

HP OpenView Select Identity

Installation Guide

Software Version: 3.0

**UNIX® (Sun Solaris) and Windows®
Operating Systems**



July 2004

© Copyright 2004 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

© Copyright 2002, 2004 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.

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.

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, Institut 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

Chapter 1	Welcome to Select Identity	8
	System Architecture	10
	Product Documentation	12
Chapter 2	Installing Select Identity	14
	Overview of System Requirements	15
	For the Database Server	15
	For the Web Application Server	17
	For the Select Identity Interface	17
	Configuring the Database Server	18
	For Microsoft SQL Server	18
	For Oracle	21
	Copying Files to the Web Application Server	23
	Configuring the Web Application Server	25
Chapter 3	Uninstalling Select Identity	34
	Uninstalling Select Identity from the Web Server	34
	Deleting the lmz File	34
	Deleting the Connectors	35
	Deleting the Data Source	35
	Deleting the Connection Pool	35
	Deleting the Mail Session	36
	Uninstalling the Select Identity Database	36
	Uninstall From Microsoft SQL Server	36
	Uninstall From Oracle	37

Appendix A Troubleshooting	38
Appendix B Logging	41
Appendix C Configuring TruAccess.properties	44
Glossary	51
Index	60

Welcome to Select Identity

HP OpenView Select Identity is the first truly scalable solution for managing identity within and between large enterprises. The Select Identity solution automates the process of provisioning and managing user accounts and access privileges across platforms, applications, and corporate boundaries. Along with robust workflow, user self-service, reporting, and delegated administration capabilities, Select Identity is the most comprehensive identity management system available.

Select Identity is designed to address the formidable challenges of managing identity within complex, multi-organizational business processes. Traditional identity management systems employ the user-centric model of roles to distribute access to users. In an extended enterprise, roles proliferate exponentially to accommodate the large number of complex business relationships that exist between users, organizations, resources and security policies.

Select Identity's Contextual Identity Management (CIM)™ is a dramatic advancement in identity management. CIM provides a service-centric approach to managing identity. In any company, its employees, customers, and partners participate in a number of services or business processes that comprise the operation of the company. For example, these processes might include "order processing" or "accounts receivable." Each Service may consist of a number of applications or resources that require unique access privileges

depending on the Service, its participants, and corporate policy. CIM incorporates these complex relationships and leverages them to automate the tasks associated with managing identity, including the following:

- Provisioning accounts and privileges
- Approving workflows
- Delegating administrative rights
- Enforcing security policy
- Reporting

CIM mitigates the limitations of the traditional role and rule-based identity management, enabling scalability throughout the extended enterprise while reducing deployment times and management costs.

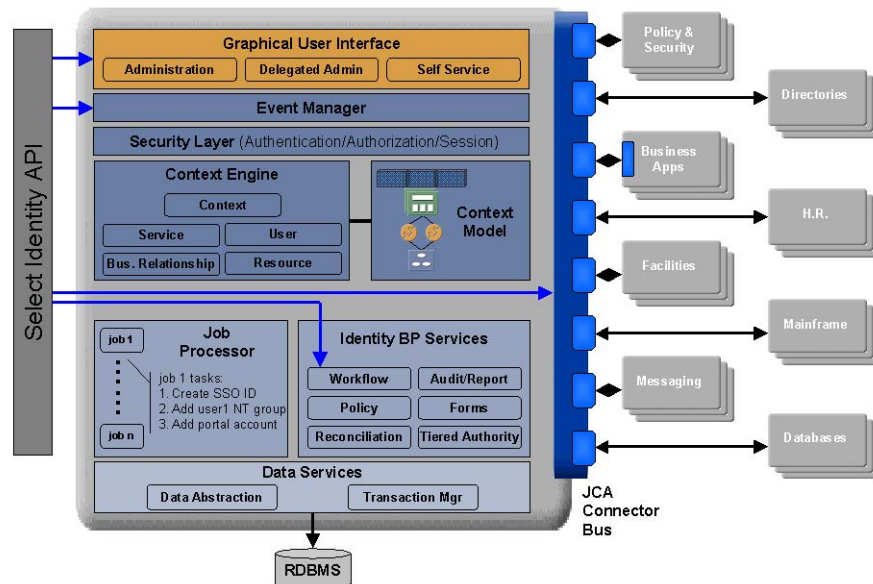
Key features of Select Identity include the following:

- Centralized Management – Provides a single point of control for the management of users and entitlements
- Provisioning – Automates the creation, update, and deletion of accounts and entitlements on information systems across the enterprise
- Extreme Delegation – Enables administrative rights to be distributed to multiple tiers of functional departments, customers, and partners
- User Self-Service – Enables end users to initiate access to services, change passwords, set password hints, and update general identity information through a simple web interface
- Workflow – Automates identity-related processes such as access approval and provisioning, and integrates these processes with other business processes
- Password and Profile Management – Manages and distributes password and user profile information across and between enterprise information systems
- Audit and Reporting – Provides standardized and on-demand reporting on permissions, actions, and user account activity

With Select Identity, provisioning and management of user accounts and privileges is no longer a barrier to realizing the efficiencies and competitive advantage of extending system access to ever greater numbers of employees, customers and partners.

System Architecture

The following illustration provides a high-level view of the Select Identity system and its components.



All requests to and from the system use the HTTP protocol. Select Identity manages a single, logical identity for each user and administrator. These logical identities are mapped to the users' various accounts on back-end systems and services. The logical identities, as well as their corresponding accounts and privileges, are governed by Select Identity system functions and permissions. Accounts are also governed by security policies that are defined by an administrator based on the access requirements of the company's products and services.

The Context Engine and Identity Business Process Services components of the Select Identity architecture are of particular importance to administrators and personnel responsible for deploying and maintaining the Select Identity

system. These components contain the functions that administrators use most. These functions include the following:

- **Context Management**

Maintains the Context structure that defines identities and access for all users and resources in the extended enterprise.

- **Service Management**

Provides a business-centric abstraction over resources, entitlements, and other identity-related entities. Services represent the products and services that you offer to customers, partners, and employees.

- **Business Relationship Management**

Provides granular control over how groups of users access services.

- **User Management**

Provides consistent account creation and management across products and services.

- **Resource Management**

Provides a connection to the physical information systems on which your products and services rely for user account data.

- **Workflow Management**

Enables the definition of identity-related business processes that can be executed for access to services or any other event within the Select Identity system.

- **Policy Management**

Defines the security and access requirements for each of the products and Services that you provide.

- **Reconciliation**

Ensures the proper coordination of provisioning workflow across multiple resources.

- **Auditing and Reporting**

Provides robust standard and custom reporting facilities over user entitlements and system event history.

- **Forms**

Automates the creation of electronic forms used by end users to register for access to services, change their passwords, set password hints, and update personal information.

- **Tiered Authority**

Enables the secure, multi-tiered delegation of administrative tasks, such as management of identity profiles and entitlements, to functional departments, customers, and partners.

Leveraging an open, standard, J2EE Connector Architecture (JCA) bus, Select Identity uses predefined connectors to access back-end system data stores. Connectors are configured during the installation process and are easy to deploy. If you wish to create your own connectors, Select Identity offers a software developer's kit (SDK) that enables you to do so.

Product Documentation

The chapters in this guide provide installation and configuration information. All installation prerequisites, system requirements, and procedures are covered in detail in this guide. There is also a section that covers the uninstallation procedure for Select Identity. Troubleshooting methods are included in an appendix.

The Select Identity product documentation also includes the following:

- Detailed procedures for deployment and system management are documented in the *HP OpenView Select Identity Administrator Guide* and Select Identity online help system. In the help system provided with the Select Identity client, tasks are grouped by the administrative functions that govern them.
- If you need to develop workflow templates, you can refer to the *HP OpenView Select Identity Workflow Studio Guide*. This guide describes how to use Workflow Studio to create templates and how to configure reporting for workflows.
- The *HP OpenView Select Identity External Call Guide* provides detailed information about creating calls to third-party applications. These calls can then be deployed in Select Identity to constrain attribute values or facilitate workflow processes.

- If you need to develop connectors, which enable you to connect to external systems for provisioning, refer to the *HP OpenView Select Identity Connector Guide*. This document provides an overview of the Connector API and the steps required to build a connector. The audience of this guide is developers familiar with Java.
- Each resource connector has an *HP OpenView Connector Installation Guide*. These are located on the Select Identity Connector CD.
- An independent, web-based, API help system is available for integration development and customization tasks. To view this help, double-click the `index.htm` file in the `Docs/API_Help` directory. This directory resides on the HP OpenView Select Identity CD.

Installing Select Identity

The following provides an overview of the Select Identity server installation process:

- Load the Select Identity schema into the database
- Configure general settings for the Select Identity server and interface
- Configure the web application server for use with Select Identity

The Microsoft SQL Server and Oracle are supported as database servers. BEA WebLogic Server is the supported application server.

Refer to the following sections for system requirements, prerequisite steps, and installation procedures.

Overview of System Requirements

Select Identity is supported on the following database and server configurations:

Database	Supported Application Server and Platform
Microsoft SQL Server	BEA WebLogic Server on Solaris BEA WebLogic Server on Windows
Oracle	BEA WebLogic Server on Solaris BEA WebLogic Server on Windows

It is *strongly* recommended that the database server and web application server be installed on separate systems, if possible, for optimal performance and ease of management.

The following sections provide an overview of the *minimum* requirements for the database and web application servers. Recommend parameters are noted where possible.

For the Database Server

Microsoft SQL Server on Windows 2000	
Version and edition	Microsoft SQL Server 2000, Enterprise Edition
Operating system	Windows 2000 Server with service pack 3
Processor	Intel Pentium or compatible, 500 MHz
Memory (RAM)	512 MB minimum, 1 GB recommended
Disk space	600 MB
JDBC driver	WebLogic jDrivers Type 4 JDBC driver from BEA

Oracle on Solaris	
Version	Oracle Database, version 9.2
Operating system	Sun Solaris 8 with the latest patch cluster
Processor	UltraSPARC, 330 MHz
Memory (RAM)	512 MB minimum, 1 GB recommended
Disk space	3 GB
JDBC driver	Oracle's Thin driver (the classname is <code>oracle.jdbc.driver.OracleDriver</code>)

Oracle on Windows	
Version	Oracle Database, version 9.2
Operating system	Windows 2000 Server with service pack 3
Processor	Intel Pentium or compatible, 500 MHz
Memory (RAM)	512 MB minimum, 1 GB recommended
Disk space	1.5 GB
JDBC driver	Oracle's Thin driver (the classname is <code>oracle.jdbc.driver.OracleDriver</code>)

For the Web Application Server

BEA WebLogic on Solaris	
Version	BEA WebLogic Server, version 8.1 with service pack 2
Operating system	Sun Solaris 8 with the latest patch cluster
Processor	UltraSPARC, 168 MHz
Memory (RAM)	256 MB minimum, 512 MB recommended
Disk space	600 MB

BEA WebLogic on Windows 2000	
Version	BEA WebLogic Server, version 8.1 with service pack 2
Operating system	Windows 2000 Server with service pack 3
Processor	Intel Pentium or compatible, 200MHz
Memory (RAM)	256 MB minimum, 768 MB recommended
Disk space	600 MB

For the Select Identity Interface

The Select Identity interface requires Microsoft Internet Explorer (IE), version 5.5 or higher, with JavaScript and cookies enabled. No installation steps are required to install the Select Identity interface. The web application server that is configured for Select Identity will serve the Select Identity interface pages.

Configuring the Database Server

To enable Select Identity to store its data in a database, you must load the schema into the chosen database server. Before loading the schema, you or the database administrator must ensure that the database server meets the *minimum* requirements. Then, create a database and user that Select Identity can use to access the database server. The following sections provide requirements and procedures for each database server.

For Microsoft SQL Server

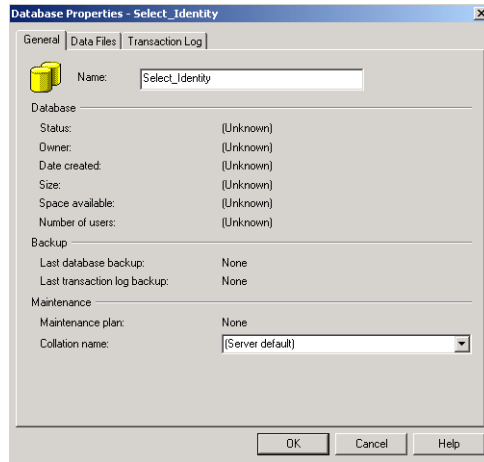
You can create a database for use by Select Identity by running SQL scripts.



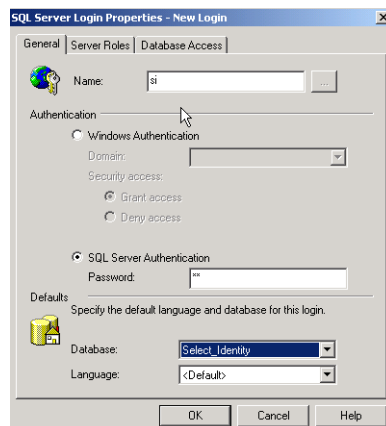
An important step in configuring the database takes place after you configure the web application server. You must ensure that the **truaccess.repository.type** entry in the `TruAccess.properties` file is set to **mssql**. The default setting is for oracle. See [Copying Files to the Web Application Server on page 23](#) for more information about this file.

Complete the following to create a SQL Server database:

- 1 Create a `Select_Identity` directory on the server.
- 2 Copy the `concerto_ddl.sql` and `concerto_dml.sql` files from the Database directory on the Select Identity CD to the `Select_Identity` directory on the SQL Server system.
- 1 Log in to the Microsoft SQL Server Enterprise Manager interface.
- 2 In Enterprise Manager, expand **Microsoft SQL Server** → **SQL Server Group** → **server**, where **server** is the name of the SQL Server instance.
- 3 Right-click **Databases**, and select **New Database....**



- 4 Enter a name for the database, such as `Select_Identity`. Click **OK** to finish creating the database.
- 5 Create a database user that can be used to manage the Select Identity database. Complete the following steps to do so:
 - a Select the **Microsoft SQL Server** → **SQL Server Group** → **server** → **Security** folder in the Enterprise Manager tree.
 - b Create a new login for the new database by right-clicking on **Logins** and selecting **New Login**. The the SQL Server Login Properties dialog displays.



- c** On the **General** tab, enter a user name such as **Select_Identity**, enter a password, and select **SQL Server Authentication** as the authentication type.
 - d** Select the new database (`Select_Identity`) from the Database list. Keep remaining default settings.
 - e** Click **OK**. You are prompted to confirm your password.
 - f** Select the **Database Access** tab on the SQL Server Login Properties dialog.
 - g** Select the **Permit** check box next to the Select Identity database user.
 - h** Assign the **db_owner** and **public** permissions to the new user.
 - i** Click **OK** to save your settings.
 - 6** Create the schema for the Select Identity database by following these steps:
 - a** Launch the SQL Query Analyzer by selecting **Tools -> SQL Query Analyzer**.
 - b** Select the new database (`SI`) from the DB drop-down list.
 - c** Load the `concerto_ddl.sql` SQL script from the `Select_Identity` directory you created in [Step 2 on page 18](#).
 - Click the Open icon.
 - Locate the `Select_Identity` directory.
 - Select the `concerto_ddl.sql` file.
 - Click **Open**.
 - d** Run the script by clicking the **Execute Script** or play button.
 - e** Verify that an error message is not displayed.
 - 7** Insert the required default data into the Select Identity database by performing the following:
 - a** Clear the previous script by clicking the **Clear Query Window** button.
 - b** Load the `concerto_dml.sql` SQL script from the directory you created in [Step 2 on page 18](#).
 - c** Run the script clicking the **Execute Script** button.

Messages in the console indicate that rows are being created.
-

- d Verify that an error message is not displayed.
- e Close the SQL Query Analyzer and the Microsoft SQL Server Enterprise Manager.

The next step is to configure the application server, as described in [Copying Files to the Web Application Server on page 23](#).

For Oracle

You can create a database for use by Select Identity by running SQL scripts.



An important step in configuring the database takes place after you configure the web application server. You must ensure that the **truaccess.repository.type** entry in the `TruAccess.properties` file is set to **oracle**. See [Copying Files to the Web Application Server on page 23](#) for more information about this setting. See [Configuring TruAccess.properties on page 44](#) for general settings in this file.

Complete the following to create the database:

- 1 Copy the `oracle_concero_ddl.sql` and `oracle_concero_dml.sql` files from the Database directory on the Select Identity CD to a directory on the Oracle server.
- 2 Launch SQL Plus. You can perform the following steps using the Enterprise Manager Console. However, the SQL Plus steps documented below are valid on Solaris and Windows.
- 3 Create tablespace into which you will load the Select Identity tables.

```
CREATE TABLESPACE tablespace_name
DATAFILE 'c:\oracle\oradata\SID\tablespace_name.dbf'
SIZE 10M
AUTOEXTEND ON NEXT 10M
MAXSIZE unlimited;
```

where *tablespace_name* is the chosen name for the Select Identity tablespace. You will reference this name when creating the database user. This command creates 10MB of tablespace then automatically extends the tablespace as needed.

- 4 Create a user to be used by Select Identity to access the Select Identity tables:

```
CREATE USER user_name PROFILE DEFAULT
IDENTIFIED BY password DEFAULT TABLESPACE tablespace_name
ACCOUNT UNLOCK;

GRANT CONNECT TO user_name;

GRANT RESOURCE TO user_name;
```

where *user_name* is the name of the database user to be created, *password* is the user's password, and *tablespace_name* is the name of the tablespace to be used assigned as the user's default tablespace.

- ▶ The `oracle_concero_ddl.sql` script, which is run in the following step, inserts tables into the user's default tablespace. If you do not assign the Select Identity tablespace as the user's default tablespace, you must edit the script to reference the Select Identity tablespace.

- 5 Change to the newly created user by entering the following command:

```
CONNECT user_name/password
```

- 6 Create the schema for the Select Identity database by performing the following steps:

- a Execute the schema creation script by running the following command:

```
@path/oracle_concero_ddl.sql
```

where *path* is the full path to the file. If Oracle is installed on Windows, you must include the drive letter in this path.

- b Verify that no error message is displayed.

- 7 Insert the required default data into the Select Identity database as follows:

- a Run the data creation script by entering the following command:

```
@path/oracle_concero_dml.sql
```

where *path* is the full path to the file. If Oracle is installed on Windows, you must include the drive letter in this path.

- b Verify that no error message is displayed.

Copying Files to the Web Application Server

There are three files that must be copied to the web application server and configured for installation.

- 1 Create a `Select_Identity` directory on the web application server that can store Select Identity files. The product installation and connector installations will reference this directory. For example, you could create the `C:\Select_Identity` directory on Windows.
- 2 Copy the following files from the Select Identity product CD to the new directory:
 - `SI30/application/lmz.ear`
 - `SI30/properties/TruAccess.properties`
 - `SI30/connector/connector.jar`
- 3 The `logging.properties` file is also required for installation. Copy this file from the `properties` directory on the Select Identity product CD to one of the following directories:
 - The `jdk141_05/jre/lib/` directory logs all activity to the file configured by `logging.properties`.
 - If you want to designate a log file for Select Identity messages only, copy `logging.properties` into a different directory, such as a subdirectory of the library directory. This is particularly useful if you have multiple WebLogic servers configured in a domain.

After copying the file, refer to [Logging on page 41](#) for instructions on configuring this file. Note that configuring logging is crucial and Select Identity may not function properly if you do not configure the `logging.properties` file.

- 4 Ensure that the `C:\temp\log` directory exists on the application server's system. The `logging.properties` file specifies this directory as the destination of Select Identity log messages.



If this directory does not exist, Select Identity will not start. If you wish to log to a different (existing) directory, edit the `logging.properties` file as described in [Logging on page 41](#).

- 5 If necessary, create a subdirectory named `ext` in the JDK library subdirectory of the WebLogic server's installation directory (by default, this directory is in `jdk141_05/jre/lib/` on WebLogic 8.1).
- 6 Copy the contents of the `library` directory on the Select Identity CD to the `ext` subdirectory.
- 7 If you are planning to use the Auto Discovery or Reconciliation features in Select Identity, create a directory called `endorsed` under the Weblogic installation directory, `jdk141_05\jre\lib`. Copy `xalan.jar`, `xercesImpl.jar`, and `xml-apis.jar` from the Select Identity CD to this directory.
- 8 For easier access to documentation, you can copy the product documentation and help from the `docs` directory on the Select Identity CD to a directory on the application server.

There are several configuration settings contained in the `TruAccess.properties` file. Most are optional settings that determine defaults for the Select Identity client. The following settings are **required** by the installation process.

- **`truaccess.sender.email`**

Specify a general email address that will be used as the sender's address for any email that is sent by Select Identity. For example, you may want to specify `info@your_company.com` or `select_identity_admin@your_company.com`. This address must exist on the SMTP server configured for use by Select Identity's application server.

You can also specify a value for `truaccess.sender.name` to coincide with this setting.

- **`truaccess.method`, `truaccess.host`, `truaccess.port`**

Provide values that make up the URL of the Console interface. Specify the protocol, host name or IP address, and port. For example, **`http://localhost:7001/`**.

- **`truaccess.loginURL`, `truaccess.logoutPage`**

Specify the log-in page and the page that is displayed when the user logs out of Select Identity. Synchronize these settings with the values of **`truaccess.method`, `truaccess.host`, and `truaccess.port`**.

If you are using external authentication, make sure that the **`truaccess.authentication`** setting is **off**.

- **truaccess.repository.type**

This setting defines the type of database server you are using. Specify **mssql** for Microsoft SQL Server or **oracle** for Oracle. Oracle is the default setting.

This setting is case sensitive.

- **truaccess.upload.filedir**

Specify a valid location on the Select Identity server that can be used as temporary storage while Select Identity uploads files to the database.

You can configure the remaining settings in the file, though default values are provided and you can set them at your convenience. See [Configuring TruAccess.properties on page 44](#) for information about additional file settings.

Configuring the Web Application Server

Select Identity relies on the web application server to serve the Select Identity interface pages, communicate with the database server to store and retrieve data, and send email based on an action performed through the Select Identity interface. You can configure a BEA WebLogic application server for use with Select Identity.



Select Identity supports clustered servers through the application server layer. See the application server documentation for information on clustering these servers.

To configure WebLogic for use with Select Identity, complete the following steps. These steps assume that WebLogic is installed and running.

- 1 Ensure that the system where WebLogic is installed meets the *minimum* requirements, as documented on [page 17](#) (on Solaris) and on [page 17](#) (on Windows).
- 2 If necessary, upgrade the Java encryption level that WebLogic uses to 128K. To determine whether you need to upgrade the encryption level of your server, note the size of the `local_policy.jar` and `US_export_policy.jar` files in the `jdk141_05/jre/lib/security` directory. If they are 3KB, you need to upgrade the encryption. The files for 128K encryption are 5KB.

Refer to <http://java.sun.com/j2se/1.4.2/download.html> (click **DOWNLOAD** next to Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 1.4.2 under Other Downloads). Refer to the readme file that is downloaded to confirm which files to replace.

- 3 If the WebLogic server is not running, start the server and log in to the WebLogic Server Console as a system user. (Typically, the URL for the console is **http://host:7001/console**.)

The WebLogic server can be started by selecting **Start** → **Programs** → **BEA WebLogic** → **Projects** → **User Projects** then selecting **Start Server** next to the name of your domain or project.

- 4 Configure the mail session, as follows:
 - a Select **My_domain** → **Services** → **Mail** from the tree on the left-hand side of the console, where *domain_name* is the domain created during the WebLogic installation. The Mail Sessions page is displayed.
 - b Click **Configure a new Mail Session** on the Mail Sessions page. The Create a new MailSession page is displayed.

This page allows you to define the configuration of this mail session.

Name:
The name of this mail session.

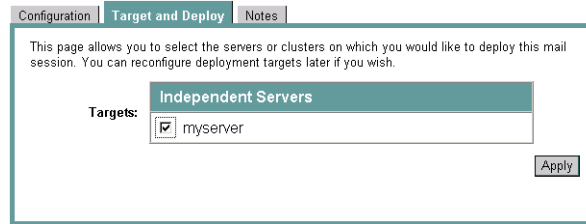
JNDIName:
The JNDI Lookup Name used to look up the `javax.mail.Session` object for this mail session.

Properties:
The properties to use to configure this mail session. The property names are specified in the JavaMail API Design Specification.

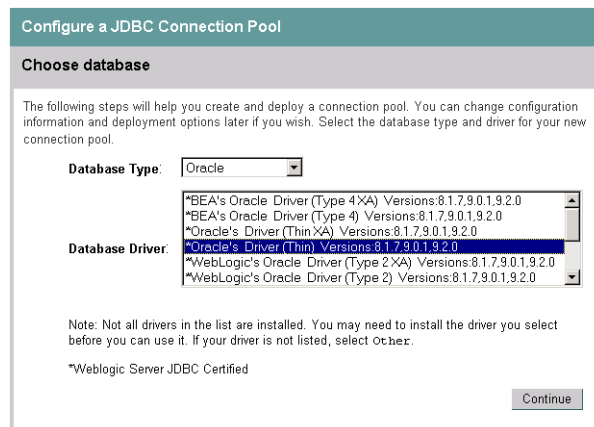
- c On the Configuration tab, provide the following information:

Field	Value
Name	Enter a name for the mail session.
JNDIName	Enter mail/TruAccess .
Properties	Enter the mail server's IP address. Here is an example: mail.smtp.host=192.168.1.52 .

- d Click **Create**. The **Target and Deploy** tab is displayed.



- e Select the WebLogic server designated for Select Identity's use and click **Apply** to finish mail session configuration.
- 5 Configure a JDBC connection pool to enable WebLogic to communicate with the database server by completing the following steps:
- a Select **My_domain** → **Services** → **JDBC** → **Connection Pools** from the tree. The JDBC Connection Pool page is displayed.
- b Click **Configure a new JDBC Connection Pool** on the JDBC Connection page. The Configure a JDBC ConnectionPool page is displayed.



- c Choose the database type and driver. If you installed the driver documented in [For the Database Server on page 15](#), choose **BEA's MS SQL Server Driver (Type 4) Versions:7.0, 2000** for SQL Server. Choose **Oracle's Driver (Thin) Versions:8.1.7,9.0.1,9.2.0** for Oracle.

Then, click **Continue**.

- d On the Define connection properties page, enter the following information:

Field	Value
Name	Enter a name for connection pool.
Database Name	The name of the database created on the database server for use by Select Identity. For example, <code>Select_Identity</code> .
Host Name	The IP address or host name of the database server.
Port	The database's port. The default port for Microsoft SQL Server is 1433. For Oracle, the default port is 1521.
Database User Name	The user created for use to administer the Select Identity database.
Password and Confirm Password	Enter the database user's password.

The following is an example:

Configure a JDBC Connection Pool

Define connection properties

Name your new connection pool and provide additional information to connect to your database.

Name:
The name of this JDBC connection pool.

Connection Properties

Database Name:
The name of the database to connect to.

Host Name:
The name or IP address of the database server.

Port:
The port on the database server used to connect to the database.

Database User Name:
The database account user name used in the physical database connection.

Password:
Confirm Password:
The database account password used in the physical database connection.

- e Click **Continue**. WebLogic displays the Test database connection page and constructs the values displayed in the fields on the page.
- f Click **Test Driver Configuration** to verify that WebLogic can connect to the database. Or, you can skip this step.
- g Click **Create and deploy** to create the JDBC connection pool.
- h Click on the newly created connection pool to verify that your server is selected on the Target and Deploy tab. You can also edit the connection properties, if you wish.

Configuration | **Target and Deploy** | Monitoring | Control | Testing | Notes

This page allows you to select the servers or clusters on which you would like to deploy this JDBC connection pool.

Targets:
 myserver

- i Click **Apply** if you change settings.

- 6 Configure JDBC data sources, as follows:
- a Select **My_domain** → **Services** → **JDBC** → **Data Sources** from the tree. The data sources configuration page is displayed.
 - b Click **Configure a new JDBC Data Source** on the JDBC Data Sources page. The Configure a JDBC Data Source page is displayed.

Configure a JDBC Data Source

Configure the data source

Define your new JDBC data source.

Name:

The name of this JDBC data source.

JNDI Name:

The JNDI path to where this JDBC data source is bound.

Honor Global Transactions

Specifies whether this data source will participate in existing global (XA) transactions. Unchecking this option while creating the data source should be done rarely and with care. This option can not be changed once the data source is created.

Emulate Two-Phase Commit for non-XA Driver

Specifies whether the JDBC resource will emulate participation in a global transaction. This option is only applicable when the associated connection pool uses a non-XA JDBC driver and when global transactions are honored in the data source.

- c Enter the following information:

Field	Value
Name	Enter a name for the new data source.
JNDI Name	Enter <code>jdbc/TruAccess</code> .

Leave the default values for the rest of the options on the page.

- d Click **Continue**.

- e Select the connection pool created in [Step 5 on page 27](#) from the **Pool Name** drop-down list.

The screenshot shows a dialog box titled "Configure a JDBC Data Source" with a sub-header "Connect to connection pool". The main text says "Associate your newly created JDBC data source with a connection pool." Below this, there is a "Pool Name:" label followed by a dropdown menu showing "SI_ConnectionPool". A small text block explains: "The JDBC connection pool associated with this data source. The connection pool you select is used to supply database connections to client applications that request a connection from this data source." At the bottom right, there is a "Continue" button.

- f Click **Continue**.
 - g Ensure your server is selected on the Target data source page. Then, click **Create**.
- 7 Stop the WebLogic server.
 - 8 Edit the `startweblogic.cmd` file to modify WebLogic's classpath, specify the location of the `TruAccess.properties` file, the location of the `connector.jar` file, and the location of the `logging.properties` file, if necessary.

The `startweblogic.cmd` file resides in the `WebLogic_home/user_projects/domains/domain/` directory on WebLogic 8.1.

- Add the `Select_Identity` directory to the classpath. This enables the web server to access the `connector.jar` file. Here is an example of the classpath after it is modified. The new entries are bolded:

```
set CLASSPATH=%WEBLOGIC_CLASSPATH%;%POINTBASE_CLASSPATH%;
%JAVA_HOME%\jre\lib\rt.jar;
%WL_HOME%\server\lib\webservices.jar;c:\Select_Identity;
c:\Select_Identity\connector.jar;%CLASSPATH%
```

- Add the `TruAccess.properties` file to the line beginning with `%JAVA_HOME%/bin/java`. Add this argument before the **weblogic.Server** entry. If the `TruAccess.properties` file is not located in a path that is relative to the `weblogic81` directory, make sure that you provide a fully-qualified path to the file.

If WebLogic is running on a UNIX server, you will also add `-Djava.awt.headless=true` to the classpath.

Here is an example of this line after it is modified. The new entry is bolded:

```
%JAVA_HOME%\bin\java %JAVA_VM% %MEM_ARGS% %JAVA_OPTIONS%
-Dweblogic.Name=%SERVER_NAME%
-Dweblogic.ProductionModeEnabled=%PRODUCTION_MODE%
-Djava.security.policy="%WL_HOME%\server\lib\weblogic.policy"
-Dcom.truologica.truaccess.property.file=c:\Select_Identity\
  TruAccess.properties weblogic.Server
```

- If you copied `logging.properties` into a directory different than WebLogic's installation directory, you must add the location to the line beginning with `%JAVA_HOME%/bin/java`. Again, add this argument before **weblogic.Server** in the line. If the file is not located in a path that is relative to the `weblogic81` directory, make sure that you provide a fully-qualified path to the file.

Here is an example of this line after it is modified. The new entry is bolded:

```
%JAVA_HOME%\bin\java %JAVA_VM% %MEM_ARGS% %JAVA_OPTIONS%
-Dweblogic.Name=%SERVER_NAME%
-Dweblogic.ProductionModeEnabled=%PRODUCTION_MODE%
-Djava.security.policy="%WL_HOME%\server\lib\weblogic.policy"
-Dcom.truologica.truaccess.property.file=c:\Select_Identity\
  TruAccess.properties -Dcom.truologica.truaccess.util.logging.
misc.config.file=c:\Select_Identity\log\logging.properties
weblogic.Server
```

- 9 Save your settings.
- 10 Restart the WebLogic server.
- 11 Deploy Select Identity on the application server as follows:
 - a Log in to the WebLogic Server Console.
 - b Select the *domain_name* → **Deployments** → **Applications** folder.
 - c Click **Deploy a new Application**.

- d Locate and select the `lmz.ear` file, which resides in the `Select_Identity` directory created in [Copying Files to the Web Application Server on page 23](#).

The Deploy an Application page displays. The fields are populated with default values.

If multiple servers are configured, click **Target Application** and select the targets of the Select Identity application.

- e Click **Deploy**.

The deployment may take several seconds to complete.

Uninstalling Select Identity

There are a number of places where Select Identity stores information. To completely uninstall the product, perform the steps in each of the following sections.

Uninstalling Select Identity from the Web Server

The following sections provide steps for a complete uninstall from the web server.

Deleting the Imz File


To uninstall Select Identity on WebLogic, you delete the `lmz.ear` file from the WebLogic server.



Make sure that all dependencies on the system are removed.

Complete the following steps:


- 1 Log in to the WebLogic Server Console.
- 2 Select the ***domain_name*** → **Deployments** → **Applications** folder.

- 3 Click the **Delete** button () next to the lmz application.
- 4 When prompted to confirm the deletion, click **Yes**.

Deleting the Connectors


You may have any number of connectors installed to support system resources. If you are completely uninstalling the Select Identity product you will want to uninstall the connectors.

Perform the following steps:

- 1 Log in to the WebLogic Server Console.
- 2 Select the **domain_name** → **Deployments** → **Connector Module** folder.
- 3 Click the **Delete** button () next to the connectors that you have installed.
- 4 When prompted to confirm the deletion, click **Yes**.
- 5 Click **Continue**.

Deleting the Data Source


Perform the following steps to delete the Select Identity data source:

- 1 Log in to the WebLogic Server Console.
- 2 Select the **domain_name** → **Services** → **JDBC** → **Data Sources** folder.
- 3 Click the **Delete** button () next to the **jdbc/TruAccess** connection.
- 4 When prompted to confirm the deletion, click **Yes**.
- 5 Click **Continue**.

Deleting the Connection Pool


Perform the following steps to delete the Select Identity connection pool:

- 1 Log in to the WebLogic Server Console.
- 2 Select the **domain_name** → **Services** → **JDBC** → **Connection Pools** folder.

- 3 Click the **Delete** button () next to the connection pool that was used by the data source.
- 4 When prompted to confirm the deletion, click **Yes**.
- 5 Click **Continue**.

Deleting the Mail Session

Perform the following steps to delete the Select Identity mail session:

- 1 Log in to the WebLogic Server Console.
- 2 Select the **domain_name** → **Services** → **JDBC** → **Mail Session** folder.
- 3 Click the **Delete** button () next to the **mail/TruAccess** connection.
- 4 When prompted to confirm the deletion, click **Yes**.
- 5 Click **Continue**.

Uninstalling the Select Identity Database

After you uninstall the product from the web server, you can uninstall the data and tables from the database.

Uninstall From Microsoft SQL Server

Perform the following steps to uninstall the Select Identity from Microsoft SQL Server:

- 1 Log in to the Microsoft SQL Server Enterprise Manager.
- 2 Under Microsoft SQL Server, locate the `Select_Identity` database instance.
- 3 Right-click on the database.
- 4 Select **Delete**.
- 5 To confirm the action, click **Yes**.

- 6 Under the **Security** heading, click **Logins**.
- 7 Right-click on the Select Identity database user name, such as *SI* in previous procedures.
- 8 Select **Delete**.
- 9 To confirm the action, click **Yes**.

Uninstall From Oracle

Perform the following steps to uninstall the Select Identity database from Oracle:

- 1 From a SQL Plus command prompt, log in to Oracle as a user with system permissions.
- 2 Enter the following command:

```
drop user Select_Identity_database_username cascade
```



Troubleshooting

This chapter provides error messages that you may encounter when configuring the web application server for use with HP OpenView Select Identity. A suggested solution is also provided for each message.

By default, trace information is displayed in the window from which the WebLogic Application Server was started.

- The WebLogic Server does not start.

Possible Cause: The `logging.properties` file is not configured properly.

Possible Solution: For more information, see [Logging on page 41](#) for details. In particular, make sure that the directory specified for the FileHandler log file (the **pattern** attribute in the message format) exists.

- The WebLogic Server does not recognize the `lmz` application.

Possible Cause: An anomaly in the installation.

Possible Solution: Add the EJBs to the WebLogic server using the WebLogic Server Console.

- When the WebLogic Server starts, the following error is displayed:

```
<Error> <JDBC> <Cannot startup connection pool
"ConceroConnectionPool" weblogic.common.ResourceException:
Could not create pool connection. The DBMS driver exception was:
java.sql.SQLException: SQL Server has been paused.
```

Possible Cause: SQL Server is not running.

Possible Solution: Start SQL Server.

- When the WebLogic Server starts, the following error is displayed:

```
<Error> <JDBC> <Cannot startup connection pool
"ConceroConnectionPool" weblogic.common.ResourceException:
Could not create pool connection. The DBMS driver exception was:
java.sql.SQLException:
Login failed for user 'sa'. Severity 14, State 1, Procedure
'null null', Line 0 Unable to connect, please check your
server's version and availability.
    at weblogic.jdbc.mssqlserver4.TdsStatement.
    microsoftLogin(TdsStatement.java:2872)
```

Possible Cause: The user ID or password is configured incorrectly for SQL Server.

Possible Solution: See [Step 5d](#) in the procedure for configuring WebLogic on [page 28](#).

- When attempting to sign in to Select Identity (through the web browser), an Error 500 -Internal Server Error is displayed on the page and the following error message is displayed in the server's window:

```
<Error> <JDBC> <Error during Data Source creation:
weblogic.common.ResourceException: DataSource(jdbc.AccessUsDB)
can't be created with non-existent Pool (connection or multi)
(ConceroConnectionPool)>
```

Possible Cause: The targets for the JDBC connection pool may not be configured correctly.

Possible Solution: See [Step 5h](#) in the procedure for configuring WebLogic on [page 29](#).

- When attempting to create an administrator, this error is displayed:

```
createAndSendMail exception : javax.mail.SendFailedException:  
Sending failed;  
nested exception is:  
javax.mail.MessagingException: Could not connect to SMTP host:  
65.70.174.236, port: 25;
```

Possible Cause: The mail server is not available or the mail server configuration is not correct.

Possible Solution: See [Step 4](#) in the procedure for configuring WebLogic on [page 26](#).

Logging

HP OpenView Select Identity implements the `java.util.logging.Logger` class, as defined by the Java 2, Standard Edition, v 1.4.1 API Specification. During installation, the `logging.properties` file is copied from the Select Identity CD to a subdirectory on the application server. This file defines how Select Identity logs messages and exceptions, according to the specification.

Here is an example of the `logging.properties` file:

```
# Set the logging level for the root of the namespace.
# This becomes the default logging level for all Loggers.
.level=WARNING

# List of global handlers
handlers = com.trulogica.truaccess.util.logging.misc.FileHandler,
com.trulogica.truaccess.util.logging.misc.ConsoleHandler

# Properties for the FileHandler
com.trulogica.truaccess.util.logging.misc.FileHandler.limit = 500000
com.trulogica.truaccess.util.logging.misc.FileHandler.count = 3
com.trulogica.truaccess.util.logging.misc.FileHandler.pattern = concero.log

# Default level for ConsoleHandler. This can be used to
# limit the levels that are displayed on the console even
# when the global default has been set to a trace level
com.trulogica.truaccess.util.logging.misc.ConsoleHandler.level = FINEST
com.trulogica.truaccess.connector.ldap.ldapv3.LDAPConnector.level = FINE
```

The following options are available for you to configure. For more detail about each option, refer to the `Logger` class in the API specification.

- **Handlers**

Defines where messages are logged. You *must* configure the following handlers in `logging.properties`: `ConsoleHandler` and `FileHandler`. In addition, the following handlers are available: `MemoryHandler` and `StreamHandler`. In the example on [page 41](#), a `FileHandler` and `ConsoleHandler` are configured (you must also configure the handler's format, as shown in the following example):

```
# List of global handlers
handlers =
com.truologica.truaccess.util.logging.misc.FileHandler,
com.truologica.truaccess.util.logging.misc.ConsoleHandler

# Properties for the FileHandler
com.truologica.truaccess.util.logging.misc.FileHandler.limit = 500000
...
```

- **Message format**

Defines the format of logged messages based on the handler type. For example:

```
# Properties for the FileHandler
com.truologica.truaccess.util.logging.misc.FileHandler.limit = 500000
com.truologica.truaccess.util.logging.misc.FileHandler.count = 3
com.truologica.truaccess.util.logging.misc.FileHandler.pattern = concero.log
```

Note the **pattern** attribute for `FileHandler`, which defines the location of the log file. The file location is relative to the user's root directory (the user under which the application server is running). This directory must exist. If it does not, `Select Identity` will not start.

For example, if you specify `log/log.txt` and the application server is running under the administrative user whose home directory is `/user/admin`, the file is written to the `/user/admin/log/log.txt` file. You can also specify an absolute path, such as `/temp/log/log.txt`.

Refer to the `Logger` class in the API specification for a list of format parameters required for each handler type.

- **Log level**

Defines the logging output. Specify a level using the level entry and set the level to `SEVERE` (the highest value), `WARNING`, `INFO`, `CONFIG`, `FINE`, `FINER`, or `FINEST` (the lowest value). You can specify a level for all messages or only those written by a specific component.

In the example, the default logging level is set to `WARNING` but a log level is also specified for the LDAP connector component (you must also specify a handler for component-specific log levels):

```
# Set the logging level for the root of the namespace.
# This becomes the default logging level for all Loggers.
.level=WARNING

# List of global handlers
...

# Properties for the FileHandler
...

# Default level for ConsoleHandler. This can be used to
# limit the levels that are displayed on the console even
# when the global default has been set to a trace level
com.trulogica.truaccess.util.logging.misc.ConsoleHandler.level = FINEST
com.trulogica.truaccess.connector.ldap.ldapv3.LDAPConnector.level = FINE
```



Configuring TruAccess.properties

You can configure general settings for the Select Identity server and interface by editing the `TruAccess.properties` file. This file provides settings for the following:

- Email that is sent as part of the provisioning process
- Provisioning retry attempts
- The Select Identity interface URL and log-in page
- Authentication
- Temporary storage needed during uploads to the database
- Auditing
- LDAP
- Account reconciliation with system resources

Each section of the file is described below. Sections that should not be edited are specified.

```
truaccess.email.new.timeinterval=120
```

Specifies the time interval in seconds that the email daemon uses to send new email.

```
truaccess.email.retry.timeinterval=900
```

Specifies the time interval in seconds that the email daemon uses for sending new email if initial attempts were unsuccessful.

```
truaccess.email.retry.maximum=1
```

Specifies the maximum number of retry attempts for sending email. Setting this to 0 causes Select Identity to retry an infinite number of times.

```
truaccess.email.to.empty=off
```

Specifies off if you do not want any email sent if the “to” email address cannot be determined. Specifies on if you want to send an email to the administrator in this event.

```
truaccess.email.userinfochange=off
```

Do not change truaccess.email.userinfochange=off, leave as is.

```
truaccess.email.redirect=off
```

```
truaccess.email.redirect.dir=C:/temp/email
```

For testing purposes if a mail server is not available, specifies on and a directory to where the email files should be written.

```
truaccess.email=on
```

```
truaccess.email.inprogresstimeout=600000
```

```
truaccess.email.batchcount=10
```

Determines whether Select Identity sends email. If set to **off**, no email is sent.

```
truaccess.sender.name=Select Identity
```

```
truaccess.sender.email=support@hp.com
```

Specifies a default name and email address to use if the sender information cannot be determined.

```
truaccess.job.retry.timeinterval=600
```

```
truaccess.job.retry.maximum=3
```

Specifies the time interval in seconds that Select Identity will wait between attempts and the maximum number of retries when trying to execute a function, such as deleting a user.

```
truaccess.postprovision.retry.timeinterval=5000
```

```
truaccess.postprovision.retry.maximum=20
```

Specifies the time (in milliseconds) to sleep before retry in post provision (adding account to the Select Identity database) and number of retry times.

```
truaccess.method=http
truaccess.host=localhost
truaccess.port=7001
```

Constructs the URL to the Select Identity system within email notifications.

```
truaccess.pageredirect.timeout=10
```

Specifies the time-out in seconds for page redirects.

```
truaccess.dataSource=jdbc/TruAccess
```

Specifies the data source JNDI name. You should not have to modify this setting.

```
truaccess.mailSession=mail/TruAccess
```

Specifies the JNDI name for the mail session id. You should not have to modify this setting.

```
truaccess.repository.type=oracle
```

Specifies the Select Identity database type. The default value is **oracle**. If using Microsoft SQL Server, specify **mssql**. This value is case-sensitive.

```
truaccess.logo=/lmz/images/TruLogica.gif
```

Specifies the relative or fully qualified path name for your logo file.

```
truaccess.homepage=http://www.trulogica.com
```

```
truaccess.customerName=TruLogica
```

Specifies your home page and name.

```
truaccess.authentication=on
```

```
truaccess.sso.token.name=ct_remote_user
```

```
truaccess.loginURL=https://localhost:7001/lmz/control/signin
```

```
truaccess.logoutPage=https://localhost:7001/lmz/control/
logout
```

If authentication is **on** then the following three attributes are ignored. If authentication is **off** then specifies the single sign on token name, the login URL and the logout URL for cleaning up the session.

```
truaccess.upload.fileDir=c:/temp
```

```
truaccess.upload.maxfilesize=10485760
```

Specifies a temp directory that the Auto Discovery process uses and the maximum file size in bytes.

```
truaccess.audit.detail=off
```

Specifies **on** or **off** to increase the level of detail stored for audit history reports. If **on**, performance may be affected.

```
truaccess.provisioning.delay=2
```

Specifies delay in seconds for asynchronous provisioning.

```
truaccess.userdiscovery.mapping.file=C:/temp/
AttributeMapping.xml
```

Specifies the location for the attribute mapping XML file for user discovery.

```
truaccess.resource.record.max=1000
```

This parameter specifies the maximum number of users during reconciliation.

```
truaccess.dateformat=yyyy-MM-dd
```

Specifies the date format.

```
truaccess.timestampformat=yyyy-MM-dd hh:mm:ss a
```

Specifies the timestamp format.

```
truaccess.version=3.0.0
```

Version number of Select Identity, do not change.

```
#truaccess.hibernate.config=/com/trulogica/truaccess/util/
persistence/mssqlserver.hibernate.cfg.xml
```

Specifies the hibernate property file. **DO NOT UNCOMMENT.**

```
truaccess.fixedtemplate.passwordreset=SelectIdentityDefaultProcess
truaccess.fixedtemplate.terminate.disable=SelectIdentity
DefaultProcess
truaccess.fixedtemplate.terminate=SelectIdentityDefaultProcess
truaccess.fixedtemplate.disable=SelectIdentityDefaultProcess
truaccess.fixedtemplate.enable=SelectIdentityDefaultProcess
truaccess.fixedtemplate.expiration=UserAccountExpirationWF
truaccess.fixedtemplate.reconciliation=Reconciliation
DefaultProcess
```

Specifies the template for password reset.

```
truaccess.expirationProcessPeriod=30
```

Specifies the manager notification sent prior to automatic account expiration. The default is 30 days.

```
#truaccess.expire.administrator.userId=sisa
truaccess.expire.administrator.adminFunc=Concero Sys Admin
```

Specifies the Select Identity system administrator login and administrative role.

```
truaccess.disable=true
truaccess.disabledays=1
```

Specifies account disable period before the account is terminated. Set this flag to **true** if accounts need to be disabled before terminated.

```
truaccess.policy.id=1
```

Specifies the default Select Identity policy identifier.

```
truaccess.user.extra=Standard\ ID,PersonNumber
truaccess.user.extra.Standard\ ID.column=stdId
#truaccess.user.extra.PersonNumber.column=PersonNumber
```

Specifies extra attributes that can be associated with users. These fields should be nullable. If these attributes are set, the TAUUser database table must be extended by adding extra columns. The extra attributes must then be mapped to those columns. This is done for search efficiency.

```
truaccess.recon.rootdir=c:/temp/reconroot
truaccess.recon.stagingdir=c:/temp/reconstaging
truaccess.recon.backupdir=c:/temp/reconbackup
truaccess.recon.filename.timeformat=yyyy_MM_dd_H_mm
truaccess.recontimer.startdelay=30
truaccess.recontimer.timeinterval=30
truaccess.recon.task.check.threshold=3
truaccess.recon.check_serviceassignment_authadd=false
```

Specifies the attributes for reconciliation.

```
truaccess.batch.inprogresstimeout=1800
truaccess.batch.ownerkey=0
```

Specifies the attributes for batch processing.

```
truaccess.batch.pickuppolicy=1
truaccess.batch.reportdir=c:/temp/reports
```

Specifies the policy to pick up the batch

- 1 - common batch only
- 2 - own batch only
- 3 - common and own batch


```
truaccess.singlevalue.attribute.delete=false
```

Specifies whether or not a user's single value attributes should be deleted.

```
#com.hp.si.webservice.auth.resource=ldap
#com.hp.si.webservice.auth.ldap.accessurl=ldap://
localhost:389
#com.hp.si.webservice.auth.ldap.uidattr=uid
#com.hp.si.webservice.auth.ldap.suffix=ou=People,dc=trulogic
a,dc=com
#com.hp.si.webservice.auth.ldap.needssl=false
```

Specifies external authentication for web service requests.

```
com.hp.si.user.attributes.dropdown.constraint.count=10
```

Specifies the user attribute drop-down value count. This determines when you can choose a user name from a drop-down list or when you must use the search function.

```
#com.hp.si.clientName=Select Identity
```

Specifies your company name.

```
#com.hp.si.clientHeadTag=/ClientPages/SI/headTags.jsp
```

If you want to have a company-specific header, provide the .jsp path here. *Work with Select Identity Professional Services to ensure that the formatting is consistent with Select Identity.*

```
#com.hp.si.clientHeader=/ClientPages/SI/header.jsp
```

If you want to have a company-specific header, provide the .jsp path here. *Work with Select Identity Professional Services to ensure that the formatting is consistent with Select Identity.*

```
#com.hp.si.clientFooter=
```

If you want to have a company-specific footer, provide the .jsp path here. *Work with Select Identity Professional Services to ensure that the formatting is consistent with Select Identity.*

```
truaccess.sqlQueryInListSize=200
```

The maximum number of positional parameters to be used in a SQL query “in” list or array.

`truaccess.batchQuerySize=500`

The maximum number of queries to be executed in a single batch insert/update statement.

`truaccess.generatedFileSizeLimit=2000000`

Indicates the size of the files (in bytes) that are generated by the reporting subsystem. This is a soft limit; the actual file size may exceed this by a small amount.

`truaccess.userdetailconfigrpt.sortattributes=UserName, FirstName, LastName, Email, Company, Department, CostCenter`

Indicates the column on which sorting should take place in the user detail configuration report.

A

Access Control List (ACL)

An abstraction that organizes entitlements and controls authorization. An ACL is list of entitlements and users that is associated with a secured object, such as a file, an operation, or an application. In an ACL-based security system, protected objects carry their protection settings in the form of an ACL.

Access Management

The process of authentication and authorization.

Action

An action represents a task that can be performed within each Select Identity capability.

See also: *capability*

Admin Role

A template that defines the administrative actions that can be performed by a user. An Administrative Service is created to provide access to roles. Users are then given access to the Service. Users with administrative roles can also grant their set of roles to another administrator within their Service context.

Approval Process

The process of approving the association, modification, or revocation of entitlements for an identity. This process is automated of these through workflow templates.

Approver

A Select Identity administrator who has been given approval actions through an Admin Role.

Attribute

An attribute is an individual field that helps define an identity profile. For each identity, an attribute has a corresponding value. For example, an attribute could be “department” with possible values of “IT,” “sales,” or “support.”

Audit Report

A report that provides regular account interaction information within the Select Identity system.

Authentication

Verification of an identity’s credentials.

Authoritative Source

A resource that has been designated as the “authority” for identity information. Select Identity accounts can be reconciled against accounts in an authoritative source.

Authorization

Real-time enforcement of an identity’s entitlements. Authentication is a prerequisite for authorization.

Auto Discovery

The process of adding user accounts to the Select Identity system for a specified Service through the use of a data file.

B**Business Relationship**

A Select Identity abstraction that defines how a logical grouping of users will access a Select Identity Service. The Select Identity Service is a superset of all the identity management elements of a business service.

Business Service

A business service is a product or facility offered by, or a core process used by, a business in support of its day-to-day operations. Example business services could include an online banking service, the customer support process, and IT infrastructure services such as email, calendaring, and network access.

See also: *service*

C

Capability

Actions that can be performed within the Select Identity client are grouped by capability, or link, in the interface.

See also: *action*

Challenge and Response

A method of supplying alternate authentication credentials, typically used when a password is forgotten. Select Identity challenges the end user with a question and the user must provide a correct response. If the user answers the question correctly, Select Identity resets the password to a random value and sends email to the user. The challenge question can be configured by the administrator. The valid response is stored for each user with the user's profile and can be updated by an authenticated user through the Self Service pages.

Configurations

The Configurations capability enables you to import and export Select Identity settings and configurations. This is useful when moving from a test to a production environment.

Configuration Reports

Configuration reports provide current system information for user, administrator, and Service management activities.

Connector

A J2EE connector that communicates with the system resources that contain your identity profile information.

Context

A Select Identity concept that defines a logical grouping of users that can access a Service.

Contextual Identity Management (CIM)

An organizational model that introduces new abstractions that simplify and provide scale to the business processes associated with identity management. These abstractions are modeled after elements that exist in businesses today and include Select Identity Services and Business Relationships.

Credentials

A mechanism or device used to verify the authenticity of an identity. For example, a user ID and password, biometrics, and digital certificates are considered credentials.

D**Data File**

An SPML file that enables you to define user accounts to be added to Select Identity through Auto Discovery or Reconciliation.

Delegated Administration

The ability to securely assign a subset of administrative roles to one or more users for administrative management and distribution of workload. Select Identity enables role delegation through the Self Service pages from one administrator to another user within the same Service context.

Delegated Registration

Registration performed by an administrator on behalf of an end user.

E**End User**

A role associated to every user in the Select Identity system that enables access to the Self Service pages.

Entitlement

An abstraction of the resource privileges granted to an identity. Entitlements are resource-specific and can be resource account IDs, resource role memberships, resource group memberships, and resource access rights and privileges. Entitlements are also considered privileges, permissions, or access rights.

External Call

A programmatic call to a third-party application or system for the purpose of validating accounts or constraining attribute values.

F**Form**

An electronic document used to capture information from end users. Forms are used by Select Identity in many business processes for information capture and system operation.

I**Identity**

The set of authentication credentials, profile information, and entitlements for a single user or system entity. Identity is often used as a synonym for “user,” although an identity can represent a system and not necessarily a person.

Identity Management

The set of processes and technologies involved in creating, modifying, deleting, organizing, and auditing identities.

M**Management**

The ongoing maintenance of an object or set of objects, including creating, modifying, deleting, organizing, auditing, and reporting.

N

Notifications

The capability that enables you to create and manage templates that define the messages that are sent when a system event occurs.

P

Password Reset

The ability to set a password to a system-generated value. Select Identity uses a challenge and response method to authenticate the user and then allow the user to reset or change a password.

Policy

A set of regulations set by an organization to assist in managing some aspect of its business. For example, policy may determine the type of internal and external information resources that employees can access.

Process

A repeatable procedure used to perform a set of tasks or achieve some objective. Whether manual or automated, all processes require input and generate output. A process can be as simple as a single task or as complicated a multi-step, conditional procedure.

See also: *approval process*

Profile

Descriptive attributes associated with an identity, such as name, address, title, company, or cost center.

Provisioning

The process of assigning authentication credentials to identities.

R

Reconciliation

The process by which Select Identity accounts are synchronized with a system resource. Accounts can be added to the Select Identity system through the use of an SPML data file.

Registration

The process of requesting access to one or more resources. Registration is generally performed by an end user seeking resource access, or by an administrator registering a user on a user's behalf.

See also: *delegated registration, self registration*

Request

An event within the Select Identity system for the addition, modification, or removal of a user account. Requests are monitored through the Request Status capability.

Resource

Any single application or information repository. Resources typically include applications, directories, and databases that store identity information.

Role

A simple abstraction that associates entitlements with identities. A role is an aggregation of entitlements and users, typically organized by job function.

See also: *administrative role*

Rule

A programmatic control over system behavior. Rules in Select Identity are typically used for programmatic assignment of Services. Rules can also be used to detect changes in system resources.

S

Self Registration

Registration performed by an end user seeking access to one or more resources.

Self Service

The ability to securely allow end-users to manage aspects of a system on their own behalf. Select Identity provides the following self-service capabilities: registration, profile management, and password management (including password change, reset, and synchronization).

Service

A business-centric abstraction representing resources, entitlements, and other identity-related entities. Services represent the products and services that you offer to customers and partners.

Service Attribute

A set of attributes and values that are available for or required by a Service. Attributes are created and managed through the Attributes pages.

See also: *Attributes*

Service View

A restricted view of a Service that is valid for a group of users. Views enable you to define a subset of Service registration fields, change field names, reorder fields, and mask field values for specific users.

Single Sign-On (SSO)

A session/authentication process that permits a user to enter one set of credentials (name and password) in order to access multiple applications. A Web SSO is a specialized SSO system for web applications.

SPML Data File

A file that is used to add and provision accounts within Select Identity.

See also: *Data File*

U

Users

The Select Identity capability that provides consistent account creation and management across Services.

W

Workflow

The tasks, procedural steps, organizations or people involved, and required input and output information needed for each step in a business process. In identity management, the most common workflows are for provisioning and approval processes.

Workflow Engine

A system component that executes workflows and advances them through their flow steps.

Workflow Studio

The Select Identity capability that enables you to create and manage workflow templates.

index

A

auditing, 11

B

business relationship management, 11

C

CIM, 8

configuring
 logging, 41

connectors, 12

context management, 11

Contextual Identity Management, 8

D

database server
 configuring Oracle, 21
 configuring SQL Server, 18

F

features, 9

forms, 12

functional components, 11

G

general settings, 24

I

interface requirements, 17

interface settings, 24

L

log files, 41

O

online help, 12

Oracle requirements, 16

P

policy management, 11

R

reconciliation, 11

reporting, 11

resource management, 11

S

server settings, 24

- service management, 11
- SQL Server requirements, 15
- system architecture, 10
- system errors on WebLogic, 38
- system requirements
 - database server, 15
 - interface, 17
 - overview, 15
 - web application server, 17

T

- tiered authority, 12
- troubleshooting, 38, 44
- truaccess.properties, 24, 44

U

- uninstalling, 34
- user management, 11

V

- virtual user ID, 10

W

- web application server, configuring
 - WebLogic, 25
- WebLogic requirements, 17
- welcome, 8
- workflow management, 11