HP OpenView Select Identity

Installation Guide
for the LDAP Connector
for Microsoft Active Directory

Software Version: 3.0

July 2004

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.

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://openview.hp.com/

There you will find contact information and details about the products, services, and support that HP OpenView offers.

You can go directly to the support web site at:

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

The support web site includes:

- Downloadable documentation
- Troubleshooting information
- Patches and updates
- Problem reporting
- Training information
- Support program information

contents

1

Installing the LDAP Connector

The LDAP connector enables HP OpenView Select Identity to manage user data in LDAP. It is a one-way connector and pushes changes made to user data in the Select Identity database to a target LDAP server. This connector is generic and can be used to connect to any LDAP data source. The mapping file controls how Select Identity fields are mapped to LDAP fields.

The LDAP connector is packaged in two JAR files: one containing the mapping files called `schema.jar` and one containing the rest of the source code called `TALDAPv3.rar`. These files are located in the `LDAP Active Dir` directory on the Select Identity Connector CD.

Installing on the Web Server

To install the LDAP connector on the Select Identity server, complete these steps.

➤ Note that this procedure installs a connector that supports LDAP for Microsoft Active Directory. Also, the application server in this example is WebLogic 8.1.

Perform this procedure after the Select Identity product installation.

1. If necessary, stop the application server.
2. A `Select_Identity` directory was created on the web server during the product installation. Copy the `TALDAPv3.rar` and `schema.jar` files from the Select Identity Connector CD to this directory.
3. Extract `schema.jar` and remove the `.jar` file when finished.
4. The `startweblogic.cmd` file was edited during the product installation to specify the location of the `Select_Identity` directory. The `startweblogic.cmd` file resides in the `*WebLogic_home*/user_projects/domains/*domain*/` directory on WebLogic 8.1.

   Ensure that this line `set CLASSPATH=%CLASSPATH%` in the `startweblogic.cmd` file references `C:\Select_Identity` in the class path.
5. If necessary, start the web server.
6. Log in to the WebLogic Administrator Console.
7. Navigate to ***My_domain*** **–> Deployments –> Connector Modules**.
8. Click **Deploy a New Connector Module**.
9. Locate and select the `TALDAPv3.rar` file from the list. It is stored in the `Select_Identity` directory.
10. Click **Target Module**.
11. Select the **My Server** (which is your server instance) check box.
12. Click **Continue**. Review your settings.
13. Keep all default settings and click **Deploy**.
14. The Status of Last Action column should display Success.

To test the connector, log in to the Select Identity application and deploy the connector through the Connector pages. See the *HP OpenView Select Identity Administrator Guide* for procedures.

Using Secure LDAP

You need to use secure LDAP (ldaps) to connect to Windows Active Directory for user password changes. Without this, passwords will not be updated in Active Directory by the LDAP connector.

The access URL that you will be using in the Resource Access Information should be changed so that you use ldaps and port 636, such as **ldaps://192.168.1.19:636**.

Download and install the required certificate from the CA for MS Active Directory. Perform the following steps to install this certificate:

1. Access `%JAVA_HOME%\jre\lib\security`.
2. Copy the certificate to this directory.
3. Import the certificate into the `cacerts` keystore, under an alias such as *CA123* using the keytool command.

   ```
   keytool -import -alias CA123 -file adcert.cer -keystore cacerts
   ```

4. Ensure that the certificate is imported by listing it.

   ```
   keytool -list -alias CA123 -keystore cacerts
   ```

2

Understanding the LDAP Mapping File

Each connector is deployed with an XML mapping file that contains the attributes required by the resource application. This file is used to map user account additions and modifications from Select Identity to the system resource. When you deploy a resource through the Select Identity Resources pages, you can review this file.

You can create attributes that are specific to Select Identity through the Attributes pages in the Select Identity client. These attributes can be used to associate Select Identity user accounts with system resources by mapping them to the connector mapping file described in this chapter. This process becomes necessary because 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.

This file does not need to be edited unless you want to map additional attributes to your resource. If attributes and values are not defined in this mapping file, they cannot be saved to the resource through Select Identity.

The LDAP connector provides three mapping files: one for an Active Directory server (`ActiveDir.xml`), one for an iPlanet server (`iPlanet.xml`), and one for ETrust (`CAEtrust.xml`). The files are created in XML, according to SPML standards, and are bundled in a JAR file called `schema.jar`.

General Information

The following operations can be performed in the mapping file:

- Add a new attribute mapping
- Delete an existing attribute mapping
- Modify attribute mappings

Here is an explanation of the elements in the XML mapping files provided by the LDAP connectors:

- **<Schema>**, **<providerID>**, and **<schemaID>**

  Provides standard elements for header information.

- **<objectClassDefinition>**

  Defines the actions that can be performed on the specified object as defined by that name attribute (in the <properties> element block) and the Select Identity-to-resource field mappings for the object (in the <memberAttributes> block). For example, the object class definition for users defines that users can be created, read, updated, deleted, reset, and expired in LDAP.

  - **<properties>**
    Defines the operations that are supported on the object. This can be used to control the operations that are performed through Select Identity. The following operations can be controlled:

    - Create (CREATE)
    - Read (READ)
    - Update (UPDATE)
    - Delete (DELETE)
    - Enable (ENABLE)
    - Disable (DISABLE)
    - Reset password (RESET_PASSWORD)
    - Expire password (EXPIRE_PASSWORD)
    - Change password (CHANGE_PASSWORD)

The operation is assigned as the name of the <attr> element and access to the operation is assigned to a corresponding <value> element. You can set the values as follows:

- true — the operation is supported by the connector
- false — the operation is not supported by the connector and will throw a permission exception
- bypass — the operation is not supported by the connector but will not throw any exception; the operation is simply bypassed

Here is an example:

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

- **<memberAttributes>**
  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.

  Each <attributeDefinitionReference> element contains the following attributes:

  - Name — the name of the reference.
  - Required— if this attribute is required in the provisioning (set to true or false).
  - Concero:tafield — the name of the Select Identity resource attribute.
  - Concero:resfield — the name of the physical resource attribute from the resource schema. If the resource does not support an explicit schema (such as UNIX), this can be a tag field that indicates a resource attribute mapping.

- Concero:isKey — An optional attribute that, when set to true, specifies that this is the key field to identify the object on the resource. Only one <attributeDefinitionReference> can be specified where isKey="true". This key field does not need to be the same as the key field of the identity object in Select Identity.
- Concero:init — An optional attribute that identifies that the attribute is initialized with the value of the attribute passed in from Select Identity.

Here is an example:

```
<memberAttributes>
  <attributeDefinitionReference name="User Name"
   required="true" concero:tafield="[User Name]"
   concero:resfield="cn" concero:isKey="true"
   concero:init="true" />
```

The interpretation of the mapping between the connector field (as specified by the Concero:tafield attribute) and the resource field (as specified by the Concero:resfield attribute) is determined by the connector. The LDAP connector has code to interpret the mappings in one way, as follows:

- The connector attribute names are specified in square braces, like this: [xyz]. The value of attribute xyz is taken from the UserModel during provisioning.
- Composite attributes can be specified in the LDAP connector mapping file. To do this, specify [attr1] xxxx [attr2] as the connector attribute. This specifies that the value of the attr1 and attr2 attributes should be combined with the string xxxx to form a mapping for the specified resource field. LDAP connector has code to handle these composite mappings.

- **<attributeDefinition>**

  Defines the properties of each object's attribute. For example, the attribute definition for the HomeDir attribute defines that it must be between zero and 100 characters in length and can contain the following letters, numbers, and characters: a-z, A-Z, 0-9, @, +, and a space.

  Here is an excerpt from the `ActiveDir.xml` file:

```
<attributeDefinition name="HomeDir" description="User Home
directory" type="xsd:string" >
  <properties>
```

```
    <attr name="minLength">
      <value>0</value>
    </attr>
    <attr name="maxLength">
      <value>128</value>
    </attr>
    <attr name="pattern">
      <value><![CDATA[[a-zA-Z0-9@]+]]> </value>
    </attr>
  </properties>
</attributeDefinition>
```

- **<concero:entitlementMappingDefinition>**

  Defines how entitlements are mapped to users.

- **<concero:objectStatus>**

  Defines how to assign status to a user.

- **<concero:relationshipDefinition>**

  Defines how to create relationships between users.

Active Directory Mapping Information

The following are the attribute mappings supported for Active Directory. These are listed in the `ActiveDir.xml` mapping file. You can add, modify, delete attributes once you are familiar with the contents of this file. See the *HP OpenView Select Identity Connector Developer Guide* for more information about attributes and mapping information.

The Select Identity resource attributes are editable. They reflect the identity information as seen within the Select Identity system.

The physical resource attributes are literal attributes of user accounts on Active Directory. These attributes cannot be changed.

SI Resource Attribute	Active Directory Attribute	Description
User Name	cn	Key field on the resource
Password	UnicodePwd	
First Name	givenname	
Last Name	sn	
User Name	samaccountname	
FirstName + LastName	displayname	
Directory	homeDirectory	
Last Name + First Name	userPrincipalName	
Address 1	streetAddress	
Address 2	postOfficeBox	
City	l	
State	st	
Zip	postalCode	
Title	title	
Business Phone	telephoneNumber	
Home Phone	homePhone	
Profile Path	profilePath	
Script Path	scriptPath	
Description	description	
Disable function	userAccountControl=514	Disables a user
Enable function	userAccountControl=512	Enables a user

3

Uninstalling the LDAP Connector

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

- All resource dependencies have been removed.
- The connector has been deleted through the Select Identity client Connectors pages.

Perform the following to delete a connector:

1. Log in to the WebLogic Server Console.
2. Navigate to ***My_Domain*** **–> Deployments –> Connector Module**.
3. Click the delete icon next to the connector that you want to uninstall.
4. Click **Yes** to confirm the deletion.
5. Click **Continue**.