HP OpenView Select Identity

Connector for BMC CONTROL-SA ESS Systems, Version 3.1.03

Installation and Configuration Guide

Connector Version: 3.2
Select Identity Version: 3.3

April 2005

Legal Notices

Warranty

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

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

Restricted Rights Legend

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

Hewlett-Packard Company
United States of America

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

Copyright Notices

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

Select Identity uses software from the Apache Jakarta Project including:

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

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

Additional third party software used by Select Identity includes:

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

Trademark Notices

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

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

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

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

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

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

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

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

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

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

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

Support

Please visit the HP OpenView web site at:

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

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

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

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

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

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

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

To find more information about access levels, go to:

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

To register for an HP Passport ID, go to:

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

contents

1

Installing the Connector

The CONTROL-SA connector enables HP OpenView Select Identity to perform the following tasks in BMC CONTROL-SA Enterprise Security Station (ESS) systems:

- 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

The CONTROL-SA connector is a one-way connector and pushes changes made to user data in the Select Identity database to a target CONTROL-SA server. The mapping file controls how Select Identity fields are mapped to CONTROL-SA fields.

The CONTROL-SA connector is packaged in the following files. These files are located in the `BMC Control-SA ESS 3/ess` directory on the Select Identity Connector CD.

- `EssConnector.rar` — contains the model DDL and DMLs for the ESS Child task status tables
- `essschema.jar` — contains the `ControlSaEss.xml` mapping file, which contains the attribute mapping file to control how Select Identity fields are mapped to BMC CONTROL-SA fields
- `essclient-scripts.zip` — contains scripts for testing connectivity to ESS server for web applications running on Windows
- `essclient-scripts.tar.gz` — contains scripts for testing connectivity to ESS server for web applications running on Solaris
- `essmodel.zip` — contains the ESS connector schema for web applications running on Windows
- `essmodel.tar.gz` — contains the ESS connector schema for web applications running on Solaris

System Requirements

The CONTROL-SA connector is supported in the following environment:

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

The connector is supported with CONTROL-SA 3.1.03 on Solaris 9.

Deploying on the Web Application Server

To install the CONTROL-SA connector on the Select Identity server, complete these steps:

1. Create a subdirectory in the Select Identity home directory where the connector's RAR file will reside. For example, you could create the `C:\Select_Identity\connectors` folder on Windows. (A connector subdirectory may already exist.)
2. Copy the `EssConnector.rar` file from the Select Identity Connector CD to the connector subdirectory.
3. Create a schema subdirectory in the Select Identity home directory where the connector's mapping file(s) will reside. For example, you could create the `C:\Select_Identity\schema` folder. (This subdirectory may already exist.)
4. Extract the contents of the `essschema.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. Start the application server if it is not currently running.
7. Log on to the WebLogic Server Console.
8. Navigate to ***My_domain*** → **Deployments** → **Connector Modules**.
9. Click **Deploy a New Connector Module**.
10. Locate and select the `EssConnector.rar` file from the list. It is stored in the connector subdirectory.
11. Click **Target Module**.
12. Select the **My Server** (your server instance) check box.
13. Click **Continue**. Review your settings.
14. Keep all default settings and click **Deploy**. The Status of Last Action column should display Success.
15. Modify the mapping file, if necessary. See Understanding the Mapping File on page 14 for details.

After installing the connector, refer to Configuring the Connector on page 20 to register and configure this connector in Select Identity.

Installing the ESS Client on the Application Server

The CONTROL-SA connector uses the ESS client to communicate with the ESS server over a secure connection. You must first install the ESS client on the Select Identity application server before using the connector within Select Identity. The client is available from BMC Software (**http://www.bmc.com**). Follow the installation steps to install the client, then use the `ESSClient` executable (Solaris) or `ESSClient.exe` file (Windows) with the connector.

Installing Scripts on the Application Server

Select Identity provides several scripts to communicate with the BMC CONTROL-SA ESS server. These scripts can be used to test connectivity with the ESS server. They are located on the Select Identity Connector CD in the `ess` directory. Perform one of the following steps to install connector scripts on the application server:

- If running the application server on a Windows server, extract the contents of `essclient-scripts.zip` to a local directory.
- If running the application server on a Solaris server, extract the contents of `essclient-scripts.tar.gz` to a local directory.

The following scripts are provided:

- `dotest` – Tests the connection to ESS.
- `adduser` – Adds a new user.
- `deleteuser` – Deletes an existing user.
- `essenv` – Sets up the environment for other scripts to run.
- `getAllJobCodes` – Lists all of the job codes of a user.
- `getLinkStatus` – Provides the status of a previously issued job code assignment request.
- `getuser` – Provides the details of a user on ESS.
- `linkuser` – Assigns a job code to a user.

- `listAllJobCodes` – Lists all job codes in the server.
- `listjobcodes` – Lists all job codes of the user.
- `modifyuser` – Modifies the attributes of an existing user.
- `restore` – Enables the user.
- `revoke` – Disables the user.
- `unlinkuser` – Unassigns the job code from a user.

Testing Connectivity to the ESS Server

After installing the client and the scripts, you must test the connection to the ESS server. This step ensures that the CONTROL-SA connector can communicate with and provision into the CONTROL-SA ESS server.

If you are running the Select Identity server and the connector on Windows or Solaris, perform the following steps to test connectivity. You will need the connection information from the ESS client installation.

1. Open and edit the `essenv` script.

   ESS_HOME – folder where the `Essclient` executable is located.

   ESS_HOST – host name of the server running ESS server.

   ESS_PORT – port number of the orbix daemon on the host, usually 1570.

   ESS_USER – user account on the server that has privileges for provisioning other users.

   ESS_PASSWORD – password of the above user.

   ESS_OWNER – owner of the ESS schema.

2. Source the `essenv` file so that the above environment is set.
3. Run the `dotest` script. This will establish a connection to the ESS server and ensure that the account is valid. If this test fails, check the environment or the ESS client installation.

Installing the ESS Task Schema

Entitlements in Select Identity are mapped to job codes in ESS. Each job code assignment to a user will cause a set of child processes to be started by the ESS server. The CONTROL-SA connector uses two tables — EssTask and EssChildTask — to track the status of these child processes.

The schema of these tables is provided in the `essmodel.zip` and `essmodel.tar.gz` files. You need to register this schema in the database, if not already done by the DDLs provided by Select Identity. Use `essmodel.dll` to create these tables.

The CONTROL-SA connector retrieves the job codes from the ESS server as follows:

- Retrieves them from ESS dynamically. The list of job codes can be retrieved from the server each time a new user is created. This is useful when the total number of job codes is less and may change often. This method uses the ESS client to retrieve the job codes.
- Retrieves them from a database table. This is preferable in cases where the list is large. A JDBC call is made to retrieve this list from a database table. The connector has configuration parameters to specify the following information:

  — Data source JNDI name of the Connection to the database

  — Table name that holds the job code information

  — Column name with the names of the job codes

  This table could be an existing table that the connector can access. If there is no such table, you can add a table locally and put in all of the job codes.

  The distribution is provided with a sample DDL for this table, `essEntitlements.ddl`, along with sample job code data in `essEntitlements.dml`.

Registering a Unique ID for ESS Schema Tables

Use the `createuid.sql` file (for SQL Server) or `createuid_oracle.sql` file (for Oracle) to register the seed for the next ID for the EssTask and EssChildTask tables in the schema. These files are located in `essmodel.zip` (for SQL Server) or `essmodel.tar.gz` (for Solaris). The connector uses these files to add rows to these tables.

2

Understanding the Mapping File

The CONTROL-SA connector is deployed with the `ControlSaEss.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 `essschema.jar`. The mapping file is used to map user account additions and modifications from Select Identity to the system 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 “username” can have a different name on 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.

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 CONTROL-SA connector:

- **<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 BMC.

  - **<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

— bypass — the operation is not supported by the connector

Here is an example:

```
<objectClassDefinition name="User" description="BMC 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="UserName"
   required="true" concero:tafield="[UserName]"
   concero:resfield="_99_default_userid"
   concero:isKey="false"
   concero:init="false" />
```

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 CONTROL-SA 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 CONTROL-SA 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. CONTROL-SA 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 `ControlSaEss.xml` file:

```
<attributeDefinition name="Email" description="Email"
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.

CONTROL-SA Mapping Information

The following are the attribute mappings supported for BMC CONTROL-SA. These are listed in the `ControlSaEss.xml` mapping file. 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 physical resource attributes are literal attributes of user accounts in CONTROL-SA. These attributes cannot be changed.

Select Identity Resource Attribute	CONTROL-SA Attribute	Description
Person Number	user_id	Key field on the resource
UserName	_99_default_userid	
Password	password	

Select Identity Resource Attribute	CONTROL-SA Attribute	Description
Last Name, First Name, Middle Name	user_name	
Email	_99_user_email	
Company Code	_99_COMPANY_CODE	
Cost Center	_99_COST_CENTER	
Company Code, Cost Center	_99_Department	
Hierarchy Code	org_parent	
Business Phone	_99_Business_Number	
Home Phone	_99_Home_Number	
Home Fax	_99_Home_Fax	
SSN	_99_user_ssn	
Default Shell	_99_default_shell	
Street Address	_99_Address	
MailCode	_99_Mailbox_ID	
City	_99_City	
State	_99_State	
Client Status	_99_user_status	
Client Type	_99_user_type	
groupname	jc_name	

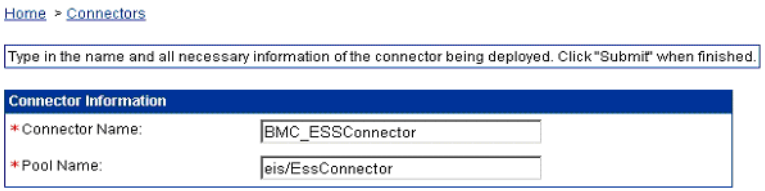
3

Configuring the Connector

After you deploy the connector on the application server, you must configure Select Identity to use the connector by deploying it in the Select Identity client. The following provides an overview of the procedures you must complete in order to deploy your connector. It also provides connector-specific information you must provide when configuring Select Identity to use the connector.

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

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



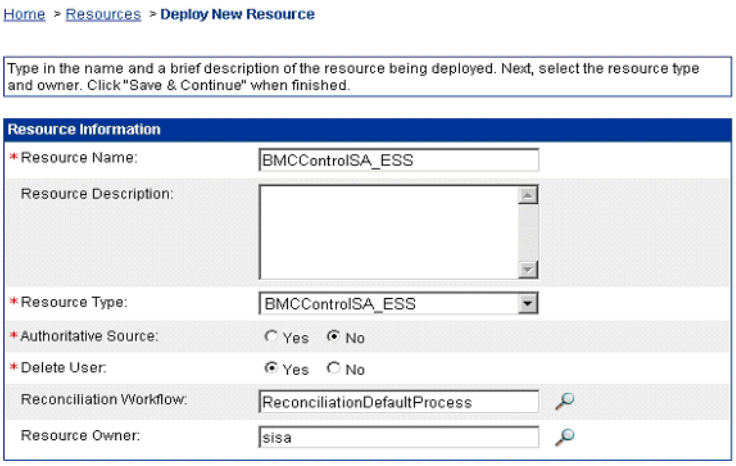
2 Deploy a resource that uses the newly created connector. On the Resources home page, click the **Deploy New Resource** button. When configuring the resource, refer to the following table for parameters specific to this connector:

Field Name	Sample Values	Description
Resource Name	control-sa_server	Name given to the resource.
Resource Type	Control-SA	The connector that was deployed in Step 1 on page 20.
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 the CONTROL-SA connector, select this option.
Admin User Name	admin_user	The administrative login name.
Admin Password	password123	Administrative account password.
Name of the ESS Schema owner	essid	The Enterprise Security Station (ESS) schema owner.
Host Name of the ESS Server	server.company.com	The ESS server host.
Port Number of the ESS Server	1570	The port number of the orbix daemon.
ESSClient	/essclient.exe	The location of the ESS client on the web server.

Field Name	Sample Values	Description
Max Wait Time	2	The maximum time that the server should wait for a response in minutes. This is used for job code assignments.
Entitlements Table Data Source	jdbc/TruAccess	The data source for entitlements information.
Entitlements Table	essEntitlements	The database table for entitlements.
Column Name	entitlements	The database column for entitlements.
Mapping File	ControlSAEss.xml	The attribute mapping file for the resource.

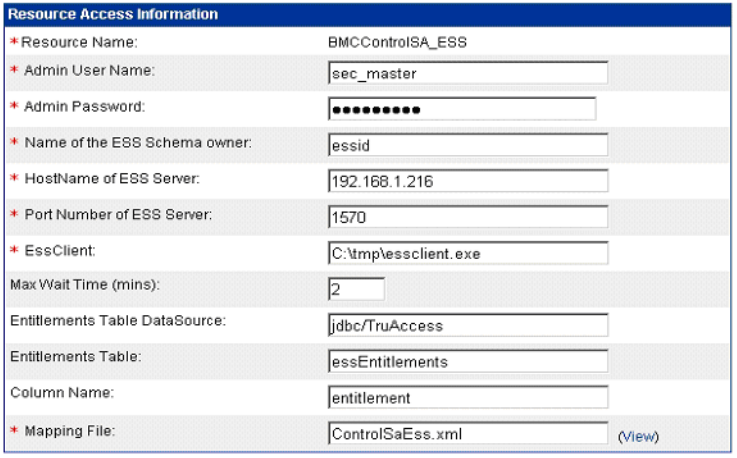
Complete the steps in this procedure as described in the “Resources” chapter of the *HP OpenView Select Identity Administrator Guide*. After

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



The Additional Info page of the resource properties will look similar to this:

The Access Info page of the resource properties will look like this:



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

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.

4

Uninstalling the Connector

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

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

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**.

After deleting the connector, you can remove the scripts as well. Remove them from the directory on the Select Identity server where they were extracted (see Installing Scripts on the Application Server on page 10).