

HP OpenView Select Identity

**Connector for Netegrity SiteMinder
Version 5.5**

Installation and Configuration Guide

**Connector Version: 3.4
Select Identity Version: 3.3.1**



August 2005

© 2005 Hewlett-Packard Development Company, L.P.

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

© 2005 Hewlett-Packard Development Company, L.P.

No part of this document may be copied, reproduced, or translated into another language without the prior written consent of Hewlett-Packard Company. The information contained in this material is subject to change without notice.

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

Select Identity uses software from the Apache Jakarta Project including:

- Commons-beanutils.
- Commons-collections.
- Commons-logging.
- Commons-digester.
- Commons-httpclient.

- Element Construction Set (ecs).
- Jakarta-poi.
- Jakarta-regexp.
- Logging Services (log4j).

Additional third party software used by Select Identity includes:

- JasperReports developed by SourceForge.
- iText (for JasperReports) developed by SourceForge.
- BeanShell.
- Xalan from the Apache XML Project.
- Xerces from the Apache XML Project.
- Java API for XML Processing from the Apache XML Project.
- SOAP developed by the Apache Software Foundation.
- JavaMail from SUN Reference Implementation.
- Java Secure Socket Extension (JSSE) from SUN Reference Implementation.
- Java Cryptography Extension (JCE) from SUN Reference Implementation.
- JavaBeans Activation Framework (JAF) from SUN Reference Implementation.
- OpenSPML Toolkit from OpenSPML.org.
- JGraph developed by JGraph.
- Hibernate from Hibernate.org.
- BouncyCastle engine for keystore management, bouncycastle.org.

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

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

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

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

Trademark Notices

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.

UNIX is a registered trademark of The Open Group.

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://passport2.hp.com/hpp/newuser.do>

contents

| | | |
|------------------|---|----|
| Chapter 1 | Installing the Connector | 7 |
| | System Requirements. | 8 |
| | Deploying on the Web Application Server. | 9 |
| | Configuring the Application Server. | 10 |
| | Configuring SiteMinder | 11 |
| Chapter 2 | Understanding the Mapping File | 18 |
| | General Information. | 19 |
| | SiteMinder Mapping Information | 23 |
| Chapter 3 | Configuring the Connector | 24 |
| Chapter 4 | Uninstalling the Connector | 29 |

Installing the Connector

The Netegrity SiteMinder connector — hereafter referred to as the SiteMinder connector — enables HP OpenView Select Identity to perform the following tasks in SiteMinder:

- Add, update, and remove users
- Retrieve user attributes
- Enable and disable users
- Verify a user's existence
- Change user passwords
- Reset user passwords
- Retrieve all entitlements
- Retrieve a list of supported user attributes
- Assign and unassign entitlements to and from users

It is a one-way connector and pushes changes made to user data in the Select Identity database to a target SiteMinder server and its configured user store (such as iPlanet LDAP Directory Server). The mapping file controls how Select Identity fields are mapped to SiteMinder fields.

The SiteMinder connector is packaged in the following files:

- `NetSmSchema.jar` – contains the mapping files and a sample `WebAgent.conf` file
- `NetSmConnector.rar` – contains the connector binary files

These files are located in the `Netegrity SiteMinder` directory on the Select Identity Connector CD.

System Requirements

The SiteMinder connector is supported in the following environment:

| Select Identity Version | Application Server | Database |
|-------------------------|--------------------------------|-----------------|
| 3.0.2 | WebLogic 8.1.2 on Windows 2000 | SQL Server 2000 |
| | 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 |
| 3.3 | WebLogic 8.1.4 on Windows 2003 | SQL Server 2000 |
| 3.3.1 | WebLogic 8.1.4 on Windows 2003 | SQL Server 2000 |

This connector is supported with Netegrity SiteMinder, version 5.5, on Windows 2000 and Solaris 8.

Deploying on the Web Application Server

To install the SiteMinder 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 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\schemas` folder. (This subdirectory may already exist.)
- 3 Copy the `NetSmConnector.rar` file from the Select Identity Connector CD to the connector subdirectory.
- 4 Extract the contents of the `NetSmSchema.jar` file (on the Select Identity Connector CD) to the schema subdirectory.
- 5 Ensure that the CLASSPATH environment variable in the WebLogic server startup script references the schema subdirectory.
- 6 Modify the mapping file, if necessary. See [Understanding the Mapping File on page 18](#) for details.
- 7 Start the application server if it is not currently running.
- 8 Log on to the WebLogic Server Console.
- 9 Navigate to *My_domain* → **Deployments** → **Connector Modules**.
- 10 Click **Deploy a New Connector Module**.
- 11 Locate and select the `NetSmConnector.rar` file from the list. It is stored in the connector subdirectory.
- 12 Click **Target Module**.
- 13 Select the **My Server** (your server instance) check box.
- 14 Click **Continue**. Review your settings.
- 15 Keep all default settings and click **Deploy**. The Status of Last Action column should display Success.

After installing the connector, see [Configuring the Connector on page 24](#) about registering and configuring the connector in Select Identity.

Configuring the Application Server

The SiteMinder connector requires that you copy the SiteMinder Java API JAR files to the application server. You must also copy the SiteMinder shared libraries to the application server; these libraries are called by the SiteMinder Agent APIs, which are used by the connector for provisioning. Then, you must set the application server's CLASSPATH and PATH environment variables to reference the files. Complete the following steps:

- 1 Create a directory named `SiteMinderConnectorLib` in the Select Identity home directory.
- 2 Copy the SiteMinder Java API JAR files, which are distributed with SiteMinder, to the `SiteMinderConnectorLib` subdirectory. These files include the following:

- `Smjavaagentapi.jar`
- `Smjavasdk2.jar`

The default location of these files on the SiteMinder system is `C:\Program Files\Netegrity\SiteMinder\SDK\java\`.

- 3 Add the JAR files to the application server startup script. Here is an example:

```
set CLASSPATH=%WEBLOGIC_CLASSPATH%;
%POINTBASE_CLASSPATH%;%JAVA_HOME%\jre\lib\rt.jar;
%WL_HOME%\server\lib\webservices.jar;
c:\SelectIdentity\SiteMinderConnectorLib\smjavaagentapi.jar;
c:\SelectIdentity\SiteMinderConnectorLib\smjavasdk2.jar;
%CLASSPATH%
```

- 4 Copy the following files to the folder `SiteMinderConnectorLib` subdirectory.

- `smjavasdk2.dll`
- `smjavaagentapi.dll`

These files can be copied from the SiteMinder system, where they typically reside in `C:\Program Files\Netegrity\SiteMinder\SDK\bin\`.

5 On Solaris, copy the following files to a directory on the server:

- libsmagentapi.so
- libsmjavaagentapi.so

Then, set the `LD_LIBRARY_PATH` environment variable to this directory. This environment variable should be available for the use by the application server.

6 Add the `SiteMinderConnectorLib` folder to the `PATH` variable in the startup script.

Configuring SiteMinder

The SiteMinder connector provisions users in SiteMinder using the SiteMinder Java APIs that are bundled in the `NetSmConnector.rar` file. Therefore, no third-party libraries are required. However, you must configure SiteMinder to work with the connector. The following steps explain the necessary configuration changes.



Note that the steps assume the following environment:

- SiteMinder Policy Server 5.5 on Solaris 8 and Windows 2000 Server
- SiteMinder SDK 5.5 on Solaris 8 and Windows 2000 Server
- SiteMinder WebAgents configured for the following web servers:
 - iPlanet 6.0 on HP-UX11i, Solaris 9, and Windows
 - Microsoft IIS on Windows
- iPlanet Directory Server 5.1 on Solaris 9 and Windows

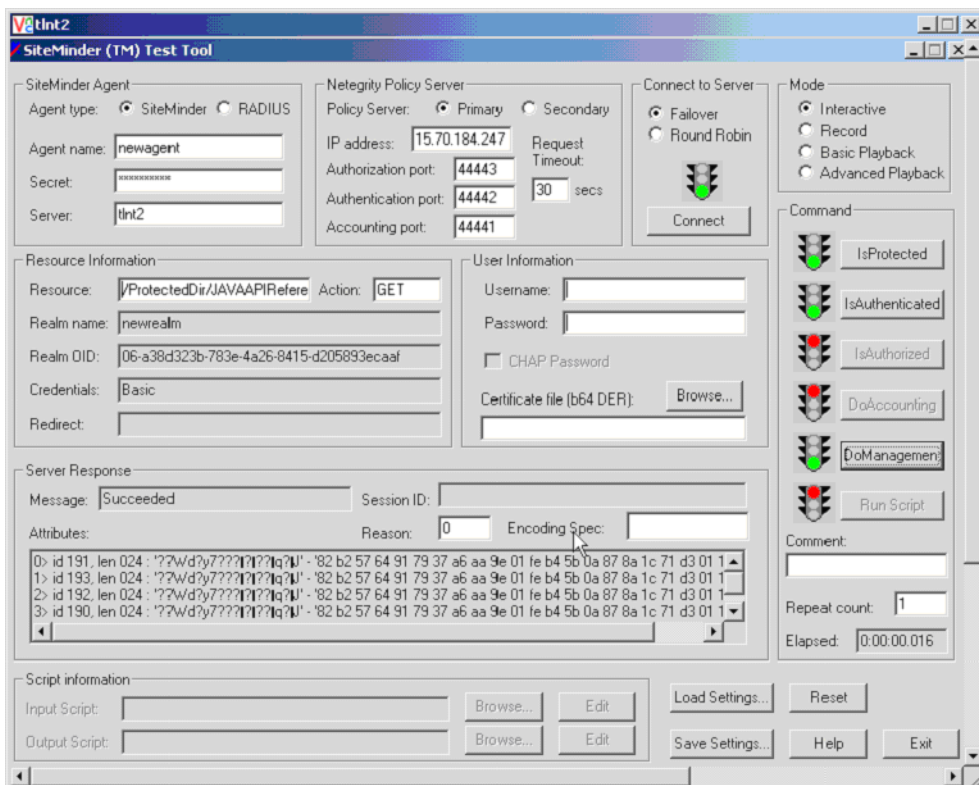
Complete these steps to configure SiteMinder for use with the connector:

- 1 Install the SiteMinder Policy Server and WebAgent as explained in the SiteMinder Installation Guides. Note that for the correct operation of SiteMinder DMS APIs, which are used in the SiteMinder Connector, it is assumed that the SiteMinder WebAgent is installed on the same machine

as the Select Identity server. The following configuration steps provide instructions using the iPlanet Directory Server as the SiteMinder user store.

▶ When installing the WebAgent, you must enable support for 4.x WebAgents, which will require providing a shared secret between the WebAgent and the Policy Server. See [Step 5 on page 13](#) for more information on generating the shared secret.

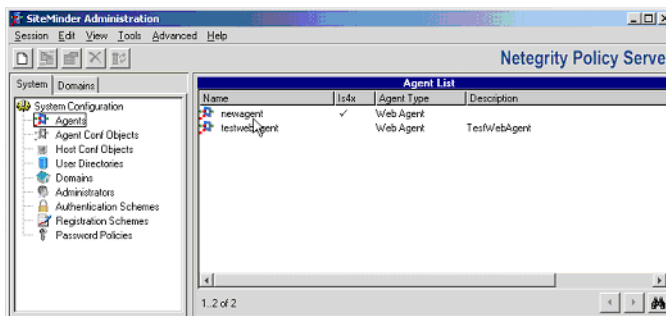
- 2 On the SiteMinder Policy Server, select **Start** → **Program Files** → **SiteMinder** → **SiteMinder Test Tool**.
- 3 Enter the Policy Server IP address, login and password, and ensure that the connection to the SiteMinder Policy Server is functioning properly. Refer to the SiteMinder documentation if issues arise for this step (or steps 1 and 2).



- 4 Locate the `WebAgent.conf` file that was extracted from the `NetSmSchema.jar` file. Update the `WebAgent.conf` file by comparing it to the working `WebAgent.conf` and `SmHost.conf` files for the configured `WebAgent`. Specifically, update the following variables: `hostname`, `defaultagentname`, `agentname`, `polycyserver`, and `sharedsecret`. Here are example values:

```
hostname="sisun2"
defaultagentname="sisun2agent"
agentname="sisun2agent, 15.70.184.29"
enablefailover="NO"
maxsocketsperport="20"
minsocketsperport="2"
newsocketstep="2"
requesttimeout="10"
polycyserver="15.70.184.29,44441,44442,44443"
sharedsecret="{RC2}zrd6db4uwkhF77zuaEKYDN4avJnuuobIItgJ2JRuOV1b
Fl/m5wBd/2oIL/f7PtAID5XcNtXNodLAPUuKb9m1RxOvOFF9U975B2oS/
ztqSXz1fXIQc2Xo3u019fSTIU9GbkLcG1/
+zj6JV0WbssfvpIWodxch6lCMc4AkOZG8YLZzaN11zQbE/O6KLGagn9j"
```

- 5 To update the shared secret, complete the following steps:
- Click **Agents** on the left side of the SiteMinder Administration window.



- Double-click the configured agent on the left side of the window. The SiteMinder Agent Dialog displays.

- c Select the **Support 4.x Agent** option and provide the values of the shared secret, which are used between this agent and the Policy Server.

- d Click **OK**.
- 6 Configure SiteMinder such that it knows the user password attribute name in the configured LDAP User Store.
- a On the left side of the window, select the configured user store.
 - b Right-click to display the User Directory Properties dialog.

- c Configure the Directory Setup tab as shown in the following snapshot:

The screenshot shows the 'User Directory Properties' dialog box for a 'Planet LDAP User Store'. The 'Directory Setup' sub-tab is active, showing the following configuration:

- NameSpace:** LDAP:
- Server:** 15.70.184.247
- LDAP Search:**
 - Root:** dc=india, dc=hp, dc=com
 - Scope:** Subtree
 - Max time:** 30 seconds
 - Max results:** 0
- LDAP User DN Lookup:**
 - Start:** (empty)
 - End:** (empty)
 - Example:** (button)

Buttons at the bottom include 'View Contents...', 'OK', 'Cancel', and 'Apply'. The status bar at the bottom reads 'User Directory iPlanet LDAP User Store'.

- d Configure the Credentials and Connection tab as shown here:

The screenshot shows the 'User Directory Properties' dialog box for 'iPlanet LDAP user Store'. The 'Credentials and Connection' tab is active. The 'Name' and 'Description' fields both contain 'iPlanet LDAP user Store'. The 'Administrator Credentials' section has the following settings:

- Require Credentials
- *Username: cn=Directory Manager
- *Password: [masked]
- *Confirm Password: [masked]

Other options include:

- Run in Authenticated User's Security Context
- Secure Connection

Buttons at the bottom include 'View Contents...', 'OK', 'Cancel', and 'Apply'. The status bar at the bottom reads 'User Directory iPlanet LDAP user Store'.

- e Configure the User Attributes tab as shown here:

SiteMinder User Directory Dialog

User Directory Properties HELP

*Name: iPlanet LDAP User Store Description: iPlanet LDAP User Store

Directory Setup Credentials and Connection **User Attributes**

These are the names of directory user profile attributes SiteMinder will use. These attributes must be available in the directory and must not be used by any other application. Attributes marked R must be readable; attributes marked RW must be read-write.

Universal ID (R):

Disabled Flag (RW):

Password Attribute (RW):

Password Data (RW):

Anonymous ID (RW):

Email (R):

Challenge/Response (RW):

View Contents...

OK Cancel Apply

User Directory iPlanet LDAP user Store

- 7 Ensure that SiteMinder continues to function correctly and operation continues to happen smoothly by repeating [Step 2](#).

For troubleshooting, use the SiteMinder Test Tool. Or, to troubleshoot connectivity and configuration of Policy Server, WebAgent, and LDAP User Store, execute the sample Java programs that are provided with the SiteMinder SDK. View the log files for details. The return code and error message may help to troubleshoot any issue that might arise.

Understanding the Mapping File

The SiteMinder connector is deployed with the `netSm.xml` mapping file, which describes the attributes required by the system. The file is created in XML, according to SPML standards, and is bundled in a JAR file called `NetSmSchema.jar`. The mapping file is used to map user account additions and modifications from Select Identity to the SiteMinder resource. When you deploy a resource using the Resources page of the Select Identity client, you can review this file.

You can create attributes that are specific to Select Identity using the Attributes page in the Select Identity client. These attributes can be used to associate Select Identity user accounts with system resources by editing the connector mapping file described in this chapter. This process becomes necessary because, for example, a single attribute “owner” can have a different name on different resources, such as “user” for UNIX, “UID” for a database, and “ownerID” 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.

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 file:

- **<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, and reset in SiteMinder.

- **<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)
- 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
- bypass — the operation is not supported by the connector

Here is an example:

```
<objectClassDefinition name="User" description="User Info">
  <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. In general, the attribute assigned to tafield should be the same as the physical resource attribute, or at least the connector attribute. For example, it is recommended to have the following:

```
<attributeDefinitionReference name="FirstName"
  required="false" concero:tafield="[givenname]"
  concero:resfield="givenname" concero:init="true"
  concero:isMulti="true"/>
```

instead of this:

```
<attributeDefinitionReference name="FirstName"
required="false" concero:tafield="[FirstName]"
concero:resfield="givenname" concero:init="true"
concero:isMulti="true"/>
```

- **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.

Also, the attribute name may be case-sensitive; for example, if the attribute is defined in all uppercase letters on the resource, be sure to specify it in all uppercase letters here.

- **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 `<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.

Note that for a key field mapping where `isKey="true"` and `tafield` is not assigned the `UserName` attribute, `UserName` should not be used in any other mapping. That is, `UserName` can be assigned to `tafield` only in cases where it is mapped to the key field in the resource. Example:

```
<attributeDefinitionReference name="UserName"
required="true" concero:tafield="[UserName]"
concero:resfield="uid" concero:isKey="true"
concero:init="true"/>
```

- **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="UserName"
    required="true" concero:tafield="User Name"
    concero:resfield="schema=newxj,table=T_USERINFO,
    column=USERSTATUS" concero:isKey="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 SiteMinder 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 SiteMinder 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. SiteMinder connector has code to handle these composite mappings.

- **<attributeDefinition>**

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.

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

```
<attributeDefinition name="UserName" description="UserName"
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>
```

- **<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.

SiteMinder Mapping Information

The SiteMinder connector supports the following identify information to be provisioned on the SiteMinder system. The User Name attribute is unique the the primary key for the user. You can add, modify, or delete attributes once you are familiar with the contents of this file. You can edit the Select Identity resource attributes; they reflect the identity information as seen in Select Identity. The SiteMinder attributes are literal attributes of user accounts on the SiteMinder system. These attributes cannot be changed.

| Select Identity Resource Attribute | Connector Attribute | SiteMinder Attribute | Description |
|---|----------------------------|-----------------------------|--|
| User Name | Username | uid | Unique field |
| Password | Password | userpassword | User's password |
| First Name | Firstname | givenname | First name |
| Last Name | Lastname | sn | Surname |
| Email | Email | mail | Email ID |
| Common Name | Firstname | cn | Common name |
| AccountLockStatus | AccountLockStatus | nsaccountlock | The default value provided in the mapping file |

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:

[Home](#) > [Connectors](#) > **Modify Connector : MarchSMConnector**



| Connector Information | |
|-----------------------|------------------|
| * Connector Name: | MarchSMConnector |
| * Pool Name: | eis/NetSm |

- 2 Deploy a resource that uses the newly created connector. On the Resources home page, click the **Deploy New Resource** button. When configuring the resource, refer to the following table for parameters specific to this connector:

| Field Name | Sample Values | Description |
|-------------------------------------|-------------------------------|--|
| Resource Name | SiteMinder | Name given to the resource. |
| Resource Type | SiteMinder | The connector that was deployed in Step 1 . |
| Authoritative Source | No | Whether this resource is a system that is considered to be the authoritative source for user data in your environment. You must specify No because the connector cannot synchronize account data with the Select Identity server. |
| Associate to Group | Selected | Whether the system uses the concept of groups. For this connector, select this option. |
| SiteMinder Admin | SiteMinder | SiteMinder superuser account name. |
| Admin Password | Admin123 | SiteMinder superuser password. |
| SiteMinder User Directory | SiteMinder LDAP_User_Store | Name of the user directory as configured in SiteMinder. |
| SiteMinder Root Organizational Unit | Ou=smtesting | The name of the node in the LDAP user store. (Additional OUs are provided below this level.) |
| OrgUnit Object Class | Top, organizationalunit | Object class of the Org Unit in LDAP user store. |
| Group Object Class | Top, groupofuniquenames | Object class of the groups in LDAP user store. |

| Field Name | Sample Values | Description |
|-------------------|--|--|
| Group Suffix | Ou=Groups | Group suffix name. This is one level below the "SiteMinder Root Organizational Unit" in the LDAP user store. |
| User Object Class | Top, person, organizationalperson, inetorgperson | Object class of the user entry in LDAP user store. |
| User Suffix | Ou=People | User suffix name. This is one level below the "SiteMinder Root Organizational Unit" in the LDAP user store. |
| Mapping File | netism.xml | The attribute mapping XML file. |

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 connector, the Basic Info page of the resource properties will look similar to this:

| Resource Information | |
|--------------------------|--|
| * Resource Name: | MarchSmResource |
| Resource Description: | Sample SiteMinder Resource |
| * Resource Type: | MarchSMConnector |
| * Authoritative Source: | <input type="radio"/> Yes <input checked="" type="radio"/> No |
| * Delete User: | <input checked="" type="radio"/> Yes <input type="radio"/> No |
| Reconciliation Workflow: | <input type="text"/>  |
| Resource Owner: | isisa  |

The Additional Info page will look similar to this:

| Resource Information | |
|---|-------------------------------------|
| Resource Name: | MarchSmResource |
| <input checked="" type="checkbox"/> Manage User | |
| Associate to Group: | <input checked="" type="checkbox"/> |

The Access Info page will look similar to this:

| Resource Access Information | |
|--|---|
| * Resource Name: | MarchSmResource |
| * Siteminder Admin: | <input type="text" value="SiteMinder"/> |
| * Admin Password: | <input type="text" value="admin123"/> |
| * Siteminder User Directory: | <input type="text" value="localiplanet_dir"/> |
| * Siteminder Root Organizational unit: | <input type="text" value="ou=smtesting"/> |
| OrgUnit Object Class: | <input type="text" value="top,organizationalUnit"/> |
| * Group Object Class: | <input type="text" value="top,groupofuniquenames"/> |
| * Group Suffix: | <input type="text" value="ou=Groups"/> |
| * User Object Class: | <input type="text" value="top,person,organizationalPerson,inetor"/> |
| * User Suffix: | <input type="text" value="ou=People"/> |
| * Mapping File: | <input type="text" value="netsm.xml"/> (View) |

- 3 Create the AccountLockStatus attribute. This attribute is used internally by the connector to enable or disable the user in the SiteMinder LDAP user store. After a user is disabled, he or she will not be able to log in to the iPlanet LDAP resource. The mapping file contains an attribute called nsaccountlock to which you will map this attribute.

Create other 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.

After you configure attributes for the SiteMinder resource, the attributes look similar to this:

| Resource Attribute | Min.Length | Max.Length | Mapped To | Authoritative |
|-------------------------------|------------|------------|-------------------------------|-------------------------------------|
| NetSm3.3Resource_ENTITLEMENTS | 1 | 255 | NetSm3.3Resource_ENTITLEMENTS | <input checked="" type="checkbox"/> |
| NetSm3.3Resource_KEY | 1 | 255 | NetSm3.3Resource_KEY | <input checked="" type="checkbox"/> |
| [AccountLockStatus] | 1 | 64 | AccountLockStatus | <input type="checkbox"/> |
| [Email] | 1 | 64 | Email | <input type="checkbox"/> |
| [FirstName] | 1 | 64 | FirstName | <input type="checkbox"/> |
| [LastName] | 1 | 64 | LastName | <input type="checkbox"/> |
| [Password] | 1 | 64 | Password | <input type="checkbox"/> |
| [UserName] | 1 | 64 | UserName | <input type="checkbox"/> |

- 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.

Note the following when creating Service views for the SiteMinder connector:

- Do not add the AccountLockStatus attribute to any Service view; it is for internal use by the connector.
- Do not add the password attribute as part of the Service view; it is used for user-modification.

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 using the Select Identity client Connectors pages.

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**.