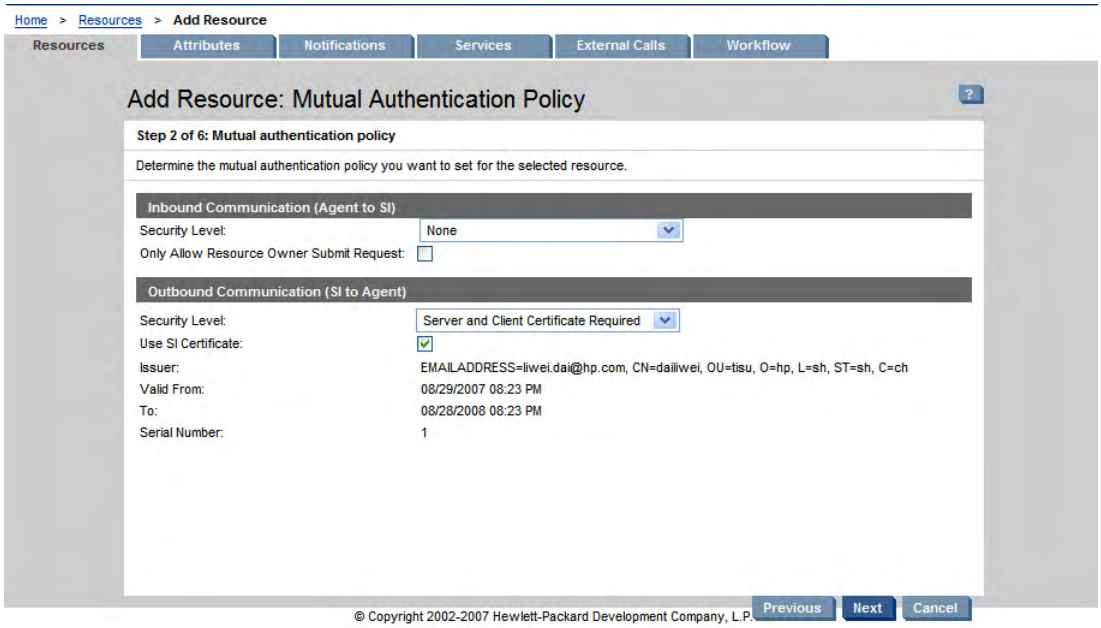
To specify a mutual authentication policy, perform the following steps:

a Select the inbound security level from the **Security Level** list.

- **None** – indicates that the resource does not use PKI (Public Key Infrastructure) for secure inbound communication. If Client Certificate Required is selected on the System Security page, you cannot select None. A client certificate is required.
- **Client Certificate Required** – indicates that the client must present a certificate when connecting to Select Identity.

b Optional. Select the **Only Allow Resource Owner Submit Request** option.

If checked, the Resource Owner must be defined and the resource owner must have a certificate. When selected, only the owner of the resource can submit a reconciliation request.



c Select the outbound security level from the **Security Level** list:

- **None** – indicates that the resource does not use PKI for secure outbound communication.
- **Server Certificate Required** – indicates that the server must present a certificate when Select Identity connects to this server. The Select Identity connector must also request the server's certificate and validate it.
- **Server and Client Certificate Required** - indicates that the Select Identity connectors must submit a request for the server certificate and validate it, and that the Select Identity connectors must present a certificate to the server for authentication.

  If you select Server and Client Certificate Required, complete the following fields as appropriate:

Field	Description
Use Select Identity Certificate	Indicates that the Select Identity certificate is required and is set up, see Configure System Security.
Client Certificate	Specifies the client certificate to use.
Use key store password	Indicates that the password for the key store is used.
Password	Specifies the password to use if not using the key store password.

  Select Identity displays additional information about the certificate including valid from/to dates, serial number, and the issuer of the certificate.

  For more information about Mutual Authentication, refer to *HP Select Identity Administration Online Help*.

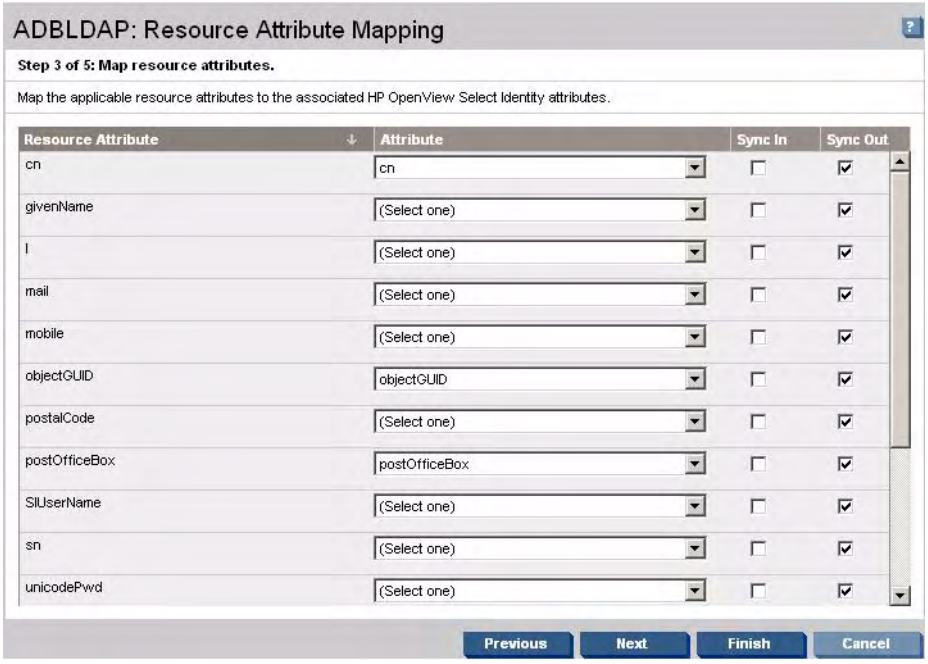
4 Fill in the parameters given in *Table 5: Resource Configuration Parameters* in the connector's Installation and Configuration Guide.

5 Click **Next.** Add New Resource: Resource Access Information page appears. Select Identity displays the Resource Access Information page in the following format (the image of Resource Access Information page for Active Directory Bidirectional LDAP connector is shown below for example):

6 Fill in the parameters given in *Table 5: Resource Configuration Parameters* in the connector's Installation and Configuration Guide.
Some of the connectors, like database connectors, are supported by attribute mapping utility. These connectors may not have a mapping file packaged with it. In case of such connectors, click **Edit** link next to Mapping File text box. Attribute Mapper page appears. The mapping xml file can be generated from attribute mapping page.

Refer to the *HP Select Identity Administration Online Help* for more information on creating mapping file by using attribute mapping utility.

After typing the resource access parameters in the Resource Access Information page, click **Next**. The Resource Attribute Mapping page appears.

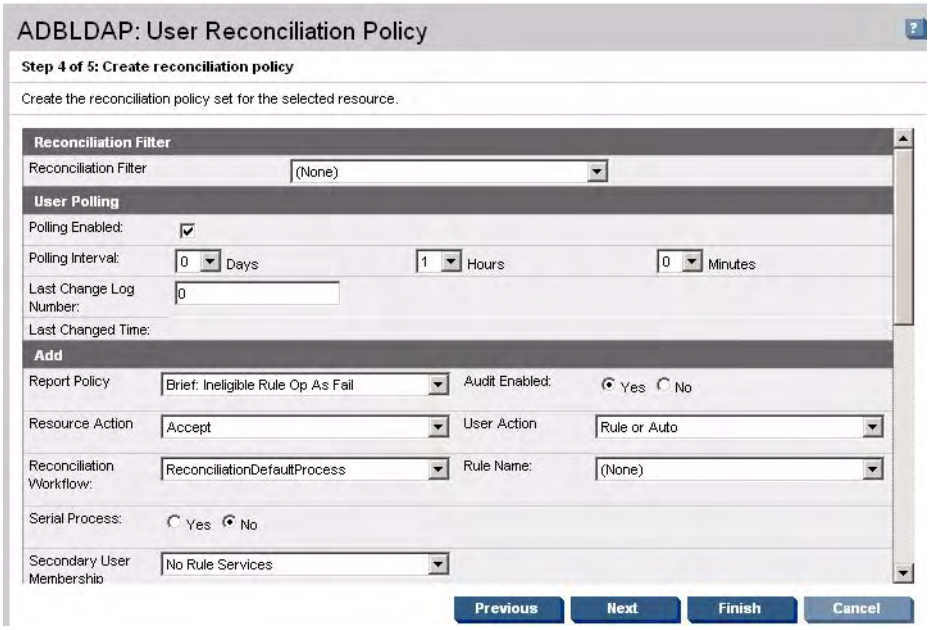


7 Map each resource attribute to Select Identity Attribute by using drop-down list.

While mapping the resource attributes on Select Identity, refer to *Table6* in the connector's Installation and Configuration Guide for connector specific mapping information.

If the connector supports reverse synchronization for that particular attribute, you must check **Sync In** checkbox to reflect the changes on that attribute made on the resource end to Select Identity.
Attributes updated at Select Identity are set to **Sync Out** to reflect the change at resource end.

8 Click **Next.** The User Reconciliation Policy page appears. (The image of User Reconciliation Policy page for Active Directory Bidirectional LDAP connector is shown below for example).



9 Perform the following in this page, as required:

— Under the Add and Modify section, select an available rule from the Rule Name drop-down box.

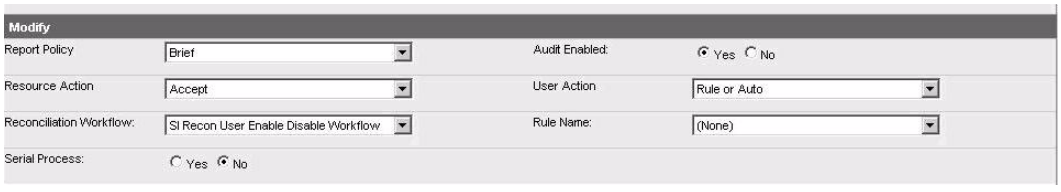
— In case of connectors where reverse synchronization is achieved by using agent, select ExtendedSpmlRequestFilter from the Reconciliation Filter drop-down box.

— In case of connectors where reverse synchronization is achieved by polling, such as bidirectional LDAP based connectors, select the **Polling Enabled** check box, set the polling interval, and then click **Apply.** You must set the polling interval as mentioned in the connector's Installation and Configuration Guide. If the connector does not employ polling for reverse synchronization, skip to the next step. Refer to the connector's Installation and Configuration Guide to find out if the connector needs polling to be enabled.

— Some of the connectors, such as bidirectional LDAP connectors, need the User Enable/ Disable Workflow External Call to be configured. In case of those connectors, modify the following fields on User Reconciliation Policy page.

  – Under the Add section, select Select Identity Recon User Enable Disable Workflow from Reconciliation Workflow drop-down box.

  – Under the Modify section, select Select Identity Recon User Enable Disable Workflow from Reconciliation Workflow drop-down box.



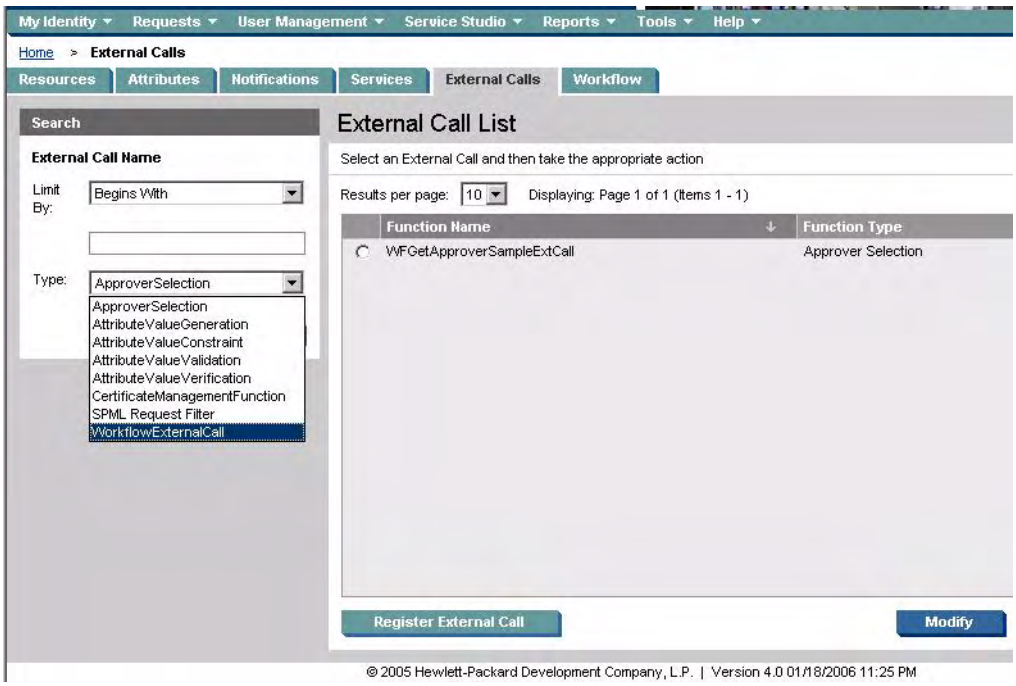
Refer to the respective connector Installation and Configuration Guide to find out if the connector needs these fields to be modified.

To use the connector, you must associate the newly added resource to a service. Refer to the *Service Studio* chapter of the *HP Select Identity Administration Online Help* for information on services.

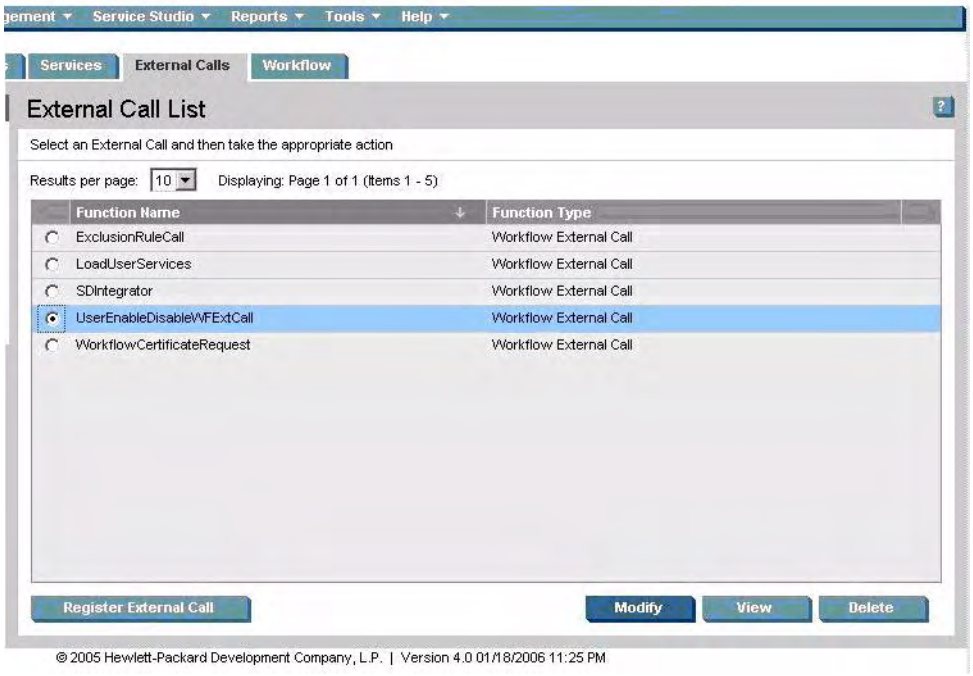
Configuring User Enable/Disable Workflow External Call

Some of the connectors, such as PeopleSoft connector or bidirectional LDAP based connectors, require user enable/ disable workflow external call to be modified. Refer to the respective connector's Installation and Configuration Guide to find out if this is required.
Perform the following steps to configure user enable/disable workflow external call.

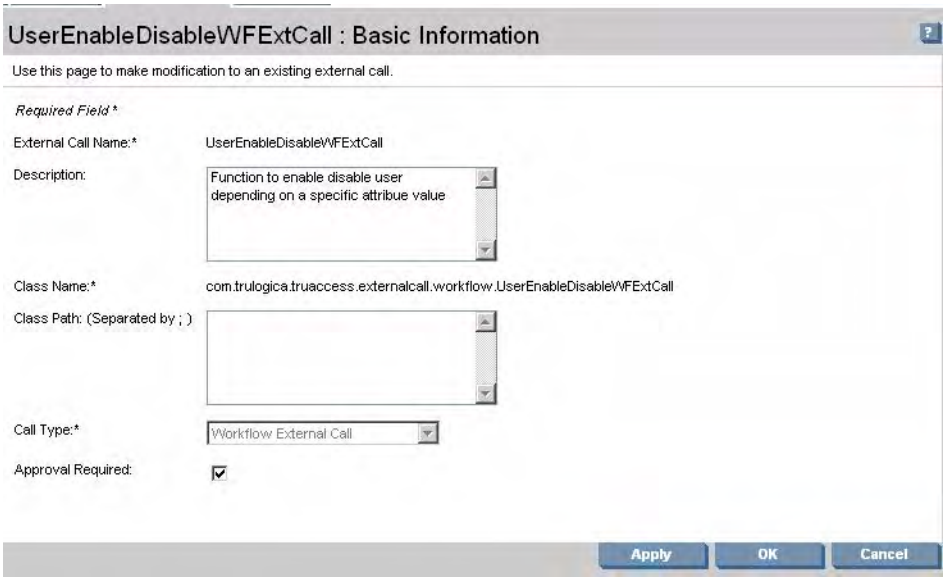
1 On Select Identity home page, click **Service Studio** → **External Calls**. External Call List page appears.



2 On left pane, select **WorkflowExternalCall** from Type drop-down box, and then click **Search**. On right pane, a list of workflow external calls appear.

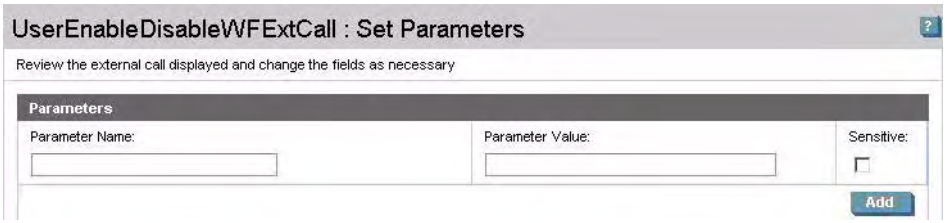


3 On right pane, select the **UserEnableDisableWFExtCall** radio button, and then click **Modify**. The UserEnableDisableWFExtCall: Basic Information page appears.



4 Enter the description, and then click **Apply**.

5 Click **Parameters** link on left pane. UserEnableDisableWFExtCall: Set Parameters page appears.



6 You must enter the following parameters and add, one after another.

a `AttributeName`

b `EnableValue`

c `DisableValue`

d `UserName`

e `Password`

f `Url`

7 Click **OK**.

Refer to the *Table 7: UserEnableDisableWFExtCall Parameters* in the connector's Installation and Configuration Guide for exact value of the parameters.

In order to use the connector, you must associate the newly added resource to a service. Refer to the *Service Studio* chapter of the *HP Select Identity Administration Online Help* for information on services.

Configuring Connector on Non-English Platforms

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

- When entering user attributes to provision (in Select Identity), you cannot enter local language characters 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.
- 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.
- The log messages are in English.
- 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 if the connector is internationalized. While provisioning users on the 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

6 Uninstalling the Connector

To uninstall a connector from Select Identity, perform the following:

1. Delete the connector from Select Identity home page.
2. Delete the connector from application server.

Deleting the Connector from Select Identity

Before deleting a connector, remove all dependencies on the connector. Perform the following steps to delete a connector from Select Identity.

1. Click **Service Studio** → **Resources** on Select Identity homepage. Resource List page appears.
2. Click **Manage Connectors.** Manage Connectors page appears.
3. Select the connector, which you want to delete.
4. Click **Delete**.

Deleting the Connector from WebLogic

Perform the following to delete a connector from WebLogic:

1. Log on to the WebLogic Server Console.
2. Click **Deployments** menu on the left pane, and then double click on **Connector Modules**
3. The right hand pane of the console displays a table showing all the deployed applications, including connectors. Select the connector you want to remove, click **Stop**, then choose **Force Stop Now** from the popup menu. Click **Yes** in the next page.
4. Click **Lock & Edit** in the left pane. Choose the connector in the right pane, click **Delete**. Click **Yes** in the next page. Click **Yes** to confirm the deletion.
5. Click **Active Changes** in the left pane.

Deleting the Connector from WebSphere

Perform the following steps to uninstall the connector from WebSphere:

1. Log on to the WebSphere Application Server Console.
2. Navigate to **Resources** → **Resource Adapters** → **Resource Adapters**.

3 Select the connector to uninstall.

4 Click **Delete**.

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

6 If it is a cluster setup, click **Browse Nodes** to select other available nodes and perform step 3 to step 6 for each node.

Deleting the Connector from JBoss

Perform the following steps to uninstall the connector from JBoss:

1 Remove contents of the schema JAR or ZIP file from <*SI_HOME_DIR*>`/schema` subdirectory.

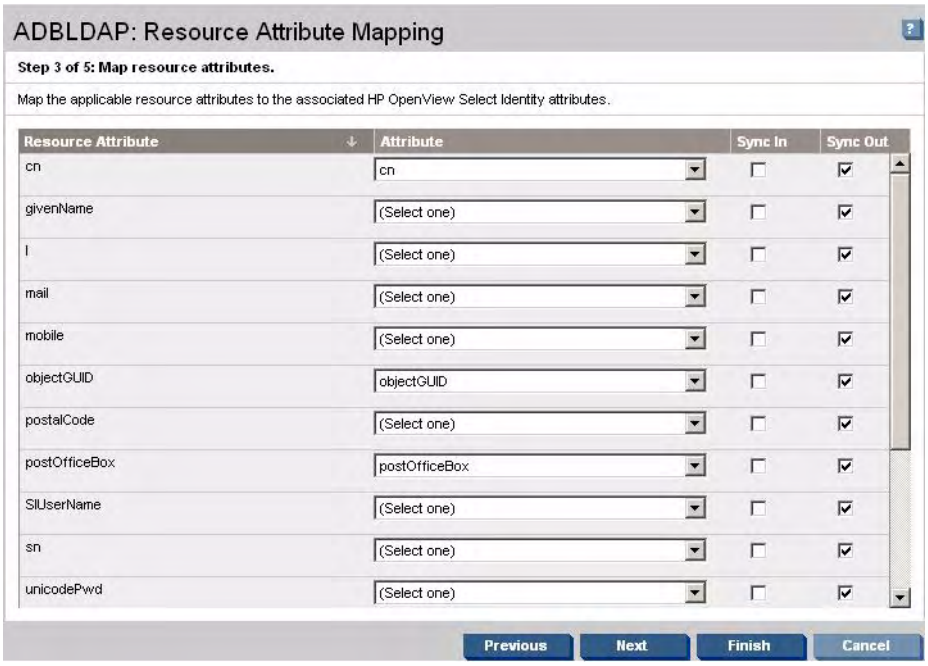
2 Remove the connector's RAR file from <*JBoss_Install_Dir*>`/server/profile/deploy` directory.

3 Remove the connector xml file from <*JBoss_Install_Dir*>`/server/profile/deploy` directory.

4 Restart the JBoss application server.

A Mapping Files

User profile in a resource has a number of attributes, for example, username, first name, last name, and so on. You must map these resource attributes to the Select Identity attributes. Every connector is associated with a mapping file, which contains resource-specific attributes. While mapping resource attributes, Select Identity fetches the attributes from the connector's mapping file, and displays under Resource Attribute column. The Attribute column displays the drop-down boxes, which list all the Select Identity attributes.



A connector may have more than one mapping file. Mapping files are usually XML files or properties files.

In addition to the mapping file, if you configure the connector for reverse synchronization, you must have an XSL transformation file. The mapping file(s) and the transformation file for the connector are usually bundled with Schema JAR or ZIP file. Some of the connectors do not have any Schema JAR or ZIP file bundled. In that case, you must generate the XML mapping file and XSL transformation by using attribute mapping utility of Select Identity.

If reverse synchronization is configured, Select Identity server receives SPML requests that contain the attribute changes. The names of the attributes in the SPML request are defined by the resource. To transform the attribute names to Select Identity attribute names, the request is parsed by Select Identity by using the XSL file.

XML Mapping File

An XML mapping file typically contains the following elements.

- **<Schema>** element — It is the first element in the mapping file. Entire content of the mapping file is contained in this tag. The schema element in the mapping file for TAM connector is shown below.

```
<Schema
  xmlns="urn:oasis:names:tc:SPML:1:0"
  xmlns:spml="urn:oasis:names:tc:SPML:1:0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:dsml="urn:oasis:names:tc:DSML:2:0:core"
xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
  xmlns:concero="http://www.trulogica.com/concero/v21"
xsi:schemaLocation="urn:oasis:names:tc:SPML:1:0 file://C:/SPML/
cs-pstc-spml-schema-1.0.xml"
  majorVersion="1.0" minorVersion="1.0" >
```

- **<providerID>** and **<schemaID>** elements — These elements provide standard elements for header information. These elements in the mapping file for TAM connector is shown below.

```
<providerID providerIDType="urn:oasis:names:tc:SPML:1:0#URN" >trulogica's
URN from oasis</providerID>

 <schemaID
schemaIDType="urn:oasis:names:tc:SPML:1:0#GenericString">TivoliAccessMana
ger</schemaID>
```

- **<objectClassDefinition>** element — This element defines the actions that can be performed on the specified object and the Select Identity-to-resource field mappings for the object. This element consists of the elements:

  — **<properties>** element — It defines the operations that are supported on the object. This can be used to control the operations that are performed through Select Identity. The operation is assigned as the name of the <attr> element and access to the operation is assigned to a corresponding <value> element.

  — **<memberAttributes>** element — It defines the attribute mappings. This element contains <attributeDefinitionReference> elements that describe the mapping for each attribute. Each <attributeDefinitionReference> must be followed by an <attributeDefinition> element that specifies details such as minimum length, maximum length, and so on.

  The <objectClassDefinition> element in the mapping file for TAM connector is shown below.

```
<objectClassDefinition name="User" description="TAM User">
    <properties>
      <attr name="CREATE">
        <value>true</value>
      </attr>
      <attr name="READ">
        <value>true</value>
      </attr>
      <attr name="UPDATE">
        <value>true</value>
```

```
    </attr>
       <attr name="DELETE">
         <value>true</value>
       </attr>
       <attr name="RESET_PASSWORD">
         <value>true</value>
       </attr>
       <attr name="EXPIRE_PASSWORD">
         <value>false</value>
       </attr>
</properties>
    <memberAttributes>
      <!-- This is the Key for the user in TAM and the Directory store -->
      <attributeDefinitionReference name="UserDn" required="true"
        concero:tafield="[First Name] [Last Name]-[GUID]"
concero:resfield="cn"
concero:isDn="true"/>
     <!-- This is Concero UserId, Also user can login into TAM using this
-->    <attributeDefinitionReference name="User Name" required="true"
concero:tafield="User Name" concero:resfield="uid" concero:isKey="true"/>
<attributeDefinitionReference name="Password" required="true"
concero:tafield="Password" concero:resfield="password"/>
<attributeDefinitionReference name="First Name" required="true"
concero:tafield="First Name" concero:resfield="fname" />
<attributeDefinitionReference name="Last Name" required="true"
concero:tafield="Last Name" concero:resfield="lname"/>
    <attributeDefinitionReference name="Description" required="false"
concero:tafield="Description" concero:resfield="description"/>
    </memberAttributes>
</objectClassDefinition>
```

`concero:tafield` attribute of `<attributeDefinitionReference>` element specifies the name of the Select Identity attribute to which the resource attribute needs to be mapped.

`concero:resfield` attribute of `<attributeDefinitionReference>` element specifies the name of the attribute from the resource schema, which has to be mapped to a Select Identity attribute. If the resource does not support an explicit schema (such as UNIX), this can be a tag field that indicates a resource attribute mapping.

- **<attributeDefinition>** element — It defines the properties of each object's attribute. For example, the attribute definition for the Directory attribute defines that it must be between one and 50 characters in length and can contain the following letters, numbers, and characters: a-z, A-Z, 0-9, @, +, and a space. An example of this element in the mapping file of TAM connector is shown below.

```
<attributeDefinition name="User Name" description="userId"
type="xsd:string" >

  <properties>
      <attr name="minLength">
        <value>1</value>
      </attr>
      <attr name="maxLength">
        <value>100</value>
      </attr>
      <attr name="pattern">
        <value><![CDATA[[a-zA-Z0-9@]+]]> </value>
```

```
    </attr>
  </properties>
</attributeDefinition>
```

Properties Mapping File

Some of the connectors, such as Domino connector or agent based Active Directory connector, use properties file to hold mapping information, instead of an XML file. The mapping information in properties file is given in the following format.

Select Identity Attribute | *Resource Attribute*

For example, in `aduser.properties` file of agent based Active Directory connector, the mapping information is given as

`Email | mail` where `Email` is an Select Identity attribute and `mail` is a resource attribute.

Attributes can be concatenated. The attribute names and the separators must

not contain the | delimiter. For concatenation, the format is as follows:

[*Select Identity Attribute*]<separator>[*Select Identity Attribute*] |*Resource Attribute*

For example, in `aduser.properties` file,

```
[City] [addr1]|street
```

XSL Transformation File

If the value of a resource attribute, which has been mapped to Select Identity, is changed at the resource end, the change can be reflected to Select Identity by reverse synchronization.

The Select Identity server receives SPML requests that contain the attribute changes, which are parsed by Select Identity by using the XSL file. In Attribute name mappings section of the XSL file, mapping relationship between the resource attribute and Select Identity attributes are defined.

Resource side attribute is represented as:

```
<xsl:variable name="RES_ATTR0" select="'xxxxxxxxxxx'"/>
```

The mapped attribute for `ATTR0` in Select Identity is represented as:

```
<xsl:variable name="SI_ATTR0" select="'xxxxxxxxxxx'"/>
```

For example, in `domino.xsl`, the XSL file for Domino connector, `ATTR0` is defined as

```
<xsl:variable name="RES_ATTR0" select="'fullname'"/>
<xsl:variable name="SI_ATTR0" select="'SIUSERKEY'"/>
```