Sun Java™ System

Identity Manager 6.0 Resources Reference

2005Q4M3

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 819-4520-10

Contents

Resources Reference

Synchronizing LDAP Passwords

Active Directory Synchronization Failover

Preface

This guide provides reference and procedural information to help you load and synchronize account information from a resource into Sun Java™ System Identity Manager.

How to Find Information in this Guide

The guide is organized in these sections:

- Chapter 1. Resources Reference — Identifies installation, configuration, and implementation information for Sun Java™ System Identity Manager (Identity Manager) resources.
- Chapter 2. AttrParse — Provides information needed to customize the AttrParse facility, which mainframe-based resource adapters use to extract information from the resource.
- Chapter 3. Adding Actions to Resources — Describes how to create and implement actions on UNIX, Windows NT, and Windows Active Directory resources in Identity Manager.
- Chapter 4. Synchronizing LDAP Passwords — Describes the Identity Manager product enhancements to support password synchronization from the Sun Java™ System Directory Server to the Identity Manager system.
- Chapter 5. Active Directory Synchronization Failover — Describes how to limit the number of repeated events that occur when you switch to a new domain controller.

Related Documentation and Help

Sun Microsystems provides additional printed and online documentation and information to help you install, use, and configure Identity Manager:

- *Identity Manager Installation*

  Step-by-step instructions and reference information to help you install and configure Identity Manager and associated software.

- *Identity Manager Upgrade*

  Step-by-step instructions and reference information to help you upgrade and configure Identity Manager and associated software.

- *Identity Manager Administration*

  Procedures, tutorials, and examples that describe how to use Identity Manager to provide secure user access to your enterprise information systems.

- *Identity Manager Technical Deployment Overview*

  Reference and procedural information that describes how to customize Identity Manager for your environment.

- *Identity Manager Workflows, Forms, and Views*

  Reference and procedural information that describes how to use the Identity Manager workflows, forms, and views — including information about the tools you need to customize these objects.

- *Identity Manager Deployment Tools*

  Reference and procedural information that describes how to use different Identity Manager deployment tools; including rules and rules libraries, common tasks and processes, dictionary support, and the SOAP-based Web service interface provided by the Identity Manager server.

- *Identity Manager Audit Logging*

  Reference and procedural information that describes how to load and synchronize account information from a resource into Sun Java™ System Identity Manager.

- *Identity Manager Tuning, Troubleshooting, and Error Messages*

  Reference and procedural information that describes Identity Manager error messages and exceptions, and provide instructions for tracing and troubleshooting problems you might encounter as you work.

- *Identity Manager Help*

  Online guidance and information that offers complete procedural, reference, and terminology information about Identity Manager. You can access help by clicking the Help link from the Identity Manager menu bar. Guidance (field-specific information) is available on key fields.

Product Support

If you have problems with Identity Manager, contact customer support using one of the following mechanisms:

- The online support web site at http://www.sun.com/service/online/us
- The telephone dispatch number associated with your maintenance contract

We'd Like to Hear from You!

We would like to know what you think of this guide and other documentation provided with Identity Manager. If you have feedback - positive or negative - about your experiences using this product and documentation, please send us a note:

Sun Microsystems.
5300 Riata Park Court
Austin, TX 78727
Attn: Identity Manager Information Development

Email: idm-idd@sun.com

We'd Like to Hear from You!

1 Resources Reference

This chapter describes the resource adapters that are provided with your Identity Manager installation.

The following table lists these adapters (sorted by type) and provides an overview of supported versions, Active Sync support, connection methods, and communication protocols for each adapter:

Resource	Supported Versions	Active Sync Support	Gateway?	Communication Protocols
CRM and ERP Systems				
Oracle Applications (see page 1-182)	Oracle Financials on Oracle Applications 11.5.9, 11.5.10	No	No	JDBC
PeopleSoft Component (see page 1-215)	PeopleTools 8.1 – 8.42 with HRMS 8.0 – 8.8	Yes Smart polling, Listener	No	Client connection toolkit (Sync Only)
PeopleSoft Component Interface	PeopleTools 8.1 through 8.4.	No	No	Client connection toolkit (Read/Write)
SAP (see page 1-243)	SAP R/3 4.5, 4.6, 4.7	No	No	BAPI via SAP Java Connector
	SAP HR 4.5, 4.6, 4.7	Yes Smart polling, Listener		ALE
SAP Enterprise Portal (see page 1-268)	6.20 SP2+	No	No	SAP User Management Engine
Siebel CRM (see page 1-305)	6.0, 7.0, 7.7	No	No	Siebel Data API
Databases				
DB2 (see page 1-89)	7.0, 7.2, 8.1, 8.2	No	No	JDBC, SSL
Microsoft SQL Server (see page 1-152)	2000	No	No	JDBC, SSL
MySQL (see page 1-157)	4.1	No	No	JDBC, SSL

Resource	Supported Versions	Active Sync Support	Gateway?	Communication Protocols
Databases (continued)				
Oracle (see page 1-182)	9i, 10g	No	No	JDBC, SSL
Sybase (see page 1-323)	12.*x*	No	No	JDBC, SSL
Directories				
LDAP (see page 1-134)	3.0	Yes Smart polling, Listener	No	LDAP v3, JNDI, SSL
Microsoft Active Directory (see page 1-43)	2000 SP4, 2003	Yes Smart polling	Yes	ADSI
NetWare NDS (see page 1-165)	Netware 5.1 SP6 Netware6.0 with eDirectory 8.7.1 Novell SecretStore 3.0	Yes Smart polling	Yes	NDS Client, LDAP, SSL
Message Platforms				
Lotus Domino Gateway (see page 1-93)	5.0, 6.5	Yes Smart polling	Yes	RMI, IIOP using Toolkit for Java, CORBA
Microsoft Exchange (see page 1-108)	5.5	No	Yes	ADSI
	Note: Support for the Microsoft Exchange 5.5 resource adapter has been deprecated. Use the Active Directory resource for Exchange 2000/20003, which is integrated with Exchange.			
Novell GroupWise (see page 1-115)	5.5, 6.0	No	Yes	NDS Client, LDAP, SSL
Miscellaneous				
Database Table (see page 1-84)		Yes Smart polling	No	JDBC
Flat File ActiveSync (see page 1-109)		Yes Smart polling (Internal Diff engine)	No	

Resource	Supported Versions	Active Sync Support	Gateway?	Communication Protocols
Miscellaneous (continued)				
INISafe Nexess (see page 1-124)	1.1.5		com.initech. eam.api Classes	
JMS Listener (see page 1-128)	1.1 or later	Yes	No	Varies, per resource
Microsoft Identity Integration Server (see page 1-149)	2003	No	No	JDBC
Remedy Help Desk (see page 1-238)	4.5, 5.0	Yes Smart polling	Yes	Remedy APIs
Scripted Gateway (see page 1-272	Not applicable		Yes	Varies, per resource
Scripted Host (see page 1-277)	Not applicable		No	TN3270
Sun Java™ System Communications Services (see page 1-332)		Yes	No	JNDI over SSL or TCP/IP
Operating Systems				
AIX (see page 1-75)	4.3.3, 5.2, 5.3	No	No	Telnet, SSH
HP-UX (see page 1-118)	11.0, 11i v1, 11i v2	No	No	Telnet, SSH
OS/400 (see page 1-194)	V4r3, V5r1	No	No	Java toolkit for AS400
Red Hat Linux (see page 1-232)	Linux 8.0, 9.0	No	No	Telnet, SSH
	Advanced Server 2.1, 3.0, 4.0			
Solaris (see page 1-317)	2.7, 7, 8, 9, 10	No	No	Telnet, SSH
SuSE Linux (see page 1-232)	Enterprise 9	No	No	Telnet, SSH
Windows NT, 2000, and 2003 (see page 1-367)	NT, 2000, 2003	No	Yes	ADSI

Resource	Supported Versions	Active Sync Support	Gateway?	Communication Protocols
Security Managers				
ACF2 (see page 1-24)	6.4, 6.5sp2, TSO 5.2, 5.3, CICS 2.2	No	No	Secure TN3270
ActivCard (see page 1-38)	5.0 (AIMS 3.6)	No	No	AIMS SDK, HTTPS
ClearTrust (see page 1-81)	5.01	No	No	Server Proxy API, JNDI, SSL
Natural (see page 1-160)		No	No	Secure TN3270
RACF (see page 1-224)	1.*x*, 2.*x*	No	No	Secure TN3270
SecurID ACE/Server (see page 1-296)	5.0, 6.0 for Windows	No	Yes	SecurID Admin API
	5.1, 6.0 for UNIX		SecurID TCL Interface	
Top Secret (see page 1-352)	5.3	Yes Smart polling (Filtered TSS Audit Events)	No	Secure TN3270
Web Single Sign On (SSO)				
IBM/Tivoli Access Manager (see page 1-17)	4.1, 5.1, 7	No	No	JNDI, SSL
Netegrity Siteminder (see page 1-312)	Admin 5.5	No	No	Netegrity SDK, JNDI, SSL
	LDAP 5.5			JNDI, SSL
	Table 5.5			JDBC, JNDI, SSL

Resource	Supported Versions	Active Sync Support	Gateway?	Communication Protocols
Sun Java System Access Manager (see page 1-323)	Sun ONE Identity Server 6.0, 6.1, 6.2	No	No	JNDI, SSL
		Note: Support for the Sun ONE Identity Server resource adapter has been deprecated. Use the Sun Java System Access Manager resource adapter instead.		
	Sun Java System Identity Server 2004Q2	No	No	JNDI, SSL
	Sun Java System Access Manager 6 2005Q1, 7 2005Q4			

Note The Identity Manager adapters can be often be used in their default state.

To enable an adapter,

1. Follow the installation and configuration procedures provided in the adapter's *Identity Manager Installation Notes* section in this chapter.
2. Add the resource to Identity Manager by using the Resource Wizard, as described in *Sun Java™ System Identity Manager Administration*.

Note See *Sun Java™ System Identity Manager Data Loading and Synchronization* for information about customizing adapters.

How the Adapter Sections are Organized

The resource adapter sections in this chapter are organized as follows:

- **Introduction —** Lists supported resource versions. (Refer to the Readme file supplied with your latest service pack version for updates to this list.)
- **Resource Configuration Notes** — Lists additional steps you must perform on the resource to allow you to manage the resource from Identity Manager.
- **Identity Manager Installation Notes** — Details the installation and configuration steps that you must follow to work with the resource.
- **Usage Notes** — Lists dependencies and limitations related to using the resource.
- **Security Notes** — Describes the types of connection supported as well as the authorizations needed on the resource to perform basic tasks.

- **Provisioning Notes** — Lists whether the adapter can perform tasks such as enable/disable accounts, rename accounts, and whether it allows pass-through authentication.
- **Account Attributes** — Describes default user attributes supported for the resource.
- **Resource Object Management** — Lists objects the adapter can manage.
- **Identity Template** — Provides notes about how to construct or work with the resource identity template.
- **Sample Forms** — Shows the location of a sample form you can use to construct a custom Create/Update User form. Unless otherwise indicated, sample forms are located in the *InstallDir*\idm\sample\forms\ directory.
- **Troubleshooting** — Lists the classes that can be used for tracing and debugging.

A detailed description of each topic is provided in the remainder of this section.

Topic Descriptions

This section describes the information provided for each adapter, and the topics are organized as follows:

- *Introduction*
- *Resource Configuration Notes*
- *Identity Manager Installation Notes*
- *Usage Notes*
- *Active Sync Configuration*
- *Security Notes*
- *Provisioning Notes*
- *Account Attributes*
- *Resource Object Management*
- *Identity Template*
- *Sample Forms*
- *Troubleshooting*

Introduction

The introductory section lists the versions of the resource supported by the adapter. Other versions might be supported, but they have not been tested.

This section also lists the adapter's Java class name. The class name is always used for tracing. In addition, if the resource is a custom resource, the class name must be specified on the Configure Managed Resources page. See *Identity Manager Installation Notes* on page 1-7 for more information about custom resources.

Some resources have multiple adapters. For example, Identity Manager provides adapters for Windows Active Directory and Windows Active Directory ActiveSync. In these cases, a table similar to the following is listed in the introductory section:

GUI Name	Class Name
Windows 2000 / Active Directory	`com.waveset.adapter.ADSIResourceAdapter`
Windows 2000 / Active Directory ActiveSync	`com.waveset.adapter.ActiveDirectoryActiveSyncAdapter`

The GUI name is displayed on the drop-down menu on the Resources page. Once the resource has been added to Identity Manager, this name is also displayed in the resource browser.

Resource Configuration Notes

This section lists additional steps you must perform on the resource to allow you to manage the resource from Identity Manager. (It is assumed that the resource is fully functional before you attempt to establish a connection with Identity Manager.) If there are no configuration tasks, the section will be blank or say "None".

Identity Manager Installation Notes

From an installation perspective, there are two types of adapters:

- Identity Manager adapters
- Custom adapters

Identity Manager adapters do not require additional installation procedures. Use the following steps to display the resource on the drop-down menu on the Resource page:

1. From the Identity Manager Administrative interface, click **Configure**, and then click **Managed Resources**.
2. Click the appropriate check boxes in the Identity Manager Resources section.
3. Click the **Save** button at the bottom of the page.

Custom adapters require additional installation procedures. Typically, you must copy one or more jar files to the *InstallDir*\idm\WEB-INF\lib directory and add the adapter's Java class to the list of adapters. The jar files are usually available on the installation media, or via download on the internet.

The following example from the DB2 resource adapter illustrates this procedure:

1. Copy the db2java.jar file to the *InstallDir*\idm\WEB-INF\lib directory.
2. From the Identity Manager Administrative interface, click **Configure**, and then click **Managed Resources**.
3. Click the **Add Custom Resource** button near the bottom of the page.
4. Enter the full class name of the adapter in the bottom text box, such as com.waveset.adapter.DB2ResourceAdapter.
5. Click the **Save** button at the bottom of the page.

The following table lists the adapters that require jar files to be installed on the Identity Manager server.

Adapter	Files Required
Access Manager	pd.jar
ACF2	habeans.jar —OR— • habase.jar • hacp.jar • ha3270.jar • hassl.jar • hodbase.jar
ClearTrust	ct_admin_api.jar If using SSL, these .jar files: • asn1.jar • certj.jar • jce1_2-do.jar • jcert.jar • jnet.jar • jsafe.jar • jsaveJCE.jar • jsse.jar • rsajsse.jar • sslj.jar
DB2	db2java.jar
INISafe Nexess	• concurrent.jar • crimson.jar • external-debug.jar • INICrypto4Java.jar • jdom.jar • log4j-1.2.6.jar

Adapter	Files Required
MS SQL Server	• msbase.jar • mssqlserver.jar • msutil.jar Third-party driver required for SQL Server 7.
MySQL	mysqlconnector-java-3.0.*x*-stable-bin.jar
Natural	habeans.jar —OR— • habase.jar • hacp.jar • ha3270.jar • hassl.jar • hodbase.jar
Oracle and Oracle ERP	oraclejdbc.jar
PeopleSoft Component and PeopleSoft Component Interface	psjoa.jar
RACF	habeans.jar —OR— • habase.jar • hacp.jar • ha3270.jar • hassl.jar • hodbase.jar
SAP	• jco.jar • sapidoc.jar
SAP HR ActiveSync	• jco.jar • sapidoc.jar • sapidocjco.jar

Adapter	Files Required
Scripted Host	habeans.jar —OR— • habase.jar • hacp.jar • ha3270.jar • hassl.jar • hodbase.jar
Siebel CRM	**Siebel 6:** • SiebelDataBean.jar • SiebelTC_enu.jar • SiebelTcCommon.jar • SiebelTcOM.jar **Siebel 7.0:** • SiebelJI_Common.jar • SiebelJI_enu.jar • SiebelJI.jar **Siebel 7.7** • Siebel.jar • SiebelJI_enu.jar
SiteMinder	• smjavaagentapi.jar • smjavasdk2.jar
Sun Java System Access Manager	• am_sdk.jar • am_services.jar
Sybase	jconn2.jar
Top Secret	habeans.jar —OR— • habase.jar • hacp.jar • ha3270.jar • hassl.jar • hodbase.jar

Usage Notes

This section lists dependencies and limitations related to using the resource. The contents of this section varies between adapters.

Active Sync Configuration

This section provides resource-specific configuration information that can be viewed on the General Active Sync Settings page of the Active Sync Wizard. The following attributes are applicable to most Active Sync adapters.

Parameter	Description
Process Rule	Either the name of a TaskDefinition, or a rule that returns the name of a TaskDefinition, to run for every record in the feed. The process rule gets the resource account attributes in the activeSync namespace, as well as the resource ID and name. This parameter overrides all others. If this attribute is specified, the process will be run for every row regardless of any other settings on this adapter.
Correlation Rule	If no Identity Manager user's resource info is determined to own the resource account, the Correlation Rule is invoked to determine a list of potentially matching users/accountIDs or Attribute Conditions, used to match the user, based on the resource account attributes (in the account namespace). The rule returns one of the following pieces of information that can be used to correlate the entry with an existing Identity Manager account: • Identity Manager user name • WSAttributes object (used for attribute-based search) • List of items of type AttributeCondition or WSAttribute (AND-ed attribute-based search) • List of items of type String (each item is the Identity Manager ID or the user name of an Identity Manager account) If more than one Identity Manager account can be identified by the correlation rule, a confirmation rule or resolve process rule will be required to handle the matches. For the Database Table, Flat File, and PeopleSoft Component Active Sync adapters, the default correlation rule is inherited from the reconciliation policy on the resource.

Parameter	Description
Confirmation Rule	Rule which is evaluated for all users returned by a correlation rule. For each user, the full user view of the correlation Identity Manager identity and the resource account information (placed under the “account.” namespace) are passed to the confirmation rule. The confirmation rule is then expected to return a value which may be expressed like a Boolean value. For example, “true” or “1” or “yes” and “false” or “0” or null. For the Database Table, Flat File, and PeopleSoft Component Active Sync adapters, the default confirmation rule is inherited from the reconciliation policy on the resource.
Delete Rule	A rule that can expect a map of all values with keys of the form `activeSync.` or `account.` A LighthouseContext object (`display.session`) based on the proxy administrator’s session is made available to the context of the rule. The rule is then expected to return a value which may be expressed like a Boolean value. For example, “true” or “1” or “yes” and “false” or “0” or null. If the rule returns true for an entry, the account deletion request will be processed through forms and workflow, depending on how the adapter is configured.
Resolve Process Rule	Either the name of the TaskDefinition or a rule that returns the name of a TaskDefinition to run in case of multiple matches to a record in the feed. The Resolve Process rule gets the resource account attributes as well as the resource ID and name. This rule is also needed if there were no matches and **Create Unmatched Accounts** is not selected. This workflow could be a process that prompts an administrator for manual action.
Create Unmatched Accounts	If set to true, creates an account on the resource when no matching Identity Manager user is found. If false, the account is not created unless the process rule is set and the workflow it identifies determines that a new account is warranted. The default is true.
Populate Global	If set to true, populates the global namespace in addition to the activeSync namespace. The default value is false.

Security Notes

The Security Notes section provides connection and authorization information.

Supported Connections - Lists the type of connection used to communicate between Identity Manager and the resource. The following types of connections are commonly used:

- Sun Identity Manager Gateway
- Secure Shell (SSH)
- Java Database Connectivity (JDBC) over Secure Sockets Layer (SSL)
- Java Naming and Directory Interface (JNDI) over SSL
- Telnet/TN3270

Other connection types are possible.

Required Administrative Privileges - Lists the privileges the administrator account must have to create users and perform other tasks from within Identity Manager. The administrator account is specified on the Resource Attributes page.

Provisioning Notes

This section contains a table that summarizes the provisioning capabilities of the adapter. These capabilities include

- **Enable/Disable Account** — The ability to enable and disable user accounts is determined by the resource. For example, on some UNIX systems, an account is disabled by changing the password to a random value.
- **Rename Account** — The ability to rename user accounts is determined by the resource.
- **Pass-Through Authentication** — A Identity Manager feature that enables resource users to login to the Identity Manager User interface.
- **Before/After Actions** — Actions are scripts that run within the context of a managed resource, if native support exists for scripted actions.

  For example, on UNIX systems, actions are sequences of UNIX shell commands. In Microsoft Windows environments, actions are DOS-style console commands that can execute within the CMD console.
- **Data Loading Methods** — Indicates how data can be loaded into Identity Manager. The following methods are supported:
  - **ActiveSync** — Allows information that is stored in an “authoritative” external resource (such as an application or database) to synchronize with Identity Manager user data. The adapter can push or pull resource account changes into Identity Manager.
  - **Discovery** (load from resource) — Initially pulls resource accounts into Identity Manager, without viewing before loading. Resource account information can also be imported from or exported to a file.

- **Reconciliation** — Periodically pull resource accounts into Identity Manager, taking action on each account according to configured policy. Use the reconciliation feature to highlight inconsistencies between the resource accounts on Identity Manager and the accounts that actually exist on a resource, and to periodically correlate account data.

Account Attributes

The account attributes, or schema map, maps Identity Manager account attributes to resource account attributes. The list of attributes varies for each resource. You may remove unused attributes from the schema map page. However, adding attributes might require editing the user forms or other code.

The Identity Manager User Attributes can be used in rules, forms, and other Identity Manager-specific functions. The Resource User Attributes are used only when the adapter communicates with the resource.

Resource Object Management

Lists the objects on the resource that can be managed through Identity Manager.

Identity Template

Defines account name syntax for users. For most resources, the syntax is the same as the account ID. However, the syntax is different if the resource uses hierarchical namespaces.

Sample Forms

A form is an object associated with a page that contains rules about how the browser should display user view attributes on that page. Forms can incorporate business logic and are often used to manipulate view data before it is presented to the user.

Forms can be edited with the Identity Manager Business Process Editor (BPE). The BPE is a standalone, Swing-based Java application that allows you to create and edit forms. By selecting form and field definitions from various dialogs and menus, you can quickly customize the content and appearance of Identity Manager pages. For more information, see the *Identity Manager Workflows, Forms, and Views*.

Built-In Forms

Some forms are loaded into the Identity Manager repository by default. To view a list of forms in the repository, perform the following steps:

1. From a web browser, go to `http://IdentityManagerHost/idm/debug`

   The browser displays the System Settings page.
2. From the options menu adjacent to **List Objects**, select **Type: ResourceForm**.
3. Click **List Objects**. The **List Objects of Type: ResourceForm** page is displayed. This page lists all editable forms that reside in the Identity Manager repository.

Also Available

Identity Manager provides many additional forms that are not loaded by default. These forms are located in the `InstallDir\idm\sample\forms\` directory.

Troubleshooting

Trace output can be helpful when identifying and resolving problems with any adapter. Generally, these are the steps you will follow when using tracing to help identify and resolve problems:

1. Turn on tracing.
2. Reproduce the problem and evaluate the results.
3. Optionally turn tracing on for additional packages or classes, or turn up the tracing level and repeat steps 2 and 3 as needed.
4. Turn off tracing.

To turn tracing on, follow these steps:

1. Log in to Identity Manager as the Configurator account
2. Go to the Debug page: `http://IdentityManagerHost/idm/debug`
3. Click **Show Trace**
4. Ensure that Trace Enabled is checked
5. Enter the full class name in the **Method/Class** text box.
6. Enter a trace level (1-4). Each level captures different types of information:
   - **1** – Entry and exit of public methods, plus major exceptions.
   - **2** – Entry and exit of all methods.
   - **3** – Significant informational displays (such as the value of variables that control flow) that occur only once per method invocation.
   - **4** – Informational displays that occur *n* times per method invocation.
7. Fill out the rest of the page as desired. Click **Save** when you are ready to begin tracing.

To disable tracing, either deselect the Show Trace option, or delete the class name from the Method/Class text box.

Access Manager

The Tivoli Access Manager resource adapter is defined in the `com.waveset.adapter.AccessManagerResourceAdapter` class.

This resource adapter supports the following versions of Access Manager:

- 4.1, 5.1
- 7

Resource Configuration Notes

This section provides instructions for configuring Access Manager resources; including:

- General instructions for setting up the IBM Tivoli Access Manager resource for use with Identity Manager
- Instructions for using Access Manager as the Web Access Control for Identity Manager

General Configuration

Follow these steps when setting up the IBM Tivoli Access Manager resource for use with Identity Manager:

1. Install the IBM Tivoli Access Manager Java Runtime Component on the Identity Manager server.
2. Set your PATH variable to include the path to the JVM for your application server. For example,
   - If you have a WebLogic 7.x install on a UNIX server, set your path to:

     ```
     PATH=$WLHOME/bea/jdk131_04/bin:$WLHOME/bea/jdk131_04/jre/
     bin:$PATH
     ```

   - If you have a Websphere 4.x install on a Windows 2000 server, set your path to:

     ```
     set PATH=%WebSphere%\AppServer\java\bin;%WebSphere%\
     AppServer\java\jre\bin;%PATH%
     ```

3. Run the `pdjrtecfg -action config` command to install the following Access Manager `.jar` files to the JRE's `lib/ext` directory:
   - `ibmjceprovider.jar`
   - `ibmjsse.jar`
   - `ibmpkcs.jar`
   - `jaas.jar`
   - `local_policy.jar`
   - `PD.jar`
   - `US_export_policy.jar`
   - `ibmjcefw.jar`

   Note For more information, see the *IBM Tivoli Access Manager Base Installation Guide*.

4. Remove the following jar files from the *`InstallDir`*`\idm\WEB-INF\lib` directory (depending on your application server, these files may have been removed during the Identity Manager product installation):
   - `jsse.jar`
   - `jcert.jar`
   - `jnet.jar`
   - `cryptix-jce-api.jar`
   - `cryptix-jce-provider.jar`
5. Modify the `java.security` file:

   ```
   security.provider.2=com.ibm.crypto.provider.IBMJCE
   security.provider.3=com.ibm.net.ssl.internal.ssl.Provider
   ```

6. Add the VM parameter to the application server:

   ```
   -Djava.protocol.handler.pkgs= \
     com.ibm.net.ssl.internal.www.protocol
   ```

   Note If necessary, you can add multiple packages by delimiting with a `|` (pipe symbol). For example:

   ```
   -Djava.protocol.handler.pkgs=sun.net.www.protocol| \
     com.ibm.net.ssl.internal.www.protocol
   ```

7. Make sure the IBM Tivoli Access Manager Authorization Server is configured and running.

8. Run the command:

   ```
   SvrSslCfg
   ```

   For example:

   ```
   java com.tivoli.pd.jcfg.SvrSslCfg -action config \
   -admin_id sec_master -admin_pwd secpw \
   -appsvr_id PDPermissionjapp -host amazn.myco.com \
   -mod local -port 999 -policysvr ampolicy.myco.com:7135:1 \
   -authzsvr amazn.myco.com:7136:1 -cfg_file c:/am/configfile \
   -key_file c:/am/keystore -cfg_action create
   ```

   The "am" directory must already exist. Successful completion creates these files in the `c:\am` directory:

   - `configfile`
   - `keystore`

Note For more information, see *IBM Tivoli Access Manager Authorization Java Classes Developer's Reference* and *IBM Tivoli Access Manager Administration Java Classes Developer's Reference*.

Setting up Web Access Control

The following procedure describes the general configuration steps to use Tivoli Access Manager as the Web Access Control for Identity Manager. Some of the following steps require detailed knowledge of the Tivoli Access Manager software.

1. Install and configure IBM Tivoli Access Manager Java Runtime Component on the Identity Manager server.
2. Configure the JDK Security Settings on the Identity Manager server.
3. Create the Access Manager SSL Config files on the Identity Manager server.
4. Create a Junction in Access Manager for the Identity Manager URLs. Refer to the Tivoli Access Manager product documentation for more details.

   The following example `pdadmin` command illustrates how to create a junction:

   ```
   pdadmin server task WebSealServer create -t Connection /
   -p Port -h Server -c ListOfCredentials -r /
   -i JunctionName
   ```

5. Configure the Identity Manager Base HREF property for the WebSeal Proxy Server.
6. Setup the Access Manager resource adapter.
7. Load the Access Manager users into Identity Manager.
8. Configure Pass-Through Authentication for Access Manager in Identity Manager.

When a user attempts to access the Identity Manager URLs via Access Manager, the user's identity is passed in the HTTP header to Identity Manager. Identity Manager then uses that identity to verify the user exists in Access Manager and in Identity Manager. If the user is trying to access the Identity Manager Administrator interface, Identity Manager checks the Identity Manager Security configuration for the user to make sure they have Identity Manager administrative rights. End users are also verified against Access Manager, and whether they have a Identity Manager account.

Identity Manager Installation Notes

Note If you are installing IBM Tivoli Access Manager with a WebSphere application server, do not copy the `jsse.jar`, `jcert.jar`, and `jnet.jar` files during Identity Manager installation to the `WEB-INF\lib` directory; otherwise, a conflict results.

The Access Manager resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. Copy the `pd.jar` file from the Access Manager installation media to the `$WSHOME/WEBINF/lib` directory.
2. Add the following value in the Custom Resources section of the Configure Managed Resources page:

   ```
   com.waveset.adapter.AccessManagerResourceAdapter
   ```

Usage Notes

This section lists dependencies and limitations related to using the Access Manager resource adapter.

Notes:

- When accessing Identity Manager through Access Manager, the browser must use a JRE Version 1.3.1 or earlier for applets to display correctly.
- If you want to use the Identity Manager single sign-on or pass-through authentication features with this resource, you must use Access Manager as the Identity Manager proxy server. For more information on proxy servers, see , *Identity Manager Deployment Tools*.

Creating GSO Credentials

To configure GSO Web Resource or GSO Resource Group credentials from the Identity Manager Create User page, perform the following steps:

1. Select **`Add GSO Web Credentials`** or **`GSO Resource Group Credentials`**.
2. Select a target from the appropriate GSO credential drop-down menu.
3. Enter a resource user ID and password in the text fields.
4. You may edit the resource credential user ID and/or password by editing the appropriate field. For security reasons, the credential password is never retrieved.

Deleting GSO Credentials

To delete a credential, select it from the table and then click the corresponding **Remove** button.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses JNDI over SSL to communicate with Access Manager.

Required Administrative Privileges

The administrative user must have sufficient privileges to create, update, and delete users, groups, web resources, and resource groups.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	No
Pass-through authentication	Yes
Before/after actions	No
Data loading methods	• Import directly from resource • Reconciliation

Account Attributes

The following table provides information about Access Manager account attributes.

Attribute	Data Type	Description
firstname	string	Required. The user's first name.
lastname	string	Required. The user's last name.
registryUID	string	Required. The account name stored in the user registry.
description	string	Text describing the user.
groups	string	The Access Manager groups that the user is a member of.
noPwdPolicy	boolean	Indicates whether a password policy is enforced.
ssoUser	boolean	Indicates whether the user has single sign-on abilities.
expirePassword	boolean	Indicates whether the password will be expired.
importFromRgy	boolean	Indicates whether to import group data from the user registry.
deleteFromRgy	boolean	Indicates whether the user should be deleted.

Attribute	Data Type	Description
syncGSOCreds	boolean	Indicates whether to synchronize GSO passwords to the Access Manager password.
gsoWebCreds	string	A list of web resource credentials the user has access to.
gsoGroupCreds	string	A list of resource group credentials the user has access to.

Resource Object Management

Identity Manager supports the following objects:

Resource Object	Features Supported	Attributes Managed
Group	Create, find, update, delete	name, description, registry name, member

Identity Template

The account name syntax is:

```
$accountId$
```

Sample Forms

Identity Manager provides the `AccessManagerUserForm.xml` sample form.

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

```
com.waveset.adapter.AccessManagerResourceAdapter
```

ACF2

The ACF2 resource adapter supports management of user accounts and memberships on an OS/390 mainframe via the IBM Host Access Class Library APIs. The adapter manages ACF2 over a TN3270 emulator session.

The ACF2 adapter supports the following versions:

- **ACF2**: 6.4, 6.5 SP2
- **TSO**: 5.2, 5.3
- **CICS**: 2.2

The ACF2 resource adapter is defined in the `com.waveset.adapter.ACF2ResourceAdapter` class.

Resource Configuration Notes

None

Identity Manager Installation Notes

The ACF2 resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. To add the ACF2 resource to the Identity Manager resources list, you must add the following value in the Custom Resources section of the Configure Managed Resources page.

   ```
   com.waveset.adapter.ACF2ResourceAdapter
   ```

2. The Identity Manager mainframe adapters use the IBM Host Access Class Library (HACL) to connect to the mainframe. The HACL is available in IBM Websphere Host On-Demand (HOD). The recommended jar containing HACL is `habeans.jar` and is installed with the HOD Toolkit (or Host Access Toolkit) that comes with HOD. The supported versions of HACL are in HOD V7.0, V8.0, and V9.0.

   However, if the toolkit installation is not available, the HOD installation contains the following jars that can be used in place of the `habeans.jar`:

   - `habase.jar`
   - `hacp.jar`
   - `ha3270.jar`

- `hassl.jar`
- `hodbase.jar`

Copy the `habeans.jar` file or all of its substitutes into the `WEB-INF/lib` directory of your Identity Manager installation. See http://www.ibm.com/software/webservers/hostondemand/ for more information.

Usage Notes

This section lists dependencies and limitations related to using the ACF2 resource adapter.

Administrators

TSO sessions do not allow multiple, concurrent connections. To achieve concurrency for Identity Manager ACF operations, you must create multiple administrators. Thus, if you create two administrators, two Identity Manager ACF operations can occur at the same time. We recommend that you create at least two (and preferably three) administrators.

If you are running in a clustered environment, you must define an admin for each server in the cluster. This applies even if it is the same admin. For TSO, there must be a different admin for each server in the cluster.

If clustering is not being used, the server name should be the same for each row (the name of the Identity Manager host machine).

Note Host resource adapters *do not* enforce maximum connections for an affinity administrator across multiple host resources connecting to the same host. Instead, the adapter enforces maximum connections for affinity administrators within each host resource.

If you have multiple host resources managing the same system, and they are currently configured to use the same administrator accounts, you might have to update those resources to ensure that the same administrator is not trying to perform multiple actions on the resource simultaneously.

Resource Actions

The ACF2 adapter requires login and logoff resource actions. The login action negotiates an authenticated session with the mainframe. The logoff action disconnects when that session is no longer required.

See the Usage Notes for the Top Secret adapter on page 1-353 for more information about creating login and logoff resource actions.

SSL Configuration

This section provides information about configuring SSL, including:

- *Connecting the Adapter to a Telnet/TN3270 Server using SSL or TLS*
- *Generating a PKCS #12 File*
- *Troubleshooting the SSL Connection*

Connecting the Adapter to a Telnet/TN3270 Server using SSL or TLS

Use the following steps to connect ACF2 resource adapters to a Telnet/TN3270 server using SSL/TLS.

1. Obtain the Telnet/TN3270 server's certificate in the PKCS #12 file format. Use `hod` as the password for this file. Consult your server's documentation on how to export the server's certificate. The procedure "Generating a PKCS #12 File" below for some general guidelines.
2. Create a `CustomizedCAs.class` file from the PKCS #12 file. If you are using a recent version of HOD, use the following command to do this.

   ```
   ..\hod_jre\jre\bin\java -cp ../lib/ssliteV2.zip;../lib/sm.zip
   com.ibm.eNetwork.HOD.convert.CVT2SSLIGHT CustomizedCAs.p12 hod
   CustomizedCAs.class
   ```
3. Place the `CustomizedCAs.class` file somewhere in the Identity Manager server's classpath, such as `$WSHOME/WEB-INF/classes`.
4. If a resource attribute named **`Session Properties`** does not already exist for the resource, then use the BPE or debug pages to add the attribute to the resource object. Add the following definition in the `<ResourceAttributes>` section:

   ```
   <ResourceAttribute name='Session Properties' displayName='Session
   Properties' description='Session Properties' multi='true'>
   </ResourceAttribute>
   ```
5. Go to the Resource Parameters page for the resource and add the following values to the **`Session Properties`** resource attribute:

   ```
   SESSION_SSL
   true
   ```

Generating a PKCS #12 File

The following procedure provides a general description of generating a PKCS #12 file when using the Host OnDemand (HOD) Redirector using SSL/TLS. Refer to the HOD documentation for detailed information about performing this task.

1. Create a new `HODServerKeyDb.kdb` file using the IBM Certificate Management tool. As part of that file, create a new self-signed certificate as the default private certificate.

   If you get a message that is similar to “error adding key to the certificate database” when you are creating the `HODServerKeyDb.kdb` file, one or more of the Trusted CA certificates may be expired. Check the IBM website to obtain up-to-date certificates.
2. Export that private certificate as Base64 ASCII into a `cert.arm` file.
3. Create a new PKCS #12 file named `CustomizedCAs.p12` with the IBM Certificate Management tool by adding the exported certificate from the `cert.arm` file to the Signer Certificates. Use `hod` as the password for this file.

Troubleshooting the SSL Connection

You can enable tracing of the HACL by adding the following to the Session Properties resource attribute:

```
SESSION_TRACE
ECLSession=3 ECLPS=3 ECLCommEvent=3 ECLErr=3 DataStream=3 Transport=3
ECLPSEvent=3
```

Note The trace parameters should be listed without any new line characters. It is acceptable if the parameters wrap in the text box.

The Telnet/TN3270 server should have logs that may help as well.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses TN3270 connections to communicate with ACF2.

Required Administrative Privileges

The administrators that connect to ACF2 must be assigned sufficient privileges to create and manage ACF2 users.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	Yes
Pass-through authentication	No
Before/after actions	Yes
Data loading methods	• Import directly from resource • Reconciliation

Account Attributes

The following table provides information about ACF2 account attributes.

Resource User Attribute	Data Type	Description
NAME	string	The user name displayed on logging and security violation reports
PHONE	string	The user's telephone number
ACCESS.ACC-CNT	string	The number of system accesses made by this logonid since it was created
ACCESS.ACC-DATE	string	The date of this user's last system access
ACCESS.ACC-SRCE	string	The logical or physical input source name or source group name where this logonid last accessed the system
ACCESS.ACC-TIME	string	The time of this user's last system access
CANCEL/SUSPEND.CANCEL	boolean	The logonid is canceled and denied access to the system

Resource User Attribute	Data Type	Description
CANCEL/SUSPEND.CSDATE	string	The date when the CANCEL or SUSPEND field was set
CANCEL/SUSPEND.CSWHO	string	The logonid that set the CANCEL, SUSPEND, or MONITOR field
CANCEL/SUSPEND.MON-LOG	boolean	ACF2 writes an SMF record each time this user enters the system
CANCEL/SUSPEND.MONITOR	boolean	CA-ACF2 sends a message to the security console and to a designated person (CSWHO) each time this user enters the system
CANCEL/SUSPEND.SUSPEND	boolean	The logonid is suspended and denied access to the system
CANCEL/SUSPEND.TRACE	boolean	All data references by this user are traced and logged
CICS.ACF2CICS	boolean	Indicates that CA-ACF2 CICS security is to be initialized in any CICS/ESA 4.1 or later region running with this address space logonid
CICS.CICSCL	string	CICS operator class
CICS.CICSID	string	CICS operator ID
CICS.CICSKEY	string	The first three bytes of transaction security key values to support CICS Release 1.6 and later.
CICS.CICSKEYX	string	The last five bytes of transaction security key values to support CICS Release 1.6 and later.
CICS.CICSPRI	string	CICS operator priority
CICS.CICSRSL	string	CICS resource access key
CICS.IDLE	string	The maximum number of minutes permitted between terminal transactions for this user
IMS.MUSDLID	string	The default logonid for a MUSASS address space.
IDMS.IDMSPROF	string	The name of the sign-on profile CLIST executed when the user signs on to CA-IDMS
IDMS.IDMSPRVS	string	The version of the sign-on profile CLIST executed when the user sign on to CA-IDMS

Resource User Attribute	Data Type	Description
MUSASS.MUSID	string	Groups IMS records in the Infostorage database to ensure that IMS records are associated with the proper control region.
MUSASS.MUSIDINF	boolean	The MUSID field should be used to restrict access to a MUSASS region for CA-ACF2 Info type system entry calls.
MUSASS.MUSOPT	string	The name of the CA-ACF2 CA-IDMS options module that controls the CAIDMS address space.
MUSASS.MUSPGM	string	The name of the CA-IDMS start up program
MUSASS.MUSUPDT	boolean	Allows the user to update the CA-ACF2 databases.
PRIVILEGES.ACCOUNT	boolean	The user can insert, delete, and change logonids, as limited by a scope
PRIVILEGES.ACTIVE	string	The logonid is automatically activated one minute after midnight on the date contained in this field
PRIVILEGES.AUDIT	boolean	With this privilege, a user can inspect, but not modify, the parameters of the CAACF2 system.
PRIVILEGES.AUTODUMP	boolean	Dump created when a data set or resource violation occurs.
PRIVILEGES.AUTONOPW	boolean	This virtual machine can be autologged without specifying a password.
PRIVILEGES.BDT	boolean	This logonid’s address space belongs to the Bulk Data Transfer (BDT) product.
PRIVILEGES.CICS	boolean	The logonid has the authority to sign on to CICS.
PRIVILEGES.CMD-PROP	boolean	This indicates that the user can override the global CPF target list by using the SET TARGET command or the TARGET parameter
PRIVILEGES.CONSULT	boolean	The user can display other logonids.
PRIVILEGES.DUMPAUTH	boolean	This user can generate a dump even when the address space is in an execute-only or path control environment
PRIVILEGES.EXPIRE	string	The date when .temporary. logonids expire

Resource User Attribute	Data Type	Description
PRIVILEGES.IDMS	boolean	The logonid has the authority to sign on to CA-IDMS.
PRIVILEGES.JOB	boolean	The user can enter batch and background Terminal Monitor Program (TMP) jobs.
PRIVILEGES.JOBFROM	boolean	The user can use the //*JOBFROM control statement.
PRIVILEGES.LEADER	boolean	The user can display and alter certain fields of other logonids for other users.
PRIVILEGES.LOGSHIFT	boolean	A user can access the system outside the time period specified in the SHIFT field of the logonid record.
PRIVILEGES.MAINT	boolean	A user can use a specified program executed from a specified library to access resources without loggings or validation.
PRIVILEGES.MUSASS	boolean	This logonid is a multiple user single address space system (MUSASS).
PRIVILEGES.NO-INH	boolean	A network job cannot inherit this logonid from its submitter.
PRIVILEGES.NO-SMC	boolean	Step-must-complete (SMC) controls are bypassed; a job is considered noncancelable for the duration of the sensitive VSAM update operation.
PRIVILEGES.NO-STORE	boolean	This user is unauthorized to store or delete rule sets
PRIVILEGES.NON-CNCL	boolean	A user can access all data, even if a rule prohibits this access.
PRIVILEGES.PGM	string	The specified APF-authorized program to submit jobs for this logonid.
PRIVILEGES.PPGM	boolean	The user can execute those protected programs specified in the GSO PPGM record.
PRIVILEGES.PRIV-CTL	boolean	Checks privilege control resource rules when the user accesses the system to see what additional privileges and authorities the user has.
PRIVILEGES.PROGRAM	string	The specified APF-authorized program to submit jobs for this logonid.

Resource User Attribute	Data Type	Description
PRIVILEGES.READALL	boolean	The logonid has only read access to all data at the site.
PRIVILEGES.REFRESH	boolean	This user is authorized to issue the F ACF2,REFRESH operator command from the operator.s console.
PRIVILEGES.RESTRICT	boolean	This restricted logonid is for production use and does not require a password for user verification
PRIVILEGES.RSRCVLD	boolean	Specifies that a resource rule must authorize any accesses that a user makes.
PRIVILEGES.RULEVLD	boolean	An access rule must exist for all data this user accesses.
PRIVILEGES.SCPLIST	string	The infostorage scope record that restricts accesses for this privileged user.
PRIVILEGES.SECURITY	boolean	This user is a security administrator who, in the limits of his scope, can create, maintain, and delete access rules, resource rules, and infostorage records.
PRIVILEGES.STC	boolean	Only started tasks use this logonid
PRIVILEGES.SUBAUTH	boolean	Only an APF-authorized program can submit jobs specifying this logonid.
PRIVILEGES.SYNCNODE	string	The node where the synchronized logonid for this logonid is found in the Logonid database
PRIVILEGES.TAPE-BLP	boolean	This user can use full bypass label processing (BLP) when accessing tape data sets
PRIVILEGES.TAPE-LBL	boolean	This user has limited BLP when accessing tape data sets.
PRIVILEGES.TSO	boolean	This user is authorized to sign on to TSO.
PRIVILEGES.VAX	boolean	This logonid has associated VAX (UAF) infostorage records.
PRIVILEGES.VLDRSTCT	boolean	Turning on this field for a RESTRICT logonid indicates that PROGRAM and SUBAUTH are to be validated even when the logonid is inherited

Resource User Attribute	Data Type	Description
PASSWORD.MAXDAYS	string	The maximum number of days permitted between password changes before the password expires. If the value is zero, no limit is enforced
PASSWORD.MINDAYS	string	The minimum number of days that must elapse before the user can change the password
PASSWORD.PSWD-DAT	string	The date of the last invalid password attempt
PASSWORD.PSWD-EXP	boolean	The user's password was manually expired (forced to expire).
PASSWORD.PSWD-INV	string	The number of password violations that occurred since the last successful logon
PASSWORD.PSWD-SRCE	string	The logical or physical input source name or source group name where the last invalid password for this logonid was received
PASSWORD.PSWD-TIM	string	The time when the last invalid password for this logonid was received
PASSWORD.PSWD-TOD	string	The date and time the password was last changed
PASSWORD.PSWD-VIO	string	The number of password violations occurring on PSWD-DAT.
PASSWORD.PSWD-XTR	boolean	The password for this logonid is halfway-encrypted and can be extracted by an APF-authorized program
RESTRICTIONS.AUTHSUP1 through AUTHSUP8	boolean	These fields can activate extended user authentication (EUA) for each designated system user
RESTRICTIONS.GROUP	string	The group or project name associated with this user.
RESTRICTIONS.PREFIX	string	The high-level index of the data sets that this user owns and can access.
RESTRICTIONS.SHIFT	string	The shift record that defines when a user is permitted to log on to the system.
RESTRICTIONS.SOURCE	string	The logical or physical input source name or source group name where this logonid must access the system

Resource User Attribute	Data Type	Description
RESTRICTIONS.VMACCT	string	A loginid field that holds the default account number for a virtual machine.
RESTRICTIONS.VMIDLEMN	string	The number of minutes that this user can be idle on the system before idle terminal processing begins.
RESTRICTIONS.VMIDLEOP	string	The type of idle terminal processing to perform when the user exceeds the idle time limit.
RESTRICTIONS.ZONE	string	The name of the Infostorage Database zone record defining the time zone where this logonid normally accesses the system (that is, the user's local time zone)
STATISTICS.SEC-VIO	string	The total number of security violations for this user.
STATISTICS.UPD-TOD	string	The date and time that this logonid record was last updated.
TSO.ACCTPRIV	boolean	Indicates the user has TSO accounting privileges
TSO.ALLCMDS	boolean	The user can enter a special prefix character to bypass the CA-ACF2 restricted command lists.
TSO.ATTR2	string	The IBM program control facility (PCF) uses the PSCBATR2 field for command limiting and data set protection.
TSO.CHAR	string	The TSO character-delete character for this user
TSO.CMD-LONG	boolean	Indicates that only the listed command and aliases are accepted when using TSO command lists.
TSO.DFT-DEST	string	The default remote destination for TSO spun SYSOUT data sets
TSO.DFT-PFX	string	The default TSO prefix that is set in the user's profile at logon time.
TSO.DFT-SOUT	string	The default TSO SYSOUT class
TSO.DFT-SUBC	string	The default TSO submit class
TSO.DFT-SUBH	string	The default TSO submit hold class

Resource User Attribute	Data Type	Description
TSO.DFT-SUBM	string	The default TSO submit message class
TSO.INTERCOM	boolean	This user is willing to accept messages from other users through the TSO SEND command.
TSO.JCL	boolean	This user can submit batch jobs from TSO and use the SUBMIT, STATUS, CANCEL, and OUTPUT commands
TSO.LGN-ACCT	boolean	This user can specify an account number at logon time.
TSO.LGN-DEST	boolean	The user can specify a remote output destination at TSO logon that overrides the value specified in the DFT-DEST field
TSO.LGN-MSG	boolean	This user can specify message class at logon time.
TSO.LGN-PERF	boolean	This user can specify a performance group at logon time.
TSO.LGN-PROC	boolean	This user can specify the TSO procedure name at logon time.
TSO.LGN-RCVR	boolean	This user can use the recover option of the TSO or TSO/E command package.
TSO.LGN-SIZE	boolean	This user is authorized to specify any region size at logon time.
TSO.LGN-TIME	boolean	This user can specify the TSO session time limit at logon time.
TSO.LGN-UNIT	boolean	This user can specify the TSO unit name at logon time.
TSO.LINE	string	The TSO line-delete character
TSO.MAIL	boolean	Receive mail messages from TSO at logon time.
TSO.MODE	boolean	Receive modal messages from TSO.
TSO.MOUNT	boolean	This user can issue mounts for devices.
TSO.MSGID	boolean	Prefix TSO message IDs.
TSO.NOTICES	boolean	Receive TSO notices at logon time.
TSO.OPERATOR	boolean	This user has TSO operator privileges.

Resource User Attribute	Data Type	Description
TSO.PAUSE	boolean	Causes a program to pause when a command executed in a CLIST issues a multilevel message.
TSO.PMT-ACCT	boolean	Forces this user to specify an account number at logon time.
TSO.PMT-PROC	boolean	Forces this user to specify a TSO procedure name at logon time.
TSO.PROMPT	boolean	Prompt for missing or incorrect parameters.
TSO.RECOVER	boolean	Use the recover option of the TSO or TSO/E command package.
TSO.TSOACCT	string	The user's default TSO logon account
TSO.TSOCMDS	string	The name of the TSO command list module that contains the list of the commands that this user is authorized to use.
TSO.TSOFSCRN	boolean	This user has the full-screen logon display.
TSO.TSOPERF	string	The user's default TSO performance group
TSO.TSOPROC	string	The user's default TSO procedure name
TSO.TSORBA	string	The mail index record pointer (MIRP) for this user
TSO.TSORGN	string	The user's default TSO region size (in K bytes) if the user does not specify a size at logon time.
TSO.TSOSIZE	string	The user's maximum TSO region size (in K bytes) unless the user has the LGS-SZE field specified
TSO.TSOTIME	string	The user's default TSO time parameter
TSO.TSOUNIT	string	The user's default TSO unit name
TSO.VLD-ACCT	boolean	Indicates CA-ACF2 is to validate the TSO account number.
TSO.VLD-PROC	boolean	Indicates CA-ACF2 is to validate the TSO procedure name.
TSO.WTP	boolean	Displays write-to-programmer (WTP) messages.

Resource Object Management

None

Sample Forms

`ACF2UserForm.xml`

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.HostAccess`
- `com.waveset.adapter.ACF2ResourceAdapter`

See the Troubleshooting section for the Top Secret adapter on page 1-365 for more information about troubleshooting the HostAccess class.

ActivCard

The ActivCard resource adapter is defined in the `com.waveset.adapter.ActivCardResourceAdapter` class.

This adapter supports the following version of ActivCard AIMS:

- 5.0 (AIMS Enterprise SDK version 3.6)

Resource Configuration Notes

The paths to the client and root certificate files on the machine running Identity Manager are required, as well as the client certificate passphrase and keystore type. In addition, the following ActivCard configuration information is required:

- URL of AIMS server (which must match the certificate exactly)
- Port on AIMS server for communication (via https)
- Base DN for users created by ActivCard within its repository
- Objectclass of ActivCard users within ActivCard's repository
- Unique ID used by ActivCard within its repository

To view the name of the base node from within the ActivCard Identity Management System, click the Configuration tab, then click the Repositories link. Information about the directory can be displayed by clicking on the View link on that page. To view the User ID attribute, click Configuration, then the Customization link, then select “Directories” from the “Select a Topic” drop down list.

Identity Manager Installation Notes

You must install the ActivCard adapter on one of the following types of application servers:

- Java 1.4 using JSSE, such as Tomcat 5
- WebLogic 8

Identity Manager supports the ActivCard adapter without configuring the System Configuration object if your application server runs on Java 1.4 with JSSE.

If the application server is WebLogic 8, then add the following attribute in the System Configuration object in the top-level System settings (along with the other Attribute definitions).

```
<Attribute name='httpsHandler'
value='com.waveset.util.HttpsUtilImpl_Weblogic8'/>
```

In a single-server environments, specify the attribute as a top-level setting. In a clustered environment, the `httpsHandler` attribute can be specified in either location.

Note The value of the `httpsHandler` attribute can also be `com.waveset.util.HttpsUtilImpl_JSSE_1_4`. This value is supported by default.

Access to the AIMS server is controlled through certificates that must be installed on the machine running Identity Manager. The client and root certificates are required. Do not move these files without reconfiguring their location in the Identity Manager administrator interface, as the certificates are not copied into the system configuration. Instead, the certificates are accessed when needed.

Certificates must be in the following formats:

Application Server Type	Format
Java 1.4 with JSSE	JKS or PCKS12
WebLogic 8	PEM

Usage Notes

This section lists dependencies and limitations related to using the ActivCard resource adapter.

- The ActivCard adapter accomplishes provisioning by using the ActivCard AIMS-Enterprise SDK, which communicates with a secure web server to send and retrieve information. A certificate with associated operator privileges in ActivCard is used to access the server, and only one connection per certificate is allowed at a time. If the same certificate is used to access the ActivCard operator interface, the adapter will be unable to communicate with the server during that time. It is recommended to have a different certificate for each Identity Manager server accessing ActivCard.

- If the `cryptix-jce-api.jar` and `cryptix-jce-provide.jar` files are present `%WSHOME%/WEB-INF/lib` directory, there may be a problem using the certificate. The test connection might fail with a message to check the paths and passphrase. In this situation, stop the application server, delete these JAR files, and restart the application server.
- Be sure the port number is correct when configuring the resource adapter. If you specify an incorrect port, the test connection will take a long time to fail.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses HTTPS to communicate with ActivCard.

Required Administrative Privileges

Administrators must have operator-level access within ActivCard.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	No
Pass-through authentication	No
Before/after actions	No
Data loading methods	• Import directly from resource • Reconcile with resource

Account Attributes

The following attributes are displayed on the Account Attributes page for the ActivCard resource adapters. All attributes are of type String.

Any attribute present in the objectclass specified for the ActivCard adapter to use can also be added. The attribute value is returned from the directory used by ActivCard. ActivCard uses an attribute (configurable within ActivCard) to store the device information, so care must be taken to not overwrite this information by exposing the attribute to update by Identity Manager.

Identity Manager User Attribute	Resource User Attribute	Description
accountId	userID	Required. The user's login ID.
lastname	sn	The user's last name (surname).
firstname	givenname	The user's first (given) name.
fullname	cn	Required. The user's full name.
email	mail	Required. The user's full name.
device ID	device ID	The serial number on the smart card.
device type	device type	Currently, OP_2.0 is the only supported value.

Resource Object Management

Not applicable

Identity Template

```
$accountId$
```

Sample Forms

```
ActivCardUserForm.xml
ActivCardUserViewForm.xml
```

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

```
com.waveset.adapter.ActivCardResourceAdapter
```

Additionally, you can set the following Identity Manager Active Sync logging parameters for the resource instance:

- Maximum Log Archives
- Maximum Active Log Age
- Log File Path
- Maximum Log File Size
- Log Level

Active Directory

The following resource adapters support Windows Active Directory 2000 SP3 and later and Windows Active Directory 2003:

GUI Name	Class Name
Windows 2000 / Active Directory	`com.waveset.adapter.ADSIResourceAdapter`
Windows 2000 / Active Directory Active Sync	`com.waveset.adapter.ActiveDirectoryActiveSyncAdapter`

The Windows 2000 / Active Directory resource adapter is defined in the `com.waveset.adapter.ADSIResourceAdapter` class.

This adapter supports the following versions:

- Windows Active Directory 2000 SP4
- Windows Active Directory 2003

Note The Windows 2000 / Active Directory Active Sync adapter (`com.waveset.adapter.ActiveDirectoryActiveSyncAdapter` has been deprecated as of Identity Manager 5.0 SP1. All features in this adapter are now in the Windows 2000/ Active Directory adapter. Although existing instances of the Active Sync adapter will still function, new instances of these can no longer be created.

Resource Configuration Notes

This section provides instructions for configuring the following Active Directory resources for use with Identity Manager, including the following:

- *Sun Identity Manager Gateway Location*
- *Sun Identity Manager Gateway Service Account*
- *Out of Office Messages*

Sun Identity Manager Gateway Location

The Gateway system should be running Windows 2000 or later. Although it might be possible to manage Active Directory (AD) from a Gateway system running Windows NT with the Active Directory Client Extension installed, this is not recommended.

Unless the LDAP Hostname resource attribute is set, the Gateway will perform a serverless bind to the directory. In order for the serverless bind to work, the Gateway needs to be installed on a system that is in a domain and that “knows” about the domain/directory to be managed. Generally, if the Gateway is in a domain that is in the same forest as the domain to be managed, or there is a trust relationship between the domains, then the serverless bind will succeed.

The LDAP Hostname resource attribute tells the Gateway to bind to a particular DNS hostname or IP address. This is the opposite of a serverless bind. However, the LDAP Hostname does not necessarily have to specify a specific domain controller. The DNS name of an AD domain can be used. If the Gateway system's DNS server is configured to return multiple IP addresses for that DNS name, then one of them will be used for the directory bind. This avoids having to rely on a single domain controller.

Sun Identity Manager Gateway Service Account

By default, the Gateway service runs as the local System account. This is configurable through the Services MMC Snap-in.

If you run the Gateway as an account other than Local System, then Gateway service account requires the “Act As Operating System” and “Bypass Traverse Checking” user rights. It uses these rights for pass-through authentication and for changing and resetting passwords in certain situations.

Most of the management of AD is done using the administrative account specified in the resource. However, some operations are done as the Gateway service account. This means that the Gateway service account must have the appropriate permissions to perform these operations. Currently, these operations are:

- Creating home directories
- Running actions (including before and after actions)

The Authentication Timeout resource attribute (provided for pass-through authentication only) prevents the adapter from hanging if a problem occurs on the Gateway side.

Out of Office Messages

The outOfOfficeEnabled and outofOfficeMessage account attributes can be used to enable the out of office autoreply function and set the out-of-office message, respectively. These can be used for Exchange 200x accounts. These attributes are only set on account updates and not account creates.

The adapter requires that the Messaging Application Programming Interface (MAPI) be installed on the gateway machine. There are at least two ways to install the MAPI subsystem. The simplest way is to install the Microsoft Outlook client on the gateway machine. No other configuration is necessary.

Another way is to install the Exchange System Management Tools, which are located on the Exchange Server CD. The management tools are installed as a component of the normal Exchange Server install. However, this installs the MAPI subsystem files, but it does not complete the configuration.

The `mapisvc.inf` file (typically located in `c:\winnt\system32`) contains the available MAPI services, and it must be updated to include the Exchange message service entries. The `msems.inf` file, which is contained in the gateway zip file, contains the entries that need to be merged into the `mapisvc.inf` file to configure the Exchange message server. The `msems.inf` file can be merged into the `mapisvc.inf file` manually using a text file editor such as notepad. Alternatively, a tool named `MergeIni.exe` is available on the Microsoft Platform SDK and can be found in the Windows Core SDK in the `Microsoft SDK\Bin` directory.

Use the following command to run `MergeIni`:

```
MergeIni msems.inf -m
```

Identity Manager Installation Notes

No additional installation procedures are required on this resource.

Usage Notes

This section lists dependencies and limitations related to using the Active Directory resource adapter, including:

- *Checking Password History*
- *Supporting Microsoft Exchange Servers*
- *Configuring Active Sync*

Checking Password History

To check the password history for an Active Directory account when an end-user changes his or her password, the user must provide an AD password. This functionality is enabled on an AD resource by setting the `User Provides Password On Change` resource attribute to `1` and adding the `WS_USER_PASSWORD` attribute to the account attributes with type `encrypted`. `WS_USER_PASSWORD` must be added as a Identity Manager User Attribute and as a Resource User Attribute.

The `sources.`*`ResourceName`*`.hosts` property in the `waveset.properties` file can be used to control which host or hosts in a cluster will be used to execute the synchronization portion of a resource adapter using Active Sync. *`ResourceName`* must be replaced with the name of the Resource object.

Supporting Microsoft Exchange Servers

To support Microsoft Exchange Server 2000 and later, the following account attributes must be enabled:

- homeMDB
- homeMTA
- mailNickname
- msExchHomeServerName

The following account attributes are displayed in the schema map by default and are also used for managing Exchange accounts:

- garbageCollPeriod
- mDBOverHardQuotaLimit
- mDBOverQuotaLimit
- mDBStorageQuota
- mDBUseDefaults

If your Active Directory resource is not being used to manage Exchange Server attributes, then you must remove these attributes from the schema map for these adapters to successfully provision Active Directory accounts with Identity Manager.

The Active Directory adapter can be modified to support printer, computer, or other Active Directory objects. The following example illustrates how to modify the XML code in the appropriate Java class to support printer objects.

```
<ObjectType name='Printer' icon='group'>
   <ObjectClasses operator='AND'>
      <ObjectClass name='printQueue'/>
   </ObjectClasses>
   <ObjectFeatures>
      <ObjectFeature name='create'/>
      <ObjectFeature name='update'/>
      <ObjectFeature name='delete'/>
   </ObjectFeatures>
   <ObjectAttributes idAttr='distinguishedName' displayNameAttr='cn'
descriptionAttr='description'>
      <ObjectAttribute name='cn' type='string'/>
      <ObjectAttribute name='description' type='string'/>
      <ObjectAttribute name='managedby' type='string'/>
      <ObjectAttribute name='distinguishedName' type='string'/>
   </ObjectAttributes>
</ObjectType>
```

In addition, you must create at least one new form to support printer objects.

The Windows Active Directory resource can manage Exchange 2000 contacts by changing the object class to `contact` and removing the `password`, `accountId`, and `expirePassword` resource attributes.

Configuring Active Sync

Before Identity Manager 5.5, if the **`Process deletes as updates`** check box was selected, Identity Manager would disable a deleted Identity Manager user as well as all resource accounts and mark the user for later deletion. By default, this check box was selected. In Identity Manager 5.5 and beyond, this functionality is configured by setting the Delete Rule set to None.

If the checkbox was previously deselected, then the Delete Rule will be set to **`ActiveSync has isDeleted set`**.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

The recommended approach for connecting to an Active Directory resource is with the Gateway service. The Gateway service uses ADSI and a TCP/IP socket connection (3 DES) for exchanging password information on the network.

You can also use LDAP over SSL or TCP/IP to connect to the Active Directory server. In this scenario, use the LDAP resource adapter.

Required Administrative Privileges

This section describes Active Directory permission and reset password permission requirements.

Active Directory Permissions

The administrative account configured in the Active Directory resource must have the appropriate permissions in Active Directory.

Identity Manager Functionality	Active Directory Permissions
Create Active Directory User accounts	Create User Objects To create the account enabled, you must have the ability to Read/Write the userAccountControl property. To create with the password expired, you must be able to Read/Write the Account Restrictions property set (includes the userAccountControl property).
Delete Active Directory User accounts	Delete User Objects
Update Active Directory User accounts	• Read All Properties • Write All Properties **Note**: If only a subset of the properties are to managed from Identity Manager, then Read/Write access can be given to just those properties.

Identity Manager Functionality	Active Directory Permissions
Change/Reset AD User account passwords Unlock AD User accounts Expire AD User accounts	User Object permissions: • List Contents • Read All Properties • Read Permissions • Change Password • Reset Password User Property permissions: • Read/Write lockoutTime Property • Read/Write Account Restrictions Property set • Read accountExpires Property To set permissions for the lockoutTime property, you should use the cacls.exe program available in the Windows 2000 Server resource kit.

Reset Password

The permissions to perform Create, Delete, and Update of resource objects are as expected. The account needs the Create and Delete permissions for the corresponding object type and you need appropriate Read/Write permissions on the properties that need to be updated.

Pass-Thru Authentication

To support Active Directory (AD) pass-thru authentication:

- When configuring the Gateway to run as a user, that user account must have the “Act As Operating System” and “Bypass Traverse Checking” user rights. By default, the Gateway runs as the Local System account, which should already have these rights. Also, the “Bypass Traverse Checking” user right is enabled for all users by default.

Note If you must update user rights, there might be a delay before the updated security policy is propagated. Once the policy has been propagated, you must restart the Gateway.

- Accounts being authenticated must have “Access This Computer From The Network” user rights on the Gateway system.

The Gateway uses the `LogonUser` function with the `LOGON32_LOGON_NETWORK` log-on type and the `LOGON32_PROVIDER_DEFAULT` log-on provider to perform pass-thru authentication. The `LogonUser` function is provided with the Microsoft Platform Software Development Kit.

Accessing Deleted Objects

The administrative account must have access to the Deleted Objects container in the active directory. By default, only Administrators and the System account have access to this container. Other users can be granted access to this container. For information on granting access to the Deleted Objects container, see Microsoft Knowledge Base article 892806.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	Yes
Pass-through authentication	Yes **Note**: The Authentication Timeout resource attribute (provided for pass-through authentication only) prevents the Active Directory adapter from hanging if a problem occurs on the Gateway side.
Before/after actions	Yes. The Active Directory resource supports before and after actions, which use batch scripts to perform activities on the Active Directory gateway system during a user create, update, or delete request. See *Chapter 3, "Adding Actions to Resources,"* for more information.
Data loading methods	• Import directly from resource • Reconcile with resource • Active Sync

Account Attributes

The syntax (or type) of an attribute usually determines whether the attribute is supported. In general, Identity Manager supports boolean, string, and integer syntaxes. Binary strings and similar syntaxes are not supported.

Attribute Syntax Support

This section provides information about supported and unsupported account syntaxes.

Supported Syntaxes

The following table lists the Active Directory syntax supported by Identity Manager:

AD Syntax	Identity Manager Syntax	Syntax ID	OM ID	ADS Type
Boolean	Boolean	2.5.5.8	1	ADSTYPE_BOOLEAN
Enumeration	String	2.5.5.9	10	ADSTYPE_INTEGER
Integer	Int	2.5.5.9	2	ADSTYPE_INTEGER
DN String	String	2.5.5.1	127	ADSTYPE_DN_STRING
Presentation Address	String	2.5.5.13	127	ADSTYPE_CASE_IGNORE_STRING
IA5 String	String	2.5.5.5	22	ADSTYPE_PRINTABLE_STRING
Printable String	String	2.5.5.5	19	ADSTYPE_PRINTABLE_STRING
Numeric String	String	2.5.5.6	18	ADSTYPE_NUMERIC_STRING
OID String	String	2.5.5.2	6	ADSTYPE_CASE_IGNORE_STRING
Case Ignore String (teletex)	String	2.5.5.4	20	ADSTYPE_CASE_IGNORE_STRING
Unicode String	String	2.5.5.12	64	ADSTYPE_OCTET_STRING
Interval	String	2.5.5.16	65	ADSTYPE_LARGE_INTEGER
LargeInteger	String	2.5.5.16	65	ADSTYPE_LARGE_INTEGER

Unsupported Syntaxes

The following table lists the Active Directory syntaxes that are not supported by Identity Manager:

Syntax	Syntax ID	OM ID	ADS Type
DN with Unicode string	2.5.5.14	127	ADSTYPE_DN_WITH_STRING
DN with binary	2.5.5.7	127	ADSTYPE_DN_WITH_BINARY
OR-Name	2.5.5.7	127	ADSTYPE_DN_WITH_BINARY
Replica Link	2.5.5.10	127	ADSTYPE_OCTET_STRING
NT Security Descriptor	2.5.5.15	66	ADSTYPE_NT_SECURITY_DESCRIPTOR
Octet String	2.5.5.10	4	ADSTYPE_OCTET_STRING
SID String	2.5.5.17	4	ADSTYPE_OCTET_STRING
UTC Time String	2.5.5.11	23	ADSTYPE_UTC_TIME
Object(Access-Point)	2.5.5.14	127	n/a

Account Attribute Support

This section provides information about the Active Directory account attributes that are supported and those not supported by Identity Manager.

Supported Account Attributes

The following table lists the account attributes supported by Identity Manager:

Schema Name	Attribute Type	Description
accountExpires	String	The date when the user's account expires.
AccountLocked	Boolean	Whether or not an account is locked out. Cannot be set to true; only the Windows system can set to true.
accountNameHistory	String	The length of time that the account has been active. Read-only
aCSPolicyName	String	String name of an ACS policy that applies to this user.
adminCount	String	Indicates that a given object has had its ACLs changed to a more secure value by the system because it was a member of one of the administrative groups (directly or transitively).Set by system. Read-only.
adminDescription	String	The description displayed on admin screens.
adminDisplayName	String	The name to be displayed on admin screens.
altSecurityIdentities	String	Contains mappings for X.509 certificates or external Kerberos user accounts to this user for the purpose of authentication.
assistant	String	The distinguished name of a user's administrative assistant.
badPasswordTime	String	The last time the user tried to log onto the account using an incorrect password.
badPwdCnt	String	Read-only. Number of login attempts with incorrect password. The value may only be for those logins that failed at the domain controller that is being queried.
businessCategory	String	Describes the kind of business performed by an organization.
c	String	The two-character country code in the address of the user.

Schema Name	Attribute Type	Description
cn	String	Common Name. This attribute is set from the CN value in the DN. Read-only.
co	String	Text-Country (country name)
company	String	The user's company name.
codePage	Int	Specifies the code page for the user's language of choice.
countryCode	String	Specifies the country code for the user's language of choice.
defaultClassStore	String	The default Class Store for a given user.
department	String	Contains the name for the department in which the user works.
description	String	Contains the description to display for an object. This value is treated as single-valued by the system.
desktopProfile	String	The location of the desktop profile for a user or group of users.
destinationIndicator	String	Not used by Active Directory.
displayName	String	The name displayed in the address book for a particular user. This is usually the combination of the user’s first name, middle initial, and last name.
displayNamePrintable	String	Printable version of the displayName.
distinguishedName	String	Cannot be set directly. Read only. Set the DN on create using the DN template or the accountId account attribute.
division	String	The user's division.
dynamicLDAPServer	String	DNS name of server handing dynamic properties for this account.
employeeID	String	The ID of an employee.
extensionName	String	The name of a property page used to extend the UI of a directory object.
facsimileTelephoneNumber	String	Contains telephone number of the user's business fax machine.
flags	Int	To be used by the object to store bit information.

Schema Name	Attribute Type	Description
garbageCollPeriod	Int	This attribute is located on the CN=Directory Service,CN=Windows NT,CN=Services,CN=Configuration,... object. It represents the period in hours between DS garbage collection runs.
generationQualifier	String	Indicates a person's generation. For example, Jr. or II.
givenName	String	Contains the given name (first name) of the user.
groupPriority	String	Not used
groups	String	Windows security and distribution groups
groupsToIgnore	String	Not used
homeDirectory	String	The user's home directory. If homeDrive is set and specifies a drive letter, homeDirectory should be a UNC path. The path must be a network UNC path of the form \\server\share\directory. This value can be a null string. The user's home directory will be created if: 1. The value is a UNC path that is not a share name (it specifies a directory on a share) 2. Any and all parent directories exist 3. The Create Home Directory resource attribute is set to 1 4. The user that the gateway service is running as must have permission to create the directory The user will be given Full Control of the created directory.
homeDrive	String	The drive letter (including the colon) that the home directory should be mapped to (for example, "Z:"). It should only be specified if homeDirectory is a UNC path.
homeMDB	String	The distinguished name of the message database (MDB) for this mailbox. It has a format similar to CN=Mailbox Store (*SERVERNAME*),CN=First Storage Group, CN=InformationStore,CN=*SERVERNAME*,CN=Servers, CN=First Administrative Group,CN=Administrative Groups, CN=*EXCHANGE ORG*, CN=Microsoft Exchange,CN=Services,CN=Configuration,DC=*DOMAIN*, DC=*YOURCOMPANY*,DC=com'

Schema Name	Attribute Type	Description
homeMTA	String	Points to the message transfer agent (MTA) that services this object. It has a format similar to CN=Microsoft MTA,CN=*SERVERNAME*,CN=Servers, CN=First Administrative Group,CN=Administrative Groups, CN=*EXCHANGE ORG*, CN=Microsoft Exchange,CN=Services,CN=Configuration,DC=*DOMAIN*, DC=*YOURCOMPANY*,DC=com
homePhone	String	The user's main home phone number.
homePostalAddress	String	A user's home address.
info	String	The user's comments. This string can be a null string.
initials	String	Contains the initials for parts of the user's full name.
internationalISDNNumber	String	Specifies an International ISDN number associated with an object.
ipPhone	String	The TCP/IP address for the phone. Used by Telephony.
l	String	Contains the locality, such as the town or city, in the user's address.
lastLogon	String	The last time the user logged on at a DC.
lastLogonTimestamp	String	The time that the user last logged into the domain. This value is only updated when the user logs in if a week has passed since the last update.
lastLogoff	String	The last time the user logged off.
legacyExchangeDN	String	The distinguished name previously used by Exchange.
localeID	Int	This attribute contains a list of locale IDs supported by this application. A locale ID represents a geographic location like France.
lockoutTime	String	The number of minutes to wait before resetting the invalid logon count.
logonCount	Int	The number of successful times the user tried to log on to this account. This property is maintained separately on each domain controller in the domain.
mail	String	One or more email addresses.
mailNickName	String	Exchange nickname.

Schema Name	Attribute Type	Description
managedObjects	String	Contains the list of objects that are managed by the user.Set by the system. Read only.
manager	String	Directory name of the user's manager.
maxStorage	String	The maximum amount of disk space the user can use.
mDBOverHardQuotaLimit	String	The maximum mailbox size, in KB, over which sending and receiving mail is disabled.
mDBOverQuotaLimit	String	The mailbox quota overdraft limit, in KB.
mDBStorageQuota	String	The message database quota, in KB.
mDBUseDefaults	String	Indicates whether the store should use the default quota, rather than the per-mailbox quota.
mhsORAddress	String	X.400 address.
middleName	String	The user’s middle name.
mobile	String	The primary cell phone number.
msCOM-PartitionSetLink	String	A link used to associate a COM+ Partition with a COM+ PartitionSet object. Read only.
msCOM-UserLink	String	A link used to associate a COM+ PartitionSet with a User object. Read only.
msCOM-UserPartitionSetLink	String	A link used to associate a User with a COM+ PartitionSet. Read only.
msDS-AllowedToDelegateTo	String	Contains a list of Service Principal Names (SPN). This attribute is used to configure a service to be able to obtain service tickets usable for Constrained Delegation.
ms-DS-Approx-Immed-Subordinates	Int	The approximate number of subordinates for this user. Read only.
msDS-Cached-Membership-Time-Stamp	String	Used by the Security Accounts Manager for group expansion during token evaluation. Read only.
mS-DS-ConsistencyChildCount	Int	This attribute is used to check consistency between the directory and another object, database, or application, by comparing a count of child objects.
msExchHomeServerName	String	The name of the Exchange server. It has a format similar to /o=*EXCHANGEORG*/ou=First Administrative Group/cn=Configuration/cn=Servers/cn=*SERVERNAME*

Schema Name	Attribute Type	Description
ms-DS-KeyVersionNumber	Int	The Kerberos version number of the current key for this account. This is a constructed attribute. Read only.
ms-DS-Mastered-By	String	Back link for msDS-hasMasterNCs. Read only.
ms-DS-Members-For-Az-Role-BL	String	Back-link from member application group or user to Az-Role object(s) linking to it. Read only.
ms-DS-NC-Repl-Cursors	String	A list of past and present replication partners, and how up to date we are with each of them. Read only.
ms-DS-NC-Repl-Inbound-Neighbors	String	Replication partners for this partition. This server obtains replication data from these other servers, which act as sources. Read only.
ms-DS-NC-Repl-Outbound-Neighbors	String	Replication partners for this partition. This server sends replication data to these other servers, which act as destinations. This server will notify these other servers when new data is available. Read only.
ms-DS-Non-Members-BL	String	Back link from non-member group/user to Az group(s) linking to it. Read only.
ms-DS-Operations-For-Az-Role-BL	String	Back-link from Az-Operation to Az-Role object(s) linking to it. Read only.
ms-DS-Operations-For-Az-Task-BL	String	Back-link from Az-Operation to Az-Task object(s) linking to it. Read only.
ms-DS-Repl-Attribute-Meta-Data	String	A list of metadata for each replicated attribute. Read only.
ms-DS-Repl-Value-Meta-Data	String	A list of metadata for each value of an attribute. Read only.
ms-DS-Tasks-For-Az-Role-BL	String	Back-link from Az-Task to Az-Role object(s) linking to it. Read only.
ms-DS-Tasks-For-Az-Task-BL	String	Back-link from Az-Task to the Az-Task object(s) linking to it. Read only.
ms-DS-User-Account-Control-Computed	Int	A computed attribute to expose user password expired and user account locked out.
msExchMailboxSecurityDescriptor	String	This attribute determines Exchange Mailbox rights for the user. For more information, see “Managing ACL Lists” on page 65
ms-Exch-Owner-BL	String	The back-link to the owner attribute. Contains a list of owners for an object. Read only.

Schema Name	Attribute Type	Description
ms-IIS-FTP-Dir	String	The user home directory relative to the file server share. It is used in conjunction with ms-IID-FTP-Root to determine the FTP user home directory.
ms-IIS-FTP-Root	String	This attribute determines the file server share. It is used in conjunction with ms-IID-FTP-Dir to determine the FTP user home directory.
name	String	The Relative Distinguished Name (RDN) of the user. Cannot be set directly. Read only. Set the RDN on create using the DN template or the accountId account attribute. Do not use “name” for the left-hand side of the schema map as it is a reserved attribute name.
networkAddress	String	The TCP/IP address for a network segment.
nTSecurityDescriptor	String	The NT security descriptor for the schema object. For more information, see “Managing ACL Lists” on page 65.
o	String	The name of the company or organization.
objectCategory	N/A	An object class name used to groups objects of this or derived classes. Set by the system. Read-only.
objectClass	N/A	The list of classes from which this class is derived. The value of this attribute should be set using the Object Class resource attribute. Read-only.
objectVersion	Int	A version number for the object.
operatorCount	Int	The number of operators on the computer.
otherFacsimileTelephoneNumber	String	A list of alternate facsimile numbers.
otherHomePhone	String	A list of alternate home phone numbers.
otherIpPhone	String	The list of alternate TCP/IP addresses for the phone. Used by Telephony.
otherLoginWorkstations	String	Non-NT or LAN Manager workstations from which a user can login.
otherMailbox	String	Contains other additional mail addresses in a form such as CCMAIL: JohnDoe.
otherMobile	String	Additional mobile phone numbers
otherPager	String	Additional pager numbers
otherTelephone	String	Additional telephone numbers

Schema Name	Attribute Type	Description
ou	String	Organizational unit
outOfOfficeEnabled	Boolean	Enables the out-of-office autoreply function
outOfOfficeMessage	String	The text of an out-of-office message.
pager	String	Pager number
personalTitle	String	User’s title
PasswordNeverExpires	Boolean	Indicates whether the user’s password will expire.
physicalDeliveryOfficeName	String	The office where deliveries are routed to.
postalAddress	String	The office location in the user's place of business.
postalCode	String	The postal or zip code for mail delivery.
postOfficeBox	String	The P.O. Box number for this object.
preferredDeliveryMethod	String	The X.500. preferred way to deliver to addressee
preferredOU	String	The Organizational Unit to show by default on user' s desktop.
primaryGroupID	Int	If the user is not already a member of the group, then the promaryGroupID must be set in 2 steps: add the user to the group then set the primaryGroupId.
primaryInternationalISDNNumber	String	The primary ISDN number.
primaryTelexNumber	String	The primary telex number.
profilePath	String	Specifies a path to the user's profile. This value can be a null string, a local absolute path, or a UNC path.
proxyAddresses	String	A proxy address is the address by which a Microsoft Exchange Server recipient object is recognized in a foreign mail system. Proxy addresses are required for all recipient objects such as custom recipients and distribution lists.
pwdLastSet	String	This attribute indicates the last time the user modified the password. This value is stored as a large integer that represents the number of seconds elapsed since 00:00:00, January 1, 1601 (FILETIME). If this value is set to zero and the user account has the password never expires property set to false, then the user must set the password at the next logon.
revision	Int	The revision level for a security descriptor or other change. Read only.

Schema Name	Attribute Type	Description
rid	Int	The relative Identifier of an object. Read only.
sAMAccountName	String	Login name.
sAMAccountType	Int	This attribute contains information about every account type object. Set by system. Read only.
scriptPath	String	The path for the user's logon script. The string can be null.
seeAlso	String	DNs of related objects
serialNumber	String	User's serial number. Not used by Active Directory.
servicePrincipalName	String	List of distinguished names that are related to an object.
showInAddressBook	String	This attribute is used to indicate which MAPI address books an object will appear in. It is normally maintained by the Exchange Recipient Update Service.
showInAdvancedViewOnly	Boolean	True if this attribute is to be visible in the Advanced mode of the UI.
sn	String	Family or last name
st	String	State or province name
street	String	Street address
Structural-Object-Class	String	Stores a list of classes contained in a class hierarchy, including abstract classes. Read only.
telephoneNumber	String	Primary telephone number.
Terminal Services Initial Program	String	The path of the initial program that runs when the user logs on.
Terminal Services Initial Program Directory	String	The path of working directory for the initial program
Terminal Services Inherit Initial Program	Boolean	Indicates whether the client can specify an initial program true - The client can specify program. false - The **Terminal Services Initial Program** value is used and client is logged off when exiting that program.
Terminal Services Allow Logon	Boolean	false - The user cannot logon. true - The user can logon.

Schema Name	Attribute Type	Description
Terminal Services Active Session Timeout	Integer	Duration in milliseconds. A value of 0 indicates the connection timer is disabled.
Terminal Services Disconnected Session Timeout	Integer	The maximum duration, in milliseconds, that a terminal server retains a disconnected session before the logon is terminated. A value of 0 indicates the disconnection timer is disabled.
Terminal Services Idle Timeout	Integer	The maximum idle time, in milliseconds. If there is no keyboard or mouse activity for the specified interval, the user's session is disconnected or terminated depending on the value specified in Terminal Services End Session On Timeout Or Broken Connection. A value of 0 indicates the idle timer is disabled.
Terminal Services Connect Client Drives At Logon	Boolean	Indicates whether the terminal server automatically reestablishes client drive mappings at logon. false - The server does not automatically connect to previously mapped client drives. true - The server automatically connects to previously mapped client drives at logon.
Terminal Services Connect Client Printers At Logon	Boolean	indicates whether the terminal server automatically reestablishes client printer mappings at logon. false - The server does not automatically connect to previously mapped client printers. true - The server automatically connects to previously mapped client printers at logon.
Terminal Services Default To Main Client Printer	Boolean	Indicates whether the client printer is the default printer. false - The client printer is not the default printer. true - The client printer is the default printer.
Terminal Services End Session On Timeout Or Broken Connection	Boolean	Specifies the action when the connection or idle timers expire, or when a connection is lost due to a connection error. false - The session is disconnected. true - The session is terminated.

Schema Name	Attribute Type	Description
Terminal Services Allow Reconnect From Originating Client Only	Boolean	Indicates how a disconnected session for this user can be reconnected. false - The user can log on to any client computer to reconnect to a disconnected session. true - The user can reconnect to a disconnected session by logging on to the client computer used to establish the disconnected session.
Terminal Services Callback Settings	Integer	Indicates the configuration for dialup connections in which the terminal server hangs up and then calls back the client to establish the connection. 0 - Callback connections are disabled. 1 - The server prompts the user to enter a phone number and calls the user back at that phone number. 2 - The server automatically calls the user back at the phone number specified by the Terminal Services Callback Phone Number attribute.
Terminal Services Callback Phone Number	String	The phone number to use for callback connections.
Terminal Services Remote Control Settings	Integer	Indicates whether the user session can be shadowed. Shadowing allows a user to remotely monitor the on-screen operations of another user. 0 - Disable 1 - Enable input, notify 2 - Enable input, no notify 3 - Enable no input, notify 4 - Enable no input, no notify
Terminal Services User Profile	String	The path of the user's profile for terminal server logon.
Terminal Services Local Home Directory	String	The path of the user's home directory for terminal server logon.

Schema Name	Attribute Type	Description
Terminal Services Home Directory Drive	String	A drive name (a drive letter followed by a colon) to which the UNC path specified in the Terminal Services Local Home Directory attribute is mapped.
textEncodedORAddress	String	Supports X.400 addresses in a text format.
title	String	Contains the user's job title. This property is commonly used to indicate the formal job title, such as Senior Programmer, rather than occupational class, such as programmer. It is not typically used for suffix titles such as Esq. or DDS.
userAccountControl	Int	Specifies flags that control password, lockout, disable/enable, script, and home directory behavior for the user. This property also contains a flag that indicates the account type of the object. The flags are defined in LMACCESS.H.
userParameters	String	Parameters of the user. Points to a Directory string that is set aside for use by applications. This string can be a null string, or it can have any number of characters before the terminating null character.
userPassword	Encrypted	The user's password in UTF-8 format. This is a write-only attribute.
userPrincipalName	String	An Internet-style login name for a user based on the Internet standard RFC 822. The UPN is shorter than the distinguished name and easier to remember. By convention, this should map to the user e-mail name.
userSharedFolder	String	Specifies a UNC path to the user's shared documents folder. The path must be a network UNC path of the form \\server\share\directory. This value can be a null string.
userSharedFolderOther	String	Specifies a UNC path to the user's additional shared documents folder. The path must be a network UNC path of the form \\server\share\directory. This value can be a null string.
userWorkstations	String	NetBIOS or DNS names of computers user can log into, separated by commas.
usnChanged	String	USN value assigned by the local directory for the latest change, including creation. Read only.
usnCreated	String	USN-Changed value assigned at object creation.
USNIntersite	Int	The USN for inter-site replication.

Schema Name	Attribute Type	Description
uSNLastObjRem	String	Indicates when the last object was removed from a server. Read only.
uSNSource	String	Value of the USN-Changed attribute of the object from the remote directory that replicated the change to the local server.Read only.
WS_PasswordExpired	Boolean	Indicates whether to expire the user's password.
WS_USER_PASSWORD	Encrypted	Contains the user password. See the Usage Notes for more information.
wbemPath	String	References to objects in other ADSI namespaces.
whenChanged	String	The date when this object was last changed. Read only.
whenCreated	String	The date when this object was created. Read only.
wWWHomePage	String	The user's primary web page.
url	String	A list of alternate web pages.
x121Address	String	The X.121 address for an object.

Managing ACL Lists

The `nTSecurityDescriptor` and the `msExchMailboxSecurityDescriptor` attribute values contain ACL lists that you must specify in a special way.

For example, the following shows a user form a company might use to assign a default set of permissions to each user they provision:

```
<Field name='attributes[AD].nTSecurityDescriptor' hidden='true'>
  <Expansion>
      <list>
        <s>Domain Admins|983551|0|0|NULL|NULL</s>
        <s>NT AUTHORITY\SYSTEM|983551|0|0|NULL|NULL</s>
        <s>Account Operators|983551|0|0|NULL|NULL</s>
        <s>NT AUTHORITY\Authenticated Users|131220|0|0|NULL|NULL</s>
        <s>NT AUTHORITY\Authenticated Users|256|5|0|
{AB721A55-1E2F-11D0-9819-00AA0040529B}|NULL</s>
        <s>NT AUTHORITY\SELF|131220|0|0|NULL|NULL</s>
      </list>
  </Expansion>
</Field>
```

Here is a description of the preceding format:

```
Trustee|Mask|aceType|aceFlags|objectType|InheritedObjectType
```

Where:

- **Trustee** is the `DOMAIN\Account` of the user.
- **Mask** is a flag specifying access permissions (read, write, etc.).
- **aceType** is a flag indicating the access-control entry (`ACE`) types.
- **aceFlags** is a flag specifying whether other containers or objects can inherit the `ACE` from the ACL owner.
- **objectType** is a flag indicating the ADSI object type. the `objectType` value is a GUID to a property or an object in string format.
  - The GUID refers to a property when you use `ADS_RIGHT_DS_READ_PROP` and `ADS_RIGHT_DS_WRITE_PROP` access masks.
  - The GUID specifies an object when you use `ADS_RIGHT_DS_CREATE_CHILD` and `ADS_RIGHT_DS_DELETE_CHILD` access masks.
- **InheritedObjectType** is a flag indicating the child object type of an ADSI object. The `InheritedObjectType` value is a GUID to an object in string format. When you set such a GUID, the `ACE` applies only to the object referred to by the GUID.

The following information (found in MSDN) is provided to help you further understand some of these fields:

- **aceType**:

  ```
  ADS_ACETYPE_ACCESS_ALLOWED = 0,
  ADS_ACETYPE_ACCESS_DENIED = 0x1,
  ADS_ACETYPE_SYSTEM_AUDIT = 0x2,
  ADS_ACETYPE_ACCESS_ALLOWED_OBJECT = 0x5,
  ADS_ACETYPE_ACCESS_DENIED_OBJECT = 0x6,
  ADS_ACETYPE_SYSTEM_AUDIT_OBJECT = 0x7,
  ADS_ACETYPE_SYSTEM_ALARM_OBJECT = 0x8
  ADS_ACETYPE_ACCESS_ALLOWED
  ```

  Where:

  - **ADS_ACETYPE_ACCESS_ALLOWED**: The `ACE` is of the standard `ACCESS ALLOWED` type, where the `ObjectType` and `InheritedObjectType` fields are `NULL`.
  - **ADS_ACETYPE_ACCESS_DENIED**: The `ACE` is of the standard system-audit type, where the `ObjectType` and `InheritedObjectType` fields are `NULL`.

- **ADS_ACETYPE_SYSTEM_AUDIT**: The ACE is of the standard system type, where the ObjectType and InheritedObjectType fields are NULL.
- **ADS_ACETYPE_ACCESS_ALLOWED_OBJECT**: On Windows 2000, ACE grants access to an object or a subobject of the object, such as a property set or property.

  ObjectType, InheritedObjectType, or both contain a GUID that identifies a property set, property, extended right, or type of child object.
- **ADS_ACETYPE_ACCESS_DENIED_OBJECT**: Windows 2000, ACE denies access to an object or a subobject of the object, such as a property set or property.

  ObjectType, InheritedObjectType, or both contain a GUID that identifies a property set, property, extended right, or type of child object.
- **ADS_ACETYPE_SYSTEM_AUDIT_OBJECT**: Windows 2000, ACE audits access to an object or a subobject of the object, such as a property set or property.

  ObjectType, InheritedObjectType, or both contain a GUID that identifies a property set, property, extended right, or type of child object.
- **ADS_ACETYPE_SYSTEM_ALARM_OBJECT**: Not used on Windows 2000/XP at this time.

- **aceFlags**

```
ADS_ACEFLAG_INHERIT_ACE = 0x2,
ADS_ACEFLAG_NO_PROPAGATE_INHERIT_ACE = 0x4,
ADS_ACEFLAG_INHERIT_ONLY_ACE = 0x8,
ADS_ACEFLAG_INHERITED_ACE = 0x10,
ADS_ACEFLAG_VALID_INHERIT_FLAGS = 0x1f,
ADS_ACEFLAG_SUCCESSFUL_ACCESS = 0x40,
```

  Where:

  - **ADS_ACEFLAG_FAILED_ACCESS = 0x80**
    ADS_ACEFLAG_INHERIT_ACE: Indicates child objects that will inherit this access-control entry (ACE).

    The inherited ACE is inheritable unless you set the ADS_ACEFLAG_NO_PROPAGATE_INHERIT_ACE flag.
  - **ADS_ACEFLAG_NO_PROPAGATE_INHERIT_ACE**: Causes the system to clear the ADS_ACEFLAG_INHERIT_ACE flag for the inherited ACEs of child objects, which prevents the ACE from being inherited by subsequent generations of objects.

- **ADS_ACEFLAG_INHERIT_ONLY_ACE**: Indicates an inherit-only ACE that does not exercise access control on the object to which it is attached.

  If you do not set this flag, the ACE is an effective ACE that exerts access control on the object to which it is attached.
- **ADS_ACEFLAG_INHERITED_ACE**: Indicates whether the ACE was inherited. The system sets this bit.
- **ADS_ACEFLAG_VALID_INHERIT_FLAGS**: Indicates whether the inherited flags are valid. The system sets this bit.
- **ADS_ACEFLAG_SUCCESSFUL_ACCESS**: Generates audit messages for successful access attempts, used with ACEs that audit the system in a system access-control list (SACL).
- **ADS_ACEFLAG_FAILED_ACCESS**: Generates audit messages for failed access attempts, used with ACEs that audit the system in a SACL.

- **objectType** and **InheritedObjectType**: Specifies the GUID of other objects in the form:

  {BF9679C0-0DE6-11D0-A285-00AA003049E2}

  The object/attribute GUID is wrapped in brackets { }. This format is returned during a fetch. Within ADSI there are GUIDs to represent specific attributes to grant access and also a way to describe an inherited relationship.

For more detailed information, reference the following website:

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dsportal/dsportal/directory_services_portal.asp

The best method in which to find the correct string to pass down, is to do the following:

1. Add the attribute to your schema, and then add the following field to your user form, as follows:

```
<Field name='accounts[AD].nTSecurityDescriptor'>
  <Display class='TextArea'>
    <Property name='title' value='NT User Security Descriptor'/>
    <Property name='rows' value='20'/>
    <Property name='columns' value='100'/>
  </Display>
</Field>
```

   or

```
<Field name='accounts[AD].msExchMailboxSecurityDescriptor'>
  <Display class='TextArea'>
    <Property name='title' value='Mailbox Security Descriptor'/>
    <Property name='rows' value='20'/>
    <Property name='columns' value='100'/>
  </Display>
</Field>
```

2. Edit a user's object in Active Directory and set the corresponding ACL lists for all users to establish a baseline.
3. Edit the user in Identity Manager and on the Edit user form.

   You should see a text area with the corresponding values, which have been pulled from the user object in Active Directory.

Using the preceding method will help you determine which values you must add to the form, for the settings you want.

Unsupported Attributes

The following table lists the account attributes that are not supported by Identity Manager:

Schema Name	Notes
allowedAttributes	Operational attribute
allowedAttributesEffective	Operational attribute
allowedChildClasses	Operational attribute
alowedChildClassesEffective	Operational attribute
bridgeheadServerListBL	System usage.
canonicalName	Operational attribute
controlAccessRights	String(Octet)
createTimeStamp	String(UTC-Time)
dBCSPwd	String(Octet)
directReports	System usage. Set using the manager attribute of the users that are managed by this user.
dSASignature	Object(Replica-Link)
dSCorePropagationData	String(UTC-Time)
fromEntry	Operational attribute
frsComputerReferenceBL	System usage
fRSMemberReferenceBL	System usage
fSMORoleOwner	System usage

Schema Name	Notes
groupMembershipSAM	String(Octet)
instanceType	System usage
isCriticalSystemObject	System usage
isDeleted	System usage
isPrivilegeHolder	System usage
lastKnownParent	System usage
lmPwdHistory	String(Octet)
logonHours	String(Octet)
logonWorkstations	String(Octet)
masteredBy	System usage.
memberOf	System usage. Use the “groups” attribute.
modifyTimeStamp	String(UTC-Time)
MS-DRM-Identity-Certificate	String(Octet)
ms-DS-Cached-Membership	String(Octet)
mS-DS-ConsistencyGuid	String(Octet)
mS-DS-CreatorSID	String(Sid)
ms-DS-Site-Affinity	String(Octet)
mSMQDigests	String(Octet)
mSMQDigestsMig	String(Octet)
mSMQSignCertificates	String(Octet)
mSMQSignCertificatesMig	String(Octet)
msNPAllowDialin	Use RAS MPR API to read and update values.
msNPCallingStation	Use RAS MPR API to read and update values.
msNPSavedCallingStationID	Use RAS MPR API to read and update values.
msRADIUSCallbackNumber	Use RAS MPR API to read and update values.
msRADIUSFramedIPAddress	Use RAS MPR API to read and update values.
msRADIUSFramedRoute	Use RAS MPR API to read and update values.

Schema Name	Notes
msRADIUSServiceType	Use RAS MPR API to read and update values.
msRASSavedCallbackNumber	Use RAS MPR API to read and update values.
msRASSavedFramedIPAddress	Use RAS MPR API to read and update values.
msRASSavedFramedRoute	Use RAS MPR API to read and update values.
netbootSCPBL	System usage
nonSecurityMemberBL	System usage
ntPwdHistory	System usage
objectGUID	String(Octet). The GUID is stored in the Identity Manager user object in the ResourceInfo for the account.
objectSid	String(Sid)
otherWellKnownObjects	Object(DN-Binary)
partialAttributeDeletionList	System usage
partialAttributeSet	System usage
possibleInferiors	System usage
proxiedObjectName	Object(DN-Binary)
queryPolicyBL	System usage
registeredAddress	String(Octet)
replPropertyMetaData	System usage
replUpToDateVector	System usage
repsFrom	System usage
repsTo	System usage
sDRightsEffective	Operational attribute
securityIdentifier	String(Sid)
serverReferenceBL	System usage
sIDHistory	String(Sid)
siteObjectBL	System usage

Schema Name	Notes
subRefs	System usage
subSchemaSubEntry	System usage
supplementalCredentials	System usage
systemFlags	System usage
telexNumber	String(Octet)
teletexTerminalIdentifier	String(Octet)
terminalServer	String(Octet)
thumbnailPhoto	String(Octet)
thumbnailLogo	String(Octet)
tokenGroups	String(Sid) / Operational attribute
tokenGroupsGlobalAndUniversal	String(Sid)
tokenGroupsNoGCAcceptable	String(Sid) / Operational attribute
unicodePwd	String(Octet). Use userPassword to set the user's password.
userCert	String(Octet)
userCertificate	String(Octet)
userSMIMECertificate	String(Octet)
wellKnownObjects	Object(DN-String)
x500uniqueIdentifier	String(Octet)

Resource Object Management

Identity Manager supports the following Active Directory objects:

Resource Object	Supported Features	Attributes Managed
Group	Create, update, delete	cn, samAccountName, description, managedby, member, mail, groupType, authOrig, name
DNS Domain	Find	dc
Organizational Unit	Create, delete, find	ou
Container	Create, delete, find	cn, description

The attributes that can be managed on resource objects are also generally dictated by the attribute syntaxes. The attributes for these object types are similar as those for user accounts and are supported accordingly.

Identity Template

Windows Active Directory is a hierarchically based resource. The identity template will provide the default location in the directory tree where the user will be created. The default identity template is

```
CN=$fullname$,CN=Users,DC=mydomain,DC=com
```

The default template must be replaced with a valid value.

Sample Forms

This section lists the sample forms provided for the Active Directory resource adapter.

Built-In

- ActiveDirectory ActiveSync Form
- Windows Active Directory Create Container Form
- Windows Active Directory Create Group Form
- Windows Active Directory Create Organizational Unit Form
- Windows Active Directory Create Person Form
- Windows Active Directory Create User Form
- Windows Active Directory Update Container Form
- Windows Active Directory Update Group Form

- Windows Active Directory Update Organizational Unit Form
- Windows Active Directory Update Person Form
- Windows Active Directory Update User Form

Also Available

`ADUserForm.xml`

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

`com.waveset.adapter.ADSIResourceAdapter`

In addition, tracing can be enabled on the Gateway service via the Identity Manager debug pages. (*`InstallDir`*`\idm\debug\Gateway.jsp`). This page allows you to specify the level of trace, location of the trace file, and the maximum size of the trace file. This page also allows you to remotely retrieve the gateway trace file and display the version information for the Gateway.

The Gateway service may also be started from the console with debug tracing via various command line switches. Use `-h` to review the usage for the Gateway service.

AIX

The AIX resource adapter is defined in the `com.waveset.adapter.AIXResourceAdapter` class.

This adapter supports the following versions of AIX:

- 4.3.3
- 5.2, 5L 5.3

Resource Configuration Notes

If you will be using SSH (Secure Shell) for communication between the resource and Identity Manager, set up SSH on the resource before configuring the adapter.

Identity Manager Installation Notes

No additional installation procedures are required on this resource.

Usage Notes

The AIX resource adapter primarily provides support for the following AIX commands:

- `mkuser, chuser, rmuser`
- `mkgroup, chgroup, rmgroup`
- `passwd, pwdadm`

Note For more information about supported attributes and files, refer to the AIX manual pages for these commands.

The Bourne-compliant shell (sh, ksh) must be used as the root shell when connecting to a UNIX resource (AIX, HP-UX, Solaris, or Linux).

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses the following connections to communicate with the AIX adapter:

- Telnet
- SSH (SSH must be installed independently on the resource.)

Required Administrative Privileges

Managing users and groups require that the administrator be root or a member of the security group.

The adapter supports logging in as a standard user, then performing a su command to switch to root (or root-equivalent account) to perform administrative activities. Direct logins as root user are also supported.

The adapter also supports the sudo facility (version 1.6.6 or later), which can be installed on AIX from the AIX Toolbox. The sudo facility allows a system administrator to give certain users (or groups of users) the ability to run some (or all) commands as root or another user.

In addition, if sudo is enabled for a resource, its settings will override those configured on the resource definition page for the root user and admin user.

If you are using sudo, you must set the `tty_tickets` parameter to true for the commands enabled for the Identity Manager administrator. Refer to the man page for the sudoers file for more information.

The administrator must be granted privileges to run the following commands with sudo:

User, Group, and Security Commands		NIS Commands	Miscellaneous Commands	
• chgroup • chgrpmem • chsec • chuser • lsgroup • lssec • lsuser • mkgroup • mkuser	• rmgroup • rmuser • passwd • pwdadm	• make • ypcat • ypmatch • yppasswd	• awk • cat • cd • chmod • chown • cp • cut • diff • echo	• grep • ls • mv • rm • sed • sleep • sort • tail • touch

In addition, the NOPASSWORD option must be specified for each command.

You can use a test connection to test whether

- These commands exist in the administrator user's path
- The administrative user can write to /tmp
- The administrative user have rights to run certain commands

Note A test connection can use different command options than a normal provision run.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	No
Pass-through authentication	Yes
Before/after actions	Yes
Data loading methods	• Import directly from resource • Reconcile with resource

You can define resource attributes to control the following tasks for all users on this resource:

- Create a home directory when creating the user
- Copy files to the user's home directory when creating the user
- Delete the home directory when deleting the user

Account Attributes

The following table lists the AIX user account attributes.

Notes:

- Attributes are optional unless noted in the description.
- All attributes are Strings.

Resource User Attribute	mkuser Equivalent	Description
accountId	login_name	Required. The user's login name.
account_locked	account_locked=[true | false]	Indicates if the user account is locked.
admin	admin=[true|false]	Defines the administrative status of the user.
daemon	daemon=[true|false]	Indicates whether the user can run programs using the cron or src daemon.
expires	expires=*MMDDhhmmyy*	The expiration date of the account.
gecos	gecos=*String*	General information about the user.
groups	groups=*GroupNames*	A comma-separated list of group names the user belongs to.
home	home=*PathName*	The full path to the user's home directory.
id	id=*Integer*	A unique integer string that specifies the user ID.
login	login=[true | false]	Indicates whether the user can log in to the system with the login command.
loginretries	loginretries=*attempts*	The number of unsuccessful login attempts allowed after the last successful login before the system locks the account.
maxage	maxage=*weeks*	The maximum age, in weeks, of a password.

Resource User Attribute	mkuser Equivalent	Description
maxexpired	maxexpired=*weeks*	The maximum time, in weeks, beyond the maxage value that a user can change an expired password.
pgrp	pgrp=*GroupName*	The user's primary group.
rlogin	rlogin=[true | false]	Permits access to the account from a remote location with the telnet or rlogin commands.
shell	shell=*PathName*	The program run for the user at session initiation.
su	su=[true | false]	Indicates whether another user can switch to the specified user account with the su command.
umask	umask=*Value*	Sets file permissions.

Resource Object Management

Identity Manager supports the following native AIX objects:

Resource Object	Features Supported	Attributes Managed
Group	Create, update, delete, save as	groupName, admin, users

Identity Template

$accountId$

Sample Forms

Built-In

- AIX Group Create Form
- AIX Group Update Form

Also Available

`AIXUserForm.xml`

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.AIXResourceAdapter`
- `com.waveset.adapter.ScriptedConnection`

ClearTrust

The ClearTrust resource adapter is defined in the `com.waveset.adapter.ClearTrustResourceAdapter` class.

This adapter supports the following version of ClearTrust:

- 5.0.1

Resource Configuration Notes

You must edit the ClearTrust `eserver.conf` file to configure SSL mode. Change the `cleartrust.eserver.api_port.use_ssl` setting.

Note For more information, refer to ClearTrust documentation.

Identity Manager Installation Notes

The ClearTrust resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. To add this resource to the Identity Manager resources list, you must add the following value in the Custom Resources section of the Configure Managed Resources page.

   ```
   com.waveset.adapter.ClearTrustResourceAdapter
   ```

2. Copy the `ct_admin_api.jar` file to the `WEB-INF\lib` directory.
3. If using SSL, copy the following files to the `WEB-INF\lib` directory.
   - `asn1.jar`
   - `certj.jar`
   - `jce1_2-do.jar`
   - `jcert.jar`
   - `jnet.jar`
   - `jsafe.jar`
   - `jsaveJCE.jar`
   - `jsse.jar`
   - `rsajsse.jar`
   - `sslj.jar`

Usage Notes

The ClearTrust API is split for users and administrators. (Users are not granted access to servers; administrators are users with administrative rights to the ClearTrust server.) Identity Manager does not create or manage ClearTrust administrative users.

There are three types of entitlements in ClearTrust: Application, Application Function and URL. Identity Manager supports Application Function only; other entitlements are ignored. We recommend that entitlements be assigned to groups and the groups assigned to the user (which is supported by the adapter).

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses JNDI over SSL to communicate with the ClearTrust adapter.

Required Administrative Privileges

None

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	
Pass-through authentication	Yes
Before/after actions	No
Data loading methods	• Reconciliation • Import from resource

Account Attributes

The following table provides information about ClearTrust account attributes.

Identity Manager User Attribute	Resource User Attribute	Description
accountId	accountName	Required. The unique account ID for this user.
isAdminLockout	isAdminLockout	Boolean.
externalDN	externalDN	The external domain name for this user.
email	emailAddress	The user's email address.
endDate	endDate	The end date for this user.
startDate	startDate	The start date for the user.
firstname	firstName	The user's first name.
lastname	lastName	The user's last name.
userGroup	userGroup	The groups assigned to the user.

Resource Object Management

None

Identity Template

```
$accountId$
```

Sample Forms

```
ClearTrustUserForm.xml
```

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

```
com.waveset.adapter.ClearTrustResourceAdapter
```

Database Table

The Database Table adapter is defined in the `com.waveset.adapter.DatabaseTableResourceAdapter` class.

This adapter supports any relational database that has a JDBC driver.

The Database Table resource adapter is designed to guide you through a series of steps to connect to and manage users that are located in a single custom database table. The adapter also supports Active Sync to poll for account changes.

Note This resource is not designed to manage the DBMS system accounts which are typically found in multiple tables. (The adapter does not support join operations.) For those resources, continue to use the Oracle, SQL Server, DB2, Sybase, and MySQL resources.

Resource Configuration Notes

None

Identity Manager Installation Notes

No additional installation procedures are required on this resource.

Usage Notes

This section provides configuration notes related to using the Database Table resource adapter, including:

- General configuration notes
- Active Sync configuration notes

General Configuration

Use the following steps to set up a new Database Table resource:

1. Specify the database access parameters. Include the database type, connection information, and the database name where the table to be managed is located.
2. All of the available tables for that database are displayed on the Database Tables page. Select the table where the resource accounts for this resource are stored.

3. Select the columns from the table that Identity Manager will manage. One of these columns will be designated as the Key and be used as the account name attribute for the users and one column will be designated as the Password and be used as the account password. Other columns can be selected as attributes to be managed.
4. The resource schema map page will list just those attributes that were selected to be managed. It will not list the Key and Password attributes. These attributes will be implicitly managed.
5. The Active Sync Configuration page allows you to optionally specify the Active Sync-related Database Table attributes. If you are not using the adapter as an Active Sync, you can skip these values. See the *Active Sync Configuration* section below for additional details.
6. Specify the identity template used for this resource. This is the Identity Manager attribute name that will be used for the Key attribute.
7. Specify the Identity Manager resource parameters for this resource. This includes information like the resource name, Active Sync scheduling and logging, and approvers for the resource.

Active Sync Configuration

Note The Active Sync adapter does not detect account deletions. As a result, you must reconcile to detect these deletions.

During its Active Sync poll, the Database Table adapter selects resource accounts (from the specified database table) for passing to the user form (or instead to the workflow if specified).

The **`Static Search Predicate`** parameter specifies the optional static predicate used to qualify the accounts to be returned from the database. (A predicate is an SQL expression that is evaluated.) The parameter must be expressed in the native SQL syntax.

The following example illustrates the use of this parameter:

```
syncState = 'P'
```

This example requires that a column named `syncState` exists and that `P` is a possible value. This value is combined with the **`Last Fetched Predicate`** parameter to form the complete qualifier.

The **`Last Fetched Conjunction`** parameter is the value AND or OR. It specifies the conjunction prepended to the Last Fetched Predicate.

The **Last Fetched Predicate** parameter specifies another optional predicate, but this predicate can contain one or more user attributes defined in Identity Manager. This feature allows you to construct a predicate in native SQL syntax that compares values returned in a previous poll to values returned in the current poll. For example, if the `lastMod` column contains a timestamp, then this value can be compared on each poll. Then, if the value is higher on the current poll than on the previous poll, return information about the database entry. The following expression illustrates this feature:

```
'lastMod' > "$(lastmod)"
```

The value specified between the parentheses must be an Identity Manager User Attribute defined on the schema map page. The `$(lastmod)` token will be replaced with the value returned on the previous poll. An example value might be `2004-06-20 6:23:00`.

Note The first time the adapter polls, the **`Last Fetched Filter`** is not applied, because there are no previously fetched values. The filter will be run in all subsequent polls.

The Database Table adapter concatenates the **`Static Search Predicate`**, **`Last Fetched Conjunction`**, and **Last Fetched Predicate** resource parameters and sends a search expression similar to the following:

```
syncState = 'P' AND lastMod > '2004-06-20 6:23:00'
```

The **ORDER BY** parameter allows you to provide a native SQL ORDER BY clause to force the poll to process the rows in the specified order. Do not include the words `ORDER BY` in the value. For example, if you specify a value of `lastMod`, the rows are sorted based on the `lastMod` column, in an ascending order.

The optional **`Process to run with changes`** parameter, if specified, identifies the Identity Manager workflow to launch with each qualified account returned from the database. The map of values passed to the workflow is keyed by the attributes on the left-hand side of the schema map. If this value is not specified, then the update will be performed via the standard Active Sync user form processing.

Security Notes

Refer to the adapter documentation for the database to determine the supported connections and required administrative privileges.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	No
Rename account	No
Pass-through authentication	No
Before/after actions	No
Data loading methods	• Import from resource • Active Sync • Reconciliation

Account Attributes

The Resource User Attributes are populated by the wizard during the creation or editing of the resource. The values of these columns for selected users are then mapped with their corresponding attribute names found in the Identity Manager User Attributes.

The `sources.`*`ResourceName`*`.hosts` property in the `waveset.properties` file can be used to control which host or hosts in a cluster will be used to execute the synchronization portion of an Active Sync adapter. *`ResourceName`* must be replaced with the name of the Resource object.

Resource Object Management

None

Identity Template

```
$accountId$
```

Sample Forms

None

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

```
com.waveset.adapter.DatabaseTableResourceAdapter
```

Additionally, you can set the following Identity Manager Active Sync logging parameters for the resource instance:

- Maximum Log Archives
- Maximum Active Log Age
- Maximum Log File Size
- Log File Path
- Log Level

DB2

The DB2 resource adapter is defined in the `com.waveset.adapter.DB2ResourceAdapter` class.

This adapter supports the following versions of IBM DB2 Universal Database for Linux, UNIX, and Windows:

- 7.0, 7.2
- 8.1, 8.2

Use this adapter to support user accounts for logging into DB2. If you have a custom DB2 table, see *Database Table* on page 1-84 for information about using the Resource Adapter Wizard to create a custom DB2 table resource.

Resource Configuration Notes

DB2 offers two types of JDBC access, each of which requires a different driver.

- **The application driver** (`COM.ibm.db2.jdbc.app.DB2Driver`) requires local client software and a local database instance.

  Because DB2 runs on a separate (often dedicated) host in most production environments, the local database instance usually contains an alias to the remote database instance. In this configuration, the local database instance uses a DB2-specific protocol to communicate with the remote database instance. This type of driver is the default on the DB2 Resource Parameters page.
- **The network driver** (`COM.ibm.db2.jdbc.net.DB2Driver`) does not require local client software or a local database.

  This driver does require that the DB2 Java Daemon (`db2jd`) be running on the target server. (In most production environments, the target server is a separate host, but the network driver works as well with a local database instance.)

  This daemon is not started by default, but the database administrator can start it manually or configure it to start automatically when the database instance starts.

Identity Manager Installation Notes

The DB2 resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. To add this resource to the Identity Manager resources list, you must add the following value in the Custom Resources section of the Configure Managed Resources page.

   ```
   com.waveset.adapter.DB2ResourceAdapter
   ```

2. Unzip the `Db2\java\db2java.zip` file.
3. Copy the `db2java.jar` file to the *`InstallDir`*`\idm\WEB-INF\lib` directory.

Usage Notes

DB2 performs authentication externally, and authorization internally. Authentication is performed through an accountID/password that is passed on to an external certifier. By default, the operating system performs the authentication, but other programs can be used for this purpose.

Authorization is done by mapping the accountID internally to various permissions at the database, index, package, schema, server, table, and/or table space level. Granting authorization does not automatically authenticate the accountID. (Thus, you can authorize nonexistent accounts.) Revoking authorization does not remove publicly available authority from an accountID.

In general, you should place the DB2 application in a resource group that also includes the machine upon which it is installed.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses JDBC over SSL to communicate with the DB2 adapter.

Required Administrative Privileges

The administrator must have SYSADM authority to grant DBADM authority. To grant other authorities, either DBADM or SYSADM authority is required.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	No
Rename account	No
Pass-through authentication	No
Before/after actions	No
Data loading methods	Import from resource

Account Attributes

The following table lists the DB2 user account attributes. All attributes are Strings.

Resource User Attribute	Description
accountId	Required.
grants	Required. Any comma-separated list of valid grants. For example: `CONNECT ON MySchema.MyTable,DELETE ON MySchema.MyTable,INSERT ON MySchema.MyTable,SELECT ON MySchema.MyTable,UPDATE ON MySchema.MyTable`

Resource Object Management

None

Identity Template

```
$accountId$
```

Sample Forms

None

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

`com.waveset.adapter.DB2ResourceAdapter`

Domino

The Domino resource adapter is defined in the `com.waveset.adapter.DominoResourceAdapter` class.

This adapter supports the following versions of Lotus Domino Server:

- 5.0
- 6.5

Note The Domino Active Sync adapter (`com.waveset.adapter.DominoActiveSyncAdapter` has been deprecated as of Identity Manager 5.0 SP1. All features in this adapter are now in the Domino Gateway adapter. Although existing instances of the Domino Active Sync adapter will still function, new instances of these can no longer be created.

Resource Configuration Notes

This section provides instructions for configuring Domino resources for use with Identity Manager, including:

- General instructions for setting up the Domino resource for use with Identity Manager
- Instructions for installing the Gateway to support Domino

General Configuration Instructions

Use these procedures to set up a Domino resource adapter:

1. Create the Identity Manager administrator in Domino. Use a certifier ID that has access to all organizations needed to manage users.
2. Add the user to the access control list (ACL) of the address book for the server, `names.nsf`.
   a. Give the user Editor access.
   b. Assign the user the following roles:
      - GroupModifier
      - UserCreator
      - UserModifier
3. Add the user to the ACL of the registration log, `certlog.nsf`, with Depositor access.

4. Add the user to the ACL of the Administration Requests, `admin4.nsf`, with Depositor access.
5. Add the newly-created user to server security:
   a. Open the **Security** panel to edit the server configuration.
   b. If access to the Domino server is restricted, make sure the Identity Manager proxy account has access to the server. This is done by specifying the account name or a group to which the proxy account belongs in the **`Access Server`** field.
   c. If there is a before or after action that calls a Domino agent, the user might need to be added to the **`Run unrestricted LotusScript/Java agents`** or **`Run restricted LotusScript/Java agent`** field, depending on how the agent being called is configured.

Installing the Gateway to Support Domino

For the gateway to talk with Domino there must be a Notes client already installed on the gateway machine

Add the following string values to `HKEY_LOCAL_MACHINE\SOFTWARE\Waveset\Lighthouse\Gateway` in the Windows registry to ensure Domino works properly:

`notesInstallDir` - This is the location where the client is installed and where the notes.dll file is location. Typically, the location is something like `C:\Lotus\Notes\`.

`notesIniFile` - The full path to the Lotus Notes initialization file, including the file name. We recommend that you copy the file from its default location (such as `C:\Lotus\Notes\notes.ini`) to the directory containing the Identity Manager gateway. Therefore, you should set the value of this registry key to a value similar to `C:\`*`GatewayDir`*`\notes.ini`.

Note Make sure the Notes client is running with a network-enabled profile. If you change the network connection after you copy the ini file, you must re-copy it or run the client through the command line, as in:

```
C:\Lotus\Notes\notes.exe =PathToIniFile
```

Identity Manager Installation Notes

No additional installation procedures are required on this resource.

Usage Notes

This section provides information related to using the Domino resource adapter, which is organized into the following sections:

- *Recertification Process*
- *Changing Passwords*
- *Disabling and Enabling*
- *ID File*
- *Rename/Move*
- *Resource Names*
- *Active Sync Configuration*

Recertification Process

The recertification process is done using the Boolean user attribute named “recertify.” During an update operation the attribute is checked; if enabled, the user ID is recertified.

The recertification process is done via the adminp process, meaning we generate an adminp request and the recertification of the id gets done at some point afterwards. The timing of the recertification will depend on configuration of the Domino server. In 5.0, if the address book entry gets recertified, then the next time the user logs into the system it will fix up the ID file with the new digest keys.

Changing Passwords

Administrative password changes and resets are not supported. Users must change their own passwords because the current password is required when changing.

The current password must be defined in the schema map as an account attribute named WS_USER_PASSWORD and needs to be of the encrypted type.

Disabling and Enabling

In the Domino database, there isn’t a native disable flag for each user, so each user disabled is placed in a DENY GROUP. When enabled, they are removed as members of any of the defined groups. DENY GROUP has a maximum number of members threshold so the group has to be specified as an account attribute to the resource. This requires an additional DenyGroups account attribute to be passed to the resource. DenyGroups can be set during a Disable, Enable, or Deprovision, but will not be fetched without additional coding.

When deprovisioning or disabling, you must send a list of DenyGroups that the user will be added to. When enabling, you must send a list of DenyGroups that the user will be removed from.

The available DenyGroups can be fetched from the resource with the following code:

```
<invoke name='listResourceObjects' class='com.waveset.ui.FormUtil'>
   <ref>:display.session</ref>
   <s>DenyLists</s>
   <s>YourResourceName</s>
   <null/>
   <s>false</s>
</invoke>
```

The currently assigned DenyGroups can be fetched on a disable, enable, or deprovision form with this code:

```
<invoke name='getList'>
   <invoke name='getView'>
      <ref>display.session</ref>
      <concat>
         <s>UserViewer:</s>
         <ref>resourceAccounts.id</ref>
      </concat>
      <map>
         <s>TargetResources</s>
         <list>
            <s>YourResourceName</s>
         </list>
      </map>
   </invoke>
   <s>accounts[YourResourceName].DenyGroups</s>
</invoke>
```

In the enable, disable, and deprovision forms, you must address the DenyGroups attribute as:

```
resourceAccounts.currentResourceAccounts
[YourResourceName].attributes.DenyGroups
```

The following example defines a field in the disable form that lists the available DenyGroups in the left hand side of a multi-select box:

```
<Field name='resourceAccounts.currentResourceAccounts
[YourResourceName].attributes.DenyGroups'>
   <Display class='MultiSelect'>
      <Property name='title' value='Deny Groups'/>
      <Property name='required'>
         <Boolean>false</Boolean>
      </Property>
```

```
      <Property name='allowedValues'>
         <invoke name='listResourceObjects'
class='com.waveset.ui.FormUtil'>
            <ref>:display.session</ref>
            <s>DenyLists</s>
            <s>YourResourceName</s>
            <null/>
            <s>false</s>
         </invoke>
      </Property>
      <Property name='availableTitle' value='Available Deny Groups'/>
      <Property name='selectedTitle' value='Assigned Deny Groups'/>
   </Display>
</Field>
```

The following example defines a field in the enable form that lists the assigned DenyGroups in a derivation rule of a hidden field:

```
<Field name='resourceAccounts.currentResourceAccounts
[YourResourceName].attributes.DenyGroups'>
   <Derivation>
      <invoke name='getList'>
         <invoke name='getView'>
            <ref>display.session</ref>
            <concat>
               <s>UserViewer:</s>
               <ref>resourceAccounts.id</ref>
            </concat>
            <map>
               <s>TargetResources</s>
               <list>
                  <s>YourResourceName</s>
               </list>
            </map>
         </invoke>
         <s>accounts[YourResourceName].DenyGroups</s>
      </invoke>
   </Derivation>
</Field>
```

ID File

The gateway machine generates new IDs for users that are newly registered. They may be placed on a UNC path that is accessible to the gateway process/service. So, specifying `\\machine\ids\myidfile.id` would put it on the network share.

There might be a need for the gateway to run as a user when configured as a service to get access to the share specified when a user is created. You can assign SYSTEM to have access to shares, but it depends on how the gateway network environment looks.

You can specify that the ID file be stored in the address book also by setting the Store ID In Addr Book resource attribute to TRUE/FALSE.

Rename/Move

The move/rename actions are also preformed by the `adminp` process. A move can be initiated from the rename form by changing the `certifierOrgHierarchy` attribute and providing the original `certifierId` file and password for that `id` file. The move request will create a “Name Move Request” in the requests database and must be completed by the new certifier that represents the user's new organization. A move can be initiated by changing the user's first/last name.

Note You cannot perform a rename and a move at the same time; the `adminp` process will not allow this since the request references the canonical name which will be changed in both cases.

Resource Names

The gateway requires that all Domino resources be named uniquely. If you have multiple Identity Manager deployments and they “point” to the same gateway, all of the Domino resources that exist on the deployments must have unique resource names.

Active Sync Configuration

Before Identity Manager 5.5, if the Active Sync **`Process deletes as updates`** check box was selected, Identity Manager would disable a deleted Identity Manager user as well as all resource accounts and mark the user for later deletion. By default, this check box was selected. In Identity Manager 5.5 and beyond, this functionality is configured by setting the Delete Rule set to None.

If the checkbox was previously deselected, then the Delete Rule will be set to **`ActiveSync has isDeleted set`**.

Additional Information

This section provides some additional, useful information related to this adapter, including:

- *ListAllObjects*
- *Form Updates*
- *searchFilter*
- *Other Form Issues*
- *Attributes Configured to be Passed Into Views*
- *Actions*

ListAllObjects

You can list any object specified in Domino. Pass in the view name as the “type” to the `listAllObjects` call.

Form Updates

Since some of these operations require additional attributes, default forms must be updated to include these attributes.

The resource definition already defines the attributes that should be passed to the various views.

- Enable, Disable — DenyGroups
- Deprovision — DenyGroups (optional)
- Expired Login, Change Password, Change My Password Form — HTTPPassword (must be secret), id file
- Rename — certifierIDFile, credentials (must be secret)

searchFilter

The following sample UserForm illustrates how the searchFilter option for the getResourceObjects method can be implemented for Domino. This form finds all users with the last name Smith on the resource MyResource.

```
<Form name='Domino searchFilter Form' objectType=UserForm'>
   <Display class='EditForm'/>
   <Field name='rcwfield'>
      <Display class='MultiSelect'>
         <Property name='title' value='My Lister'/>
         <Property name='availableTitle' value='Listing available
items'/>
         <Property name='selectedTitle' value='Selected Item(s)'/>
         <Property name='allowedValues'>
            <block trace='true'>
               <invoke name='getResourceObjects'
class='com.waveset.ui.FormUtil'>
                  <ref>:display.session</ref>
                  <s>People</s>
                  <s>MyResource</s>
                  <Map>
                     <MapEntry key='searchAttrsToGet'>
                        <List>
                           <String>LastName</String>
                           <String>ShortName</String>
                           <String>MailFile</String>
                        </List>
                     </MapEntry>
                     <MapEntry key='searchFilter'
value='@IsAvailable(LastName) &amp;
@Contains(@LowerCase(LastName);"smith")'/>
                  </Map>
               </invoke>
            </block>
         </Property>
      </Display>
      <Disable>
         <i>0</i>
      </Disable>
   </Field>
</Form>
```

Other Form Issues

- Only the HTTPPassword can be changed or reset via the administrator. If you do not want to change only the HTTPPassword, the default tables must filter the Domino adapter.
- The Change My Password, Change Password, and Expired Login forms generate a column named “Forgot Old Password?” This column must be removed for Domino resources since Identity Manager does not support administrator password updates.

Attributes Configured to be Passed Into Views

- idFile — Password, LoginChange
- DenyGroups — Enable, Disable, Delete
- certifierIdFile, credentials — Rename
- HTTPPassword — Password, LoginChange

Actions

The following variables are available for use in before and after actions:

- WSUSER_accountId
- WSUSER_UNID

The WSUSER_UNID variable refers to the Lotus Notes universal ID. This variable cannot be referenced until after the account has been created.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses the Sun Identity Manager Gateway to communicate with Domino.

Required Administrative Privileges

None

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	Yes
Pass-through authentication	No
Before/after actions	Yes
Data loading methods	• Import from resource • Reconciliation • Active Sync

Account Attributes

The following table provides information about Domino account attributes. The default data type is string, unless otherwise indicated.

Resource User Attribute	Description
alternateOrgUnit	The organizational unit for the user in the alternate language.
AltFullName	The user's full name, in the user's native language
AltFullNameLanguage	The language associated with the alternate full name.
Assistant	The name of an assistant.
CalendarDomain	The domain name for the calendar.
CellPhoneNumber	The user's cell phone number.
certifierIDFile	Path to the certifier ID file relative to the gateway machine (overrides value on resource)
CertifierOrgHierarchy	Path of certifier's organization hierarchy, such as /US1 (overrides value on resource)

Resource User Attribute	Description
CheckPassword	Integer. 1 = check 0 = no check
Children	The name or names of the employee's children.
City	The city of the user's home address.
Comment	A comment about the user.
CompanyName	The company the user works for.
Country	The country of the user's home address.
credentials	Password for the certifier ID file (overrides value on resource)
dbQuotaSizeLimit	Specifies the maximum size of the user's mail database. If you specify a value less than 1000, then the maximum size is in megabytes (MB). If the value is 1000 or greater, then the maximum size is expressed in bytes. Values between 1001 and 1023 are rounded up to 1024 bytes. The proxy administrator must be listed as an Administrator in the Server document to set this attribute.
dbQuotaWarningThreshold	Specifies the size of a user's mail database at which point a warning about the size of the database is generated. If you specify a value less than 1000, then the threshold is in megabytes (MB). If the value is 1000 or greater, then the threshold is expressed in bytes. Values between 1001 and 1023 are rounded up to 1024 bytes. The proxy administrator must be listed as an Administrator in the Server document to set this attribute.
defaultPasswordExp	Number of days for new certificates to be issued (create, recertify operations)

Resource User Attribute	Description
deleteMailFileOption	Overrides the resource attribute: • **0**: Do not delete mail file • **1**: Delete just mail file specified in person record • **2**: Delete mail file specified in person record and all replicas **Note**: If configured to delete the `mailfile` and `adminp` request will be queued and must be approved natively before it is deleted.
DenyGroups	
Department	The department name or number of the user.
DisplayName	The user's displayed name.
EmployeeID	The unique employee ID for the user.
firstname	The user's first name.
HomeFAXPhoneNumber	The user's home fax/phone number
HTTPPassword	Password to be used when accessing a Notes server from a web browser or other HTTP client.
idFile	Full qualified path to the ID file relative to the gateway machine.
gateway machine	
InternetAddress	
JobTitle	The user's job title.
lastModified	A string representation of the last date and time the user was modified.
lastname	The user's last name
Location	Office location or mail stop
MailAddress	The user's e-mail address.
MailDomain	Domain name of user's mail server

Resource User Attribute	Description
MailFile	The name of the mail file, such as MAIL\JSMITH
mailOwnerAccess	Indicates the access control level for the mailbox owner. Possible values are 0 (manager), 1(desginer), and 2 (editor). This attribute is not in the schema map by default. The attribute is applicable only when creating users.
MailServer	The user's mail server name.
MailTemplate	Name of mail template. Only valid during create.
Manager	The user's manager.
MiddleInitial	Middle initial with a trailing period.
NetUserName	The user's network account name.
NotesGroups	
objectGUID	The user's NotesID.
OfficeCity	The city of the user's work address.
OfficeCountry	The country of the user's work address.
OfficeFAXPhoneNumber	The fax number of the user's work address.
OfficeNumber	The office number of the user's work address.
OfficePhoneNumber	The phone number of the user's work address.
OfficeState	The state or province of the user's work address.
OfficeStreetAddress	The street address of the user's work address.
OfficeZIP	The postal code of the user's work address.
orgUnit	
password	The user's password

Resource User Attribute	Description
PasswordChangeInterval	Integer. The number of days after which the user must supply a new password.
PasswordGracePeriod	The number of days after the password has expired before the user is locked out.
PhoneNumber	The user's home telephone number.
PhoneNumber_6	
Profiles	
Recertify	Boolean. Flag to indicate you would like to recertify a user.
SametimeServer	Hierarchical name of the user's sametime server.
ShortName	Short user name commonly used by a foreign mail system.
Spouse	The name of the user's spouse.
State	The state or province in the user's home address.
StreetAddress	The address of the user's home address.
Suffix	The user's generational qualifier
Title	The user's title
WebSite	The user's web site.
WS_USER_PASSWORD	Attribute used to send user's current password during user change password requests.
x400Address	
Zip	The postal code of the user's home address.

Identity Template

Domino stores the identity of each user in the `userid` file. However, that same user name is stored in the user record in the `FullName` attribute. That attribute is multi-valued, and the first one in the list is unique. The first name in the list is stored in canonical format and is similar to the following:

```
CN=Joe T Smith/O=MyCompany
```

Using this name we can get to the record of the Name and address book. Identity Manager stores this string on the `resourceInfo` in its “nice” form, which looks like:

```
Joe T Smith/MyCompany
```

Domino has built-in functions to convert names back and forth at the API level. Identity Manager also stores the `NOTEID` as the `GUID` attributes, and whenever possible uses this global identifier to look up users in Domino.

The default identity template is:

```
$firstname$ $MiddleInitial$ $lastname$$CertifierOrgHierarchy$
```

Depending on the environment, the middle initial may not be not included.

Sample Forms

DominoActiveSyncForm.xml

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

```
com.waveset.adapter.DominoResourceAdapter
```

Exchange 5.5

Support for the Microsoft Exchange resource adapter has been deprecated.

Note Use the Active Directory resource for Exchange 2000/20003, which is integrated with Exchange.

Flat File Active Sync

The Flat File Active Sync adapter is defined in the `com.waveset.adapter.FlatFileActiveSyncAdapter` class.

The flat file Active Sync adapter provides the ability to read from the following types of files:

- Comma-separated value (CSV)
- Pipe-delimited (|)
- LDAP Data Interchange Format (LDIF), if the Netscape `ldapjdk.jar` is provided in the class path.

Custom parsers can also be used, if the parser class implements the `com.waveset.util.FlatFileIterator` interface.

This adapter is a source-only adapter; it will not write back out to a file.

The following cases are some examples in which it might be appropriate to use the Flat File Active Sync adapter:

- A direct API or other programmatic interface does not exist.
- No resource adapter exists for the specific resource.
- Data stored in one or more resources must be pre-processed before being read into Identity Manager.
- The resource owner does not allow direct connections to the resource.
- No direct connectivity is available to the resource.

Resource Configuration Notes

The flat file to be read in by the adapter must be available to the application server (or all application servers, if running a cluster) on a local hard drive, network share, or mounted drive, depending on the platform. The log directory must also be visible to the application server(s) and writable by the account under which the application server process is running.

The most reliable configuration (and recommended practice) is to store the flat file on a drive that is local to the application server. The log file should also be written to a local directory. If using multiple Identity Manager instances on different hosts, choose one host on which to run the flat file Active Sync adapter, and specify that host in the `waveset.properties` file as the value of sources.hosts. Setting this property will ensure that the polling operation on the adapter will always run one or more particular hosts.

Note If the log file cannot be written, the flat file Active Sync adapter will not start.

Identity Manager Installation Notes

No additional installation procedures are required on this resource.

Usage Notes

This section provides configuration notes related to using the Flat File Active Sync resource adapter, which is organized into the following sections:

- *General Notes*
- *Active Sync Configuration*
- *Supported Example Files*

General Notes

If you are polling an LDIF file, the LDAP API converts attribute names to lower case. Therefore, if you have an attribute name that contains a capital letter, such as `accountId`, the LDAP API converts it to `accountid`. The following error is logged when you start Active Sync.

```
com.waveset.util.WavesetException: No name attribute found for user
based on Resolve Identity Rule or schema map.
```

To correct this situation, in your schema map, set your resource user attribute to `accountid`.

You can have a comma-separated field in a CSV file by double quoting the field, or by using a pipe delimiter. The data would need to be transformed into a list by the Active Sync form or in subsequent process.

You might encounter the same error message when you import a file that does not directly set the accountId via a column in the file. To avoid this error message, change the Active Sync User Form by adding a Field for `global.accountId` and adding logic to build the accountId within that field. The following example field sets accountId to be `firstname.lastname`, but only on `create` operations.

```
<Field name='waveset.accountId'>
   <Expansion>
      <concat>
         <ref>activeSync.firstname</ref>
         <s>.</s>
         <ref>activeSync.lastname</ref>
      </concat>
   </Expansion>
   <Disable>
      <neq>
         <ref>feedOp</ref>
         <s>create</s>
      </neq>
   </Disable>
</Field>
```

Active Sync Configuration

The Flat File Active Sync adapter can track the timestamp of a flat file. In addition, the adapter can archive the last file processed and then compare it to the most recent version. Identity Manager will then act on the accounts that are different in the two files.

If these features are enabled, the first time Identity Manager polls the source flat file, the system copies the file and places it in the same directory. The copied (archived) file is named `FFAS_`*`timestamp`*`.FFAS`, with the timestamp indicating the last time the original file was changed. The format of the timestamp is determined by the operating system on which the source file resides.

On each subsequent poll, Identity Manager compares the timestamp on the original file with the most recent timestamp. If the new timestamp value is the same as the previous value, then the file has not changed, and no further processing is performed until the next poll. If the timestamp values are different, Identity Manager checks for the presence of the FFAS file. If the file does not exist, Identity Manager processes the updated source file as if it were a new file.

If the timestamps are different and the archived FFAS file exists, Identity Manager compares the source file with the archived file. The comparison will filter any users that have not changed. If a user has changed, then it will be sent through the adapter in the normal manner, and the configured process, correlation and delete rules determine what to do with the user.

To facilitate these rules, the adapter will add an additional attribute to indicate the situation discovered by the difference mechanism. If any users exist only in the newly-updated source file, the user record will have an additional attribute `diffAction` which will have the value of `create`. If any entries were updated in the source file, the attribute `diffAction` will be added and the value set to `update`. If any users were deleted then `diffAction` will be `delete`.

After the comparison of the two files is complete and all account processing has taken place, Identity Manager deletes the original FFAS file and copies the current source file to a new FFAS file. The timestamp on this file will be different than the previous FFAS file.

Supported Example Files

The following example files are supported by the adapter.

Comma-Separated Values

```
accountId,firstname,lastname,email,street address
kb323441,Kevin,Brown,Kevin.Brown@example.com,”1234 Pecan Ave., Ste 30”
pc432343,Penelope,Carter,Penelope.Carter@example.com,4234 Main St.
```

Pipe-Delimited

```
accountId|firstname|lastname|email|street address
kb323441|Kevin|Brown|Kevin.Brown@example.com|1234 Pecan Ave., Ste 30
pc432343|Penelope|Carter|Penelope.Carter@example.com|4234 Main St.
```

LDAP Interchange Format

```
dn: cn=Kevin Brown,ou=People,dc=example,dc=com
changetype: add
objectClass: top
objectClass: person
objectClass: organizationalperson
objectClass: inetorgperson
employeeNumber: kb323441
cn: Kevin Brown
sn: Brown
departmentNumber: 7013
description: Production
displayName: Kevin
givenName: Kevin
mail: Kevin.Brown@example.com
o: Acme
ou: Production
postalAddress: 1234 Pecan Ave., Ste 30
postalCode: 43231
st: CA
```

```
street: 1234 Pecan Ave, Ste 30
title: Production Assistant
```

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

See the “Resource Configuration Notes” on page 109.

Required Administrative Privileges

None

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	No
Rename account	No
Pass-through authentication	No
Before/after actions	No
Data loading methods	Active Sync

Account Attributes

The resource adapter schema definition is dependent on the contents of the flat file. If no attributes are specified, the adapter will use the attribute names pulled from the flat file. In the case of a CSV or pipe-delimited file, these values will correspond to the column heads. If different Identity Manager attribute names should be mapped to the column names, specify one or more of those mappings in the schema map.

Resource Object Management

Not applicable

Identity Template

The identity template is ignored by this adapter.

Sample Forms

None

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

`com.waveset.adapter.FlatFileActiveSyncAdapter`

GroupWise

The GroupWise resource adapter is defined in the `com.waveset.adapter.GroupWiseResourceAdapter` class.

This adapter supports the following versions of Novell GroupWise:

- 5.5
- 6.0

Resource Configuration Notes

None

Identity Manager Installation Notes

No additional installation procedures are required on this resource.

Usage Notes

None

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses the following to communicate with the GroupWise adapter:

- NDS Client
- LDAP
- SSL

Required Administrative Privileges

None

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Not applicable
Rename account	
Pass-through authentication	Yes
Before/after actions	
Data loading methods	• Reconciliation • Import from resource

Account Attributes

The following table provides information about GroupWise account attributes.

Notes:

- Attribute Types are String unless indicated otherwise.
- Unless specifically indicated, attributes are not required. If required, check the corresponding box in the Required column of the Administrator Interface Resource Wizard Account Attributes page.

Attribute Type (Syntax)	Resource User Attribute	Required?	GroupWise Name (Display name as it appears on the native resource)
	NetID	Y	
	GivenName		
	Surname		
Encrypted	userPassword	Y	
	Department		
	FaxNumber		
	GatewayAccess		

Attribute Type (Syntax)	Resource User Attribute	Required?	GroupWise Name (Display name as it appears on the native resource)
	MailboxExpDate		
	PhoneNumber		
	Title		

Resource Object Management

Identity Template

$accountId$

Sample Forms

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

`com.waveset.adapter.GroupWiseResourceAdapter`

HP-UX

The HP-UX resource adapter is defined in the `com.waveset.adapter.HPUXResourceAdapter` class.

This adapter supports the following versions of HP-UX:

- 11.0
- 11i v1
- 11i v2

Resource Configuration Notes

If you will be using SSH (Secure Shell) for communication between the resource and Identity Manager, set up SSH on the resource before configuring the adapter.

Identity Manager Installation Notes

No additional installation procedures are required on this resource.

Usage Notes

The HP-UX resource adapter primarily provides support for the following HP-UX commands:

- `useradd, usermod, userdel`
- `groupadd, groupmod, groupdel`
- `passwd`

For more information about supported attributes and files, refer to the HP-UX manual pages for these commands.

When a rename of a user account is executed on a HP-UX resource, the group memberships are moved to the new user name. The user's home directory is also renamed if the following conditions are true:

- The original home directory name matched the user name.
- A directory matching the new user name does not already exist.

The Bourne-compliant shell (sh, ksh) must be used as the root shell when connecting to a UNIX resource (AIX, HP-UX, Solaris, or Linux).

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses the following connections to communicate with the HP-UX adapter.

- Telnet
- SSH (SSH must be installed independently on the resource.)

Required Administrative Privileges

The adapter supports logging in as a standard user, then performing a su command to switch to root (or root-equivalent account) to perform administrative activities. Direct logins as root user are also supported.

The adapter also supports the sudo facility (version 1.6.6 or later), which can be installed on HP-UX 11i from the HP-UX Internet Express CD. sudo allows a system administrator to give certain users (or groups of users) the ability to run some (or all) commands as root or another user.

In addition, if sudo is enabled for a resource, its settings will override those configured on the resource definition page for the root user.

If you are using sudo, you must set the `tty_tickets` parameter to true for the commands enabled for the Identity Manager administrator. Refer to the man page for the sudoers file for more information.

The administrator must be granted privileges to run the following commands with sudo:

User and Group Commands	NIS Commands	Miscellaneous Commands	
• groupadd • groupdel • groupmod • last • listusers • logins • passwd • useradd • userdel • usermod	• make • ypcat • ypmatch • yppasswd	• awk • cat • chmod • chown • cp • cut • diff • echo • grep	• ls • mv • rm • sed • sleep • sort • tail • touch • which

In addition, the NOPASSWORD option must be specified for each command.

You can use a test connection to test whether

- These commands exist in the administrator user's path
- The administrative user can write to /tmp
- The administrative user have rights to run certain commands

Note A test connection can use different command options than a normal provision run.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	HP-UX does not natively support Identity Manager enable and disable actions. Identity Manager simulates enabling and disabling accounts by changing the user password. The changed password is exposed on enable actions, but it is not exposed on disable actions. As a result, enable and disable actions are processed as update actions. Any before or after actions that have been configured to operate on updates will execute.
Rename account	Yes
Pass-through authentication	Yes
Before/after actions	Yes
Data loading methods	• Import directly from resource • Reconcile with resource

You can define resource attributes to control the following tasks for all users on this resource:

- Create a home directory when creating the user
- Copy files to the user's home directory when creating the user
- Delete the home directory when deleting the user

Account Attributes

The following table lists the HP-UX user account attributes.

Notes:

- These attributes are optional unless noted in the description.
- All attributes are Strings.

Resource User Attribute	useradd Equivalent	Description
accountId	login	Required. The user's login name.
comment	-c *comment*	The user's full name.
dir	-d *directory*	The user's home directory.
expire	-e *expiration date*	Last date the account can be accessed.
group	-g *group*	The user's primary group.
inactive	-f *days*	Number of days the account can be inactive before it is locked
secondary_group	-G *group*	The user's secondary group or groups.
shell	-s /*Path*	The user's login shell.
time_last_login	Obtained from the last command.	The date and time of the last login. This value is read-only.
uid	-u *User ID*	The user ID, in digit form.

Resource Object Management

Identity Manager manages the following native HP-UX objects:

Resource Object	Supported Features	Attributes Managed
Group	Create, update, delete, rename, save as	groupName, gid, users

Identity Template

```
$accountId$
```

Sample Forms

Built-In

- HP-UX Group Create Form
- HP-UX Group Update Form

Also Available

`HP-UXUserForm.xml`

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.HPUXResourceAdapter`
- `com.waveset.adapter.SVIDResourceAdapter`
- `com.waveset.adapter.ScriptedConnection`

INISafe Nexess

The INISafe Nexess resource adapter is defined in the `com.waveset.adapter.INISafeNexessResourceAdapter` class.

This adapter supports Nexess 1.1.5.

Resource Configuration Notes

None

Identity Manager Installation Notes

The INISafe Nexess resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. Add the following value in the Custom Resources section of the Configure Managed Resources page.
   - `com.waveset.adapter.INISafeNexessResourceAdapter`
2. Copy the following JAR files to the *`%WSHOME%`*`\WEB-INF\lib` directory:

JAR Name	How to Obtain
concurrent.jar	http://www.jboss.org/products/jbosscache
crimson.jar	http://ant.apache.org/bindownload.cgi
external-debug.jar	Contact INITECH support.
INICrypto4Java.jar	Installed with INISafe Nexess or contact INITECH support.
jdom.jar	http://jdom.org/downloads/index.html
log4j-1.2.6.jar	http://logging.apache.org/log4j/docs/download.html

Usage Notes

This adapter supports only create, update and delete of users. You cannot perform reconciliation or load data from the resource.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Communication with INISafe Nexess is conducted through the `com.initech.eam.api` classes.

Required Administrative Privileges

The administrator must have access to the Nexess Daemon and Login Server.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	No
Pass-through authentication	No
Before/after actions	No
Data loading methods	Not applicable. This adapter only allows you to create, delete, and update users individually.

Account Attributes

The following table lists the INISafe Nexess account attributes.

Resource User Attribute	Data Type	Description
accountId	string	Required. The user's account ID.
password	Encrypted	Required. The user's password.
fullname	string	Required. The user's full name.
email	string	Required. The user's e-mail address.
enable	string	Indicates whether the user is enabled. This attribute is not displayed by default.

If you add other account attributes, the resource user attribute name must be in one of the following formats:

- `Account.`*`name`*
- `Attribute.`*`name`*
- `Field.`*`name`*

For example, a field named `sn` must have resource user attribute name of `Field.sn`

If the resource has accounts, then you may need to add a resource user attribute named `Account.accounts`. Account names are serialized as comma-separated value (CSV) strings with three fields:

```
ServiceName, accountId, password
```

Your user form will need to construct and deconstruct these strings.

Resource Object Management

None

Identity Template

```
$accountId$
```

Sample Forms

None

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

```
com.waveset.adapter.INISafeNexessResourceAdapter
```

JMS Listener

The JMS Listener adapter is a JMS (Java Message Service) client which provides the ability to perform Active Sync processing on messages from a JMS-compliant messaging system queue or topic.

This adapter is a source-only adapter; it cannot write messages back to a queue or topic.

The JMS Listener resource adapter is defined in the `com.waveset.adapter.JmsListenerResourceAdapter` class.

Resource Configuration Notes

The JMS Listener adapter can only interact with a messaging system which supports the JMS (Java Message Service) open standard, version 1.1 or later.

The adapter interacts with the source JMS messaging system topic or queue through standard JNDI lookups of a specified connection factory and destination. Therefore, the messaging system administrator must ensure that the connection factory and destination have been previously created and are available through standard JNDI lookups.

Identity Manager Installation Notes

The JMS Listener resource adapter works only in an application server environment that supports the following:

- Client API for JMS, version 1.1 or later
- JNDI (Java Naming and Directory Interface) API 1.1 or later

The application server administrator must ensure that the Identity Manager web application can successfully bind via JNDI to the JMS connection factory and destination objects appropriate for the source JMS messaging system.

Usage Notes

Connections

When Active Sync processing begins, a connection to the source messaging system is first made using the connection factory specified with the **JNDI name of Connection factory** resource parameter field. If specified, the **User** and **Password** fields are used for authentication when establishing the connection. If the fields are not specified, the connection are established using the default authentication.

The JMS Listener adapter operates in an asynchronous mode. It establishes an asynchronous listener on the queue or topic destination specified by the **JNDI name of Destination** field. When a qualified message arrives at the destination, the adapter immediately processes the message. Messages can be (optionally) additionally qualified by defining a valid JMS message selector string for the **Message Selector** field.

The connection factory and destination attributes must specify objects which correspond to the specified destination type. If a destination type of Durable Topic is specified, the additional fields of **Durable Topic ClientID** and **Durable Topic Subscription Label** are used to configure the durable subscription.

Message Mapping

When the adapter processes a qualified message, the received JMS message is first converted to a map of named values using the mechanism specified by the **Message Mapping** field. Refer to this resulting map as the *message value map*.

The message value map is then translated to the final value map using the account attributes schema map. If the adapter has account attributes specified, the adapter searches the message value map for key names that also appear as a resource user attribute in the schema map. If present, the value is copied to the final value map, but the entry name in the final value map is translated to the name specified in the Identity system user attribute column in the schema map.

If the message value map has an entry that cannot be translated using the account attributes schema map, then the entry from the message value map is copied unaltered to the final value map.

Guaranteed Delivery/Reliable Processing

The responsibility of guaranteed delivery lies with the sender of the message. Only messages sent persistently will be stored until delivered by the messaging system. This guarantees that the message will not be lost due to a crash or shutdown of the messaging system. This is referred to as once-and-only-once delivery.

The **Reliable Messaging Support** field indicates the form of reliable message processing the adapter should perform.

- If set to LOCAL, then the JMS session for the adapter is transacted. The session is always committed after the message is processed, regardless of any errors encountered during the processing stages. This ensures that the message is processed only once.
- If set to AUTO, then the session is not transacted, but the message is automatically acknowledged immediately according to the JMS definition of AUTO_ACK.
- If set to DUPS_OK, then the session is not transacted, but the message is automatically acknowledged immediately according to the JMS definition of DUPS_OK_ACK.
- If set to CLIENT, then the session is not transacted, and the message is not acknowledged by the adapter. Instead, it is expected that a lifecycle listener specified by the **Message LifeCycle Listener** field acknowledges the message as needed. The lifecycle listener is called with an AWAITING_CLIENT_ACK lifecycle event at the typical point that an acknowledgement is expected. It is rare that this mode is needed.

LifeCycle Listener

An optional lifecycle listener class can be registered with the adapter with the **Message LifeCycle Listener** field. The lifecycle listener can be used to perform:

- Custom logging of the processing stages of the adapter
- Custom manipulation of data during processing stages of the adapter
- Custom acknowledgement of messages received with CLIENT_ACK mode

Reconnections

If connection is lost to the messaging system (for example, the messaging system server has been shut down), the adapter can be configured to periodically attempt to reconnect with the messaging system to re-establish the listener.

The **Re-initialize upon exception** check box enables reconnect behavior. You can set the frequency to attempt reconnect with the **Connection Retry Frequency (secs)** field.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Many messaging systems support the capability to encrypt messages between clients and brokers. The configuration is specific to each messaging system. However, typically the encryption is abstracted so that the choice of a specially-configured connection factory is sufficient to enable encryption between the JMS Listener adapter and the messaging system broker.

Required Administrative Privileges

The user and password configured for the JMS Listener adapter must be an authenticated user in the JMS messaging system, and that user must be granted sufficient privilege to read messages from the JMS destination.

It is recommended that the messaging system administrator protect the JMS connection by disabling default authentication. For further protection, the messaging system administrator should configure the authorization (access control) to optimize security.

Provisioning Notes

The following table summarizes the provisioning capabilities of the JmsListener adapter.

Feature	Supported?
Create account	No
Update account	No
Delete account	No
Enable/disable account	No

Feature	Supported?
Rename account	No
Pass-through authentication	No
Before/after actions	No
Data loading methods	None

Account Attributes

The JMS Listener adapter does not provide default account attributes because the account attributes vary greatly, depending on the semantics of the messages read from the topic or queue.

You must define an account attribute in which the Identity System user attribute is named `accountId`.

Resource Object Management

Not supported.

Identity Template

None. You must supply the identity template with a valid value.

Sample Forms

`JmsListenerActiveSync.xml`

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

`com.waveset.adapter.JmsListenerResourceAdapter`

You may also set the following Active Sync logging parameters for the resource instance:

- Maximum Log Archives
- Maximum Active Log Age
- Maximum Log File Size

- Log File Path
- Log Level

The Test Configuration button in the resource wizard when creating or editing a resource of type JMS Listener does an extensive check. It is valuable to troubleshoot configuration issues.

Additionally, a simple tool to send or publish messages to a queue or topic is available in a report called Send JMS Message. To use the report, first import the exchange file `$WSHOME/sample/SendJMSMessageReport.xml`. You can then create instances of the Send JMS Message report. When an instance of this report is run, it writes the specified message to the specified queue or topic.

LDAP

Identity Manager provides the following resource adapters for supporting Lightweight Directory Access Protocol (LDAP) v3:

GUI Name	Class Name
LDAP	`com.waveset.adapter.LDAPResourceAdapter`
LDAP Listener Active Sync	`com.waveset.adapter.LDAPListenerActiveSyncAdapter`

The LDAP adapter provides provisioning services for standard LDAP installations.It can also read the replication changelog of an LDAP server and apply those changes to Identity Manager users or custom workflows.

The LDAP Listener Active Sync adapter uses an LDAP Listener to discover changes as they are made on the server, queues them, and processes them at the scheduling interval. The listener is primarily intended for demos as it requires the Identity Manager server to be connected at all times. Any changes made when the adapter is not running will be lost.

Note The LDAP ChangeLog Active Sync adapter has been deprecated. All functionality of this adapter has been merged into the LDAP resource adapter. Although existing instances of resources using the deprecated adapter will still function, new instances of resources using the LDAP ChangeLog Active Sync adapter can no longer be created.

Resource Configuration Notes

To setup a Sun Java™ System Directory Server resource for use with the LDAP adapter, you must configure the server to enable the change log and enable tracking of modifier information. This is done from the directory server configuration tab.

1. Click on the Replication folder, then select the “Enable change log” box. For 5.0 and later servers, you must also enable the RetroChangelog Snapin. On the configuration tab go to the plugin object, select the Retro change log plugin and enable it.
2. To verify that the server is configured to maintain special attributes for newly created or modified entries, in the Directory Server console, click Configuration > select the root entry in the navigation tree in the left pane.
3. Click Settings > verify that the Track Entry Modification Times box is checked.

The server adds the following attributes to a newly created or modified entry to determine if an event was initiated from Identity Manager.

- **creatorsName**: The DN of the person who initially created the entry.
- **modifiersName**: The DN of the person who last modified the entry.

To connect to a directory server via SSL in which a self-signed certificate has been implemented, perform the following procedure:

1. Export the CA certificate from the directory server to a temporary file. For example, on Sun Java™ System Directory, enter the following command:

   ```
   certutil -L -d DB_Directory -P slapd-HostName- -n Nickname -a > ds-cert.txt
   ```

2. Import this certificate into your keystore.

   ```
   cd $JAVA_HOME/jre/lib/security
   keytool -import -file PathTo/ds-cert.txt -keystore ./cacerts
   -storepass changeit -trustcacerts
   ```

Identity Manager Installation Notes

No additional installation procedures are required on this resource.

Usage Notes

This section provides information related to using the LDAP resource adapter, which is organized into the following sections:

- *General Notes*
- *Virtual List View Support*
- *Active Sync Configuration*

For information about enabling password synchronization on an LDAP resource, see *Synchronizing LDAP Passwords* on page 4-1.

General Notes

It is recommend that you create an Identity Manager service account to connect to LDAP, rather than using the administrator account CN=Directory Manager. Use your LDAP Directory Server management tool to set permissions via an ACI (access control instructions) at each base context.

Set the permissions in the ACI based on the source. If the adapter is connecting to an authoritative source, then set read, search, and possibly compare permissions only. If the adapter is used to write back, then you will need to set write and possibly delete permissions.

Note If the account will be used for the monitoring the changelog, an ACI should also be created on `cn=changelog`. The permissions should be set to read and search only, because you cannot write or delete changelog entries.

For the Listener adapter, the changes by users listed in the “Filter changes by” resource parameter will be ignored. Add the User DN used by any adapter to make changes through Identity Manager. This avoids loops where a change is made through Identity Manager, and then the change is detected and reapplied. If this field is blank, changes from any administrator are processed and will be filtered by the Identity Manager provisioning engine if they are unnecessary.

The `sources.`*`ResourceName`*`.hosts` property in the `waveset.properties` file can be used to control which host or hosts in a cluster will be used to execute the synchronization portion of an Active Sync resource adapter. *`ResourceName`* must be replaced with the name of the Resource object.

Virtual List View Support

Note This discussion assumes that Identity Manager connects to the LDAP resource as a non-RootDN user. If you are connecting as a RootDN user, the procedures described are applicable, but additional LDAP attribute values might be possible. Consult the Directory Server documentation for more information.

In Directory Server, the `nsLookThroughLimit` and `nsslapd-sizelimit` attributes define how many LDAP entries can be searched and returned, respectively. The default value for `nsLookThroughLimit` is 5,000, while the default for `nsslapd-sizelimit` is 2,000. Both attributes can be set to -1 to disable limits. You must restart Directory Server if you change the value of these attributes.

It is not always desirable to change the default values. To improve performance on LDAP searches, you can enable the LDAP Virtual List View (VLV) control. VLV returns partial results of a search, rather than returning all results at once.

The **`Use Blocks`** resource attribute enables Identity Manager to stay within the query result size limit by using the VLV control. The **`Block Count`** resource attribute specifies how many users to return, but this value must be less than or equal to the value set in the `nsslapd-sizelimit` attribute.

A VLV index (also known as a browsing index) must be created, or the `nsslapd-sizelimit` size limit will still be in effect. Using a VLV index significantly improves the performance of iterating over accounts, so you should set up the index if you plan to reconcile, load from resource, or export to file frequently.

Refer to the Directory Server documentation for detailed instructions on creating a VLV index. The basic process follows:

1. Create a `vlvsearch` object with the following properties:

   ```
   vlvbase: YourBaseContext
   vlvfilter: (&(objectclass=top)(objectclass=person)
   (objectclass=organizationalPerson) (objectclass=inetorgperson))
   vlvscope: 2
   ```

   The `vlvbase` attribute must match the value specified in the **`Base Context`** resource attribute. The `vlvfilter` attribute must contain the classes specified in the **`Object Classes`** resource attribute in the format shown. The `vlvscope` value of 2 indicates subtree searches.
2. Create a `vlvindex` component as a subobject of `vlvsearch`. The `vlvsort` attribute must be set to `uid`.
3. Build the VLV index using the `vlvindex` command or other mechanism.
4. Set permissions via access control instructions (ACI) for the following:
   - vlvsearch object
   - vlvindex
   - the directory the index was created for.

To set up VLV for the changelog, use the following general steps. Refer to the Directory Server documentation for detailed instructions.

1. If you have not already done so, create a browsing index for the changelog. If you use the Directory Server user interface, then by default, a `vlvsearch` object named `"MCC cn=changelog"` and a `vlvindex` object named `"SN MCC cn=changelog"` will be created.
2. Set permissions via access control instructions (ACI) so that the Identity Manager account has read, compare, and search rights for the following:
   - The changelog (`cn=changelog`)
   - The `vlvsearch` object (`cn="MCC cn=changelog",cn=config,cn=ldbm`)

- The `vlvindex` object (`"SN MCC cn=changelog",cn=config,cn=ldbm`)

Note On some versions of Directory Server, the changelog `nsLookThroughLimit` attribute has a hard-coded value of 5,000. To avoid hitting the changelog lookthrough limit, restrict the maximum number of changelog entries that are kept on the server to less than 5,000. To avoid losing changelog entries, set the polling frequency for the adapter to a short interval.

Active Sync Configuration

Before Identity Manager 5.5, the LDAP Active Sync adapters used the **`Process to run with changes`** field to determine which process to launch when a change was detected. The process specified in this field is now specified in the Active Sync **`Resolve Process Rule`**.

In addition, before Identity Manager 5.5, if the **`Process deletes as updates`** check box was selected, Identity Manager would disable a deleted Identity Manager user as well as all resource accounts and mark the user for later deletion. By default, this check box was selected. In Identity Manager 5.5 and beyond, this functionality is configured by setting the Delete Rule set to None.

If the checkbox was previously deselected, then the Delete Rule will be set to **`ActiveSync has isDeleted set`**.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses Java Naming and Directory Interface (JNDI) over TCP/IP or SSL to communicate with the LDAP adapter.

- If you are using TCP/IP, specify port 389 on the Resource Attributes page.
- If you are using SSL, specify port 636.

Required Administrative Privileges

If the value `cn=Directory Manager` is specified in the User DN resource parameter, then the Identity Manager administrator has the necessary permissions to manage LDAP accounts. If a different distinguished name is specified, that user must have the ability to read, write, delete, and add users.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	Yes
Pass-through authentication	Yes
Before/after actions	No
Data loading methods	• Import directly from resource • Reconcile with resource

Account Attributes

The syntax (or type) of an attribute usually determines whether the attribute is supported. In general, Identity Manager supports boolean, string, and integer syntaxes. Binary strings and similar syntaxes are not supported.

The following table lists the supported LDAP syntaxes. Other LDAP syntaxes might be supported, as long as it is boolean, string, or integer in nature,

LDAP Syntax	Attribute Type	Object ID
Boolean	Boolean	1.3.6.1.4.1.1466.115.121.1.7
Country String	String	1.3.6.1.4.1.1466.115.121.1.11
DN	String	1.3.6.1.4.1.1466.115.121.1.12
Directory String	String	1.3.6.1.4.1.1466.115.121.1.15
Generalized Time	String	1.3.6.1.4.1.1466.115.121.1.24
IA5 String	String	1.3.6.1.4.1.1466.115.121.1.26
Integer	Int	1.3.6.1.4.1.1466.115.121.1.27
Postal Address	String	1.3.6.1.4.1.1466.115.121.1.41
Printable String	String	1.3.6.1.4.1.1466.115.121.1.44
Telephone Number	String	1.3.6.1.4.1.1466.115.121.1.50

Default Account Attributes

The following attributes are displayed on the Account Attributes page for the LDAP resource adapters.

Note All attributes are of type String unless otherwise noted.

Identity System Attribute	Resource User Attribute	LDAP Syntax	Description
accountId	uid	Directory string	User ID
accountId	cn	Directory string	Required. The user's full name.
firstname	givenname	Directory string	The user's first (given) name.
lastname	sn	Directory string	Required. The user's last name (surname).
modifyTimeStamp	modifyTimeStamp	Generalized time	Indicates when a user entry was modified. By default, this attribute is displayed for the LDAP Listener ActiveSync adapter only.
objectClass	objectClass	OID	The object class or classes of the account This attribute is required for Active Sync to process updates correctly.
password	userPassword	Octet string	Encrypted. The user's password.

Group Management Attributes

The account attributes in the following table are not displayed in the schema by default. You must add the attribute to the schema map before you can manage groups.

Identity System Attribute	Resource User Attribute	LDAP Syntax	Description
user defined	ldapGroups	N/A	A list of distinguished names of groups the LDAP user is a member of. The resource attribute **Group Member Attr** specifies the attribute of the LDAP group entry that will be updated to contain the distinguished name of the user. The default value for the **Group Member Attr** is `uniquemember`.
user defined	posixGroups	N/A	A list of distinguished names of `posixGroups` entries the LDAP user is a member of. For an account to be assigned membership in a Posix group, it must have a value for the `uid` LDAP attribute. The `memberUid` attribute of the `posixGroup` entries will be updated to contain the `uid` of the user.

Note the following behavior when either posixGroups or ldapGroups is defined in the schema map:

- When an LDAP account is deleted, then Identity Manager removes the account's DN from any LDAP groups and the account's uid from any posixGroups.
- When the uid of an account changes, then Identity Manager replaces the old uid with the new uid in the appropriate posixGroups.
- When an account is renamed, then Identity Manager replaces the old DN with the new DN in the appropriate LDAP groups.

Person Object Class

The following table lists additional supported attributes that are defined in the LDAP Person object class. Some attributes defined in the Person object class are displayed by default.

Resource User Attribute	LDAP Syntax	Attribute Type	Description
description	Directory string	String	A short informal explanation of special interests of a person
seeAlso	DN	String	A reference to another person.
telephoneNumber	Telephone number	String	Primary telephone number

Organizationalperson Object Class

The following table lists additional supported attributes that are defined in the LDAP Organizationalperson object class. This object class can also inherit attributes from the Person object class.

Resource User Attribute	LDAP Syntax	Attribute Type	Description
destinationIndicator	Printable string	String	This attribute is used for the telegram service.
facsimileTelephoneNumber	Facsimile telephone number	String	The primary fax number.
internationaliSDNNumber	Numeric string	String	Specifies an International ISDN number associated with an object.
l	Directory string	String	The name of a locality, such as a city, county or other geographic region
ou	Directory string	String	The name of an organizational unit
physicalDeliveryOfficeName	Directory string	String	The office where deliveries are routed to.
postalAddress	Postal address	String	The office location in the user's place of business.
postalCode	Directory string	String	The postal or zip code for mail delivery.

Resource User Attribute	LDAP Syntax	Attribute Type	Description
postOfficeBox	Directory string	String	The P.O. Box number for this object.
preferredDeliveryMethod	Delivery method	String	The preferred way to deliver to addressee
registeredAddress	Postal Address	String	A postal address suitable for reception of telegrams or expedited documents, where it is necessary to have the recipient accept delivery.
st	Directory string	String	State or province name.
street	Directory string	String	The street portion of the postal address.
teletexTerminalIdentifier	Teletex Terminal Identifier	String	The teletex terminal identifier for a teletex terminal associated with an object
telexNumber	Telex Number	String	The telex number in the international notation
title	Directory string	String	Contains the user's job title. This property is commonly used to indicate the formal job title, such as Senior Programmer, rather than occupational class, such as programmer. It is not typically used for suffix titles such as Esq. or DDS.
x121Address	Numeric string	String	The X.121 address for an object.

inetOrgPerson Object Class

The following table lists additional supported attributes that are defined in the LDAP inetOrgPerson object class. This object class can also inherit attributes from the organizationalPerson object class.

Resource User Attribute	LDAP Syntax	Attribute Type	Description
businessCategory	Directory string	String	The kind of business performed by an organization.
carLicense	Directory string	String	Vehicle license or registration plate
departmentNumber	Directory string	String	Identifies a department within an organization
displayName	Directory string	String	Preferred name of a person to be used when displaying entries
employeeNumber	Directory string	String	Numerically identifies an employee within an organization
employeeType	Directory string	String	Type of employment, such as Employee or Contractor
homePhone	Telephone number	String	The user’s home telephone number.
homePostalAddress	Postal address	String	The user’s home address.
initials	Directory string	String	Initials for parts of the user's full name
labeledURI	Directory string	String	A Universal Resource Indicator (URI) and optional label associated with the user.
mail	IA5 string	String	One or more email addresses.
manager	DN	String	Directory name of the user's manager.
mobile	Telephone number	String	The user’s cell phone number.
o	Directory string	String	The name of an organization.
pager	Telephone number	String	The user’s pager number.

Resource User Attribute	LDAP Syntax	Attribute Type	Description
preferredLanguage	Directory string	String	Preferred written or spoken language for a person.
roomNumber	Directory string	String	The user's office or room number.
secretary	DN	String	Directory name of the user's administrative assistant.

The following attributes are not supported:

- audio (octet string)
- jpegPhoto (JPEG)
- photo (Fax)
- userCertificate (certificate)
- userSMIMECertificate (octet string)
- userPKCS12 (octet string)
- x500uniqueIdentifier (bit string)

Resource Object Management

Identity Manager supports the following LDAP objects by default. Any string-, integer-, or boolean-based attributes can also be managed.

Resource Object	Features Supported	Attributes Managed
Group	Create, update, delete, rename, saveas	cn, description, owner, uniqueMember
Posix Group	Create, update, delete, rename, saveas	cn, description, gid, memberUid
Domain	Find	dc
Organizational Unit	Create, delete, rename, saveas, find	ou
Organization	Create, delete, rename, saveas, find	o

The LDAP resource adapter provides management of posixGroup entries. By default, the list of accounts that are available to be assigned to a posixGroup have the posixAccount object class. The LDAP Create Posix Group Form and LDAP Update

Posix Group From can be customized to list accounts other than posixAccounts. However, these accounts must have a uid attribute defined to be a member of a posixGroup.

Identity Template

The default identity template is

```
uid=$accountId$,ou=EngUsers,dc=support,dc=waveset,dc=com
```

You must replace the default template with a valid value.

Sample Forms

Built-in

- LDAP Create Group Form
- LDAP Create Organization Form
- LDAP Create Organizational Unit Form
- LDAP Create Person Form
- LDAP Create Posix Group Form
- LDAP Update Group Form
- LDAP Update Organization Form
- LDAP Update Organizational Unit Form
- LDAP Update Person Form
- LDAP Update Posix Group Form

Also Available

- `LDAPActiveSyncForm.xml`
- `LDAPGroupCreateExt.xml`
- `LDAPGroupUpdateExt.xml`
- `LDAPPasswordActiveSyncForm.xml`

The `LDAPGroupCreateExt.xml` and `LDAPGroupUpdateExt.xml` forms allow non-unique member names.

Troubleshooting

Use the Identity Manager debug pages to set trace options on one or more of the following classes:

- `com.waveset.adapter.LDAPResourceAdapterBase`
- `com.waveset.adapter.LDAPResourceAdapter`
- `com.waveset.adapter.LDAPListenerActiveSyncAdapter`

Microsoft Identity Integration Server

The Microsoft Identity Integration Server (MIIS) resource adapter is defined in the `com.waveset.adapter.MIISResourceAdapter` class.

This adapter supports the following versions of MIIS:

- 2003

The MIIS adapter is implemented as a database table resource adapter. Therefore, the MIIS adapter has the same installation requirements and requires the same administrative privileges as the underlying database.

The MIIS adapter can be used with the following database systems:

- SQL Server
- DB2
- MySQL
- Oracle

Resource Configuration Notes

None

Identity Manager Installation Notes

Note These installation notes assume that a SQL Server database table will be managed. If you are using a database other than SQL Server, copy the jar files required for that database. See the Identity Manager Installation Notes section of the appropriate database resource adapter for more information.

The MIIS resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. Select the Microsoft Identity Integration Server option from the Resources section of the Configure Managed Resources page.
2. Copy the following files to the *`InstallDir`*`\idm\WEB-INF\lib` directory. For more information obtaining these files, see the SQL page 1-323
   - `msbase.jar`
   - `mssqlserver.jar`
   - `msutil.jar`

Usage Notes

None

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses JDBC to communicate with the MIIS adapter.

Required Administrative Privileges

The user must be able to read, write, delete, and change fields in the database. See the database adapter documentation for more information.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	No
Pass-through authentication	Yes
Before/after actions	No
Data loading methods	• Import data from resource • Reconciliation

Account Attributes

The list of account attributes is determined by which database columns were selected as Managed Columns during configuration of the MIIS resource. The possible account attributes vary for each installation.

Resource Object Management

None

Identity Template

$accountId$

Sample Forms

None

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.MIISResourceAdapter`
- `com.waveset.adapter.JdbcResourceAdapter`

Microsoft SQL Server

The MIcrosoft SQL Server resource adapter is defined in the `com.waveset.adapter.MSSQLServerResourceAdapter` class.

This adapter supports the following versions of Microsoft SQL Server:

- 2000

Use this adapter to manage multiple databases on the SQL server. Logins can be managed to the server itself as well as the managed databases.

If you have a custom SQL table, see *Database Table* on page 1-84 for information about using the Resource Adapter Wizard to create a custom Microsoft SQL table resource.

Resource Configuration Notes

None

Identity Manager Installation Notes

The Microsoft SQL Server resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. To add this resource to the Identity Manager resources list, you must add the following value in the Custom Resources section of the Configure Managed Resources page.

   ```
   com.waveset.adapter.MSSQLServerResourceAdapter
   ```

2. For SQL Server 2000, download the latest “SQL Server 2000 Driver for JDBC”. Copy the following jar files from the `Program Files\2000 Microsoft SQL Server 2000 Driver for JDBC\lib` directory to the *`InstallDir`*`\idm\WEB-INF\lib` directory.

   - `msbase.jar`
   - `mssqlserver.jar`
   - `msutil.jar`

Usage Notes

You can use two types of authentication with SQL Server:

- **Windows authentication**. SQL Server relies on Windows for all authentication and security mechanisms. When a user access SQL Server, it obtains the user and password information from the user's network security attributes. If the user has been granted access to SQL Server from within Windows, the user is logged in to SQL Server automatically. Account IDs passed in to the adapter must be in the form of *Domain\accoundID*. Pass-through authentication is not supported for Windows authentication.
- **Mixed mode authentication**. In this scenario, both Windows authentication and SQL Server authentication are enabled. When a user connects with a specified login name and password from a non-trusted connection, SQL Server performs the authentication itself by checking to see if a SQL Server login account has been set up and if the specified password matches the one previously recorded. If SQL Server does not have a login account set, authentication fails and the user receives an error message.

The SQL Server resource adapter uses the following system procedures to manage user accounts:

- sp_addlogin, sp_droplogin
- sp_addrole
- sp_addrolemember, sp_droprolemember
- sp_addsrvrolemember, sp_dropsrvrolemember
- sp_grantdbaccess
- sp_helplogins
- sp_helprole
- sp_helpuser
- sp_helpsrvrolemember
- sp_password
- sp_revokedbaccess

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses JDBC over SSL to communicate with SQL Server.

Required Administrative Privileges

The following table indicates who can execute the system procedures:

System Procedure	Permissions Required
sp_addlogin	Members of the **sysadmin** and **securityadmin** fixed server roles.
sp_addrole	Members of the **sysadmin** fixed server role, and the **db_securityadmin** and **db_owner** fixed database roles.
sp_addrolemember	Members of the **sysadmin** fixed server role and the **db_owner** fixed **database** role can execute `sp_addrolemember` to add a member to fixed database roles. Role owners can execute `sp_addrolemember` to add a member to any SQL Server role they own. Members of the **db_securityadmin** fixed database role can add users to any user-defined role.
sp_addsvrrolemember	Members of the **sysadmin** fixed server role.
sp_droplogin	Members of the **sysadmin** and **securityadmin** fixed server roles.
sp_droprolemember	Only members of the **sysadmin** fixed server role, the **db_owner** and **db_securityadmin** fixed database roles can execute `sp_droprolemember`. Only a member of the **db_owner** fixed database role can remove users from a fixed database role.
sp_dropsvrrolemember	Members of the **sysadmin** fixed server role.
sp_grantdbaccess	Members of the **sysadmin** fixed server role, the **db_accessadmin** and **db_owner** fixed database roles.
sp_helplogins	Members of the **sysadmin** and **securityadmin** fixed server roles.
sp_helprole	Execute permissions default to the **public** role.

System Procedure	Permissions Required
sp_helpsrvrolemember	Execute permissions default to the **public** role.
sp_helpuser	Execute permissions default to the **public** role.
sp_password	Execute permissions default to the **public** role for a user changing the password for his or her own login. Only members of the **sysadmin** role can change the password for another user's login.
sp_revokedbaccess	Members of the **sysadmin** fixed server role, and the **db_accessadmin** and **db_owner** fixed database roles

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	No
Pass-through authentication	• Mixed mode authentication: Yes • Windows authentication: No
Before/after actions	No
Data loading methods	• Import directly from resource • Reconcile with resource

Account Attributes

The following table lists the default account attributes (all strings).

Identity Manager User Attribute	Resource User Attribute	Description
domain	IGNORE_ATTR	The domain the user belongs to.
defaultDB	defaultDB	The user’s default database.
serverRoles	serverRoles	The database roles the user is a member of.

Because multiple databases can be managed, the Identity Manager administrator must add account attributes for each database to be managed. These attributes must include the database name as part of the attribute name in order to differentiate them from attributes for other managed databases:

Identity Manager User Attribute	Data Type	Description
userName*DBName*	String	The user name of the account on the database. Setting a userName for a database will grant access to the database for the account, and clearing the userName for a database will remove access.
roles*DBName*	String	The roles for the account on the database.

Resource Object Management

None

Identity Template

```
$domain$ $accountId$
```

Sample Forms

```
MSSQLServerUserForm.xml
```

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.MSSQLServerResourceAdapter`
- `com.waveset.adapter.JdbcResourceAdapter`

MySQL

The MySQL resource adapter is defined in the `com.waveset.adapter.MySQLResourceAdapter` class. This adapter supports the following versions of MySQL:

- 4.1

Use this adapter to support user accounts for logging into MySQL. If you have a custom table, see *Database Table* on page 1-84 for information about using the Resource Adapter Wizard to create a custom MySQL table resource.

Resource Configuration Notes

None

Identity Manager Installation Notes

The MySQL resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. To add this resource to the Identity Manager resources list, you must add the following value in the Custom Resources section of the Configure Managed Resources page.

   `com.waveset.adapter.MySQLResourceAdapter`
2. Go to `http://www.mysql.com/downloads/api-jdbc-stable.html` and download the latest version of the Connector/J 3.0 JDBC driver.
3. Unzip the downloaded file.
4. Copy the `mysqlconnector-java-3.0.`*`x`*`-stable-bin.jar` file to the *`InstallDir`*`\idm\WEB-INF\lib` directory.

Usage Notes

Identity Manager creates a new user based on the account properties of the user specified in the User Model resource parameter. If the User Model parameter is left blank, then new users will be granted default MySQL privileges. The access host would be set to %, indicating the user can access the database from any host.

The MySQL resource adapter can update MySQL user passwords only.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses JDBC over SSL to communicate with MySQL.

Required Administrative Privileges

You must be the MySQL root user or have GRANT privilege to create a user. Deleting a user requires the REVOKE privilege.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	No
Rename account	No
Pass-through authentication	No
Before/after actions	No
Data loading methods	

Account Attributes

None

Resource Object Management

None

Identity Template

```
$accountId$
```

Sample Forms

None

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

`com.waveset.adapter.MySQLResourceAdapter`

Natural

The Natural resource adapter is defined in the `com.waveset.adapter.NaturalResourceAdapter` class.

Resource Configuration Notes

None

Identity Manager Installation Notes

The Natural resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. To add the Natural resource to the Identity Manager resources list, you must add the following value in the Custom Resources section of the Configure Managed Resources page.

   `com.waveset.adapter.NaturalResourceAdapter`

2. The Identity Manager mainframe adapters use the IBM Host Access Class Library (HACL) to connect to the mainframe. The HACL is available in IBM Websphere Host On-Demand (HOD). The recommended jar containing HACL is `habeans.jar` and is installed with the HOD Toolkit (or Host Access Toolkit) that comes with HOD. The supported versions of HACL are in HOD V7.0, V8.0, and V9.0.

   However, if the toolkit installation is not available, the HOD installation contains the following jars that can be used in place of the `habeans.jar`:

   - `habase.jar`
   - `hacp.jar`
   - `ha3270.jar`
   - `hassl.jar`
   - `hodbase.jar`

   Copy the `habeans.jar` file or all of its substitutes into the `WEB-INF/lib` directory of your Identity Manager installation. See http://www.ibm.com/software/webservers/hostondemand/ for more information.

Usage Notes

This section describes dependencies and limitations related to using the Natural resource adapter and provides instructions for configuring SSL for the adapter. This information is organized into the following sections:

- *Administrators*
- *Connecting the Adapter to a Telnet/TN3270 Server Using SSL or TLS.*
- *Generating a PKCS #12 File*
- *Troubleshooting*

Administrators

Host resource adapters *do not* enforce maximum connections for an affinity administrator across multiple host resources connecting to the same host. Instead, the adapter enforces maximum connections for affinity administrators within each host resource.

If you have multiple host resources managing the same system, and they are currently configured to use the same administrator accounts, you might have to update those resources to ensure that the same administrator is not trying to perform multiple actions on the resource simultaneously.

Connecting the Adapter to a Telnet/TN3270 Server Using SSL or TLS.

Use the following steps to connect Scripted Host resource adapters to a Telnet/TN3270 server using SSL/TLS.

1. Obtain the Telnet/TN3270 server's certificate in the PKCS #12 file format. Use `hod` as the password for this file. Consult your server's documentation on how to export the server's certificate. The procedure "Generating a PKCS #12 File" below for some general guidelines.
2. Create a `CustomizedCAs.class` file from the PKCS #12 file. If you are using a recent version of HOD, use the following command to do this.

   ```
   ..\hod_jre\jre\bin\java -cp ../lib/ssliteV2.zip;../lib/sm.zip
   com.ibm.eNetwork.HOD.convert.CVT2SSLIGHT CustomizedCAs.p12 hod
   CustomizedCAs.class
   ```

3. Place the `CustomizedCAs.class` file somewhere in the Identity Manager server's classpath, such as `$WSHOME/WEB-INF/classes`.

4. If a resource attribute named **`Session Properties`** does not already exist for the resource, then use the BPE or debug pages to add the attribute to the resource object. Add the following definition in the `<ResourceAttributes>` section:

   ```
   <ResourceAttribute name='Session Properties' displayName='Session
   Properties' description='Session Properties' multi='true'>
   </ResourceAttribute>
   ```

5. Go to the Resource Parameters page for the resource and add the following values to the **`Session Properties`** resource attribute:

   ```
   SESSION_SSL
   true
   ```

Generating a PKCS #12 File

The following procedure provides a general description of generating a PKCS #12 file when using the Host OnDemand (HOD) Redirector using SSL/TLS. Refer to the HOD documentation for detailed information about performing this task.

6. Create a new `HODServerKeyDb.kdb` file using the IBM Certificate Management tool. As part of that file, create a new self-signed certificate as the default private certificate.

   If you get a message that is similar to “error adding key to the certificate database” when you are creating the `HODServerKeyDb.kdb` file, one or more of the Trusted CA certificates may be expired. Check the IBM website to obtain up-to-date certificates.

7. Export that private certificate as Base64 ASCII into a `cert.arm` file.
8. Create a new PKCS #12 file named `CustomizedCAs.p12` with the IBM Certificate Management tool by adding the exported certificate from the `cert.arm` file to the Signer Certificates. Use `hod` as the password for this file.

Troubleshooting

You can enable tracing of the HACL by adding the following to the Session Properties resource attribute:

```
SESSION_TRACE
ECLSession=3 ECLPS=3 ECLCommEvent=3 ECLErr=3 DataStream=3 Transport=3
ECLPSEvent=3
```

Note The trace parameters should be listed without any new line characters. It is acceptable if the parameters wrap in the text box.

The Telnet/TN3270 server should have logs that may help as well.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses Secure TN3270 to communicate with Natural.

Required Administrative Privileges

None

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	
Pass-through authentication	
Before/after actions	Yes
Data loading methods	Import from resource

Account Attributes

The following table provides information about Natural account attributes.

Resource User Attribute	Data Type	Description
PASSWORD	String	Account's password
GROUPS	String	List of groups to which the user is assigned
USERID	String	The account name
NAME	String	The user's name

Resource User Attribute	Data Type	Description
COPYUSER	String	Name of the account to use as a template when you create an account. You must specify this attribute when creating an account.
COPYLINKS	Boolean	Indicates whether to copy the links specified in COPYUSER when creating the account. Default is false.
DEFAULT_LIBRARY	String	Name of the account's default library.

Resource Object Management

Not supported

Identity Template

```
$accountId$
```

Sample Forms

None

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

```
com.waveset.adapter.NaturalResourceAdapter
```

NetWare NDS

Identity Manager provides adapters for supporting the following Novell products:

- NetWare 5.1 SP6 or 6.0 with eDirectory 8.7.1
- Novell SecretStore 3.0

The following table summarizes the attributes of the Novell adapters:

GUI Name	Class Name
NetWare NDS	`com.waveset.adapter.NDSResourceAdapter`
NetWare NDS with SecretStore	`com.waveset.adapter.NDSSecretStoreResourceAdapter`

Note The NetWare NDS Active Sync adapter (`com.waveset.adapter.NDSActiveSyncResourceAdapter` has been deprecated as of Identity Manager 5.0 SP1. All features in this adapter are now in the NetWare NDS adapter. Although existing instances of the NetWare NDS Active Sync adapter will still function, new instances of these can no longer be created.

Resource Configuration Notes

This section provides instructions for configuring NetWare NDS resources for use with Identity Manager, including:

- Instructions for installing the Gateway location
- Instructions for configuring the Gateway service account
- Instructions for configuring a SecretStore certificate

Gateway Location

Install the Sun Identity Manager Gateway on any NDS client that can connect to the domain to be managed. Multiple gateways should be installed if pass-through authentication is enabled.

Gateway Service Account

By default, the Gateway service runs as the local System account. This is configurable through the Services MMC Snap-in.

If you run the Gateway as an account other than Local System, then the Gateway service account requires the “Act As Operating System” and “Bypass Traverse Checking” user rights. It uses these rights for pass-through authentication and for changing and resetting passwords in certain situations.

SecretStore Certificates

To support SecretStore, a SSL certificate must be exported from the NDS system to the Identity Manager application server.

One possible way to obtain this certificate is to use ConsoleOne to export the public key. To do this, start ConsoleOne and navigate to the SSL CertificateDNS object. On the Properties dialog of the SSL CertificateDNS object, select Public Key Certificate from the Certificates tab. Press the Export button to begin the process of exporting the certificate. You do not need to export the private key. Store the file in DER format.

Copy the DER file to the Identity Manager application server. Then add the certificate to the `jdk\jre\lib\security\cacerts` keyfile using keytool or other certificate management tool. The keytool utility is shipped with the Java SDK. Refer to the Java documentation for more information about the keytool utility.

Identity Manager Installation Notes

The NetWare NDS adapter does not require any additional installation procedures.

To add the NDS SecretStore resource to the resources list, perform the following procedure:

1. Add the following value in the Custom Resources section of the Configure Managed Resources page.

   `com.waveset.adapter.NDSSecretStoreResourceAdapter`

2. Copy the `jsso.jar` file to the *`InstallDir`*`\idm\WEB-INF\lib` directory. The jsso.jar file can be obtained from one of the following locations where the NDS client with either Novell SecretStore or Novell SecureLogin is installed:

   - *`NovellInstallDir`*`\ConsoleOne\version\lib\SecretStore`
   - *`NovellInstallDir`*`\ConsoleOne\version\lib\security`

Usage Notes

This section provides information related to using the NetWare NDS resource adapter, which is organized into the following sections:

- *Miscellaneous*
- *Pass-Through Authentication*
- *Managing NDS Users in GroupWise*
- *SecretStore and the Identity Manager System Configuration Object*

Miscellaneous

- The NetWare NDS adapter in Active Sync mode does not detect account deletions. As a result, you must reconcile to detect these deletions.
- The NDS adapters support template values, including user DS and FS rights, Home Directory rights, and Trustees of New Object.
- To avoid display problems on the Resources page, set the “Identity Manager User Name Attribute” parameter to `cn`.
- NDS uses periods instead of commas to mark segments of a name. Identity Manager will return an error message if you specify commas.
- To configure an NDS resource so that you can create a user's home directory, you must add two attributes to the account attributes:

  Home Directory — String. The format of this attribute is

  `VolumeDN``#`*`NameSpaceType`*`#`*`DirectoryPath`*.

  For example,

  `SERVER_SYS.MYORG#0#\Homes\bob_smith`.

  The *`NameSpaceType`* is one of:

  - 0 — DOS name space
  - 1 — Macintosh name space
  - 2 — UNIX or NFS name space
  - 3 — FTAM name space
  - 4 — OS/2, Windows 95, or Windows NT name space

  Create Home Directory — Boolean. This attribute acts as a flag to indicate whether the actual directory should be created. The directory is created when this flag is set to true.

- If you encounter the following error on the NDS adapter,

  `NWDSAddSecurityEquiv: 0xFFFFFD9B (-613): ERR_SYNTAX_VIOLATION`

  You might need to increase the following registry keys in HKEY_LOCAL_MACHINE\Software\Waveset\Lighthouse\Gateway

  - `nds_method_retry_count` (The default is 10.)
  - `nds_method_retry_sleep_interval` (The default is 1000 milliseconds.)
- The `HKEY_LOCAL_MACHINE\Software\Waveset\Lighthouse\Gateway\ExclusiveNDSContext` registry key specifies whether the NDS context is multi-threaded. The default value of 0 indicates a multi-threaded context. Set the value to 1 for a single-threaded context.
- The NetWare API is not compatible with the `searchFilter` option of the `getResourceObjects` FormUtil method.
- If the account that connects to the NDS resource is restricted by the NDS `loginMaximumSimultaneous` attribute, then set the **Connection Limit** resource parameter to a value less than or equal to the value specified by loginMaximumSimultaneous.

Pass-Through Authentication

Due to restrictions in the way NDS handles authentication, implementing pass-through authentication on NDS requires that you create a separate resource that is devoted to this purpose. If the same client host and gateway is used to perform pass-through authentication and provisioning, an ERR_DIFF_OBJ_ALREADY_AUTHED error message might be returned.

Another Sun Identity Manager Gateway must be installed on the client host that connects to the resource that will be used for pass-through authentication. (You cannot simply create a different resource object in Identity Manager that points to the same NDS client.) The Admin **`User DN`** and **`Base Context`** fields should be the same on both resources.

Note The pass-through authentication resource must NOT be reconciled or otherwise contain user accounts. The standard resource will continue to be used for provisioning and other administrative tasks.

Use the following procedure to configure Identity Manager to enable pass-through authentication on NDS. For this example, the provisioning resource will be named `NDS_Resource`, and the resource for pass-through authentication will be named `NDS_Passthrough`.

1. On the NDS_Resource system, make sure the value of the registry key HKEY_LOCAL_MACHINE\Software\Waveset\Lighthouse\Gateway\ExclusiveNDSContext is set to the default value of 0 (multi-threaded).

   On NDS_Passthrough, set the value of ExclusiveNDSContext to 1 (single-threaded).

2. Create a new login module group that contains a separate login module for each resource. Set the **Login success requirement** field to sufficient for both login modules. Then set the order of the login modules so that the module for NDS_Passthrough is listed before the module for NDS_Resource.

3. Add the common resources attribute to the System Configuration object. This attribute indicates the users defined on listed systems have resources have synchronized user IDs and passwords.

   The following example adds the two resources to the NDS Group

```
<Attribute name='common resources'>
   <Object>
      <Attribute name='NDS Group'>
         <List>
            <String>NDS_Resource</String>
            <String>NDS_Passthrough</String>
         </List>
      </Attribute>
   </Object>
</Attribute>
```

   NDS_Resource is listed first because it is the resource through which user accounts are managed.

All provisioning functions will be handled by NDS_Resource, and all pass-through authentication calls will go through NDS_Passthrough.

Managing NDS Users in GroupWise

When integration with GroupWise is enabled, the NDS adapter can manage the GroupWise attributes of NDS users. The NDS adapter supports adding and removing NDS users from a GroupWise Post Office. It also retrieves or modifies other GroupWise account attribute, including AccountID, GatewayAccess, and DistributionLists.

Enabling GroupWise Integration

To activate the integration with GroupWise, you must define a value in the GroupWise Domain DN resource attribute. This value specifies the DN of the GroupWise domain which will managed. An example value for this attribute is

```
CN=gw_dom.ou=GroupWise.o=MyCorp
```

The NDS Tree resource attribute defines the NDS tree under which the GroupWise domain is expected to reside is. That is, the GroupWise domain must be in the same tree as the NDS users managed by the adapter.

Managing a NDS User's GroupWise Post Office

The account attribute GW_PostOffice represents the GroupWise Post Office.

To add an NDS user into a GroupWise Post Office, set the GW_PostOffice account attribute to the name of an existing Post Office that is associated with the GroupWise domain.

To move an NDS user to a different GroupWise Post Office, set the GW_PostOffice account attribute to the name of the new Post Office that is associated with the GroupWise domain.

To remove an NDS user from its Post Office, set the GW_PostOffice account attribute to the same value as the GroupWise Delete Pattern resource attribute. The default value for GroupWise Delete Pattern resource attribute is *TRASH*.

SecretStore and the Identity Manager System Configuration Object

By default, you cannot use the NetWare NDS with SecretStore adapter to manage resource objects. To enable this functionality, you must edit the System Configuration Object.

Under the lines that read:

```
<!-- form mappings -->
   <Attribute name='form'>
      <Object>
```

add the following:

```
<!-- NetWare NDS with SecretStore -->
<Attribute name='NetWare NDS with SecretStore Create Group Form'
    value='NetWare NDS Create Group Form'/>
<Attribute name='NetWare NDS with SecretStore Update Group Form'
    value='NetWare NDS Update Group Form'/>
<Attribute name='NetWare NDS with SecretStore Create Organization
Form'
    value='NetWare NDS Create Organization Form'/>
<Attribute name='NetWare NDS with SecretStore Update Organization
Form'
    value='NetWare NDS Update Organization Form'/>
<Attribute name='NetWare NDS with SecretStore Create Organizational
Unit Form'
```

```
    value='NetWare NDS Create Organizational Unit Form'/>
<Attribute name='NetWare NDS with SecretStore Update Organizational
Unit Form'
    value='NetWare NDS Update Organizational Unit Form'/>
<Attribute name='NetWare NDS with SecretStore Create User Form'
    value='NetWare NDS Create User Form'/>
<Attribute name='NetWare NDS with SecretStore Update User Form'
    value='NetWare NDS Update User Form'/>
```

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

The recommended approach for connecting to a NetWare NDS resource is with the Gateway service. The Gateway service uses a TCP/IP socket connection (3 DES) for exchanging password information on the network.

You can also use standard LDAP or LDAP over SSLP to connect to the NetWare NDS server. In this scenario, use the LDAP resource adapter.

Required Administrative Privileges

The Identity Manager administrator must have the proper NDS rights to create a NetWare user. By default, a NetWare administrator has all rights in the Directory and in the NetWare file system.

To perform password administration, an NDS administrator must have Compare, Read, and Write rights on the following properties:

- Group Membership
- Locked By Intruder
- Login Intruder Attempts
- Login Intruder Reset Time
- Password Management

The Identity Manager administrator account performing functions with NDS SecretStore must be defined as a SecretStore administrator.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	Yes
Pass-through authentication	Yes
Before/after actions	No
Data loading methods	• Import directly from resource • Reconcile with resource • Active Sync

Account Attributes

This section provides information about the NetWare NDS account attribute support including:

- *Attribute Syntax Support*
- *Account Attribute Support*

The syntax (or type) of an attribute usually determines whether the attribute is supported. In general, Identity Manager supports boolean, string, and integer syntaxes.

The values for attributes with SYN_CI_LIST (such as Language) and SYN_PO_ADDRESS (such as Postal Address) syntaxes should be a list of strings separated by $. The values for SYN_OCTET_STRING attributes should be Base 64 encoded strings of the bytes in the octet stream.

Attribute Syntax Support

Information about attribute syntax support is provided in the following Supported Syntaxes and Unsupported Syntaxes sections.

Supported Syntaxes

The following table provides information about supported attribute syntaxes:

NDS Syntax	Attr Type	Object ID	Syntax ID
Boolean	Boolean	1.3.6.1.4.1.1466.115.121.1.7	SYN_BOOLEAN
Case Exact String	String	1.3.6.1.4.1.1466.115.121.1.26 2.16.840.1.113719.1.1.5.1.2	SYN_CE_STRING
Case Ignore List	String	2.16.840.1.113719.1.1.5.1.6	SYN_CI_LIST
Case Ignore String	String	1.3.6.1.4.1.1466.115.121.1.15	SYN_CI_STRING
Class Name	String	1.3.6.1.4.1.1466.115.121.1.38	SYN_CLASS_NAME
Counter	Int	2.16.840.1.113719.1.1.5.1.22	SYN_COUNTER
Distinguished Name	String	1.3.6.1.4.1.1466.115.121.1.12	SYN_DIST_NAME
Fax Number	String	1.3.6.1.4.1.1466.115.121.1.22	SYN_FAX_NUMBER
Integer	Int	1.3.6.1.4.1.1466.115.121.1.27	SYN_INTEGER
Interval	Int	1.3.6.1.4.1.1466.115.121.1.27	SYN_INTERVAL
Numeric String	String	1.3.6.1.4.1.1466.115.121.1.36	SYN_NU_STRING
Octet String	String	1.3.6.1.4.1.1466.115.121.1.40	SYN_OCTET_STRING
Path	String	2.16.840.1.113719.1.1.5.1.15	SYN_PATH
Postal Address	String	1.3.6.1.4.1.1466.115.121.1.41	SYN_PO_ADDRESS
Printable String	String	1.3.6.1.4.1.1466.115.121.1.44	SYN_PR_STRING
Stream	String	1.3.6.1.4.1.1466.115.121.1.5	SYN_STREAM
Telephone Number	String	1.3.6.1.4.1.1466.115.121.1.50	SYN_TEL_NUMBER
Time	Int	1.3.6.1.4.1.1466.115.121.1.24	SYN_TIME

Unsupported Syntaxes

The following table provides information about unsupported syntaxes:

NDS Syntax	Object ID	Syntax ID
Back Link	2.16.840.1.113719.1.1.5.1.23	SYN_BACK_LINK
EMail Address	2.16.840.1.113719.1.1.5.1.14	SYN_EMAIL_ADDRESS
Hold	2.16.840.1.113719.1.1.5.1.26	SYN_HOLD
Net Address	2.16.840.1.113719.1.1.5.1.12	SYN_NET_ADDRESS
Object ACL	2.16.840.1.113719.1.1.5.1.17	SYN_OBJECT_ACL
Octet List	2.16.840.1.113719.1.1.5.1.13	SYN_OCTET_LIST
Replica Pointer	2.16.840.1.113719.1.1.5.1.16	SYN_REPLICA_POINTER
Timestamp	2.16.840.1.113719.1.1.5.1.19	SYN_TIMESTAMP
Typed Name	2.16.840.1.113719.1.1.5.1.25	SYN_TYPED_NAME
Unknown	2.16.840.1.113719.1.1.5.1.0	SYN_UNKNOWN

Account Attribute Support

Information about attribute support is provided in the following Supported Account Attributes and Unsupported Account Attributes sections.

Supported Account Attributes

The following attributes are displayed on the Account Attributes page for the NDS resource adapters.

Resource User Attribute	NDS Syntax	Attribute Type	Description
Create Home Directory	Boolean	Boolean	Indicates whether to create a home directory for the user. The Home Directory Parameter must be set.
Description	Case Ignore String	String	Text that describes the user.
Facsimile Telephone Number	Facsimile Telephone Number	String	The telephone number and, optionally, the parameters for a facsimile terminal associated with a user.

Resource User Attribute	NDS Syntax	Attribute Type	Description
Full Name	Case Ignore String	String	The full name of a user.
Generational Qualifier	Case Ignore String	String	Indicates a person's generation. For example, Jr. or II.
Given Name	Case Ignore String	String	The given (first) name of a user.
Group Membership	Distinguished Name	String	A list of the groups to which the user belongs.
GW_AccountID	Not applicable	String	Account ID specified in the User Information field for GroupWise accounting.
GW_DistributionLists	Not applicable	String	Distribution lists of which the user is a member. The values must be valid distribution list distinguished names (DNs).
GW_GatewayAccess	Not applicable	String	Restricts access to GroupWise gateways. See your gateway documentation to determine if this field is applicable.
GW_Name	Not applicable	String	The GroupWise mailbox name.
GW_PostOffice	Not applicable	String	The name of an existing Post Office that is associated with the GroupWise domain.
Home Directory	Path	String	The location of a client's current working directory. See the "Usage Notes" for more information.
Initials	Case Ignore String	String	The user's middle initial.
Internet EMail Address	Case Ignore String	String	Specifies an Internet e-mail address.
L	Case Ignore String	String	A physical or geographical location.
Locked By Intruder	Boolean	Boolean	Indicates an account has been locked due to excessive failing login attempts.
Login Grace Limit	Integer	Int	The total number of times an old password can be used (after the old password has expired) to access the account.

Resource User Attribute	NDS Syntax	Attribute Type	Description
Login Maximum Simultaneous	Integer	Int	The number of authenticated login sessions a user can initiate simultaneously.
ou	Case Ignore String	String	The name of an organizational unit.
Password Allow Change	Boolean	Boolean	Determines whether the person logged in under an account can change the password for that account.
Password Expiration Interval	Interval	Int	The time interval a password can remain active.
Password Required	Boolean	Boolean	Establishes that a password is required for the user to log in.
Password Unique Required	Boolean	Boolean	Establishes that when a user password is changed, it must be different from those in the Passwords Used attribute.
Surname	Case Ignore String	String	Required. the name an individual inherits from a parent (or assumes by marriage) and by which the individual is commonly known.
Telephone Number	Telephone Number	String	The user's telephone number.
Title	Case Ignore String	String	The designated position or function of a user within an organization.
userPassword	N/A	Encrypted	Required. The user's password.

The following table lists additional supported attributes that are defined in the NDS User object class.

Resource User Attribute	NDS Syntax	Attribute Type	Description
Account Balance	Counter	Int	The amount of credit the user has to buy network services, such as connection time.
Allow Unlimited Credit	Boolean	Boolean	Indicates whether the user account has unlimited credit for using network services.
audio	Octet String	String	An audio file in binary format.
businessCategory	Case Ignore String	String	Describes the kind of business performed by an organization.
carLicense	Case Ignore String	String	Vehicle license or registration plate
departmentNumber	Case Ignore String	String	Identifies a department within an organization
displayName	Case Ignore String	String	The name to be displayed on admin screens.
Employee ID	Case Ignore String	String	Numerically identifies an employee within an organization
employeeType	Case Ignore String	String	Type of employment, such as Employee or Contractor
Entrust:User	Case Exact String	String	Specifies an Entrust user.
Higher Privileges	Distinguished Name	String	An alternative set of security access privileges.
homePhone	Telephone Number	String	The user's home telephone number.
homePostalAddress	Postal Address	String	The user's home address.
jpegPhoto	Octet String	String	A JPEG file containing a photo of the user
labeledUri	Case Ignore String	String	The user's Uniform Resource Identifier (URI).
Language	Case Ignore List	String	An ordered list of languages
Last Login Time	Time	String	The login time of the session previous to the current session.

Resource User Attribute	NDS Syntax	Attribute Type	Description
ldapPhoto	Octet String	String	A photo of the object in binary format.
Login Allowed Time Map	Octet String	String	The allowed login time periods for an account for each day of the week to a precision of one-half hour.
Login Disabled	Boolean	Int	Informs the user that the account has been disabled.
Login Expiration Time	Time	String	A date and time after which a client cannot log in.
Login Grace Remaining	Counter	Int	The number of grace logins are left before the account is locked.
Login Intruder Attempts	Counter	Int	The number of failed login attempts that have occurred in the current interval.
Login Intruder Reset Time	Time	String	The next time that the intruder attempts variable will be reset.
Login Script	Stream	String	The user's login script.
Login Time	Time	String	The login time of the current session.
manager	Distinguished Name	String	The user's supervisor.
Minimum Account Balance	Integer	Int	The minimum amount of credit (or money) a user must have in his or her account to access specified services.
mobile	Telephone Number	String	The user's cell phone number.
NDSPKI:Keystore	Octet String	String	Contains wrapped private keys.
NRD:Registry Data	Stream	String	NetWare Registry Database
NRD:Registry Index	Stream	String	The index of the NetWare Registry Database
pager	Telephone Number	String	The user's pager number.

Resource User Attribute	NDS Syntax	Attribute Type	Description
Password Expiration Time	Time	String	Specifies when the password will expire.
preferredLanguage	Case Ignore String	String	The user's preference for written or spoken language.
Print Job Configuration	Stream	String	Contains information on the specified print job configuration.
Printer Control	Stream	String	The NDS counterpart of the DOS printer definition file, NET$PRN.DAT.
Profile	Distinguished Name	String	The login profile to be used if the user doesn't specify one at login time.
Profile Membership	Distinguished Name	String	A list of profiles that the object can use.
Public Key	Octet String	String	A certified RSA public key
roomNumber	Case Ignore String	String	The user's office or room number.
secretary	Distinguished Name	String	The user's administrative assistant.
Security Equals	Distinguished Name	String	Specifies group membership and security equivalences of a user.
Security Flags	Integer	Int	The NCP Packet Signature level of the object.
Timezone	Octet String	String	The time zone offset for a user.
UID (User ID)	Integer	Int	A unique user ID for use by UNIX clients.
userCertificate	Octet String	String	A certificate for certificate management.
userSMIMECertificate	Octet String	String	The user's certificate for Netscape Communicator for S/MIME.
x500UniqueIdentifier	Octet String	String	An identifier to use in distinguishing between users when a DN has been reused.

Unsupported Account Attributes

The following account attributes are not supported:

- Login Intruder Address
- Login Script
- Network Address
- Network Address Restriction
- NRD:Registry Data
- NRD:Registry Index
- Passwords Used
- Print Job Configuration
- Printer Control
- Private Key
- Server Holds
- Type Creator Map

Resource Object Management

Identity Manager supports the following NetWare NDS objects by default. Any string-, integer-, or boolean-based attributes can also be managed.

Resource Object	Features Supported	Attributes Managed
Group	Create, update, delete	L, OU, O, CN, Description, Member, Owner
Organizational Unit	Create, update, delete	OU, Description, L, Facsimile Telephone Number, Telephone Number
Organization	Create, update, delete	dn, O, Description, L, Facsimile Telephone Number, Telephone Number

Identity Template

The default identity template is

```
CN=$accountId$.O=MYORG
```

You must replace the default template with a valid value.

Sample Forms

This section lists the sample forms that are available for this resource adapter.

Built-In

These forms are built into Identity Manager:

- NDS Group Create Form
- NDS Group Update Form
- NDS Create Organizational Unit Form
- NDS Update Organizational Unit Form
- NDS Create Organization Form
- NDS Update Organization Form

Also Available

The `NDSUserForm.xml` form is also available.

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.NDSResourceAdapter`
- `com.waveset.adapter.NDSSecretStoreResourceAdapter`
- `com.waveset.adapter.AgentResourceAdapter`

To make access to NDS through the Sun Identity Manager Gateway single-threaded or serialized, set the following registry key and value in the `HKEY_LOCAL_MACHINE\SOFTWARE\Waveset\Lighthouse\Gateway` node on the Gateway machine:

Name	Type	Data
ExclusiveNDSContext	REG_DWORD	• **0**: Disables this feature. The context is multi-threaded. • **1**: The context is single-threaded.

Oracle and Oracle ERP

Identity Manager provides resource adapters for supporting the following Oracle products:

- Oracle 9*i*, 10g
- Oracle Financials on Oracle Applications 11.5.9, 11.5.10

The following table summarizes the attributes of the Oracle adapters:

GUI Name	Class Name
Oracle	`com.waveset.adapter.OracleResourceAdapter`
Oracle ERP	`com.waveset.adapter.OracleERPResourceAdapter`

Use this adapter to support user accounts for logging into Oracle or Oracle Financials. If you have a custom Oracle table, see *Database Table* on page 1-84 for information about using the Resource Adapter Wizard to create a custom Oracle table resource.

Resource Configuration Notes

None

Identity Manager Installation Notes

The Oracle and Oracle ERP resource adapters are custom adapters. You must perform the following steps to complete the installation process:

1. To add an Oracle resource to the Identity Manager resources list, you must add the one of the following values in the Custom Resources section of the Configure Managed Resources page.

   ```
   com.waveset.adapter.OracleResourceAdapter
   com.waveset.adapter.OracleERPResourceAdapter
   ```

2. If you are using the JDBC thin driver:
   a. Copy the `oracle\jdbc\lib\classes12.zip` file from the installation media to the *`InstallDir`*`\idm\WEB-INF\lib` directory.
   b. Rename the file to `oraclejdbc.jar`.
3. If you are using a different driver, specify the driver and connection URL on the Resource Parameters page.

The Oracle ERP adapter supports version 11.5.9 without further modification; however, the following additional changes are required to support version 11.5.10:

1. Delete the responsibilities account attribute from the schema map and add the `directResponsibilities` and `indirectResponsibilities` attributes.
2. Copy the `OracleERPUserForm.xml` file and comment out the section labeled 11.5.9 and uncomment the 11.5.10 section. Then import your copy of the sample user form.

Note Remember to replace the OracleERP Resource string with site-specific ERP resource names in fields calling `listResourceObjects`.

Usage Notes

This section describes dependencies and limitations related to using the Oracle and Oracle ERP resource adapters.

Oracle

Information about user types and cascade deletes are provided in the following sections.

User Types

The Oracle database permits the following types of users:

- Local. Local users are fully managed by Oracle and require a password. Oracle manages these passwords as well. Therefore, the user name and password must fully comply with the standards set within the application.
- External. External users must be authenticated by the operating system or a third-party application. Oracle relies on the login authentication to ensure that a specific operating system user has access to a specific database user.
- Global. Global users must be authenticated by a directory service, such as LDAP or Active Directory. The user's name must be specified as a full distinguished name (DN) or as a null string. If a null string is used, the directory service will map authenticated global users to the appropriate database features.

If you are managing external or global users, you should place the Oracle resource in a resource group that also includes the machine upon which it is installed or the directory service.

Cascade Deletes

The noCascade account attribute indicates whether to perform cascade drops when deleting users. By default, cascade drops are performed. To disable cascade drops:

1. Add an entry to `updateableAttributes` section of System Configuration Object:

```
<Attribute name='Delete'>
   <Object>
      <Attribute name='all'>
         <List>
            <String>noCascade</String>
         </List>
      </Attribute>
   </Object>
</Attribute>
```

2. Add a field to the deprovision form:

```
<Field name='resourceAccounts.currentResourceAccounts
[MyOracleResource].attributes.noCascade'>
   <Display class='Checkbox'>
      <Property name='title' value='Do NOT Cascade MyOracleResource
Delete'/>
      <Property name='alignment' value='left'/>
   </Display>
   <Disable>
      <isnull>
         <ref>resourceAccounts.currentResourceAccounts[MyOracleRes
ource]</ref>
      </isnull>
   </Disable>
</Field>
```

3. Add the `noCascade` account attribute to Oracle Resource schema.

Note If the user owns objects and the “do not cascade” option has been selected, Oracle will throw an error. The user will not be deleted.

4. Add a `noCascade` field to the user form so that the attribute can be disabled. For example:

```
<Field name='global.noCascade'>
   <Disable>
      <s>TRUE</s>
   </Disable>
</Field>
```

Oracle ERP

The following resource parameters are applicable for the Oracle ERP adapter.

Oracle Client Encryption Types

This parameter can contain a list of valid Oracle support encryption algorithm names, such as RC4_56 or RC4_128. If this list is empty, all algorithms supported by Oracle for that Oracle release will be available. The client/server will negotiate on which of these algorithms to use based on Oracle Client Encryption Level setting.

Note The Oracle Server must also be configured to support this type of encryption.

For a more details on the supported algorithms, refer to the *Oracle Advanced Security Administrator's Guide*. See SQLNET.ENCRYPTION_TYPES_CLIENT for a list of valid values for the thin JDBC client.

Oracle Client Encryption Level

This value determines the level of security that the server/client negotiates and enforces. The default value, if left blank, is ACCEPTED. The valid values are REJECTED, ACCEPTED, REQUESTED and REQUIRED. For more details on use of this parameter, refer to the *Oracle Advanced Security Administrator's Guide* and the SQLNET.ENCRYPTION_CLIENT values.

The Oracle Server will need to be configured also to support this type of encryption.

Oracle ERP Admin User Responsibility

This value determines the ERP Responsibility used by the Identity Manager Oracle ERP Admin user to call the ERP application initialization routine. A list of valid responsibilities can be found in the `fnd_responsibility_vl` table. Also refer to the ERP documentation for more information.

If the Identity Manager Oracle ERP Admin user has a valid ERP system account and has a responsibility that matches the value of this parameter, the Oracle session created during connection enables the users' actions to be audited using the Oracle ERP auditing mechanism. For example, the `created_by` and the `last_updated_by` fields of the `fnd_user` table objects will be updated correctly with the user ID of the Identity Manager Oracle ERP Admin user.

Adding Securing Attributes

The `securingAttrs` account attribute supports the Securing Attributes feature in Oracle Financials. To configure Securing Attributes from the Identity Manager Create User page, perform the following steps:

1. Select the **Add Securing Attribute** checkbox.
2. Enter a search pattern to narrow the choices of available attributes in the **Enter Securing Attribute Search Pattern** text box. Use the % character as a wild card. Then click the **Load Securing Attributes** button. This will load the attributes into the **Oracle Securing Attributes** select box.
3. Select an attribute from the drop-down menu, and it will be added to the Securing Attributes table.

   You may remove securing attributes by selecting the attribute to be removed from the table and clicking the **Remove Selected Securing Attribute** button.

Enabling Users

Enabling an Oracle ERP user requires the value of the `owner` attribute to be specified. The value `CUST` is used by default unless the value is specifically added to the Enable Form and sent through the enable view. The following example changes the default owner to MYOWNER:

```
<Field name='resourceAccounts.currentResourceAccounts[MyOracleERP].
attributes.owner' type='string'>
   <Display class='Text'>
      <Property name='title' value='Owner'/>
   </Display>
   <Default>
      <s>MYOWNER</s>
   </Default>
</Field>
```

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager can use one of the following drivers to communicate with the Oracle adapters:

- JDBC thin driver
- JDBC OCI driver
- Third-party drivers

Required Administrative Privileges

To create an Oracle user, the administrator must have CREATE USER, ALTER USER, and DROP USER system privileges.

For Oracle and Oracle Applications, administrators must have SELECT permissions on the following database views:

- DBA_PROFILES
- DBA_ROLE_PRIVS
- DBA_SYS_PRIVS
- DBA_TABLESPACES
- DBA_TS_QUOTAS
- DBA_USERS

Oracle ERP Permissions

Oracle Applications require access to the following tables and stored procedures.

Note The administrator must be able to run the select command for all tables. In addition, the administrator must be able to update the apps.fnd_user table.

Tables	Stored Procedures
apps.ak_attributes apps.ak_attributes_tl apps.ak_web_user_sec_attr_values apps.fnd_application apps.fnd_application_tl apps.fnd_application_vl apps.fnd_profile apps.fnd_responsibility apps.fnd_responsibility_vl apps.fnd_security_groups apps.fnd_security_groups_tl apps.fnd_security_groups_vl apps.fnd_user apps.fnd_user_resp_groups apps.icx_parameters	apps.app_exception.raise_exception apps.fnd_global.apps_initialize apps.fnd_global.user_id apps.fnd_message.get apps.fnd_message.get_token apps.fnd_message.set_name apps.fnd_message.set_token apps.fnd_profile.get apps.fnd_user_pkg.AddResp apps.fnd_user_pkg.CreateUser apps.fnd_user_pkg.DisableUser apps.fnd_user_pkg.DelResp apps.fnd_user_pkg.UpdateUser apps.fnd_user_pkg.user_synch apps.fnd_user_pkg.validatelogin apps.fnd_user_resp_groups_api.assignment_exists apps.fnd_user_resp_groups_api.insert_assignment apps.fnd_user_resp_groups_api.update_assignment apps.fnd_web_sec.change_password apps.fnd_web_soc.create_user apps.fnd_web_sec.validation_login apps.icx_user_sec_attr_pub.create_user_sec_attr apps.icx_user_sec_attr_pub.delete_user_sec_attr

Note The adapter might access additional tables and stored procedures. Refer to the Oracle Applications documentation for additional information.

Oracle states that the Oracle ERP system, including the `fnd_user_pkg` stored procedures, were designed to be used to administer the ORACLE ERP system as the APPS user. Oracle does NOT recommend creating an alternate administrative user. However, if you need to manage Oracle ERP with a user other than `APPS`, contact Oracle for guidance.

The alternate administrative user must be granted the same access as the `APPS` user has to all Oracle data, including tables, views, and stored procedures.

The user will also need synonyms set up so the user will have access to the tables that the `APPS` user has access to. If a different user is used and the appropriate grants and synonyms have not been created for the user, the following error might be encountered:

```
Error: ORA-00942: table or view does not exist
```

Add the appropriate grants and synonyms to correct the error.

A sample `SQL*Plus` script is can be found in `$WSHOME/sample/other/CreateLHERPAdminUser.oracle`.

This script can be modified as necessary and be used to create an alternative Oracle ERP administrative user. Usage instructions are documented in the comments at the beginning of the script.

For pass-through authentication only, authority is needed to run the following SQL command:

```
create or replace function wavesetValidateFunc1 (username IN varchar2,
password IN varchar2)
RETURN varchar2 IS ret_val boolean;
BEGIN ret_val := apps.FND_USER_PKG.ValidateLogin(username, password);
IF ret_val = TRUE THEN RETURN 'valid';
ELSE RETURN NULL;
END IF;
END wavesetValidateFunc1;
```

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes. For Oracle ERP, set the password expiration date to a date in the past to disable the account.
Rename account	No
Pass-through authentication	Yes
Before/after actions	No
Data loading methods	Import directly from resource

Account Attributes

This section provides information about the Oracle and Oracle ERPaccount attributes, including:

- *Oracle Database*
- *Oracle Financials*

Oracle Database

The following table lists the Oracle database user account attributes.

Notes:

- All attributes are Strings.
- All attributes are optional.

Resource User Attribute	Description
noCascade	Indicates whether to perform a cascade delete for a user.
oracleAuthentication	Must be one of the following values: • LOCAL (default value) • EXTERNAL • GLOBAL
oracleDefaultTS	Name of the default tablespace for objects that the user creates.
oracleDefaultTSQuota	Maximum amount of default tablespace the user can allocate.
oracleGlobalName	Global name of a user. (Applicable only when oracleAuthentication is set to GLOBAL.)
expirePassword	This attribute is applicable for local Oracle accounts only.
oraclePrivs	One or more privileges assigned to the user.
oracleProfile	One or more profiles assigned to the user.
oracleRoles	One or more roles assigned to the user.
oracleTempTS	Name of the tablespace for the user's temporary segments.
oracleTempTSQuota	Maximum amount of temporary tablespace the user can allocate.

Oracle Financials

The following table lists the Oracle ERP account attributes. All attributes are optional.

Resource User Attribute	Data Type	Description
owner	string	The administrator who created the account.
start_date	string	The date the account is effective.
end_date	string	The date the account expires. Set the date to a previous date to disable an account. A null value indicates no expiration date.
description	string	A description of the user, such as the full name.
password_date	string	The datestamp of the last password change. Oracle ERP can use this datestamp when evaluating the password_lifespan_days attribute value. For example if you set the password_lifespan_days attribute to 90, then Oracle ERP will calculate 90 days out from the last password change date (password_date) to determine if the password is expired. Each time the Oracle ERP adapter performs a password change, it will set the password_date to the current date.
password_accesses_left	string	The number of times the user can use the current password.
password_lifespan_accesses	string	The number of accesses over the life of the password
password_lifespan_days	string	The total number of days the password is valid.
employee_id	string	Identifier of employee to whom the application username is assigned.
email_address	string	The e-mail address of the user.
fax	string	The fax number of the user.
customer_id	string	The customer ID of the user.
supplier_id	string	The supplier ID of the user.
responsibilities	string	The names of the responsibilities assigned to the user.

Resource User Attribute	Data Type	Description
responsibilityKeys	string	The keys associated with the user's list of responsibilities.
securingAttrs	string	Adds supports for securing attributes.
expirePassword	boolean	Indicates whether the password will be expired.

The Oracle ERP adapter allows you to add several read-only attributes that Identity Auditor can use to audit changes to responsibilities. The values returned in the auditorResps attribute are the active responsibilities for that user. All other attributes listed below are aggregates of each responsibility's sub-items, minus any menu and function exclusions that may exist.

The following table lists attributes that may be added to the schema map

Attribute	Description
auditorResps	List of a user's Active Responsibilities.
userMenuNames	Concatenates all User Menu Names.
menuIds	Concatenates all Menu IDs
userFunctionNames	Concatenates all User Function Names
functionIds	Concatenates all Function IDs
formIds	Concatenates all Form IDs. Includes values returned by readOnlyFormIds and readWriteOnlyFormIds.
formNames	Concatenates all Form Names. Includes values returned by readOnlyFormNames and readWriteOnlyFormNames/
userFormNames	Concatenates all User Form Names. Includes values returned by readOnlyUserFormNames and readWriteOnlyUserFormNames/
readOnlyFormIds	Concatenates all Read-Only Forms IDs
readOnlyFormNames	Concatenates all Read-Only Form Names
readOnlyUserFormNames	Concatenates all Read-Only User Form Names
readWriteOnlyFormIds	Concatenates all Read/Write-Only Forms Ids
readWriteOnlyFormNames	Concatenates all Read/Write-Only Form Names
readWriteOnlyUserFormNames	Concatenates all Read/Write-Only User Form Names

Resource Object Management

None

Identity Template

```
$accountId$
```

Sample Forms

Built-In

None

Also Available

```
OracleERPUserForm.xml
```

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.OracleResourceAdapter`
- `com.waveset.adapter.OracleERPResourceAdapter`
- `com.waveset.adapter.JdbcResourceAdapter`

OS/400

The OS/400 resource adapter is defined in the `com.waveset.adapter.OS400ResourceAdapter` class.

This adapter supports the following versions of IBM OS/400:

- V4r3
- V5r1

Resource Configuration Notes

None

Identity Manager Installation Notes

No additional installation procedures are required on this resource.

Usage Notes

Identity Manager supports three options for handling OS/400 objects that are associated with an account on an OS/400 resource. To enable this specialized support, you must use the OS400Deprovision form that is located in the Identity Manager sample directory. You must also edit the system configuration object; instructions for doing this are included in comments in the OS400Deprovision form. Once enabled, these options appear on the Delete Resource Accounts page when you choose to delete a user's OS/400 resource account.

Available delete options are:

- **`DLT`** - The user's resource account and associated OS/400 objects are deleted.
- **`NODLT`** - If the user has associated objects, his account is not deleted and associated OS/400 objects are not affected.
- **`CHGOWN`** - The user's resource account is deleted and associated OS/400 objects are assigned to a designated owner. CHGOWN is the default option. By default, OS/400 objects are assigned to the QDFTOWN profile.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses SSL to communicate with the OS/400 adapter.

Required Administrative Privileges

The following administrative privileges are required for this adapter:

- **CRT**: To add an OS/400 user, the administrator must have (1) *SECADM special authority, (2) *USE authority to the initial program, initial menu, job description, message queue, output queue, and attention-key-handling program if specified, and (3) *CHANGE and object management authorities to the group profile and supplemental group profiles, if specified.
- **CHG**: You must have *SECADM special authority, and *OBJMGT and *USE authorities to the user profile being changed, can specify this command. *USE authority to the current library, program, menu, job description, message queue, print device, output queue, or ATTN key handling program is required to specify these parameters.
- `DLT`: The user must have use (*USE) and object existence (*OBJEXIST) authority to the user profile. The user must have existence, use, and delete authorities to delete a message queue associated with and owned by the user profile. The user profile cannot be deleted if a user is currently running under the profile, or if it owns any objects and OWNOBJOPT(*NODLT) is specified. All objects in the user profile must first either be transferred to new owners by using the Change Object Owner (CHGOBJOWN) command or be deleted from the system. This can also be accomplished by specifying OWNOBJOPT(*DLT) to delete the objects or OWNOBJOPT(*CHGOWN user-profile-name) to change the ownership. Authority granted to the user does not have to be specifically revoked by the Revoke Object Authority (RVKOBJAUT) command; it is automatically revoked when the user profile is deleted.
- **DSP**: The user name can be specified as USRPRF(*ALL) or USRPRF(generic*-user-name) only when TYPE(*BASIC) and OUTPUT(*OUTFILE) are specified.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	No
Pass-through authentication	No
Before/after actions	Yes
Data loading methods	• Import directly from resource • Reconcile with resource

Account Attributes

The following table provides information about OS/400account attributes.

Note All attributes are strings, unless indicated otherwise.

Resource User Attribute	Description
accountId	Required. The user's logon ID.
password	Required. The user's password. This value is encrypted.
ASTLVL	Assistance level
ATNPGM	Attention-key-handling program
CCSID	Coded character set identifier
CNTRYID	Country identifier
CURLIB	Current library
DAYS_UNTIL_PASSWORD_EXPIRES	The number of days until the password expires.
DLVRY	Delivery mode
GID	Group identification number
GRPPRF	Group profile

Resource User Attribute	Description
HIGHEST_SCHEDULING_PRIORITY	
HOMEDIR	Home directory
INLMNU	Initial menu
INLPGM	Initial program
JOBD	Job description
KBDBUF	Keyboard buffering
LANGID	Language identifier
LMTCPB	Limit capabilities
LMTDEVSSN	Limit device sessions
MAXSTG	Maximum storage
MSGQ	Message queue
OUTQ	Output queue
OWNER	Owner of new objects
OWNOBJOPT	Owned object option
PRTDEV	Print device
PWDEXP	Indicates whether to set an expiration on the password.
SPCAUT	Special authority
SPCENV	Special environment
SRTSEQ	Sort sequence
STATUS	Login status of a user profile
TEXT	User description
UID	User identification number
USRCLS	User class
USROPT	User options

Resource Object Management

None

Identity Template

```
$accountId$
```

Sample Forms

```
OS400UserForm.xml
```

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

```
com.waveset.adapter.OS400ResourceAdapter
```

PeopleSoft Component

The PeopleSoft Component resource adapter is read-only. You cannot use this adapter to create or modify PeopleSoft accounts.It is used for Active Sync. The adapter is defined in the `com.waveset.adapter.PeopleSoftComponentActiveSyncAdapter` class.

This adapter supports the following versions of PeopleSoft:

- PeopleTools 8.1 through 8.4.2 with HRMS 8.0 through 8.8, using the PeopleSoft Component interface.

Note Additional versions might also be supported.

The following adapters have been deprecated:

- PeopleSoft (for JMAC interface)

  `com.waveset.adapter.PeopleSoftResourceAdapter`
- PeopleSoft Update Component

  `com.waveset.adapter.PeopleSoftComponentResourceAdapter`

  This adapter has been replaced by the PeopleSoft Component Interface adapter. See page 1-215 for more information.

Resource Configuration Notes

You must use the following PeopleSoft tools to integrate resources with the resource adapter.

- **Application Designer**: Use this tool to build and configure the Identity Manager project.
- **PeopleTools browser-based application**: Use this tool to configure component interfaces, roles, and user profiles.

Follow these steps to configure PeopleSoft for use with Identity Manager:

- Step 1: Create the New Project
- Step 2: Edit Identity Manager Objects
- Step 3: Build the Project
- Step 4: Manually Execute the `audittrigger` Script
- Step 5: Enable Auditing on Selected Tables
- Step 6: Configure PeopleTools
- Step 7: Prune the Audit Log

Step 1: Create the New Project

Create a new project with the PeopleSoft Application Designer using the following steps:

1. Create a new project in the Application Designer by selecting the **File—>New** menu. Then select **Project** from the list.
2. Name the project by performing a save. Use the **File—>Save Project As...** menu, and enter a unique name for the project, such as “IDM”.
3. Create the objects within the project by performing the tasks listed in *Edit Identity Manager Objects* below.

Step 2: Edit Identity Manager Objects

The Identity Manager project contains the following types of objects:

- Records
- Pages
- Message Agent
- Components
- Component Interfaces

You must create these objects within the Application Designer. Each of these objects are described in detail below.

Records

There are three records (two views and one table) that must be defined within the Application Designer. The following record descriptions illustrate a typical implementation. The records can be customized to the needs of the implementation by adding or changing fields.

AUDIT_EFFDT_LH View

The AUDIT_EFFDT_LH view is polled by the PeopleSoft Active Sync resource adapter. Identity Manager uses the following fields to query for events which have not yet been processed:

- AUDIT_PROC_ORDER. This field must specify the Key, Search Key, List Box Item, and From Search Field keys.
- AUDIT_PROC_END. This field must specify the Key, Search Key, List Box Item, and Through Search Field fields.
- EMPLID and EMPL_RCD. These are required non-key properties that are used by an Identity Manager query to fetch employee data.

All other fields in the AUDIT_EFFDT_LH table are optional.

The following table describes the Use Display characteristics of the AUDIT_EFFDT_LH view:

Field Name	Type	Key	Ordr	Dir	Srch	List	Sys	Default
AUDIT_PROC_ORDER	Char	Key	1	Asc	Yes	Yes	No	
AUDIT_PROC_END	Char	Key		Asc	Yes	Yes	No	
AUDIT_STAMP	DtTm				No	No	No	
EFFDT	Date				No	No	No	%date
AUDIT_OPRID	Char				No	No	No	
AUDIT_ACTN	Char				No	No	No	
AUDIT_RECNAME	Char				No	No	No	
EMPLID	Char				No	No	No	‘NEW’
EMPL_RCD	Nbr				No	No	No	

Information in the last audit entry is stored in Identity Manager as a “lastProcessed” configuration object to be used (and updated) on subsequent searches of the AUDIT_EFFDT_LH view. Maintenance of the lastProcessed Configuration object by the PeopleSoft Active Sync resource adapter prevents records from being processed more than once.

The following SQL code is used to generate the AUDIT_EFFDT_LH view:

```
SELECT audit1.AUDIT_PROC_ORDER AS AUDIT_PROC_ORDER
,audit1.AUDIT_PROC_ORDER AS AUDIT_PROC_END
,audit1.AUDIT_STAMP AS AUDIT_STAMP
,audit1.EFFDT AS EFFDT
,audit1.AUDIT_OPRID AS AUDIT_OPRID
,audit1.AUDIT_ACTN AS AUDIT_ACTN
,audit1.AUDIT_RECNAME AS AUDIT_RECNAME
,audit1.EMPLID AS EMPLID
,CAST(audit1.EMPL_RCD AS INTEGER) AS EMPL_RCD
FROM PS_AUDIT_PRS_DATA audit1
WHERE audit1.AUDIT_PROC_DATE <= %CurrentDateIn
AND NOT EXISTS (
SELECT *
FROM PS_AUDIT_PRS_DATA audit2
WHERE audit2.AUDIT_PROC_DATE <= %CurrentDateIn
AND audit2.AUDIT_PROC_ORDER > audit1.AUDIT_PROC_ORDER
AND audit2.EMPLID = audit1.EMPLID )
```

The final line in this SQL code sample prevents Identity Manager from seeing operations with effective dates until the effective date has arrived.

AUDIT_PRS_DATA Table

The AUDIT_PRS_DATA table must contain the following fields:

- AUDIT_PROC_ORDER. This field must specify the Key, Search Key, List Box Item, and From Search field keys. In addition, this field must be set to Required so that PeopleSoft puts a non-null integrity constraint on the database column.
- AUDIT_PROC_DATE. This field must specify the Alternate Search Key, List Box Item. In addition, this field must be set to Required so that PeopleSoft puts a non-null integrity constraint on the database column.
- EMPLID and EMPL_RCD. These are required non-key properties that are used by an Identity Manager query to fetch employee data.

All other fields in the AUDIT_PRS_DATA table are optional.

The following table describes the Use Display characteristics of the AUDIT_PRS_DATA view:

Field Name	Type	Key	Ordr	Dir	Srch	List	Sys	Default
AUDIT_PROC_ORDER	Char	Key	1	Asc	Yes	Yes	No	
AUDIT_PROC_DATE	Date	Alt		Asc	No	No	No	
AUDIT_STAMP	DtTm				No	No	No	%date
AUDIT_OPRID	Char				No	No	No	‘ANON’
AUDIT_ACTN	Char				No	No	No	‘C’
AUDIT_RECNAME	Char				No	No	No	‘ANON’
EMPLID	Char				No	No	No	‘NEW’
EFFDT	Date				No	No	No	%date
EMPL_RCD	Nbr				No	No	No	

PERS_SRCH_LH View

The PERS_SRCH_LH view must contain the EMPLID and EMPL_RCD fields, with the Key, Search Key, and List Box Item keys selected. All other fields provide the data that is synchronized with Identity Manager. It is up to the PeopleSoft Active Sync form to map this data into the Identity Manager user account.

The following table describes the Use Display characteristics of the PERS_SRCH_LH view:

Field Name	Type	Key	Ordr	Dir	Srch	List	Sys
EMPLID	Char	Key	1	Asc	Yes	Yes	No
EMPL_RCD	Nbr	Key	2	Asc	Yes	Yes	No
NAME	Char				No	Yes	No
LAST_NAME_SRCH	Char				No	Yes	No
SETID_DEPT	Char				No	Yes	No
DEPTID	Char				No	Yes	No
ADDRESS1	Char				No	Yes	No
PER_STATUS	Char				No	Yes	No
EMPL_STATUS	Char				No	Yes	No
FIRST_NAME	Char				No	Yes	No
LAST_NAME	Char				No	Yes	No
MIDDLE_NAME	Char				No	Yes	No
REPORTS_TO	Char				No	Yes	No
JOBCODE	Char				No	Yes	No
COMPANY	Char				No	Yes	No
NAME_INITIALS	Char				No	Yes	No
COUNTRY	Char				No	Yes	No
PHONE	Char				No	Yes	No
CITY	Char				No	Yes	No
STATE	Char				No	Yes	No
POSTAL	Char				No	Yes	No

The following SQL code is used to generate the PERS_SRCH_LH view:

Note For your convenience, the `peoplesoft/idm.zip` file on the installation media contains an SQL script file named `pers_srch_lh.sql` that duplicates the following SQL code.

```
SELECT P.EMPLID
,A.EMPL_RCD
,P.NAME
,P.LAST_NAME_SRCH
,A.SETID_DEPT
,A.DEPTID
,P.ADDRESS1
,P.PER_STATUS
,A.EMPL_STATUS
,P.FIRST_NAME
,P.LAST_NAME
,P.MIDDLE_NAME
,A.REPORTS_TO
,A.JOBCODE
,A.COMPANY
,P.NAME_INITIALS
,P.COUNTRY
,P.PHONE
,P.CITY
,P.STATE
,P.POSTAL
FROM PS_Job A
, PS_PERSONAL_DATA P
WHERE A.EMPLID = P.EMPLID
AND A.EffDt = (
SELECT MAX(C.EffDt)
FROM PS_Job C
WHERE C.EmplID = A.EmplID
AND C.EMPL_RCD = A.EMPL_RCD
AND C.EffDt <= %CurrentDateIn)
AND A.EffSeq = (
SELECT MAX(D.EffSeq)
FROM PS_Job D
WHERE D.EmplID = A.EmplID
AND D.EMPL_RCD = A.EMPL_RCD
AND D.EffDt = A.EffDt)
```

The WHERE clause returns the current employee record for the given employee ID. PeopleSoft allows multiple records for a given employee, each of which has its own effective date/effective sequence. This clause returns the record whose effective date/effective sequence pair is the latest out of all those that are already effective (whose effective date has occurred).

The WHERE clause returns null for an employee whose sunrise date is in the future.

Pages

The Identity Manager project must also contain the following pages for the Component interface only:

- LH_AUDIT_EFFDT
- LH_EMPLOYEE_DATA

LH_AUDIT_EFFDT

The LH_AUDIT_EFFDT page contains fields defined in the AUDT_EFFDT_LH table. This page is not displayed on the PeopleSoft GUI. Therefore, the layout and ordering of the fields is not important.

The following table describes the Use Display characteristics of the LH_AUDIT_EFFDT page. All items are defined in the AUDT_EFFDT_LH record.

Label	Type	Field
Unique order to process	Edit Box	AUDIT_PROC_ORDER
EmplID	Edit Box	EMPLID
Upper bound for search	Edit Box	AUDIT_PROC_END
Empl Rcd Nbr	Edit Box	EMPL_RCD
Date and Time Stamp	Edit Box	AUDIT_STAMP
Effective Date	Edit Box	EFFDT
User ID	Edit Box	AUDIT_OPRID
Action	Drop Down List	AUDIT_ACTN
Audit Record Name	Edit Box	AUDIT_RECNAME

LH_EMPLOYEE_DATA

The LH_EMPLOYEE_DATA page is the container for the fields defined in the PERS_SRCH_LH view. All items are defined in the PERS_SRCH_LH record.

The following table describes the Use Display characteristics of the LH_EMPLOYEE_DATA page:

Label	Type	Field
EmplID	Edit Box	EMPLID
Name	Edit Box	NAME
Last Name	Edit Box	LAST_NAME_SRCH
Department SetID	Edit Box	SETID_DEPT
Department	Edit Box	DEPTID
Address Line 1	Edit Box	ADDRESS1
Personnel Status	Edit Box	PER_STATUS
Employee Status	Edit Box	EMPL_STATUS
First Name	Edit Box	FIRST_NAME
Last Name	Edit Box	LAST_NAME
Middle Name	Edit Box	MIDDLE_NAME
Reports To Position	Edit Box	REPORTS_TO
Job Code	Edit Box	JOBCODE
Company	Edit Box	COMPANY
Name Initials	Edit Box	NAME_INITIALS
Country	Edit Box	COUNTRY
Telephone	Edit Box	PHONE
City	Edit Box	CITY
State	Edit Box	STATE
Postal Code	Edit Box	POSTAL
Empl Rcd Nbr	Edit Box	EMPL_RCD

Components

Components are the bridge between pages and menus. Once you have created your pages, you must add them to one or more components to use them on menus or in business processes.

Create a separate component for the each of the following pages:

- LH_AUDIT_EFFDT
- LH_EMPLOYEE_DATA

The default component names are LH_AUDIT_EFFDT and LH_EMPLOYEE_COMP

Component Interfaces

A component interface is a PeopleTools object that exposes a PeopleSoft component for synchronous access from another application, such as Identity Manager. Create a separate component interface for each component you created. The default names for the Component Interfaces are LH_AUDIT_EFFDT_COMP_INTF and LH_EMPLOYEE_COMP_INTF. These values can be modified on the General Active Sync Settings page of the Active Sync Wizard.

Step 3: Build the Project

Use this procedure to build the project and create PeopleSoft views and tables in the database.

To build the project using the Application Designer:

1. Build the project:
1. Select **Build—>Project**. The Build dialog appears.
2. In the Build Options area, select the Create Tables and Create Views options. In the Build Execute Options area, select the Execute SQL now option.
3. Click **Settings**. The Build Settings dialog appears.
4. Verify that the Recreate table if it already exists option is selected.
5. Click the Logging tab.
6. In the Logging Level area, select the Fatal errors, warnings and information messages option.
7. In the Logging Output area, enter a unique log file name.
8. Click **OK**, and then click **Build** to build the project and to create views and tables.

   Application Designer may display a warning message similar to the following:

   ```
   Potentially data destructive settings are active. Continue
   the build process?
   ```

9. Click Yes to continue to build process.

Note After importing and building the project, you must test the components in Application Designer. The reliability of the import project feature within PeopleSoft varies from release to release. Therefore, validation of the objects is very important.

Step 4: Manually Execute the audittrigger Script

The `idm.zip` file contains an Oracle SQL script named `audittrigger.oracle`. This script creates the trigger and sequence necessary to maintain the `AUDIT_PROC_DATE` and `AUDIT_PROC_ORDER` columns of the `PS_AUDIT_PRS_DATA` table.

Note The `audittrigger.oracle` script is available only for Oracle. If you are using a different database, convert the script to run on that database.

The `audittrigger.oracle` script or its equivalent must be run every time you rebuild the PeopleSoft project.

Step 5: Enable Auditing

From the Application Designer, you will enable auditing on the JOB and PERSONAL_DATA tables, and possibly on the POSITION_DATA and EMPLOYMENT tables. This is record-level auditing that writes a simple summary record with the operator and the EMPLID of the changed record.

To update your PeopleTools database objects:

1. Launch the Application Designer.
2. Select **`File –>Open`** to display the Open Object dialog.
3. Select **Record** from the Object type menu, and then type JOB in the Name field.
4. Click **Open** to open the record.
5. Select **File—>Properties** to open the record properties, and then click the Use tab.
6. In the Record Name field, select AUDIT_PRS_DATA.
7. In the Audit Options area, select the Add, Change, and Delete options. Leave the Selective option unchecked.

Repeat these steps for the PERSONAL_DATA table and other tables that will be triggers for data synchronization.

Note For more information, see “Creating Record Definitions” in the Application Designer documentation.

Step 6: Configure PeopleTools

To complete the configuration process, you must use the PeopleTools browser-based GUI to assign component interfaces to a permission list, create a role and assign permission lists to the role, and assign the role to user profiles. Refer to the PeopleTools documentation for more information about these entities.

Component Interfaces

Use of component interfaces must be authorized. To authorize a component interface:

1. Log in to the PeopleTools browser-based GUI and navigate to `Home > People Tools > Maintain Security > Use > Permission Lists`.
2. Select the Add a New Value link and enter a value such as LH_ALL
3. Click on the right arrow in the tabs section near the top of the page until the Component Interface tab is displayed. Then click on the Component Interface tab.
4. Enter an existing Component Interface, such as LH_AUDIT_EFFDT_COMP_INTF, in the text box.
5. Click the Edit link to go to the Component Interface Permissions page.
6. Click the Full Access button to enable full access for all the methods, or use the drop-down menus to assign access for individual methods. Click OK to return to the Permission Lists page.
7. Click the + (plus) button. An additional text box will be displayed.
8. Enter a different existing Component Interface, such as LH_EMPLOYEE_COMP_INTF, in the text box.
9. Repeat steps 5 and 6.
10. Save your changes.

Roles

To assign a PeopleSoft role to the Component Interfaces:

1. Navigate to `Home > People Tools > Maintain Security > Use > Roles`.
2. Select the Add a New Value link and enter a value such as LH_ROLE.
3. Click the Permission Lists tab.
4. Enter an existing Permission List, such as LH_ALL.
5. Save your changes.

User Profiles

To assign a role to a user profile:

1. Navigate to `Home > People Tools > Maintain Security > Use > User Profiles`.
2. Enter an existing user ID. This user can be specified as the user on the Resource Parameters page in Identity Manager.

Note You can also create a new user. Refer to the PeopleSoft documentation for more information about the requirements of a user account.

3. Select the Roles tab.
4. Click the + (plus) button. An additional text box will be displayed.
5. Enter the name of a role, such as LH_ALL.
6. Save your changes.

Step 7: Prune the Audit Log

Identity Manager does not delete audit events from the audit log. The PeopleSoft administrator must set up a task to prune old audit entries. This task must retain transactions with a future effective date until Identity Manager processes them. That is, entries whose `AUDIT_PROC_DATE` is in the future must NOT be pruned.

Identity Manager Installation Notes

The PeopleSoft Component resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. Copy the following file from the PeopleSoft installation media to the *`InstallDir`*`\idm\WEB-INF\lib` directory:

   `psjoa.jar`

   The version number of the jar file must match the version of PeopleSoft.
2. To add this resource to the Identity Manager resources list, you must add the following value in the Custom Resources section of the Configure Managed Resources page.

   `com.waveset.adapter.PeopleSoftComponentActiveSyncAdapter`

Usage Notes

This section provides information related to using the PeopleSoft Component resource adapter, including:

- *Controlling Hosts in a Cluster*
- *Active Sync Configuration*

Controlling Hosts in a Cluster

You can use the `sources.ResourceName.hosts` property in the `waveset.properties` file to control which host(s) in a cluster are used to execute the synchronization portion of an Active Sync resource adapter. You must replace *`ResourceName`* with the name of the Resource object.

Active Sync Configuration

Specify the Audit Component Interface Name and the Employee Component Interface Name on the General Active Sync Settings page of the Active Sync Wizard.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses the Client Connection Toolkit (Sync Only) to communicate with this adapter.

Required Administrative Privileges

The user name that connects to PeopleSoft must be assigned to a PeopleSoft role that can access the component interfaces.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Create account	No
Update account	No
Delete account	No
Enable/disable account	No
Password update	No
Rename account	No
Pass-through authentication	No
Before/after actions	No
Data loading methods	Active Sync

Account Attributes

The following table provides information about the PeopleSoft Component Active Sync adapter account attributes.

Resource User Attribute	mapName	Description
accountId	EMPLID	Required.
ACTION	ACTION	An action code of up to 3 characters
ACTION_REASON	ACTION_REASON	A reason code of up to 3 characters
AUDIT_ACTN	AUDIT_ACTN	The type of action the system audited (A=add, C=change, D=delete).
AUDIT_OPRID	AUDIT_OPRID	The operator who caused the system to trigger the audit.
AUDIT_STAMP	AUDIT_STAMP	Date and time stamp
AUDIT_RECNAME	AUDIT_RECNAME	The name of the record the system audited.

Resource User Attribute	mapName	Description
EFFSEQ	EFFSEQ	Effective sequence
EFFDT	EFFDT	Effective date
Employee ID	EMPL_ID	The key field used to uniquely identify users.
fullname	NAME	The user's full name.
firstname	FIRST_NAME	The user's first name.
lastname	LAST_NAME	The user's last name.
Middle Name	MIDDLE_NAME	The user's middle name
PS_PER_STATUS	PER_STATUS	Personnel status, such as employee or non-employee.
PS_EMPL_STATUS (Status on the AS adapter)	EMPL_STATUS	The status of the employee, such as Active, Suspended, or Terminated.
Home Address	ADDRESS1	The user's home address
Department	DEPTID	The user's department
Manager	REPORTS_TO	The user's manager
Job Title	JOBCODE	A code that identifies the user's job title.
Initials	NAME_INITIALS	The user's initials
Country	COUNTRY	3-letter country code
Company	COMPANY	Company name
Home Phone	PHONE	The user's home phone number
Home City	CITY	The city in which the user resides
Home State	STATE	The state in which the user resides
Home Zip	POSTAL	The user's home Zip or postal code.

Resource Object Management

Not applicable

Identity Template

```
$accountId$
```

Sample Forms

```
PeopleSoftForm.xml
```

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

```
com.waveset.adapter.PeopleSoftComponentActiveSyncAdapter
```

PeopleSoft Component Interface

The PeopleSoft Component Interface adapter is defined in the `com.waveset.adapter.PeopleSoftCompIntfcAdapter` class.

This resource adapter manages data in PeopleSoft via component interfaces. It can also manage additional PeopleSoft applications (such as HR and Financials) if these applications are installed on a system with a supported version of PeopleTools.

The PeopleSoft Component Interface adapter supports PeopleTools 8.1 through 8.4.2.

Resource Configuration Notes

The PeopleSoft Component Interface adapter is configured by default to support the `USER_PROFILE` and `DELETE_USER_PROFILE` component interfaces. The adapter can also use custom component interfaces to create, read, and update account data if the component interface supports the following methods:

- `Create`
- `Get`
- `Save`
- `SetPassword`

To delete accounts, the custom component interface must support the following methods:

- `Get`
- `Save`

In addition, the user specified on the Resource Parameters page must have permission to execute the methods of the invoked component interfaces.

Identity Manager Installation Notes

The PeopleSoft Component Interface adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. Copy the following file from the PeopleSoft installation media to the `$WSHOME/WEBINF/lib` directory:

   `psjoa.jar`

Note The version of the `psjoa.jar` must match the version of your installed PeopleSoft system.

2. To add this resource to the Identity Manager resources list, you must add the following value in the Custom Resources section of the Configure Managed Resources page:

   ```
   com.waveset.adapter.PeopleSoftCompIntfcAdapter
   ```

Usage Notes

The PeopleSoft Component Interface adapter accomplishes user provisioning by invoking methods and setting properties on PeopleSoft component interfaces. Component interface definitions are assigned in the PeopleSoft Component Interface configuration object. This object can be modified via the debug pages or with the Business Process Editor (BPE). You can also edit a copy of the `$WSHOME/sample/PeopleSoftComponentInterfaces.xml` file and load that file into Identity Manager.

For more information about configuring and implementing component interfaces with this adapter, see the following sections:

- Component Interface Map Definitions
- Resource Objects (see *PeopleSoft Component Interface Resource Objects* on page 1-219)

Component Interface Map Definitions

The component interface map contains the list of component interfaces available to the adapter.

- `interfaces` object — Contains a list of component interfaces. If you have a custom component interface, you must define your own component interface definition in the map. Edit the PeopleSoft Component Interfaces Configuration object and add your definition as an additional Object into the `<List>` element under the `<Attribute name='interfaces'>` element.

Each available component interface has its own definition. Key elements of a component interface definition include:

- `name` — The label of a component interface. It often matches the value of the `componentInterface` attribute, but this is not a requirement. The value will be displayed in the drop-down menu on the adapter's Resource Parameters page.
- `componentInterface` attribute — The name of the component interface, as defined in PeopleSoft.
- `getKey` attribute — The key that is used to perform a PeopleSoft GET, CREATE, and DELETE operations.

- `findKey` attribute — The key that is set when performing a PeopleSoft FIND operation.
- `key` attribute — Deprecated. Use `getKey` instead.
- `properties` attribute — A list of properties that can be read or set from the PeopleSoft component interface.

  Each Object in the *properties* list must have the following attribute:

  - name — The name of the property. This must match exactly with the name of a property exposed by the PeopleSoft component interface identified by the `componentInterface` property. The names of the properties are candidates to be listed as resource user attributes on the Account Attributes page.

  If this a collection property, then you must define additional attributes. A collection property defines its key property and its own nested set of simple and/or complex properties:

  - `isCollection` attribute — If the property is a collection, then set this to true.
  - `key` attribute — If the property is a collection, set this to the name of the property which uniquely identifies each item of the collection.
  - `properties` attribute — The list of properties which can be read/set for each item of the collection. To support arbitrary complexity, each member of this list is an Object with the same allowed attributes as the parent. That is, it can contain its own `name`, `isCollection`, `key`, and `properties` attributes.
- `disableRule` attribute — An Object that defines the logic to compute and set the user disable state. This attribute contains the following attributes
  - `property` attribute — The property to check. The value must be listed in the `properties` attribute for the `componentInterface` object.
  - `trueValue` attribute— A value that indicates the user is disabled.
  - `falseValue` attribute — A value that indicates the user is enabled.
- `supportedObjectTypes` attribute — A list of Identity Manager resource objects types that can be accessed via the adapter. Each object defines a set of features.
  - `features` attribute — A list supported features. Possible feature types include view, get, list, find, create, saveas, update, rename, and delete.

Default Component Interfaces Supported

The default Component Interface configuration object defines the following interfaces:

- USER_PROFILE — Performs create, read, and update actions.
- DELETE_USER_PROFILE — Deletes user accounts.
- ROLE_MAINT — Adds support for PeopleSoft roles.

USER_PROFILE Component Interface

The default USER_PROFLE component interface definition is used to perform create, read, and update actions. The `key` and `findKey` attributes are set to UserID, because the USER_PROFILE component interface assigns the UserID field for the GETKEYS and FINDKEYS keys.

The default definition for the USER_PROFILE component interface does not define all of the possible properties. It has been simplified to include those used in the sample user form. If you need to add more resource user attributes to the Account Attributes page, then the component interface definition must be updated first. A resource user attribute cannot be added to that page unless it is listed in the component interface definition.

Most properties are defined in USER_PROFILE are simple objects. However, the IDTypes and Roles objects are collections and can have multiple values. IDTypes contains a collection of its own, Attributes. These objects must include the `isCollection` attribute, the key name for the collection, and at least one property.

DELETE_USER_PROFILE Component Interface

The DELETE_USER_PROFILE component interface definition is used to delete user profile definitions. The OPRID key determines which user profile is to be deleted. Since the component interface does not have properties, none are listed in the definition.

ROLE_MAINT Component Interface

The ROLE_MAINT component interface definition is part of a sample implementation that illustrates how Identity Manager can be configured to list role resource objects. Other resource objects can be listed by following the general guidelines listed below and modifying the ROLE_MAINT example to match your requirements.

Note The PeopleSoft Component Interface adapter supports listing resource objects only. It does not support other object features, such as update, create, or delete.

The ROLE_MAINT component interface definition has the following characteristics of note:

- The `findKey` and `getKey` attributes are assigned to ROLENAME because ROLENAME is the primary key for FINDKEYS and GETKEYS.
- DESCR and ROLESTATUS are also keys in FINDKEYS, but since they are not primary keys, they are not listed as values for `findKey`. Instead, they are listed in the `properties` section.
- The `supportedObjectTypes` attribute defines the Role object. The Role object supports the find and get features.

PeopleSoft Component Interface Resource Objects

The XML of a PeopleSoft Component Interface resource can be edited so that resource objects can be managed. Use the debug pages or BPE to add an ObjectType element.

For example, to add support for the Role resource object, add an ObjectType element similar to the following.

```
<ObjectTypes>
<ObjectType name='Role' icon='role'>
   <ObjectFeatures>
      <ObjectFeature name='find'/>
   </ObjectFeatures>
   <ObjectAttributes idAttr='ROLENAME' displayNameAttr='ROLENAME'
descriptionAttr='DESCR'>
      <ObjectAttribute name='ROLENAME' type='string'/>
      <ObjectAttribute name='DESCR' type='string'/>
      <ObjectAttribute name='ROLESTATUS' type='string'/>
   </ObjectAttributes>
</ObjectType>
</ObjectTypes>
```

The ObjectType `name` (for example, Role) must match the name of one of the objects in the `supportedObjectTypes` list of exactly one component interface definition. Each ObjectFeature (for example, find) must have a corresponding feature in the `features` list in that same `supportedObjectTypes`. The matched component interface will be the one used to perform the resource feature. (If there are multiple matches, the first one found will be used.)

The following example is part of the component interface definition for the ROLE_MAINT component interface in the component interface map. Note that the Object name Role is found and that an item in the features list is named find.

```
<Attribute name='supportedObjectTypes' >
   <List>
      <Object name='Role'>
         <Attribute name='features' >
            <List>
               <Object name='find' />
               <Object name='get' />
            </List>
         </Attribute>
      </Object>
   </List>
</Attribute>
```

User Form

The following user form fragment can be used to retrieve a list of PeopleSoft roles. Note that ROLENAME and DESCR attributes are being fetched.

```
<invoke name='getResourceObjects' class='com.waveset.ui.FormUtil'>
   <ref>:display.session</ref>
   <s>Role</s>
   <s>PeopleSoft Component Interface</s>
   <map>
      <s>searchAttrsToGet</s>
      <list>
         <s>ROLENAME</s>
         <s>DESCR</s>
      </list>
   </map>
</invoke>
```

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses the Client Connection Toolkit (Read/Write) to communicate with this adapter.

Required Administrative Privileges

The user that connects to PeopleSoft must be assigned to a PeopleSoft role which can access the methods of the managed component interface(s).

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Create account	Yes
Update account	Yes
Delete account	Yes
Enable/disable account	Yes, if Component Interface Map defines enable/disable logic
Password update	Yes
Rename account	No
Pass-through authentication	No
Before/after actions	No
Data loading methods	• Import directly from resource • Reconciliation

Account Attributes

The account attributes for the PeopleSoft Component Interface resource depend on the component interface being managed.

Each entry of the schema map should have a Resource User Attribute name that matches one of the entries in the “properties” list defined for the component interface in the Component Interface Map. When editing the schema map, you can click the **`Test Configuration`** button to verify an appropriate match can be found.

If the Resource User Attribute name matches a collection property in the component interface map, the value for the account attribute will be an XML string representation of the collection. For examples of manipulating collection properties, see the sample user form field `accounts[PeopleSoft Component Interface].ps_roles`.

Note The default schema map entries that are defined for a new resource instance are appropriate only when used with the default USER_PROFILE and DELETE_USER_PROFILE component interface maps. If you change these maps, or create your own, then you must change your schema map accordingly.

All account attributes are of type String.

Identity Manager User Attribute	Resource User Attribute	Description
email	EmailAddress	The user's e-mail address.
description	UserDescription	A description of the user.
symbolicId	SymbolicID	Required. The user's symbolic ID.
idTypes	IDTypes	A list of user types assigned to the user.
ps_roles	Roles	A list of rules assigned to the user.

Resource Object Management

None

Identity Template

```
$accountId$
```

Sample Forms

`sample/forms/PeopleSoftCompIntfcUserForm.xml`

This user form will function as expected only if the `USER_PROFILE` component interface is being managed, and if the default account attributes are used.

If you are managing a different component interface or using a different schema map, the user form must be changed.

Troubleshooting

Use the debug pages to set trace options on the following class:

```
com.waveset.adapter.PeopleSoftCompIntfcAdapter
```

RACF

The RACF resource adapter supports management of user accounts and memberships on an OS/390 mainframe via the IBM Host Access Class Library APIs. The adapter manages RACF over a TN3270 emulator session.

The RACF resource adapter is defined in the `com.waveset.adapter.RACFResourceAdapter` class.

Resource Configuration Notes

None

Identity Manager Installation Notes

The RACF resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. To add the RACF resource to the Identity Manager resources list, you must add the following value in the Custom Resources section of the Configure Managed Resources page.

   `com.waveset.adapter.RACFResourceAdapter`

2. The Identity Manager mainframe adapters use the IBM Host Access Class Library (HACL) to connect to the mainframe. The HACL is available in IBM Websphere Host On-Demand (HOD). The recommended jar containing HACL is `habeans.jar` and is installed with the HOD Toolkit (or Host Access Toolkit) that comes with HOD. The supported versions of HACL are in HOD V7.0, V8.0, and V9.0.

   However, if the toolkit installation is not available, the HOD installation contains the following jars that can be used in place of the `habeans.jar`:

   - `habase.jar`
   - `hacp.jar`
   - `ha3270.jar`
   - `hassl.jar`
   - `hodbase.jar`

   Copy the `habeans.jar` file or all of its substitutes into the `WEB-INF/lib` directory of your Identity Manager installation. See http://www.ibm.com/software/webservers/hostondemand/ for more information.

Usage Notes

This section provides information related to using the RACF resource adapter, which is organized into the following sections:

- *Administrators*
- *Resource Actions*
- *SSL Configuration*

Administrators

TSO sessions do not allow multiple, concurrent connections. To achieve concurrency for Identity Manager RACF operations, you must create multiple administrators. Thus, if two administrators are created, two Identity Manager RACF operations can occur at the same time. We recommend that you create at least two (and preferably three) administrators.

If you are running in a clustered environment, you must define an admin for each server in the cluster. This applies even if it is the same admin. For TSO, there must be a different admin for each server in the cluster.

If clustering is not being used, the server name should be the same for each row (the name of the Identity Manager host machine).

Note Host resource adapters *do not* enforce maximum connections for an affinity administrator across multiple host resources connecting to the same host. Instead, the adapter enforces maximum connections for affinity administrators within each host resource.

If you have multiple host resources managing the same system, and they are currently configured to use the same administrator accounts, you might have to update those resources to ensure that the same administrator is not trying to perform multiple actions on the resource simultaneously.

Resource Actions

The RACF adapter requires login and logoff resource actions. The login action negotiates an authenticated session with the mainframe. The logoff action disconnects when that session is no longer required.

See the Usage Notes for the Top Secret adapter on page 1-353 for more information about creating login and logoff resource actions.

SSL Configuration

This section describes how to configure SSL for this adapter, including:

- *Connecting the Adapter to a Telnet/TN3270 Server using SSL or TLS*
- *Generating a PKCS #12 File*
- *Troubleshooting*

Connecting the Adapter to a Telnet/TN3270 Server using SSL or TLS

Use the following steps to connect RACF resource adapters to a Telnet/TN3270 server using SSL/TLS.

1. Obtain the Telnet/TN3270 server's certificate in the PKCS #12 file format. Use `hod` as the password for this file. Consult your server's documentation on how to export the server's certificate. The procedure "Generating a PKCS #12 File" below for some general guidelines.
2. Create a `CustomizedCAs.class` file from the PKCS #12 file. If you are using a recent version of HOD, use the following command to do this.

   ```
   ..\hod_jre\jre\bin\java -cp ../lib/ssliteV2.zip;../lib/sm.zip
   com.ibm.eNetwork.HOD.convert.CVT2SSLIGHT CustomizedCAs.p12 hod
   CustomizedCAs.class
   ```

3. Place the `CustomizedCAs.class` file somewhere in the Identity Manager server's classpath, such as `$WSHOME/WEB-INF/classes`.
4. If a resource attribute named **`Session Properties`** does not already exist for the resource, then use the BPE or debug pages to add the attribute to the resource object. Add the following definition in the `<ResourceAttributes>` section:

   ```
   <ResourceAttribute name='Session Properties' displayName='Session
   Properties' description='Session Properties' multi='true'>
   </ResourceAttribute>
   ```

5. Go to the Resource Parameters page for the resource and add the following values to the **`Session Properties`** resource attribute:

   ```
   SESSION_SSL
   true
   ```

Generating a PKCS #12 File

The following procedure provides a general description of generating a PKCS #12 file when using the Host OnDemand (HOD) Redirector using SSL/TLS. Refer to the HOD documentation for detailed information about performing this task.

6. Create a new `HODServerKeyDb.kdb` file using the IBM Certificate Management tool. As part of that file, create a new self-signed certificate as the default private certificate.

   If you get a message that is similar to “error adding key to the certificate database” when you are creating the `HODServerKeyDb.kdb` file, one or more of the Trusted CA certificates may be expired. Check the IBM website to obtain up-to-date certificates.
7. Export that private certificate as Base64 ASCII into a `cert.arm` file.
8. Create a new PKCS #12 file named `CustomizedCAs.p12` with the IBM Certificate Management tool by adding the exported certificate from the `cert.arm` file to the Signer Certificates. Use `hod` as the password for this file.

Troubleshooting

You can enable tracing of the HACL by adding the following to the Session Properties resource attribute:

```
SESSION_TRACE
ECLSession=3 ECLPS=3 ECLCommEvent=3 ECLErr=3 DataStream=3 Transport=3
ECLPSEvent=3
```

Note The trace parameters should be listed without any new line characters. It is acceptable if the parameters wrap in the text box.

The Telnet/TN3270 server should have logs that may help as well.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses TN3270 to communicate with the RACF adapter.

Required Administrative Privileges

To define or change information in a non-base segment of a user profile, including your own, you must have the SPECIAL attribute or at least UPDATE authority to the segment through field-level access checking.

To list the contents of a user profile or the contents of individual segments of the user profile, use the LISTUSER command.

To display the information in a non-base segment of a user profile, including your own, you must have the SPECIAL or AUDITOR attribute or at least READ authority to the segment through field-level access checking.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	Yes
Pass-through authentication	No
Before/after actions	Yes
Data loading methods	• Import directly from resource • Reconciliation

Account Attributes

The following table provides information about RACF account attributes.

Resource User Attribute	Data Type	Description
GROUPS	String	The groups assigned to the user
GROUP-CONN-OWNERS	String	Group connection owners
USERID. Required	String	Required. The user's name
MASTER CATALOG	String	Master catalog
USER CATALOG	String	User catalog
CATALOG ALIAS	String	Catalog alias
OWNER	String	The owner of the profile
NAME	String	The user's name
DATA	String	Installation-defined data

Resource User Attribute	Data Type	Description
DFLTGRP	String	The user's default group
EXPIRED	Boolean	Indicates whether to expire the password
PASSWORD INTERVAL	String	Password interval
TSO.ACCTNUM	String	The user's default TSO account number at logon
TSO.COMMAND	String	The default command at logon
TSO.HOLDCLASS	String	The user's default TSO hold class
TSO.JOBCLASS	String	The user's default TSO job class
TSO.MAXSIZE	Int	The maximum TSO region size the user can request during logon
TSO.MSGCLASS	String	The user's default TSO message class
TSO.PROC	String	The name of the user's default TSO logon procedure
TSO.SIZE	Int	The minimum TSO region size if the user does not request a region size during logon
TSO.SYSOUTCLASS	String	The user's default TSO SYSOUT class
TSO.UNIT	String	The default name of a TSO device or group of devices that a procedure uses for allocations
TSO.USERDATA	String	Installation-defined data
OMVS.ASSIZEMAX	Int	User's OMVS RLIMIT_AS (maximum address space size)
OMVS.CPUTIMEMAX	Int	User's OMVS RLIMIT_CPU (maximum CPU time)
OMVS.FILEPROCMAX	Int	User's OMVS maximum number of files per process
OMVS.HOME	String	The user's0 OMVS home directory path name
OMVS.MMAPAREAMAX	Int	User's OMVS maximum memory map size
OMVS.PROCUSERMAX	Int	User's OMVS maximum number of processes per UID
OMVS.PROGRAM	String	The user's initial OMVS shell program
OMVS.THREADSMAX	Int	User's OMVS maximum number of threads per process

Resource User Attribute	Data Type	Description
OMVS.UID	String	The user's OMVS user identifier
CICS.OPCLASS	String	The CICS operator classes for which the user will receive BMS (basic mapping support) messages
CICS.OPIDENT	String	The user's CICS operator identifier
CICS.OPPRTY	String	The user's CICS operator priority
CICS.TIMEOUT	String	The amount of time that the user can be idle before being signed off by CICS
CICS.XRFSOFF	String	A setting that indicates whether the user will be signed off by CICS when an XRF takeover occurs
NETVIEW.CONSNAME	String	MCS console identifier
NETVIEW.CTL	String	Specifies GLOBAL, GENERAL, or SPECIFIC control
NETVIEW.DOMAINS	String	Domain identifier
NETVIEW.IC	String	Initial command or list of commands to be executed by NetView when this NetView operator logs on
NETVIEW.MSGRECVR	String	Indicates whether the operator will receive unsolicited messages (NO or YES)
NETVIEW.NGMFADMN	String	Indicates whether this operator can use the NetView graphic monitor facility (NO or YES)
NETVIEW.NGMFVSPN	String	
NETVIEW.OPCLASS	String	Class of the operator

Identity Template

```
$accountId$
```

Sample Forms

Built-In

None

Also Available

`RACFUserForm.xml`

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.RACFResourceAdapter`
- `com.waveset.adapter.HostAccess`

See the Troubleshooting section for the Top Secret adapter on page 1-365 for more information about troubleshooting the HostAccess class.

Red Hat Linux and SuSE Linux

The Red Hat Linux and SuSE Linux resource adapter are two separate adapters defined in the `com.waveset.adapter.RedHatLinuxResourceAdapter` and `com.waveset.adapter.SUSELinuxResourceAdapter` classes, respectively.

The Red Hat Linux adapter supports the following versions:

- Red Hat 8.0, 9.0
- Red Hat Advanced Server 2.1, 3.0, 4.0

The SuSE Linux adapter supports the following version:

- SuSE Enterprise 9

Resource Configuration Notes

If you will be using SSH (Secure Shell) for communication between the resource and Identity Manager, set up SSH on the resource before configuring the adapter.

Identity Manager Installation Notes

No additional installation procedures are required on this resource.

Usage Notes

The Linux resource adapters primarily provide support for the following commands:

- `useradd, usermod, userdel`
- `groupadd, groupmod, groupdel`
- `passwd`

For more information about supported attributes and files, refer to the Linux manual pages for these commands.

When a rename of a user account is executed on a Linux resource, the group memberships are moved to the new user name. The user's home directory is also renamed if the following conditions are true:

- The original home directory name matched the user name.
- A directory matching the new user name does not already exist.

The Bourne-compliant shell (sh, ksh) must be used as the root shell when connecting to a Linux resource.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager can use the following connections to communicate with this adapter:

- Telnet
- SSH (SSH must be installed independently on the resource.)

Required Administrative Privileges

The adapter supports logging in as a standard user, then performing a `su` command to switch to root (or root-equivalent account) to perform administrative activities. Direct logins as root user are also supported.

The adapter also supports the sudo facility, which allows a system administrator to give users (or groups of users) the ability to run some (or all) commands as root or another user.

In addition, if sudo is enabled for a resource, its settings will override those configured on the resource definition page for the root user.

If you are using sudo, you must set the `tty_tickets` parameter to true for the commands enabled for the Identity Manager administrator. Refer to the man page for the sudoers file for more information.

The administrator must be granted privileges to run the following commands with sudo:

User and Group Commands		NIS Commands	Miscellaneous Commands	
• chsh	• passwd	• make	• awk	• ln
• groupadd	• useradd	• ypcat	• cat	• ls
• groupdel	• userdel	• ypmatch	• chmod	• mv
• groupmod	• usermod	• yppasswd	• chown	• ps
• last			• cp	• rm
			• cut	• sed
			• diff	• sort
			• echo	• tail
			• grep	• touch

In addition, the NOPASSWORD option must be specified for each command.

You can use the **Test Connection** button to test whether

- These commands exist in the administrator user's path
- The administrative user can write to /tmp
- The administrative user has rights to run certain commands

Note A test connection can use different command options than a normal provision run.

Provisioning Notes

The following table summarizes the provisioning capabilities of these adapters.

Feature	Supported?
Enable/disable account	Linux does not natively support Identity Manager enable and disable actions. Identity Manager simulates enabling and disabling accounts by changing the user password. The changed password is exposed on enable actions, but it is not exposed on disable actions. As a result, enable and disable actions are processed as update actions. Any before or after actions that have been configured to operate on updates will execute.
Rename account	Yes
Pass-through authentication	Yes
Before/after actions	Yes
Data loading methods	• Import directly from resource • Reconcile with resource

You can define resource attributes to control the following tasks for all users on this resource:

- Create a home directory when creating the user
- Copy files to the user's home directory when creating the user
- Delete the home directory when deleting the user

Account Attributes

The following table lists the Red Hat Linux and SuSE Linux user account attributes.

Notes:

- Attributes are optional unless noted in the description.
- All attributes are Strings.

Resource User Attribute	useradd Equivalent	Description
accountId	*login*	Required. The user's login name.
comment	-c *comment*	The user's full name.
dir	-d *directory*	The user's home directory.
expire	-e *expiration date*	Last date the account can be accessed.
group	-g *group*	The user's primary group.
inactive	-f *days*	Number of days the account can be inactive before it is locked
secondary_group	-G *group*	The user's secondary group or groups.
shell	-s /*Path*	The user's login shell.
time_last_login	Obtained from the last command.	The date and time of the last login. This value is read-only.
uid	-u *User ID*	The user ID, in digit form.

Resource Object Management

Identity Manager supports the following native Linux objects:

Resource Object	Features Supported	Attributes Managed
Group	Create, update, delete, rename, save as	groupName, gid, users

Identity Template

$accountId$

Sample Forms

Built-In

- Red Hat Linux Group Create Form
- Red Hat Linux Group Update Form
- SuSE Linux Group Create Form
- SuSE Linux Group Update Form

Also Available

- RedHatLinuxUserForm.xml
- SUSELinuxUserForm.xml

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.RedHatLinuxResourceAdapter`
- `com.waveset.adapter.SUSELinuxResourceAdapter`
- `com.waveset.adapter.LinuxResourceAdapter`
- `com.waveset.adapter.SVIDResourceAdapter`
- `com.waveset.adapter.ScriptedConnection`

Remedy

The Remedy resource adapter is defined in the `com.waveset.adapter.RemedyResourceAdapter` class.

This adapter supports the following versions of Remedy Help Desk:

- 4.5
- 5.0

Note The Remedy Active Sync adapter (`com.waveset.adapter.RemedyActiveSyncResourceAdapter` has been deprecated as of Identity Manager 5.0 SP1. All features in this adapter are now in the Remedy adapter. Although existing instances of the Remedy Active Sync adapter will still function, new instances of these can no longer be created.

Resource Configuration Notes

If you set the `ARTCPPORT` and `ARRPC` environment variables, then these values will override the values specified in the **`Remedy TCP Port`** and **`Remedy RPC Socket`** resource parameters.

Identity Manager Installation Notes

No additional installation procedures are required on this resource.

Usage Notes

Before Identity Manager 5.5, the Remedy Active Sync adapter used the **`Process to run with changes`** field to determine which process to launch when a change was detected. The process specified in this field is now specified in the Active Sync Resolve Process rule. This rule is required if Active Sync is enabled.

If you do not enable the Active Sync functionality, then the Remedy adapter automates the integration of Remedy tickets into a Identity Manager workflow.

If you use the Active Sync functionality, then the adapter can be configured to support the following features:

- Querying any Remedy ticket schema
- Filtering tickets based on static criteria, such as status = ‘new’.
- Filtering tickets based on dynamic criteria, such as the most-recent fetched.

- Specifying a workflow to be launched for each matching ticket.

With Active Sync, the Remedy adapter uses the **`Update Search Filter`**, **`Last Fetched Conjunction`**, and **`Last Fetched Filter`** resource parameters to determine which tickets are returned. The **`Update Search Filter`** or **`Last Fetched Filter`**, or both, should be used.

The **Update Search Filter** parameter is an optional parameter that contains an executable Remedy search expression. This parameter can contain any valid search expression that can be entered in the Advanced Search Criteria of the Remedy User application. (Valid search expressions can contain fields, selection values, and keywords.) The adapter does not attempt to check the validity of the search expression.

The following examples illustrate search expressions that would work with the Help Desk Cases sample form provided with the Remedy User application.

- `'Status' = "New"`
- `'Case Type' = "Problem"`

Note Remedy field names are enclosed in single quotation marks, while values are enclosed in double quotation marks.

If the **Last Fetched Filter** parameter is used, then the **`Last Fetched`** **Conjunction** parameter must also be specified. The **`Last Fetched Conjection`** parameter may contain one of the following values:

- `AND` — The conditions in the **`Update search filter`** field as well as the **`Last Fetched Filter`** field must be logically True.
- `OR` —The conditions in either the **`Update search filter`** field or the **`Last Fetched Filter`** field must be logically True.

The **Last Fetched Filter** parameter specifies another Remedy search expression, but this expression can contain one or more user attributes defined in Identity Manager. This feature allows you to construct an expression that compares values returned in a previous poll to values returned in the current poll. For example, if the `Case ID+` field on your Remedy form contains an ID that is unique for every ticket, then this value can be compared on each poll. If the value is higher on the current poll than on the previous poll, then return information about the ticket. The following expression illustrates this feature:

```
'Case ID+' > "$(caseId)"
```

The value specified between the parentheses must be a Waveset User Attribute defined on the schema map page. The `$(caseId)` token will be replaced with the value returned on the previous poll. An example value might be `HD0000045`.

Note The first time the adapter polls, the **`Last Fetched Filter`** is not applied, because there are no previously fetched values. The filter will be run in all subsequent polls.

The adapter concatenates the **`Update search filter`**, **`Last Fetched Conjunction`**, and **`Last Fetched Filter`** resource parameters and sends a search expression similar to the following:

```
'Status' = "New" AND 'Case ID+' > "HD00000045"
```

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses Remedy APIs to communicate with the Remedy adapter.

Required Administrative Privileges

The account used to login to the Remedy server must be on the permission list of all Remedy objects accessed by Identity Manager.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Not applicable
Rename account	No

Feature	Supported?
Pass-through authentication	No
Before/after actions	No
Data loading methods	• Active Sync • Import from resource

Account Attributes

The Remedy adapter does not provide default account attributes. Use the following guidelines when adding custom attributes:

- The Waveset User Attribute value can be used in forms and workflows. For the Remedy Active Sync adapter, this value can be used as part of the value specified in the **`Last Fetched Filter`** resource parameter.
- The Resource User Attribute must be a valid Remedy field ID. Every field in a Remedy form must have an integer field ID that is unique within that form.

  To view the ID of field from within Remedy Administrator, open the form and select the field. The field ID is displayed in brackets in the Find Field drop down menu.
- If a Resource User Attribute corresponds to a Remedy Diary field, then the attribute value will be multi-valued. Each value in the value list is in the following format:

  Timestamp User Message

  where:

  Timestamp — An integer indicating the number of seconds since 1970-01-01 UTC.

  User — The Remedy user who added the message to the diary.

  Message — The diary entry.
- To allow the Remedy adapter to change passwords, you must do the following:
  - Select the **`Supports Passwords`** resource parameter.
  - Add an account attribute in the schema map in which the Identity system user attribute name is `password` and the attribute type is encrypted. The resource user attribute must be a Remedy field ID that holds the user password.

Resource Object Management

None

Identity Template

The identity template for Remedy is generated by the Remedy system. Any identity template established through Identity Manager is ignored.

Sample Forms

None

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

`com.waveset.adapter.RemedyResourceAdapter`

In addition, you can set the following Identity Manager logging parameters for the resource instance:

- Log File Path
- Log Level
- Maximum Archives
- Maximum Age Unit
- Maximum Age Length
- Maximum Log File Size

SAP

Identity Manager provides resource adapters for supporting the following versions of SAP:

- SAP R/3 4.5, 4.6, 4.7
- SAP HR 4.5, 4.6, 4.7 (read-only access)

The following table summarizes the attributes of the SAP adapters:

GUI Name	Class Name
SAP	`com.waveset.adapter.SAPResourceAdapter`
SAP HR Active Sync	`com.waveset.adapter.SAPHRActiveSyncAdapter`

Note As of Identity Manager 6.0 and Identity Auditor 2.0, the SAP HR Active Sync account attributes have a new format. The resource user attributes in the schema map are now separated by : (colon) instead of _ (underscore). This allows an attribute from SAP HR to be a path to arbitrarily deep attributes instead of a simple attribute within the infotype. If you are upgrading either of these products from a previous version, the default attributes are renamed by default as part of the update script. The ResourceUpdater will print a message if it had a problem converting an attribute. However, you should review your account attributes to ensure the conversion was successful.

Resource Configuration Notes

This section provides configuration notes that are unique to the SAP resource adapter and to the SAP HR Active Sync adapter. In addition, this section provides configuration instructions that are common to both adapters, including:

- Creating a Logical System
- Assigning a Client to the Logical System
- Creating a Distribution Model
- Registering the RFC Server Module with the SAP Gateway
- Creating a Port Definition
- Generating Partner Profiles
- Modifying the Port Definition
- Generating an IDoc
- Activating Change Pointers

- Scheduling a Job for Change Pointer Processing
- Scheduling a Job
- Testing the Change Pointer Configuration
- Creating a CPIC User

SAP Resource Adapter

The following resource configuration notes are applicable to the SAP resource adapter only.

To enable the ability for a user to change his or her own SAP password as well as associated password data, such as “Password Last Changed Date” and password history, perform the following steps:

1. Set the **`User Provides Password On Change`** resource attribute.
2. Add WS_USER_PASSWORD to both sides of the schema map. You do not need to modify the user form or other forms.

SAP HR Active Sync Adapter

The following resource configuration notes are applicable to the SAP HR Active Sync adapter only.

The SAP Application Link Enabling (ALE) technology enables communication between SAP and external systems, such as Identity Manager. The SAP HR Active Sync adapter uses an outbound ALE interface. In an outbound ALE interface, the base logical system becomes the sender for outbound messages and the receiver of inbound messages. A SAP HR user will likely be logged into the base logical system/client when making changes to the database (for example, hiring an employee, updating position data, terminating an employee, etc.) A logical system/client must also be defined for the receiving client. This logical system will act as the receiver of outbound messages. As for the message type between the two systems, the Active Sync adapter uses a HRMD_A message type. A message type characterizes data being sent across the systems and relates to the structure of the data, also known as an IDoc type (for example, HRMD_A05).

The following steps provide the configurations required on SAP for the Active Sync adapter to receive authoritative feeds from SAP HR:

Note You must configure the SAP system parameters to enable Application Link Enabling (ALE) processing of HRMD_A IDocs. This allows for data distribution between two application systems, also referred to as messaging.

Creating a Logical System

Depending on your current SAP environment, you might not need to create a logical system. You might only need to modify an existing Distribution Model by adding the HRMD_A message type to a previously configured Model View. It is important, however, that you follow SAP's recommendations for logical systems and configuring your ALE network. The following instructions assume that you are creating new logical systems and a new model view.

1. Enter transaction code SPRO, then display the SAP Reference IMGproject (or the project applicable to your organization).
2. Click Basis Components (or SAP Web Application Services on SAP 4.7) > Application Link Enabling (ALE) > Sending and Receiving Systems > Logical Systems > Define Logical System.
3. Click Edit > New Entries.
4. Enter a name and a description for the logical system you want to create (IDMGR).
5. Save your entry.

Assigning a Client to the Logical System

1. Enter transaction code SPRO, then display the SAP Reference IMGproject (or the project applicable to your organization).
2. Click Basis Components (or SAP Web Application Services on SAP 4.7) > Application Link Enabling (ALE) > Sending and Receiving Systems > Logical Systems > Assign Client to Logical System.
3. Select the client.
4. Click GOTO > Details to display the Client Details dialog box.
5. In the Logical System field, enter the logical system you want to assign to this client.
6. Save your entry.

Creating a Distribution Model

To create a distribution model:

1. Verify that you are logged on to the sending system/client.
2. Enter transaction code BD64. Ensure that you are in Change mode.
3. Click Edit > Model View > Create.
4. Enter the short and technical names for your view, as well as the start and end date, then click Continue.

5. Select the view you created, then click Add Message Type.
6. Define the sender/logical system name.
7. Define the receiver/server name.
8. Define the Message Type you want to use (HRMD_A), then click Continue.
9. Click Save.

Registering the RFC Server Module with the SAP Gateway

During initialization, the Active Sync adapter registers with the SAP Gateway. It uses “LIGHTHOUSERFC” for its ID. This value must match the value set in the SAP application. You must configure the SAP application so that the RFC Server Module can create a handle to it. To register the RFC Server Module as an RFC destination:

1. In the SAP application, go to transaction SM59.
2. Expand the TCP/IP connections directory.
3. Click Create (F8).
4. In the RFC destination field, enter the name of the RFC destination system. (LIGHTHOUSERFC).
5. Set the connection type to T (Start an external program via TCP/IP).
6. Enter a description for the new RFC destination, and then click Save.
7. Click the Registration button for the Activation Type.
8. Set the Program ID. We recommend that you use the same value as the RFC destination (LIGHTHOUSERFC), and then click Enter.
9. If the SAP system is a Unicode system, the port must be configured for Unicode. Click the **`Special Options`** tab, and look for the Character Width In Target System section. There is a setting for unicode and non-unicode.
10. Using the buttons at the top - **`Test Connection`** and **`Unicode Test`** - test the connection to the Identity Manager resource. You must have the adapter started for the test to pass.

Creating a Port Definition

The port is the communication channel to which IDocs are sent. The port describes the technical link between the sending and receiving systems. You should configure an RFC port for this solution. To create a port definition:

1. Enter transaction code WE21.
2. Select Transactional RFC, then click the Create icon. Enter LIGHTHOUSERFC for the RFC Destination.

3. Save your changes.

Generating Partner Profiles

The system automatically generates a partner profile or you can manually maintain the profile.

Note If you are using an existing distribution model and partner profile, you do not need to automatically generate a partner profile. Instead, you can modify it to include the HRMD_A message type. To automatically generate a partner profile:

1. Enter transaction code BD82.
2. Select the Model View. This should be the Model View previously created.
3. Ensure the Transfer IDoc immediately and Trigger Immediately radio buttons are selected.
4. Click Execute.

Modifying the Port Definition

When you generated a partner profile, the port definition might have been entered incorrectly. For your system to work properly, you need to modify the port definition.

1. Enter transaction code WE20.
2. Select Partner Type LS.
3. Select your receiving partner profile.
4. Select Outbound Parameters, then click Display.
5. Select message type HRMD_A.
6. Click Outbound Options, then modify the receiver port so it is the RFC port name you created (IDMGR).
7. From the Output Mode, select Transfer IDoc Immediately to send IDocs immediately after they are created.
8. From the IDoc Type section, select HRMD_A05 as basictype (for SAP 4.6).
9. Click Continue/Save.

Generating an IDoc

1. Enter transaction code PFAL.
2. Insert the Object Type P for person objects.
3. Enter an Employee's ID for the Object ID or select a range of employees.

4. Click Execute.
5. Ensure that the status is set to “passed to port okay.”
6. The IDoc has been created. Check the Active Sync adapter log file to verify that an update was received.

Activating Change Pointers

To activate change pointers globally:

1. Enter transaction code BD61.
2. Enable the Change Pointers Active tab.

To activate change pointers for a message type:

1. Enter transaction code BD50.
2. Scroll to the HRMD_A message type.
3. Check the HRMD_A check box, then click Save.

Scheduling a Job for Change Pointer Processing

1. Enter transaction code SE38 to begin defining the variant.
2. Select the RBDMIDOC program, select Variant, then click the Create icon.
3. Name the variant and give it a description (Make note of the variant name so you can use it when scheduling the job).
4. Select the HRMD_A message type, then click Save. You will be prompted to select variant attributes. Select the background processing attribute.
5. Click Save.

Scheduling a Job

1. Enter transaction code SM36.
2. Name the job.
3. Assign Job Class. Job Class is the priority in which jobs are processed. Class A is the highest priority and will be processed first. For a production environment, assign the class to B or C.

4. Schedule a start time. Click the Start Condition tab, then click Date and Time. Enter a scheduled start time, which must be a future event.
   a. Mark the job as a periodic job. Click the Periodic Values tab, schedule how frequently you want the job to run, then press Enter. For testing purposes, setting this period to 5 minutes.
   b. Click Save.
5. Define the job steps.
   a. Enter the ABAP program name: RBDMIDOC.
   b. Select the variant you created in the previous step.
6. Click Save (Note: Click Save once; otherwise, the job will be scheduled to run multiple times).

Testing the Change Pointer Configuration

1. From the SAP client, hire an employee.
2. Ensure that an IDoc was created. You can verify IDoc creation in two locations:
   - Enter transaction code WE02, enter search date parameters and generate a list of generated IDOCs
   - Check the SAP HR Active Sync adapter log

Creating a CPIC User

Users are client-independent. For each SAP HR Active Sync adapter that will be using the driver, a system user with CPIC access must be created.

1. From User Maintenance in SAP, enter a username in the user dialog box, then click the Create icon.
2. Click the Address tab, then enter data in the last name and format fields.
3. Click the Logon Data tab, then define the initial password and set the user type to CPIC.
4. Click the Profiles tab, then add the SAP_ALL, SAP_NEW and S_A.CPIC profiles.
5. Click Save.

Note Initially, you can create a dialog user to test your SAP system configuration. If there are processing problems, you can analyze the dialog user in the debugger. You should also log into the SAP system once to set this user's password. After the system is tested and works properly, you should switch to a CPIC user for security measures.

Identity Manager Installation Notes

The SAP resource adapters are custom adapters. The Oracle and Oracle ERP resource adapters are custom adapters. You must perform the following steps to complete the installation process:

1. Download the JCo (Java Connection) toolkit from `http://service.sap.com/connectors`. (Access to the SAP JCO download pages require a login and password.) The toolkit will have a name similar to `sapjco-ntintel-2.1.4.zip`. This name will vary depending on the platform and version selected.

Note On Solaris, use the 32-bit version of the 2.1.4 (or later) SAP JCO file. Also use the corresponding IDOC libraries.

2. Unzip the toolkit and follow the installation instructions. Be sure to place library files in the correct location and to set the environment variables as directed.
3. Copy the `sapjco.jar` file to the *`InstallDir`*`\idm\WEB-INF\lib` directory.

If you are installing the SAP HR Active Sync adapter, perform these additional steps:

4. Download the SAP Java Base IDoc Class Library. The library will be in a zip file with a name similar to `sapidoc-1.0.1.zip`.
5. Unzip the library and follow the installation instructions.
6. Copy the `sapidoc.jar` file to the *`InstallDir`*`\idm\WEB-INF\lib` directory.
7. Download the SAP Java Connector IDoc Class Library. The library will be in a zip file with a name similar to `sapidocjco-1.0.1.zip`.
8. Unzip the library and follow the installation instructions.
9. Copy the `sapidocjco.jar` file to the *`InstallDir`*`\idm\WEB-INF\lib` directory.

Usage Notes

This section provides information related to using the SAP resource adapter, which is organized into the following sections:

- *General Notes*
- *SAP JCO and RFC Tracing*
- *Active Sync Configuration*

General Notes

The following general notes are provided for the resource:

- To allow editing of to and from dates on a per activity group basis, load the `SAPUserForm_with_RoleEffectiveDates_Timezone.xml` form. This form also provides the ability to select a time zone for the user.
- The `sources.`*`ResourceName`*`.hosts` property in the `waveset.properties` file can be used to control which host or hosts in a cluster will be used to execute the synchronization portion of an Active Sync resource adapter. *`ResourceName`* must be replaced with the name of the Resource object.
- The sample user forms `SAPUserForm.xml` and `SAPUserForm_with_RoleEffectiveDates_Timezone.xml` now contain a definition for a field that pre-expires the user's password. If this field's value is `true`, and an Identity Manager administrator creates or changes a user's password, the user must specify a new password upon logging in to SAP.

SAP JCO and RFC Tracing

The SAPResourceAdapter and the SAPHRActiveSyncAdapter provide resource attributes for SAP JCO and RFC tracing. They can be used to trace Identity Manager's communication with the SAP system. The attributes are JCO Trace Level and JCO Trace Directory.

The following environment variables can be set in the environment to enable SAP RFC tracing. These variables must be set in the environment before starting the application server. They control the shared library that JCO uses to communicate with the SAP system.

- RFC_TRACE: 0 or 1
- RFC_TRACE_DUMP: 0 or 1
- RFC_TRACE_DIR: Path to the directory for the trace files
- CPIC_TRACE_DIR: Path to the directory for the trace files

Note If no JCO tracing is desired, set RFC_TRACE to 0 to ensure that no trace files are created.

Active Sync Configuration

Before Identity Manager 5.5, the SAP HR Active Sync adapter used the **`Process to run with changes`** field to determine which process to launch when a change was detected. The process specified in this field is now specified in the Active Sync Resolve Process rule.

In addition, before Identity Manager 5.5, if the **`Process deletes as updates`** check box was selected, Identity Manager would disable a deleted Identity Manager user as well as all resource accounts and mark the user for later deletion. By default, this check box was selected. In Identity Manager 5.5 and beyond, this functionality is configured by setting the Delete Rule set to None.

If the checkbox was previously deselected, then the Delete Rule will be set to **`ActiveSync has isDeleted set`**.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses BAPI over SAP Java Connector (JCo) to communicate with the SAP adapters.

Required Administrative Privileges

The user name that connects to SAP HR must be assigned to a role that can access the SAP HR users.

Provisioning Notes

The default SAP HR Active Sync adapter is read-only. You cannot use this adapter to create or modify accounts.

Feature	Supported?
Enable/disable account	Basis accounts can be enabled and disabled with the SAP resource adapter. The SAP HR Active Sync adapter cannot enable or disable accounts.
Rename account	No
Pass-through authentication	No
Before/after actions	No
Data loading methods	• Import directly from resource • Active Sync (SAP HR Active Sync adapter only) • Reconciliation

Account Attributes

The following table provides information about SAP and SAP HR Active Sync account attributes.

SAP Attributes

The following attributes are applicable for the SAP resource adapter only.
All attribute types are String.

User Attribute	Resource Attribute Name	Description
accountId	USERNAME->BAPIBNAME	Required. The user's account ID.
firstname	ADDRESS->FIRSTNAME	User's first name
fullname	ADDRESS->FULLNAME	User's first and last name
email	ADDRESS->E_MAIL	User's e-mail address
lastname	ADDRESS->LASTNAME	Required. User's last name
personNumber	ADDRESS->PERS_NO	Internal key for identifying a person
addressNumber	ADDRESS->ADDR_NO	Internal key for identifying an address for central address management
birthName	ADDRESS->BIRTH_NAME	Maiden name or name given at birth

User Attribute	Resource Attribute Name	Description
middleName	ADDRESS->MIDDLENAME	User's middle name
secondLastName	ADDRESS->SECONDNAME	Second last name
academicTitle	ADDRESS->TITLE_ACA1	An academic title, such as Dr. or Prof.
academicTitle2	ADDRESS->TITLE_ACA3	A second academic title
namePrefix	ADDRESS->PREFIX1	A prefix to a last name, such as von, van der, or de la
namePrefix2	ADDRESS->PREFIX1	A second prefix to a last name
nameSupplement	ADDRESS->TITLE_SPPL	Name supplement, for example noble title, such as Lord or Lady
nickname	ADDRESS->NICKNAME	User's nickname
initials	ADDRESS->INITIALS	Middle initial or initials
nameFormat	ADDRESS->NAMEFORMAT	The sequence in which name components are assembled to present the name of a person in a complete form. The sequence can vary for each country.
nameFormatCountry	ADDRESS->NAMCOUNTRY	The country used to determine the name format
languageKey	ADDRESS->LANGU_P	The language used to enter and display text
iso639Language	ADDRESS->LANGUP_ISO	ISO 639 language code
sortKey1	ADDRESS->SORT1_P	A search term
sortKey2	ADDRESS->SORT2_P	A secondary search term
department	ADDRESS->DEPARTMENT	The department in a company as part of the company address
function	ADDRESS->FUNCTION	The user's job functionality
buildingNumber	ADDRESS->BUILDING_P	The building number where the user's office is located
buildingFloor	ADDRESS->FLOOR_P	The floor where the user's office is located
roomNumber	ADDRESS->ROOM_NO_P	The room number where the user's office is located

User Attribute	Resource Attribute Name	Description
correspondenceCode	ADDRESS->INITS_SIG	A correspondence code
inhouseMailCode	ADDRESS->INHOUSE_ML	An internal mail code
communicationType CUA	ADDRESS->COMM_TYPE	States how the user wants to exchange documents and messages with a business partner.
title	ADDRESS->TITLE	A title, such as Mr. or Mrs.
title2	ADDRESS->TITLE_P	A title, such as Mr. or Mrs.
personName	ADDRESS->NAME	Name of an address
personName2	ADDRESS->NAME_2	Second line in a name of an address
personName3	ADDRESS->NAME_3	Third line in a name of an address
personName4	ADDRESS->NAME_4	Fourth line in a name of an address
careOfName	ADDRESS->C_O_NAME	Part of the address if the recipient is different from the occupant (c/o = care of)
city	ADDRESS->CITY	User’s city
district	ADDRESS->DISTRICT	City or district supplement
cityNumber	ADDRESS->CITY_N	City code
districtNumber	ADDRESS->DISTRCT_NO	District code
cityPostalCode	ADDRESS->POSTL_COD1	User’s postal code
cityPostalCode2	ADDRESS->POSTL_COD2	Postal code required for unique assignment of the PO Box.
cityPostalCode3	ADDRESS->POSTL_COD3	Postal code which is assigned directly to a company.
poBox	ADDRESS->PO_BOX	The user’s post office box
poBoxCity	ADDRESS->PO_BOX_CIT	Post office box city
poBoxCityNumber	ADDRESS->PBOXCIT_NO	The PO Box city, if it is different from the address city.
postalDeliveryDistrict	ADDRESS->DELIV_DIS	Postal delivery district
transportZone	ADDRESS->TRANSPZONE	Regional zone of a goods recipient or supplier

User Attribute	Resource Attribute Name	Description
street	ADDRESS->STREET	The user's street
streetCode	ADDRESS->STREET_NO	A street code
streetAbbreviation	ADDRESS->STR_ABBR	A street abbreviation
houseNumber	ADDRESS->HOUSE_NO	The number portion of a street address
houseNumber2	ADDRESS->HOUSE_NO2	A secondary address number
street2	ADDRESS->STR_SUPPL1	Additional address field printed above the Street line.
street3	ADDRESS->STR_SUPPL2	Additional address field printed above the Street line.
street4	ADDRESS->STR_SUPPL3	Additional address field printed below the Street line.
street5	ADDRESS->LOCATION	Additional address field printed below the Street line.
oldBuilding	ADDRESS->BUILDING	Number or ID for the building in a contact person address.
floor	ADDRESS->FLOOR	The floor number of an address
roomNumber	ADDRESS->ROOM_NO	The room number in an address
countryCode	ADDRESS->COUNTRY	The country in an address
countryCodeISO	ADDRESS->COUNTRYISO	The two-letter ISO code for the country in an address
languageKey	ADDRESS->LANGU	The language used to enter and display text
languageKeyISO	ADDRESS->LANGU_ISO	ISO 639 language code
region	ADDRESS->REGION	State or province
sort2	ADDRESS->SORT2	A secondary search term
timeZone	LOGONDATA->TZONE	The time difference of the time zone in hours/minutes relative to the UTC
taxJurisdictionCode	ADDRESS->TAXJURCODE	the tax authority to which taxes must be paid. It is always the city to which the goods were delivered.

User Attribute	Resource Attribute Name	Description
telephoneNumber	ADDRESS->TEL1_NUMBR	Telephone number, including the area code, but no country code
telephoneExtension	ADDRESS->TEL1_EXT	Telephone number extension
faxNumber	ADDRESS->FAX_NUMBER	Fax number, including the area code, but no country code
faxExtension	ADDRESS->FAX_EXTENS	Fax number extension
buildingNumber	ADDRESS->BUILD_LONG	Number or abbreviation of a building in an address.
cuaSystems	SYSTEMS->CUASYSTEMS	Central User Administration system names
profiles	PROFILES->BAPIPROF	Profiles assigned to the user.
activityGroups	ACTIVITYGROUPOBJECTS	Roles assigned to the user.
lastLoginTime	LOGONDATA->LTIME	Read only attribute that lists the most recent login time.

SAP HR Active Sync Attributes

The acccount attributes in the schema map are now separated by **:** (colon) instead of _ (underscore). This allows an attribute from SAP HR to be a path to arbitrarily deep attributes instead of a simple attribute within the infotype.

The basic format of an attribute path is as follows:

infoType:*subType*:*iDocDef*:*attrName*

Note The *iDocDef* (IDoc definition) and *attrName* segments of an attribute path can be expanded.

An example valid attribute path is `0105:MAIL:E2P0105001:ID`. The *infoType* is `0105`, the *subType* is `MAIL`, the *iDocDef* is `E2P0105001` and the *attrName* is `ID`.

If the desired attribute is deeper than the first IDoc definition, an arbitrary number of IDoc definitions can be specified before the *attrName*, as long as each one is separated by the delimiter : (colon). For example, `0002::E2P0002001:E2Q0002002:PERNR` has the following elements:

infoType - 0002

subType - None. If an attribute does not have a subtype, use a null field or blank.

*iDocDef*1 - E2P0002001

iDocDef2 - E2Q0002002

attrName - PERNR

The IDoc Definition object can also be returned as a GenericObject. Using the above example, to get the IDoc Definition of E2Q0002002 as a GenericObject, the resource user attribute would be specified as `0002::E2P0002001:E2Q0002002` in the schema map.

In addition, `[]` (left and right brackets) can be appended to the pathname to indicate the attribute is a list. For example, if it is possible for a particular attribute to have multiple values, that attribute's values will be returned as a list by appending `[]` to the attribute name. This example would be similar to the following:

```
1001:B008:E2P1001001:VARYF[]
```

If the attribute has multiple values but [] is not appended to the attribute name, the last value will be used as the value of the attribute.

By default, the following infotypes are supported:

Infotype	Name	Supported Subtypes
0000	Actions	Not applicable
0001	Organizational Assignment	Not applicable
0002	Personal Data	Not applicable
0006	Addresses	01 (permanent residence), 03 (home residence)
0105	Communication	EMAIL (email address), 0010 (internet address)

Actions Attributes

User Attribute	Resource Attribute Name	Description
actions_end_date	0000::E2P0000001:ENDDA	End date
actions_start_date	0000::E2P0000001:BEGDA	Start date
actions_sequence_number	0000::E2P0000001:SEQNR	Number of Infotype record with same key
actions_last_changed_by	0000::E2P0000001:UNAME	Name of person who changed object

User Attribute	Resource Attribute Name	Description
actions_last_changed	0000::E2P0000001:AEDTM	Last changed on
actions_change_reason	0000::E2P0000001:PREAS	Reason for changing master data
actions_flag1	0000::E2P0000001:FLAG1	Reserved Field/Unused Field
actions_flag2	0000::E2P0000001:FLAG2	Reserved Field/Unused Field
actions_flag3	0000::E2P0000001:FLAG3	Reserved Field/Unused Field
actions_flag4	0000::E2P0000001:FLAG4	Reserved Field/Unused Field
actions_reserved1	0000::E2P0000001:RESE1	Reserved Field/Unused Field of Length 2
actions_reserved2	0000::E2P0000001:RESE2	Reserved Field/Unused Field of Length 2
actions_type	0000::E2P0000001:MASSN	Action type
actions_reason	0000::E2P0000001:MASSG	Reason for action
actions_customer_status	0000::E2P0000001:STAT1	Customer-Specific Status
actions_employment_status	0000::E2P0000001:STAT2	Employment status
actions_special_payment_status	0000::E2P0000001:STAT3	Special payment status

Organizational Assignment Attributes

User Attribute	Resource Attribute Name	Description
org_admingroup	0001::E2P0001001:ADMINGROUP	Administrator Group
org_bus_area	0001::E2P0001001:BUS_AREA	Business Area
org_ch_on	0001::E2P0001001:CH_ON	Last changed on
org_changed_by	0001::E2P0001001:CHANGED_BY	Name of person who changed object
org_cnfrm_flag	0001::E2P0001001:CNFRM_FLAG	Confirmation Fields Exist
org_co_area	0001::E2P0001001:CO_AREA	Controlling Area
org_comp_code	0001::E2P0001001:COMP_CODE	Company Code
org_contract	0001::E2P0001001:CONTRACT	Work Contract
org_costcenter	0001::E2P0001001:COSTCENTER	Cost Center

User Attribute	Resource Attribute Name	Description
org_egroup	0001::E2P0001001:EGROUP	Employee Group
org_esubgroup	0001::E2P0001001:ESUBGROUP	Employee Subgroup
org_flag1	0001::E2P0001001:FLAG1	Reserved Field/Unused Field
org_flag2	0001::E2P0001001:FLAG2	Reserved Field/Unused Field
org_flag3	0001::E2P0001001:FLAG3	Reserved Field/Unused Field
org_flag4	0001::E2P0001001:FLAG4	Reserved Field/Unused Field
org_from_date	0001::E2P0001001:FROM_DATE	Start Date
org_fund	0001::E2P0001001:FUND	Fund
org_funds_ctr	0001::E2P0001001:FUNDS_CTR	Funds Center
org_hist_flag	0001::E2P0001001:HIST_FLAG	Historical Record Flag
org_infotype	0001::E2P0001001:INFOTYPE	Infotype
org_job	0001::E2P0001001:JOB	Job
org_jobtxt	0001::E2P0001001:JOBTXT	
org_leg_person	0001::E2P0001001:LEG_PERSON	Legal Person
org_lock_ind	0001::E2P0001001:LOCK_IND	Lock Indicator for HR Master Data Record
org_name	0001::E2P0001001:NAME	Formatted Name of Employee or Applicant
org_object_id	0001::E2P0001001:OBJECT_ID	Object Identification
org_objecttype	0001::E2P0001001:OBJECTTYPE	Object Type
org_org_key	0001::E2P0001001:ORG_KEY	Organizational Key
org_org_unit	0001::E2P0001001:ORG_UNIT	Organizational Unit
org_orgtxt	0001::E2P0001001:ORGTXT	
org_p_subarea	0001::E2P0001001:P_SUBAREA	Personnel Subarea
org_payarea	0001::E2P0001001:PAYAREA	Payroll Area
org_payr_admin	0001::E2P0001001:PAYR_ADMIN	Payroll Administrator
org_perno	0001::E2P0001001:PERNO	Personnel Number

User Attribute	Resource Attribute Name	Description
org_pers_admin	0001::E2P0001001:PERS_ADMIN	Administrator for HR Master Data
org_pers_area	0001::E2P0001001:PERS_AREA	Personnel Area
org_position	0001::E2P0001001:POSITION	Position
org_postxt	0001::E2P0001001:POSTXT	
org_reason	0001::E2P0001001:REASON	Reason for Changing Master Data
org_ref_flag	0001::E2P0001001:REF_FLAG	Reference Fields Exist (Primary/Secondary Costs)
org_reserved1	0001::E2P0001001:RESERVED1	Reserved Field/Unused Field of Length 2
org_reserved2	0001::E2P0001001:RESERVED2	Reserved Field/Unused Field of Length 2
org_screenctrl	0001::E2P0001001:SCREENCTRL	Infotype Screen Control
org_seqno	0001::E2P0001001:SEQNO	Number of Infotype Record With Same Key
org_sort_name	0001::E2P0001001:SORT_NAME	Employee's Name (Sortable by LAST NAME FIRST NAME)
org_subtype	0001::E2P0001001:SUBTYPE	Subtype
org_supervisor	0001::E2P0001001:SUPERVISOR	Supervisor Area
org_textflag	0001::E2P0001001:TEXTFLAG	Text Exists for Infotype
org_time_admin	0001::E2P0001001:TIME_ADMIN	Administrator for Time Recording
org_to_date	0001::E2P0001001:TO_DATE	End Date

Personal Data Resources

User Attribute	Resource Attribute Name	Description
academicgrade	0002::E2P0002001:ACADEMICGRADE	Academic title
aristrocratictitle	0002::E2P0002001:ARISTROCRATICTITLE	Name supplement, for example noble title, such as Lord or Lady
birthplace	0002::E2P0002001:BIRTHPLACE	Employee’s place of birth

User Attribute	Resource Attribute Name	Description
countryofbirth	0002::E2P0002001:COUNTRYOFBIRTH	Country where the employee was born
dateofbirth	0002::E2P0002001:DATEOFBIRTH	Employee's date of birth
employeeno	0002::E2P0002001:EMPLOYEENO	Required. A personnel number
firstname	0002::E2P0002001:FIRSTNAME	Employee's first name. Required.
formofaddress	0002::E2P0002001:FORMOFADDRESS	Form-of-address key
fullname	0002::E2P0002001:FULLNAME	Full employee name
gender	0002::E2P0002001:GENDER	Indicates the gender of the employee
idnumber	0002::E2P0002001:IDNUMBER	Personnel ID number, such as Social Security Number
initials	0002::E2P0002001:INITIALS	Employee's initials
knownas	0002::E2P0002001:KNOWNAS	Name which the employee prefers to be called.
language	0002::E2P0002001:LANGUAGE	A language key
language_iso	0002::E2P0002001:LANGUAGE_ISO	ISO 639 language code
lastname	0002::E2P0002001:LASTNAME	Employee's last name
maritalstatus	0002::E2P0002001:MARITALSTATUS	Marital status key
maritalstatussince	0002::E2P0002001:MARITALSTATUSSINCE	Validity start date for current marital status
middlename	0002::E2P0002001:MIDDLENAME	Employee's middle name
name_format_indicator	0002::E2P0002001:NAME_FORMAT_INDICATOR	Name Format ID for employee in a list
nameatbirth	0002::E2P0002001:NAMEATBIRTH	Name at birth or second name
nameofcountryofbirth	0002::E2P0002001:NAMEOFCOUNTRYOFBIRTH	Country of birth
nameofformofaddress	0002::E2P0002001:NAMEOFFORMOFADDRESS	Name of form-of-address
nameofgender	0002::E2P0002001:NAMEOFGENDER	Name of gender

User Attribute	Resource Attribute Name	Description
nameoflanguage	0002::E2P0002001:NAMEOFLANGUAGE	Name of language
nameofmaritalstatus	0002::E2P0002001:NAMEOFMARITALSTATUS	Name of marital status
nameofnationality	0002::E2P0002001:NAMEOFNATIONALITY	Name of nationality
nameofreligion	0002::E2P0002001:NAMEOFRELIGION	Name of religion
nameofsecondnationality	0002::E2P0002001:NAMEOFSECONDNATIONALITY	Name of second nationality
nameofstateofbirth	0002::E2P0002001:NAMEOFSTATEOFBIRTH	Name of state of birth
nameofthirdnationality	0002::E2P0002001:NAMEOFTHIRDNATIONALITY	Name of third nationality
nationality	0002::E2P0002001:NATIONALITY	The employee's primary nationality
numberofchildren	0002::E2P0002001:NUMBEROFCHILDREN	The number of children the employee has.
recordnr	0002::E2P0002001:RECORDNR	Number of Infotype Record With Same Key
religion	0002::E2P0002001:RELIGION	A two-character code used to identify a religious denomination.
secondacadgrade	0002::E2P0002001:SECONDACADGRADE	Second academic title
secondname	0002::E2P0002001:SECONDNAME	Second name
secondnameprefix	0002::E2P0002001:SECONDNAMEPREFIX	Second name prefix
secondnationality	0002::E2P0002001:SECONDNATIONALITY	The employee's second nationality
stateofbirth	0002::E2P0002001:STATEOFBIRTH	State or province the employee was born
surnameprefix	0002::E2P0002001:SURNAMEPREFIX	A prefix to a last name, such as von, van der, or de la

User Attribute	Resource Attribute Name	Description
thirdnationality	0002::E2P0002001:THIRDNATIONALITY	Third nationality
validbegin	0002::E2P0002001:VALIDBEGIN	Date employee data becomes valid
validend	0002::E2P0002001:VALIDEND	Date employee data is no longer valid

Addresses Resources

User Attribute	Resource Attribute Name	Description
addresstype_permanent_address	0006:1:E2P0006001:ADDRESSTYPE	Address type of the permanent address
addresstype_home_address	0006:3:E2P0006003:ADDRESSTYPE	Address type of the home address
city_permanent_address	0006:1:E2P0006001:CITY	City of permanent address
city_home_address	0006:3:E2P0006003:CITY	City of home address
coname_permanent_address	0006:1:E2P0006001:CONAME	Care of (c/o) information for the employee's permanent address.
coname_home_address	0006:3:E2P0006003:CONAME	Care of (c/o) information for the employee's home address.
country_permanent_address	0006:1:E2P0006001:COUNTRY	Country code of permanent address
country_home_address	0006:3:E2P0006003:COUNTRY	Country code of home address
district_permanent_address	0006:1:E2P0006001:DISTRICT	District of permanent address
district_home_address	0006:3:E2P0006003:DISTRICT	District of home address
nameofaddresstype_permanent_address	0006:1:E2P0006001:NAMEOFADDRESSTYPE	Address type assigned to permanent address.
nameofaddresstype_home_address	0006:3:E2P0006003:NAMEOFADDRESSTYPE	Address type assigned to home address
nameofcountry_permanent_address	0006:1:E2P0006001:NAMEOFCOUNTRY	Country of permanent address
nameofcountry_home_address	0006:3:E2P0006003:NAMEOFCOUNTRY	Country of home address

User Attribute	Resource Attribute Name	Description
nameofstate_permanent_address	0006:1:E2P0006001:NAMEOFSTATE	Name of the state or province of permanent address
nameofstate_home_address	0006:3:E2P0006003:NAMEOFSTATE	Name of the state or province of home address
postalcodecity_permanent_address	0006:1:E2P0006001:POSTALCODECITY	Postal code city of permanent address
postalcodecity_home_address	0006:3:E2P0006003:POSTALCODECITY	Postal code city of home address
recordnr_permanent_address	0006:1:E2P0006001:RECORDNR	
recordnr_home_address	0006:3:E2P0006003:RECORDNR	
scndaddressline_permanent_address	0006:1:E2P0006001:SCNDADDRESSLINE	Second address line of the permanent address.
scndaddressline_home_address	0006:3:E2P0006003:SCNDADDRESSLINE	Second address line of the home address.
state_permanent_address	0006:1:E2P0006001:STATE	State or province of permanent address
state_home_address	0006:3:E2P0006003:STATE	State or province of home address
streetandhouseno_permanent_address	0006:1:E2P0006001:STREETANDHOUSENO	Street name and number of permanent address
streetandhouseno_home_address	0006:3:E2P0006003:STREETANDHOUSENO	Street name and number of home address
telephonenumber_permanent_address	0006:1:E2P0006001:TELEPHONENUMBER	Primary phone number for permanent address
telephonenumber_home_address	0006:3:E2P0006003:TELEPHONENUMBER	Primary phone number for home address
validbegin_permanent_address	0006:1:E2P0006001:VALIDBEGIN	Date a permanent address becomes valid
validbegin_home_address	0006:3:E2P0006003:VALIDBEGIN	Date a home address becomes valid
validend_permanent_address	0006:1:E2P0006001:VALIDEND	Date a permanent address is no longer valid
validend_home_address	0006:3:E2P0006003:VALIDEND	Date a home address is not longer valid

Communication Resources

User Attribute	Resource Attribute Name	Description
commtype_communication_EMail	0105:0010:E2P0105001:COMMTYPE	Key for communication type (Internet)
commtype_communication_EMail2	0105:MAIL:E2P0105001:COMMTYPE	Key for communication type (E-mail)
delimit_date_communication_EMail	0105:0010:E2P0105001:DELIMIT_DATE	Key date for delimiting an internet address
delimit_date_communication_EMail2	0105:MAIL:E2P0105001:DELIMIT_DATE	Key date for delimiting an Email address
email_communication_EMail	0105:0010:E2P0105001:ID	Internet address
email	0105:MAIL:E2P0105001:ID	Email address
nameofcommtype_communication_EMail	0105:0010:E2P0105001:NAMEOFCOMMTYPE	Name of communication type (internet)
nameofcommtype_communication_EMail2	0105:MAIL:E2P0105001:NAMEOFCOMMTYPE	Name of communication type (e-mail)
recordnr_communication_EMail	0105:0010:E2P0105001:RECORDNR	
recordnr_communication_EMail2	0105:MAIL:E2P0105001:RECORDNR	
validbegin_communication_EMail	0105:0010:E2P0105001:VALIDBEGIN	Date internet address becomes effective
validbegin_communication_EMail2	0105:MAIL:E2P0105001:VALIDBEGIN	Date e-mail address becomes effective
validend_communication_EMail	0105:0010:E2P0105001:VALIDEND	Date internet address expires
validend_communication_EMail2	0105:MAIL:E2P0105001:VALIDEND	Date e-mail address expires

Resource Object Management

Not applicable

Identity Template

$accountId$

Sample Forms

`SAPForm.xml`

`SAPUserForm_with_RoleEffectiveDates_Timezone.xml`

`SAPHRActiveSyncForm.xml`

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.SAPResourceAdapter`
- `com.waveset.adapter.SAPHRActiveSyncAdapter`

To determine which version of the SAP Java Connector (JCO) is installed, and to determine whether it is installed correctly, run the following command:

```
java -jar sapjco.jar
```

The command returns the JCO version as well as the JNI platform-dependent and the RFC libraries that communicate with the SAP system.

If the platform-dependent libraries are not found, refer to the SAP documentation to find out how to correctly install the SAP Java Connector.

SAP Enterprise Portal

The SAP Enterprise Portal adapter is defined in the `com.waveset.adapter.SAPPortalResourceAdapter` class.

This adapter supports the following versions of SAP Enterprise Portal:

• 6.20 SP2+

Identity Manager Installation Notes

The SAP Enterprise Portal adapter does not require any additional installation procedures.

Resource Configuration Notes

The `idmservice.par` portal archive file must be deployed onto the SAP Enterprise Portal. The `idmservice.par` file can be found in the root folder of the install image.

The portal archive defines the `com.sap.portal.prt.soap.IDMService` portal service, which is required by the SAP Enterprise Portal adapter. The adapter communicates with the portal service via SOAP calls to manage the objects on the Portal.

A Portal administrator must install the `idmservice.par`. This is done through the administrative user interface for SAP Enterprise Portal by selecting the `idmservice.par` as the file to upload.

Usage Notes

The SAP Enterprise Portal adapter accomplishes user provisioning by indirectly using the SAP User Management Engine (UME). The adapter communicates with the Identity Manager portal service. The portal service in turn makes direct UME calls.

To communicate with the Identity Manager service installed on the SAP Portal, the **`Identity Manager Portal Service Endpoint`** resource attribute must be configured.

An example endpoint is:

`https://`*`myhost`*`:50000/irj/servlet/prt/soap/com.sap.portal.prt.soap.IDMService`

The **`SAP Portal Administrator`** and **`SAP Portal Administrator Password`** resource attributes define the username and password of an administrator of the SAP Portal.

The **Test Configuration** button verifies that the endpoint, username, and password are valid by performing a status call on the Identity Manager portal service.

Security Notes

To enhance security, configure the following:

- The `com.sap.portal.prt.soap.IDMService` portal service should only be accessible through an SSL-encrypted port exposed by the Portal.
- The `com.sap.portal.prt.soap.IDMService/high_safety` Security Zone should be modified to include only the SAP super_admin role.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	No
Pass-through authentication	Yes
Before/after actions	No
Data loading methods	• Import directly from resource • Reconcile with resource

Account Attributes

The following table lists the SAP Enterprise Portal user account attributes. Unless otherwise noted, the data type for all account attributes is String.

Identity Manager User Attribute	Resource User Attribute	Description
sap_groups	groups	SAP groups in which the user is a direct member
sap_roles	roles	SAP roles in which the user is a directory member
title	title	The user’s academic title or title of nobility
firstname	firstName	The user’s first name
lastname	lastName	The user’s last name
fullname	displayName	The user’s display name
email	email	The user’s default email address
telephone	telephone	The user’s default telephone number
fax	fax	The user’s default fax number
cellPhone	cellPhone	The user’s default cell phone number
street	street	The street of the user’s home address
city	city	The city of the user’s home address
state	state	The state or province of the user’s home address
zipcode	zip	The postal code of the user’s home address
country	country	The ISO-3166 two-letter uppercase code of the country where the user lives. This value does not necessarily match the country specified in the locale.
timeZone	timeZone	The user’s time zone.
locale	locale	The user's locale, such as en_US or fr_CA.
currency	currency	The three letter uppercase code of the user's currency, such as USD, EUR, or YEN
screenReader	screenReader	Boolean. Enables or disables the user's screen reading capability.
department	department	The user’s department
jobTitle	jobTitle	The user’s job title
salutation	salutation	The user’s form of address, such as Mr., Mrs., or Dr.

Resource Object Management

SAP Groups and Roles are supported.

Identity Template

$accountId$

Sample Forms

A sample form is available at `sample/forms/SAPPortalUserForm.xml` is available. When this sample form is used, you must also import `sample/rules/SAPPortalUserFormRules.xml`.

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

`com.waveset.adapter.SAPPortalResourceAdapter`

Additionally, you can set the following Identity Manager logging parameters for the resource instance:

- Log File Path
- Maximum Log File Size
- Log Level

To view the log for the portal service on the SAP Enterprise Portal server, see the `WEB-INF/portal/logs/idm.log` file on the SAP server installation file

The portal service uses the logger `idm_logger`, which is defined in the PAR in the `PORTAL-INF/logger/logger.xml` file. By default, the `idm_logger` is set to log ALL messages

Scripted Gateway

The Scripted Gateway adapter manages a resource that is controlled by batch files that are run on the Sun Identity Manager Gateway. This adapter is a general purpose adapter, and is therefore highly configurable.

This adapter is defined in the `com.waveset.adapter.ScriptedGatewayResourceAdapter` class.

Resource Configuration Notes

None

Identity Manager Installation Notes

To add the Scripted Host resource to the Identity Manager resources list, you must add the following value in the Custom Resources section of the Configure Managed Resources page.

```
com.waveset.adapter.ScriptedGatewayResourceAdapter
```

The Sun Identity Manager Gateway (`gateway.exe`) must be installed on the host specified in the **Host** field for the adapter.

Usage Notes

Resource Actions

The Scripted Gateway adapter allows you to create a set of actions that perform basic provisioning functions such as creating, updating, deleting, and retrieving user accounts. Each of these actions is defined in a Windows batch file.

The adapter supports the following provisioning actions:

Action	Purpose	Required?
create	Creates a new user.	No, but if not provided, users cannot be created.
delete	Deletes an existing user.	No, but if not provided, users cannot be deleted.

Action	Purpose	Required?
getAllUsers	Gets information about all users on the resource	No, but if not provided, operations that depend on account iteration, such as reconciliation and Load From Resource will not be available.
getUser	Fetches attributes for an existing user.	Yes.
update	Updates attributes for an existing user.	No, but if not provided, users cannot be updated.

The `$WSHOME/sample/ScriptedGateway` directory contains a set of sample resource action definitions that could be used to provision users to a theoretical gateway script-based host application. You must customize these definitions to your environment.

For general information about resource actions, see “Adding Actions to Resources” on page 1.

Scripts

The Scripted Gateway adapter implements actions as as batch files that execute on the gateway. These scripts must be written to run on the version of Windows that has been installed on the machine running the scripts. The same account that runs the Gateway also runs the scripts.

Scripts should follow Windows conventions and exit with a return code of 0, which indicates success. Returning a non-zero code (chosen by the script writer) indicates the operation may not have been correctly completed.

Scripts may output text to the Windows standard error or standard output stream. Depending on the nature of the operation, the context of the operation, and the type of failure, the text may be displayed in the results for that operation.

For the getUser and getAllUsers operations, this text is parsed in the standard output stream to determine the attributes of each user.

The following types of environment variables can be exported to the scripts:

- Any account attribute defined in the Identity System Resource Attribute column of the schema map can be made available to the script by prefixing the account attribute with `WSUSER_`. For example, if an account attribute is named Full Name, the environment variable is named `WSUSER_Full Name`.

- Adapter configuration settings can be passed with environment variables that begin with `WSRSRC_`. The most important variable is `WSRSRC_Name`, which defines the name of the adapter. If you are running the same script on different resources, this variable can be implemented to avoid maintaining multiple copies of scripts that do the same thing on different gateways.
- The `WSOBJ_ID` and `WSOBJ_NAME` variables define the account ID and name, respectively. These variables are available to the Scripted Gateway adapter only.

The following example illustrates an example generated environment:

```
WSUSER_Email=testuser@waveset.com
WSUSER_First Name=JUnit
WSUSER_Full Name=JUnit TestUser
WSUSER_Last Name=TestUser
WSUSER_User ID=USER5647
WSUSER_ws_action_type=WindowsBatch
WSOBJ_ID=testuser
WSOBJ_NAME=testuser
WSRSRC_NAME=Scripted Gateway
WSRSRC_CLASS=com.waveset.adapter.ScriptedGatewayResourceAdapter
WSRSRC_Host=localhost
WSRSRC_List Objects Timeout=900000
WSRSRC_Request Timeout=30000
WSRSRC_TCP Port=9278
WSRSRC_connectionLimit=10
```

Generally, if an attribute's value is null, the corresponding environment variable may be omitted instead of having a value of a 0-length string.

For more information about the variables available in a script, see “Adding Actions to Resources” on page 1.

Result Handling

The AttrParse mechanism processes the results returned by the getUser and getAllUsers actions via the standard output stream. See “Implementing the AttrParse Object” on page 1 for details about implementing AttrParse objects.

For getUser actions, AttrParse returns a map of user attributes. For the getAllUsers action, it generates a map of maps. Each entry for the returned map contains the following.

- A value that is a map of user attributes like normally returned by AttrParse.
- A key that is the account ID, or if that is not known, the name.

The collectCsvHeader and collectCsvLines AttrParse tokens must be used to determine attributes and values. Do not use other AttrParse tokens that perform similar operations.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

The Sun Identity Manager Gateway is required.

Required Administrative Privileges

The administrative account that the gateway runs under must be authorized for all operations defined in the scripts.

Provisioning Notes

The following table summarizes the provisioning capabilities of the Scripted Gateway adapter.

Feature	Supported?
Create account	Yes
Update account	Yes
Delete account	Yes
Enable/disable account	Yes
Rename account	No
Pass-through authentication	No
Before/after actions	No
Data loading methods	If the getAllUsers action is defined, then the following data loading methods are supported: Import directly from resource Reconciliation

Account Attributes

The Scripted Gateway adapter does not provide default account attributes because the account attributes vary greatly.

You must define an account attribute in which the Identity System user attribute is named `accountId`.

Resource Object Management

Not supported.

Identity Template

None. You must supply the identity template with a valid value.

Sample Forms

None

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

`com.waveset.adapter.ScriptedGatewayResouceAdapter`

Scripted Host

The Scripted Host resource adapter supports management of application user accounts on an OS/390 mainframe over the IBM Host Access Class Library APIs. The adapter manages host applications over a TN3270 emulator session.

This adapter is a general purpose adapter, and is therefore highly configurable. The adapter makes no assumptions about the host application being managed, and instead relies on calling out to a set of customer-supplied scripts to perform the interactions with the host application.

The Scripted Host resource adapter is defined in the `com.waveset.adapter.ScriptedHostResourceAdapter` class.

Resource Configuration Notes

None

Identity Manager Installation Notes

The Scripted Host resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. To add the Scripted Host resource to the Identity Manager resources list, you must add the following value in the Custom Resources section of the Configure Managed Resources page.

   ```
   com.waveset.adapter.ScriptedHostResourceAdapter
   ```

2. The Identity Manager mainframe adapters use the IBM Host Access Class Library (HACL) to connect to the mainframe. The HACL is available in IBM Websphere Host On-Demand (HOD). The recommended jar containing HACL is `habeans.jar` and is installed with the HOD Toolkit (or Host Access Toolkit) that comes with HOD. The supported versions of HACL are in HOD V7.0, V8.0, and V9.0.

   However, if the toolkit installation is not available, the HOD installation contains the following jars that can be used in place of the `habeans.jar`:

   - `habase.jar`
   - `hacp.jar`
   - `ha3270.jar`
   - `hassl.jar`

- `hodbase.jar`

Copy the `habeans.jar` file or all of its substitutes into the `WEB-INF/lib` directory of your Identity Manager installation. See http://www.ibm.com/software/webservers/hostondemand/ for more information.

3. The Scripted Host adapter requires customer-supplied Javascripts. These scripts must be compatible with Mozilla Rhino. Mozilla Rhino v1_5R2 ships with Identity Manager and is located at `$WSHOME/WEB-INF/lib/javascript.jar`.

   If you need improved Javascript error reporting capability, the latest version of Mozilla Rhino (http://www.mozilla.org/rhino/) offers great improvement in the messages generated for syntax errors and other errors. The default `javascript.jar` may be replaced with a newer version from Mozilla.

Usage Notes

This section provides information related to using the Scripted Host resource adapter, which is organized into the following sections:

- *Administrators*
- *Specifying Resource Actions*
- *SSL Configuration*

Administrators

Host resource adapters *do not* enforce maximum connections for an affinity administrator across multiple host resources connecting to the same host. Instead, the adapter enforces maximum connections for affinity administrators within each host resource.

If you have multiple host resources managing the same system, and they are currently configured to use the same administrator accounts, you might have to update those resources to ensure that the same administrator is not trying to perform multiple actions on the resource simultaneously.

Specifying Resource Actions

The Resource Parameters page of the resource wizard for the Scripted Host adapter contains a set of text boxes that allow you to specify a resource action for various provisioning actions, such as login, create, delete, and iterate. These fields refer to ResourceAction objects that contain Rhino Javascript and loaded into the repository.

At run-time, the adapter does the following:

1. Loads the Javascript from the ResourceAction corresponding to the current provisioning action.
2. Prepares the necessary Java input objects to make available to the Javascript.
3. Invokes the Javascript.
4. Processes the result returned (or exceptions and errors) from the Javascript.

The `$WSHOME/sample/ScriptedHost/ScreenSampleActions.xml` file contains a set of sample resource action definitions that could be used to provision users to a theoretical screen-based host application. You will need to customize these definitions to your application.

The Scripted Host adapter supports end-user scripting for the following provisioning actions:

Action	Description	Required?
create	Create a new user.	No, but if not provided, users cannot be created.
delete	Delete an existing user.	No, but if not provided, users cannot be deleted.
disable	Disable an existing user.	No, but if not provided, users cannot be disabled.
enable	Enable an existing user.	No, but if not provided, users cannot be enabled.
getAccountIterator	Return an object used to perform iteration of existing users.	No, but if neither getAccountIterator nor listAll is provided, account iteration cannot be performed.
getUser	Fetch attributes for an existing user.	Yes.
login	Login to application.	Yes.
logoff	Logoff application.	Yes.
listAll	Return a list of existing user IDs.	No, but if neither getAccountIterator nor listAll is provided, account iteration cannot be performed.
update	Update attributes for an existing user.	No, but if not provided, users cannot be updated.

Every action script receives an `actionContext` map, as defined by the java.util.Map class. The possible contents of the map vary for each action. The following sections describe each action, and provide the following information about the action:

- Context — Describes the set of entries available in the `actionContext` map added into the Javascript execution context by the adapter before the script executes.
- Error Handling — Notes describing how the script is expected to handle abnormal or error conditions

For additional information about the actions listed in the previous table, see the following sections:

- “create Action” on page 280
- “delete Action” on page 281
- “disable Action” on page 282
- “enable Action” on page 283
- “getAccountIterator Action” on page 284
- “getUser Action” on page 285
- “listAll Action” on page 287
- “login Action” on page 288
- “logoff Action” on page 289
- “update Action” on page 290

create Action

The create action creates a user in the host application. If the create action is not defined, then new users cannot be added to the host application.

Context

The actionContext map will contain the following entries:

Key	Value Type	Value Description
`hostAccess`	com.waveset.adapter.HostAccess	Provides 3270 emulation access to a mainframe.
`adapter`	com.waveset.object.ScriptedHost ResourceAdapter	Adapter instance.
`action`	java.lang.String	The string `create`.
`id`	java.lang.String	Account ID of the user to create.

Key	Value Type	Value Description
password	java.lang.String	If present, this is the decrypted password for the new user.
attributes	java.lang.Map	Map of attributes to set for the new user. The key identifies the attribute to set, and the value is the decrypted value to which the attribute should be set.
errors	java.util.List	This is initially an empty list. The script must add java.lang.String objects to this list if any errors are found during processing.
trace	com.waveset.adapter.Trace	An object used to trace execution. Scripts can use methods of this class to make itself “debuggable” in a customer environment.

Error Handling

If any application-specific errors are found in a screen or response, the script should add appropriate strings to the `errors` key. Determining that an error has occurred may require a string search for various known error strings.

The presence of any items in the `errors` List is considered a creation failure. Additionally, any throw from within the script is considered a creation failure.

delete Action

The delete action deletes a specified user from the host application. If no delete action is defined, then users cannot be deleted from the host application.

Context

The actionContext map will contain the following entries:

Key	Value Type	Value Description
id	java.lang.String	Account ID of the user to delete.
hostAccess	com.waveset.adapter.HostAccess	Provides 3270 emulation access to a mainframe.
adapter	com.waveset.object.ScriptedHostResourceAdapter	Adapter instance

Key	Value Type	Value Description
`action`	java.lang.String	The string `delete`.
`trace`	com.waveset.adapter.Trace	An object used to trace execution. Scripts can use methods of this class to make itself “debuggable” in a customer environment.
`errors`	java.util.List	This is initially an empty list. The script must add java.lang.String objects to this list if any errors are found during processing.

Error Handling

If any application-specific errors are found in a screen or response, the script should add appropriate strings to the `errors` key. Determining that an error has occurred may require a string search for various known error strings.

The presence of any items in the `errors` List is considered a deletion failure. Additionally, any throw from within the script is considered a deletion failure.

disable Action

The disable action disables an existing user within the host application. If this action is not defined, then users on the host application cannot be disabled.

Context

The actionContext map will contain the following entries:

Key	Value Type	Value Description
`hostAccess`	com.waveset.adapter.HostAccess	Provides 3270 emulation access to a mainframe.
`action`	java.lang.String	The string `disable`.

Key	Value Type	Value Description
`id`	java.lang.String	The account ID to disable
`errors`	java.util.List	This is initially an empty list. The script must add java.lang.String objects to this list if any errors are found during processing.
`trace`	com.waveset.adapter.Trace	An object used to trace execution. Scripts can use methods of this class to make itself “debuggable” in a customer environment.

Error Handling

If any application-specific errors are found in a screen or response, the script should add appropriate strings to the `errors` key. Determining that an error has occurred may require a string search for various known error strings.

The presence of any items in the `errors` List is considered a disablement failure. Additionally, any throw from within the script is considered a disablement failure.

enable Action

The enable action enables an existing user within the host application. If this action is not defined, then users on the host application cannot be enabled.

Context

The actionContext map will contain the following entries:

Key	Value Type	Value Description
`hostAccess`	com.waveset.adapter.HostAccess	Provides 3270 emulation access to a mainframe.
`action`	java.lang.String	The string `enable`.

Key	Value Type	Value Description
`id`	java.lang.String	Account ID to enable.
`errors`	java.util.List	This is initially an empty list. The script must add java.lang.String objects to this list if any errors are found during processing.
`trace`	com.waveset.adapter.Trace	An object used to trace execution. Scripts can use methods of this class to make itself “debuggable” in a customer environment.

Error Handling

If any application-specific errors are found in a screen or response, the script should add appropriate strings to the `errors` key. Determining that an error has occurred may require a string search for various known error strings.

The presence of any items in the `errors` List is considered an enablement failure. Additionally, any throw from within the script is considered an enablement failure.

getAccountIterator Action

The getAccountIterator action returns an object used to perform iteration of existing users.

If you wish to perform account iteration (reconciliation, Load From Resource), either this action or the listAll action must be defined.

If the getAccountIterator action is not defined, then account iteration will be performed by calling listAll, and then calling getUser for each ID in the list from listAll.

If the getAccountIterator action is not defined and the listAll action is not defined, then account iteration is not supported.

Inputs

The actionContext map will contain the following entries:

Key	Value Type	Value Description
`hostAccess`	com.waveset.adapter.HostAccess	Provides 3270 emulation access to a mainframe.

Key	Value Type	Value Description
`adapter`	com.waveset.object.ScriptedHost ResourceAdapter	Adapter instance
`action`	java.lang.String	The string `getAccountIterator`.
`trace`	com.waveset.adapter.Trace	An object used to trace execution. Scripts can use methods of this class to make itself “debuggable” in a customer environment.

Return Value

The script must return a Java object that implements the Java interface com.waveset.adapter.ScriptedHostAccessAdapter.ObjectIterator.

```
public interface ObjectIterator {
   public boolean hasNext();
   public void next(java.util.Map nextObj);
   public void close();
}
```

The `nextObj` Map argument to the next() method is to be populated by the script in the same manner as the `result` entry discussed in the getUser action.

Error Handling

Any throw from within the script is considered an iteration failure.

Any thrown exceptions encountered while invoking methods on the Java object returned from the script are also considered iteration failures.

getUser Action

The getUser action retrieves one of the following from the host application:

- A string of screens or responses from which the adapter can parse the user attributes for a given user.
- A map of user attributes for a given user.

The getUser action must be defined.

Context

The actionContext map will contain the following entries:

Key	Value Type	Value Description
`hostAccess`	com.waveset.adapter.HostAccess	Provides 3270 emulation access to a mainframe.
`adapter`	com.waveset.object.ScriptedHostResourceAdapter	Adapter instance
`action`	java.lang.String	The string `getUser`.
`attrsToGet`	java.util.List	List of strings identifying the user attributes to be fetch. This list is derived from the right-hand side of the schema map.
`id`	java.lang.String	Account ID of the user to fetch
`errors`	java.util.List	This is initially an empty list. The script must add java.lang.String objects to this list if any errors are found during processing.
`trace`	com.waveset.adapter.Trace	An object used to trace execution. Scripts can use methods of this class to make itself “debuggable” in a customer environment.
`result`	java.util.Map	The script adds entries to the map to return user attributes. See the entry table below.

The `result` map is expected to be populated by the script with the following entries:

Key	Value Type	Value Description
`text`	String	Contains the text to be parsed for the user attributes. This may be the contents of one or more screens or responses. The user attributes will be extracted from this string later using the AttrParse object named in the `attrParse` entry of this map. Do not put this entry into the map if no matching user is found. Do not add this field to the map. Populate the `attrMap` map instead.
`attrParse`	String	Name of an AttrParse object which will be used by the adapter to parse user attributes from the string found in the `text` entry of this map. Set this entry only in combination with setting the `text` entry.
`attrMap`	java.util.Map	If the script is capable of directly retrieving the user attributes, then the script can set this entry with a map of the user attributes. Note that this `attrMap` entry is respected by the adapter only if the `text` entry of this map is not present.

Error Handling

If there is no matching user found, then the result map should be left empty.

If any application-specific errors are found in a screen or response, the script should add appropriate strings to the `errors` key. Determining that an error has occurred may require a string search for various known error strings.

The presence of any items in the `errors` List is considered a retrieval failure. Additionally, any throw from within the script is considered a retrieval failure.

listAll Action

The listAll action retrieves a list of user IDs found for the host application.

If the listAll action is not defined, then you cannot call the FormUtil.listResourceObjects methods for this resource instance from a form.

If the listAll action is not defined and the getAccountIterator action is not defined, then account iteration (reconciliation, Load From Resource) is not supported.

Context

The actionContext map will contain the following entries:

Key	Value Type	Value Description
`hostAccess`	com.waveset.adapter.HostAccess	Provides 3270 emulation access to a mainframe.
`adapter`	com.waveset.object.ScriptedHostResourceAdapter	Adapter instance
`action`	java.lang.String	The string `listAll`.
`resultList`	java.util.List	The script adds entries to this list. Each item added to the list by the script should be a string corresponding to a host account ID.
`errors`	java.util.List	This is initially an empty list. The script must add java.lang.String objects to this list if any errors are found during processing.
`trace`	com.waveset.adapter.Trace	An object used to trace execution. Scripts can use methods of this class to make itself “debuggable” in a customer environment.

Error Handling

If any application-specific errors are found in a screen or response, the script should add appropriate strings to the `errors` key. Determining that an error has occurred may require a string search for various known error strings.

The presence of any items in the `errors` List is considered a retrieval failure. Additionally, any throw from within the script is considered a retrieval failure.

login Action

The login action negotiates an authenticated session with the host required to manage users in the custom host application. This action must be defined.

Context

The actionContext map will contain the following entries:

Key	Value Type	Value Description
`hostAccess`	com.waveset.adapter.HostAccess	Provides 3270 emulation access to a mainframe.
`action`	java.lang.String	The string `login`.
`user`	java.lang.String	User name of the host application admin user.
`password`	com.waveset.util.EncryptedData	Encrypted object that stores the password of the host application admin user. Use `decryptToString()` to convert to plain text.
`errors`	java.util.List	This is initially an empty list.The script must add java.lang.String objects to this list if any errors are found during processing.
`trace`	com.waveset.adapter.Trace	An object used to trace execution. Scripts can use methods of this class to make itself “debuggable” in a customer environment.

Error Handling

If any application-specific errors are found in a screen or response, the script should add appropriate strings to the `errors` key. Determining that an error has occurred may require a string search for various known error strings.

The presence of any items in the `errors` List is considered a login failure. Additionally, any throw from within the script is considered a login failure.

logoff Action

The logoff action performs a disconnect from the host. This is called when the connection is no longer required. This action must be defined.

Context

The actionContext map will contain the following entries:

Key	Value Type	Value Description
`hostAccess`	com.waveset.adapter.HostAccess	Provides 3270 emulation access to a mainframe.
`action`	java.lang.String	The string `logoff`.
`errors`	java.util.List	This is initially an empty list. The script must add java.lang.String objects to this list if any errors are found during processing.
`trace`	com.waveset.adapter.Trace	An object used to trace execution. Scripts can use methods of this class to make itself “debuggable” in a customer environment.

Error Handling

If any application-specific errors are found in a screen or response, the script should add appropriate strings to the `errors` key. Determining that an error has occurred may require a string search for various known error strings.

The presence of any items in the `errors` List is considered a logoff failure. Additionally, any throw from within the script is considered a logoff failure.

update Action

The update action updates a user in the host application. If the update action is not defined, then users on the host application cannot be updated.

Context

The actionContext map will contain the following entries:

Key	Value Type	Value Description
`hostAccess`	com.waveset.adapter.HostAccess	Provides 3270 emulation access to a mainframe.
`adapter`	com.waveset.object.ScriptedHostResourceAdapter	Adapter instance
`action`	java.lang.String	The string `update`.
`id`	java.lang.String	Account ID of the user to modify

Key	Value Type	Value Description
password	java.lang.String	If present, this is the new decrypted password for the user.
attributes	java.lang.Map	Map of attributes to update on the existing user. The key identifies the attribute to set, and the value is the decrypted value to which the attribute should be set.
errors	java.util.List	This is initially an empty list. The script must add java.lang.String objects to this list if any errors are found during processing.
trace	com.waveset.adapter.Trace	An object used to trace execution. Scripts can use methods of this class to make itself “debuggable” in a customer environment.

Error Handling

If any application-specific errors are found in a screen or response, the script should add appropriate strings to the `errors` key. Determining that an error has occurred may require a string search for various known error strings.

The presence of any items in the `errors` List is considered an update failure. Additionally, any throw from within the script is considered an update failure.

SSL Configuration

This section provides instructions for the following:

- *Connecting the Adapter to a Telnet/TN3270 Server using SSL or TLS.*
- *Generating a PKCS #12 File*
- *Troubleshooting*

Connecting the Adapter to a Telnet/TN3270 Server using SSL or TLS.

Use the following steps to connect Scripted Host resource adapters to a Telnet/TN3270 server using SSL/TLS.

1. Obtain the Telnet/TN3270 server's certificate in the PKCS #12 file format. Use `hod` as the password for this file. Consult your server's documentation on how to export the server's certificate. The procedure "Generating a PKCS #12 File" below for some general guidelines.
2. Create a `CustomizedCAs.class` file from the PKCS #12 file. If you are using a recent version of HOD, use the following command to do this.

   ```
   ..\hod_jre\jre\bin\java -cp ../lib/ssliteV2.zip;../lib/sm.zip
   com.ibm.eNetwork.HOD.convert.CVT2SSLIGHT CustomizedCAs.p12 hod
   CustomizedCAs.class
   ```

3. Place the `CustomizedCAs.class` file somewhere in the Identity Manager server's classpath, such as `$WSHOME/WEB-INF/classes`.
4. If a resource attribute named **`Session Properties`** does not already exist for the resource, then use the BPE or debug pages to add the attribute to the resource object. Add the following definition in the `<ResourceAttributes>` section:

   ```
   <ResourceAttribute name='Session Properties' displayName='Session
   Properties' description='Session Properties' multi='true'>
   </ResourceAttribute>
   ```

5. Go to the Resource Parameters page for the resource and add the following values to the **`Session Properties`** resource attribute:

   ```
   SESSION_SSL
   true
   ```

Generating a PKCS #12 File

The following procedure provides a general description of generating a PKCS #12 file when using the Host OnDemand (HOD) Redirector using SSL/TLS. Refer to the HOD documentation for detailed information about performing this task.

6. Create a new `HODServerKeyDb.kdb` file using the IBM Certificate Management tool. As part of that file, create a new self-signed certificate as the default private certificate.

   If you get a message that is similar to “error adding key to the certificate database” when you are creating the `HODServerKeyDb.kdb` file, one or more of the Trusted CA certificates may be expired. Check the IBM website to obtain up-to-date certificates.
7. Export that private certificate as Base64 ASCII into a `cert.arm` file.
8. Create a new PKCS #12 file named `CustomizedCAs.p12` with the IBM Certificate Management tool by adding the exported certificate from the `cert.arm` file to the Signer Certificates. Use `hod` as the password for this file.

Troubleshooting

You can enable tracing of the HACL by adding the following to the Session Properties resource attribute:

```
SESSION_TRACE
ECLSession=3 ECLPS=3 ECLCommEvent=3 ECLErr=3 DataStream=3 Transport=3
ECLPSEvent=3
```

Note The trace parameters should be listed without any new line characters. It is acceptable if the parameters wrap in the text box.

The Telnet/TN3270 server should have logs that may help as well.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses TN3270 to communicate with the Scripted Host adapter.

Required Administrative Privileges

The Identity Manager administrators that connect to the host application must be assigned sufficient privileges to create and manage users within the host application.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	No
Create account	Yes
Update account	Yes
Delete account	Yes
Pass-through authentication	No
Before/after actions	Yes
Data loading methods	• Import directly from resource • Reconciliation

Account Attributes

The Scripted Host adapter does not provide default account attributes, because the account attributes will vary, depending on the host application being managed.

Resource Object Management

Not supported

Identity Template

$accountId$

Sample Forms

None

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.ScriptedHostResourceAdapter`
- `com.waveset.adapter.HostAccess`

See the Troubleshooting for the Top Secret adapter for more information about troubleshooting the HostAccess class.

There is always a `com.waveset.adapter.Trace` object passed in the context to the Javascripts. Enabling trace on `com.waveset.adapter.ScriptedHostResourceAdapter` will enable tracing in the Javascripts.

Additionally, for temporary tracing to stdout, the Javascripts can make calls to the Java `System.out.println()` method. For example:

```
java.lang.System.out.println("Hello World");
```

SecurID ACE/Server

Identity Manager provides resource adapters for supporting the following versions of RSA SecurID ACE/Server:

- 5.0, 6.0 for Windows
- 5.1, 6.0 for UNIX

The following table summarizes the attributes of these adapters:

GUI Name	Class Name
SecurID ACE/Server	`com.waveset.adapter.SecurIdResourceAdapter`
SecurID ACE/Server UNIX	`com.waveset.adapter.SecurIdUnixResourceAdapter`

Resource Configuration Notes

If SecurID is installed on Windows, the adapter will interface with the apidemon that is shipped with the installed version of RSA ACE/Server. Copy the apidemon from the ACE/Server installation directory (by default, `c:\ace\utils\toolkit\apidemon.exe`) to `c:\winnt\system32` or `c:\windows\system32`.

The UNIX adapter uses the RSA ACE/Server Administration Toolkit TCL API. This API must be located in the *`ACEInstallDir`*`/utils/tcl/bin` directory. The value of *`ACEInstallDir`* is specified as a resource parameter. The toolkit must be configured as described in the *Customizing Your RSA ACE/Server Administration* publication provided by RSA.

In addition, ensure that the following conditions are true so that you can manage RSA Users and other ACE database objects via Identity Manager:

- The SecurID user name specified in the **`Administrator Login`** (on the Windows adapter) or the **`Login User`** (on the UNIX adapter) resource parameter exists in the ACE/Server. If not, create an ACE user with the same default login name.
- This SecurID user can login to the ACE/Server with a password instead of a tokencode. Set the RSA ACE Server user's password to the same value specified on the adapter.

  If the current RSA ACE Server system policy does not allow a password to be set using the characters you need (for example, an alphanumeric PIN), or if you need to change the default setting for user password expiration, edit the system parameters on the RSA ACE Server Database console.

A password changed via the RSA ACE Server administrator console is a one-time password that will expire the first time this user logs in. Use the RSA ACE Agent Test Authentication facility to login so that you can change the user's password to one that will not expire immediately. Note that you may change it to the same value, so it's still the same as the password specified in the resource adapter.

- On Windows, an RSA ACE Agent Host must be added for the host where the Identity Manager gateway is running. This can be configured from the Database Administration - Host Mode console interface on the system where the RSA ACE Server is running. You must configure the DNS host name and network address, and you must specify which users have access. In addition, the agent type must be set to Net OS Agent.
- If a SecurId group name or site name contains a comma, Identity Manager might not be able to parse the name correctly. Avoid using commas in SecurId group names and site names.

Identity Manager Installation Notes

If SecurID is installed on Windows, the Identity Manager gateway must be running on the same system where the RSA ACE/Server is installed.

Usage Notes

This section provides information related to using the SecurID ACE/Server resource adapter, which is organized into the following sections:

- *Enabling Pass-Through Authentication on UNIX*
- *Enabling Multiple Tokens*
- *Password Policies*

Enabling Pass-Through Authentication on UNIX

Because the RSA C API on UNIX is not supported, enabling pass-through authentication with the SecurID ACE/Server UNIX adapter is not a straightforward process. Performing pass-through authentication on this adapter requires the following interactions between components:

Identity Manager <--> SecurID Unix Resource Adapter <--> SecurID Windows Adapter <--> Sun Identity Manager Gateway <--> RSA ACE Agent for Windows <--> RSA Unix Server

Note the following configuration and implementation points when enabling pass-through authentication with the SecurID ACE/Server UNIX adapter:

- The Sun Identity Manager Gateway and the RSA ACE Agent Host must reside on the same Windows host. See the Resource Configuration Notes section for more information.
- If the UNIX RSA server lists itself as a client, the account used to authenticate users must be defined on the UNIX resource. See the Resource Configuration Notes section for more information.
- You must specify a value for the **`ACE Server Authentication Resource`** resource parameter in the SecurID ACE/Server UNIX adapter. This value must match a resource name specified in a valid SecurID ACE/Server (for Windows) adapter.
- SecurID's authentication policies require that the UNIX SecurID server must be aware of the RSA ACE Agent for Windows. The `sdconf.rec` file must be present and configured correctly on the Windows host.
- The RSA ACE Agent for Windows must be activated for users attempting to use pass-through authentication.
- Identity Manager must be configured to use the SecurID ACE/Server or SecurID ACE/Server UNIX login module.
- Candidate users for authentication must be configured with an Identity Manager role and organization.

Enabling Multiple Tokens

The default schema map for both SecurID resource adapters is set-up to allow the administrator to specify one token. If you are using the SecurID User Form provided in the *`InstallDir\samples\forms`* directory, perform the following steps to enable up to three tokens.

1. Edit the following section of the SecurID User Form:

   ```
   <FieldLoop for='tokenNum'>
      <expression>
         <ref>oneTokenList</ref>
      </expression>
   ```

   Change `oneTokenList` to `threeTokenList`.

2. Load the User Form into Identity Manager.

3. Rename the following Identity Manager User Attributes on the left side of SecurID ACE/Server schema map:

Original Identity Manager User Attribute	Renamed Identity Manager User Attribute
tokenClearPin	token1ClearPin
tokenDisabled	token1Disabled
tokenLost	token1Lost
expirePassword	token1NewPinMode
password	token1Pin
tokenResync	token1Resync
tokenFirstSequence	token1FirstSequence
tokenNextSequence	token1NextSequence
tokenSerialNumber	token1SerialNumber
tokenUnassign	token1Unassign

4. Add the following fields to the schema map to accommodate a second token:

Identity Manager User Attribute	Resource User Attribute
token2ClearPin	token2ClearPin
token2Disabled	token2Disabled
token2Lost	token2Lost
token2NewPinMode	token2NewPinMode
password	token2Pin
token2Resync	token2Resync
token2FirstSequence	token2FirstSequence
token2NextSequence	token2NextSequence
token2SerialNumber	token2SerialNumber
token2Unassign	token2Unassign

5. Add the following fields to the schema map to accommodate a third token:

Identity Manager User Attribute	Resource User Attribute
token3ClearPin	token3ClearPin
token3Disabled	token3Disabled
token3Lost	token3Lost
token3NewPinMode	token3NewPinMode
password	token3Pin
token3Resync	token3Resync
token3FirstSequence	token3FirstSequence
token3NextSequence	token3NextSequence
token3SerialNumber	token3SerialNumber
token3Unassign	token3Unassign

Password Policies

If Identity Manager uses passwords that contain alphabet characters, and SecurID does not permit alphabet characters in a PIN, the following message will be returned:

```
SecurId ACE/Server: (realUpdateObject) Sd_SetPin Error Alpha
characters not allowed
```

To correct this error, either modify the Identity Manager password policy for the resource so that it cannot contain alphabet characters, or change the PIN restrictions on the resource to permit alphabet characters.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager can use the following to communicate with the SecurID ACE/Server adapter:

- Sun Identity Manager Gateway (Windows only)
- SecurID TCL Interface (UNIX only)

Required Administrative Privileges

The user specified in the Login User resource parameter (on UNIX) or in the Administrator Login resource parameter (on Windows) must be assigned to an administrative role that has the ability to run user- and token-related tasks.

You can use a test connection to test whether

- These commands exist in the administrator user's path
- The administrative user can write to /tmp
- The administrative user have rights to run certain commands

Note A test connection can use different command options than a normal provision run.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	Yes
Pass-through authentication	Yes
Before/after actions	No
Data loading methods	• Import from resource • Reconciliation

Account Attributes

The following table provides information about SecurID ACE/Server account attributes.

Notes

- The SecurID ACE/Server adapters do not support custom account attributes (known as User Extension Data on SecurId) that contain multiple values.
- The data type for all attributes is String, unless otherwise noted.

Identity Manager User Attribute	Resource User Attribute	Description
adminGroup	adminGroup	The group the administrator is a member of. This is a read-only attribute.
adminLevel	adminLevel	The administrative level of the user. The value can be realm, site, or group. This is a read-only attribute.
adminSite	adminSite	The sites to which the administrator has access to. This is a read-only attribute.
adminTaskList	adminTaskList	The name of the set of tasks that the administrator can perform. This is a read-only attribute.
adminTaskListTasks	adminTaskListTasks	The specific tasks the administrator can perform. This is a read-only attribute.
allowedToCreatePin	allowedToCreatePin	Read-only boolean attribute that indicates that a user is allowed to specify a PIN. If the PIN is not specified, the system will generate one for the user
clients	clients	Specifies the clients a user is a member of.
accountId	defaultLogin	The account ID for the user in ACE/Server. Maximum 48 characters.
defaultShell	defaultShell	User's default shell. Maximum 256 characters.
expirePassword	WS_PasswordExpired	Indicates whether the password will be expired. When the password is expired, the SecurID account will be placed in New PIN Mode.
firstname	firstname	Required. The user's first name. Maximum 24 characters.
groups	groups	Specifies the groups a user is a member of.
lastname	lastname	Required. The user's last name. Maximum 24 characters.
remoteAlias	remoteAlias	The user's login name in their remote realm.
remoteRealm	remoteRealm	For remote users, the realm the user is part of.

Identity Manager User Attribute	Resource User Attribute	Description
requiredToCreatePin	requiredToCreatePin	Read-only boolean attribute that indicates that a user must specify a PIN.
tempEndDate	tempEndDate	Date when temporary mode ends.
tempEndHour	tempEndHour	Hour when temporary mode ends.
tempStartDate	tempStartDate	Date when temporary mode begins.
tempStartHour	tempStartHour	Hour when temporary mode begins.
tempUser	tempUser	Sets a user in or out of temporary mode.
tokenClearPin	token1ClearPin	When set on a user update, it will cause the user's PIN to be cleared.
tokenDisabled	token1Disabled	When set on a user update, it will cause the user's PIN to be disabled.
tokenLost	token1Lost	When set to true on a user update, the account will be put in emergency access mode within RSA.
tokenFirstSequence	token1FirstSequence	Specifies the original token when a token needs to be resynchronized.
tokenNewPinMode	token1NewPinMode	When the users account has been placed in New PIN Mode, specifies the user's new PIN.
tokenNextSequence	token1NextSequence	Specifies the new token when a token needs to be resynchronized.
tokenPin	token1Pin	Encrypted. The user's PIN.
tokenResync	token1Resync	Boolean. Indicates whether to resynchronize a token. This attribute enables the tokenFirstSequence and tokenNextSequence attributes.
tokenSerialNumber	token1SerialNumber	Token serial number. Must be 12 characters. Insert leading zeros as needed to meet this requirement.
tokenUnassign	token1Unassign	Specifies a token to remove from a user.
userType	userType	Must be either **`Remote`** or **`Local`**.

Resource Object Management

None

Identity Template

$accountId$

Sample Forms

SecurID User Form

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.SecurIdResourceAdapter`
- `com.waveset.adapter.SecurIdUnixResourceAdapter`
- `com.waveset.adapter.SVIDResourceAdapter`

Siebel

The Siebel resource adapter has been deprecated. Use the Siebel CRM resource adapter (described in the next section) instead.

Siebel CRM

The Siebel CRM resource adapter is defined in the `com.waveset.adapter.SiebelCRMResourceAdapter` class.

This adapter supports the following Siebel versions:

- 6.0
- 7.0
- 7.7

Identity Manager Installation Notes

The Siebel CRM resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. To add the Siebel CRM resource to the resources list, you must add the following value in the Custom Resources section of the Configure Managed Resources page.

   `com.waveset.adapter.SiebelCRMResourceAdapter`

2. Copy the appropriate `JAR` files to the *`InstallDir`*`\idm\WEB-INF\lib` directory, as listed in the following table.

   The `JAR` file versions must match the version of the Siebel CRM resource:

Siebel 6.0	Siebel 7.0	Siebel 7.7
• `SiebelDataBean.jar` • `SiebelTC_enu.jar` • `SiebelTcCommon.jar` • `SiebelTcOM.jar`	• `SiebelJI_Common.jar` • `SiebelJI_enu.jar` • `SiebelJI.jar`	• `Siebel.jar` • `SiebelJI_enu.jar`

Note Do not copy the `JAR` files for multiple versions of Siebel into the *`InstallDir`*`\idm\WEB-INF\lib` directory. You might encounter conflicts between versions.

Resource Configuration Notes

None

Usage Notes

Choosing Business Objects and Components

By default, the Siebel CRM adapter uses the *Employee* Siebel business component of the *Employee* Siebel business object for account provisioning. However, you can configure the adapter to use any Siebel business component of any Siebel business object for account provisioning.

- To use a different business object, set the **Account Business Object** resource parameter appropriately.
- To use a different business component, set the **Account Business Component** resource parameter to the name of the preferred business component.

Note You must specify the business component within the specified business object.

You can use the Siebel Tools Client to inspect your business component and to verify which attributes are available for provisioning. The default schema map has some common attributes that are useful for the default Employee business component.

You may have to add, remove, or change attributes to manage your Siebel environment – especially if you have configured the adapter to use a business object or business component other than the default.

The following steps are a basic guide to discovering which attributes Identity Manager can provision to your Siebel environment using the Siebel Tools client:

1. Open the Siebel Tools' Object Explorer.
2. Click the **Business Component** icon.
3. Scroll down or create a query to select the desired business component.
4. Select **Fields** within the Object Explorer.

   A list of fields available to the business component should display.

The field *Name* column values shown in the Object Explorer are typically used for the right-hand side (or the Resource User Attribute), within the schema map of your configured Siebel CRM resource.

In general, you can manage any of these fields to some degree. However, if you want to manage a multi-valued field or a pick-list field, you must specify a different format for the right-hand side of the schema map, as follows:

- **For a multi-valued field**: The right-hand side must use the *field*@@*keyAttr* format, where:
  - *field* represents the name of the multi-valued field
  - *keyAttr* represents the name of a field within the associated multi-valued business component used to uniquely identify each member of the multi-valued list.

  For example:

  ```
  Position@@Name
  ```
- **For a pick list field**: The right-hand side must use the *field*!!*keyAttr* format, where:
  - *field* represents the name of the pick list field
  - *keyAttr* represents the name of a field within the associated pick-list business component used to uniquely identify a member of the pick list.

  For example:

  ```
  Employee Organization!!Name
  ```

Managing Primary Values in Multi-Value Groups

The adapter performs the following actions when a multi-value group (MVG) already contains a single member that is designated as primary:

- If the incoming MVG contains a single value that is different than the value currently defined in Identity Manager, then the new value will be inserted and marked as the primary. The previous value is then removed from Identity Manager.
- If other non-primary values have been added, by default, the primary value will remain unchanged.

If there are currently multiple values in an MVG with one of the values marked as the primary:

- If any non-primary values are deleted from the set, the current primary will remain as the primary.
- If the MVG value set is replaced with a new single value, then the new single value will be inserted and marked as the primary. All previous values are then removed.
- If other non-primary values have been added, by default, the primary value will remain unchanged.

To move a primary marker from an existing value to a new value when multiple values exist, you must add an account attribute to the schema map. The name of this attribute must be in the form "`Primary` *`MVG_Name`*", where *`MVG_Name`* is a value such as `Employee Organization Id` or `Position`. Therefore, the attribute will have a name such as `Primary Employee Organization Id` or `Primary Position`. Then, in the the user form, set the `Primary` attribute to the desired value.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	No
Rename account	Yes
Create account	Yes
Update account	Yes
Delete account	Yes
Pass-through authentication	Yes
Before/after actions	No
Data loading methods	• Import directly from resource • Reconciliation

Account Attributes

The default schema map assumes that the Employee business object and Employee business component are configured. You might have to add, remove, or change attributes to manage your Siebel environment – especially if you have configured the adapter to use a business object or business component other than the default.

Identity System User Attribute	Resource User Attribute	Description
`accountId`	`Login Name`	User's login name
`firstname`	`First Name`	User's first name

Identity System User Attribute	Resource User Attribute	Description
lastname	Last Name	User's last name
Responsibility	Responsibility@@Name	Multi-value attribute that contains a list of responsibilities you want to assign to the employee. You must manage this attribute in the user form with a multi-select box. The **Responsibility** field is set as a multi-select box in the sample Siebel CRM User Form.
Position	Position@@Name	Multi-value attribute that contains a list of positions you want to assign to the employee. All assigned positions must exist in Siebel. To assign a *Primary Position*, add the Primary Position attribute to your schema map and set the attribute to the name of the position you want to make primary.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager can use HTTP or RSA to communicate with the Siebel CRM adapter. (See the Siebel user documentation for more information.)

Required Administrative Privileges

Ensure the administrator user name/password configured for the adapter is assigned sufficient privileges within Siebel to create new records and to update existing records for the specified business component.

Resource Object Management

By default, the Siebel CRM adapter supports the following Siebel objects:

Resource Object	Features Supported	Attributes Managed
`Employee:Position`	• Create • Update • Delete • Rename	• `Name` • `Division` • `Primary Employee` • `Description`

If necessary, you can manually configure the adapter to support additional resource object types by editing the resource prototype XML as follows:

1. Add a new `<ObjectType>` element to the XML, following the default *`Employee:Position`* object type example.
2. Replace `Employee` with the name of the preferred Siebel business object.
3. Replace `Position` with the name of the preferred Siebel business component.
4. Verify that the embedded `<ObjectAttributes>` element has an `idAttr` attribute that names which `<ObjectAttribute>` will be used to uniquely identity each item in the business component.

Identify Template

The default identity template is `$accountId$`.

Sample Forms

The following sample forms are provided with this resource adapter:

Form	File
SiebelCRM User Form	`sample/SiebelCRMUserForm.xml`
SiebelCRM Create Employee:Position Form	`sample/SiebelCRMpositioncreate.xml`
SiebelCRM Update Employee:Position Form	`sample/SiebelCRMpositionupdate.xml`

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

```
com.waveset.adapter.SiebelCRMResourceAdapter
```

Additionally, you can set the following Identity Manager Active Sync logging parameters for the resource instance:

- Maximum Log Archives
- Maximum Active Log Age
- Maximum Log File Size
- Log File Path
- Log Level

SiteMinder

The SiteMinder resource adapters are defined in the following classes:

- `com.waveset.adapter.SiteminderAdminResourceAdapter`
- `com.waveset.adapter.SiteminderLDAPResourceAdapter`
- `com.waveset.adapter.SiteminderExampleTableResourceAdapter`

The following table summarizes the purpose of these adapters:

GUI Name	Purpose
`SiteminderAdmin`	Manages Siteminder administrator accounts
`SiteminderLDAP`	Manages SiteMinder users when using the Siteminder LDAP repository. This is the most commonly used adapter.
`SiteminderExampleTable`	Manages SiteMinder users when using the Siteminder database table repository

The SiteMinder resource adapters support the following versions of Netegrity SiteMinder:

- 5.5

Resource Configuration Notes

Before setting up the SiteMinder resource adapter in Identity Manager, you must complete these steps in SiteMinder:

1. Register the trusted host:
   a. Create the host configuration object for your Web application server (copy of default settings with Policy Server IP).
   b. Use `smreghost` (from the agent installation directory) to register your application server.

2. Create the agent:
   a. Enter a name for the agent.
   b. Select “Support 4.x Agents”.
   c. Select “Siteminder / WebAgent” as the agent type.
   d. Enter the IP address of the client.
   e. Enter a shared secret.

Note To successfully configure a SiteMinder resource adapter in Identity Manager, you must know the agent name and shared secret.

Identity Manager Installation Notes

The SiteMInder resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. Add the one of the following values in the Custom Resources section of the Configure Managed Resources page.
   - `com.waveset.adapter.SiteminderAdminResourceAdapter`
   - `com.waveset.adapter.SiteminderLDAPResourceAdapter`
   - `com.waveset.adapter.SiteminderExampleTableResourceAdapter`
2. Download and save one or more files to support the adapter.

Files Needed	• `smjavaagentapi.jar` • `smjavasdk2.jar`
Product Location	`Netegrity\Siteminder\SDK-2.2\java` **Note**: We recommend that you obtain the `.jar` files from the Web agent directory, to ensure there is no version conflict. If you cannot locate the `.jar` files in your Web agent directory, they are also located in the `Netegrity\SiteMinder\SDK-2.2\java` directory.
Installation Notes	Copy the `.jar` files to the `WEB-INF\lib` directory.
Class Name	• `com.waveset.adapter.SiteminderAdminResourceAdapter` • `com.waveset.adapter.SiteminderLDAPResourceAdapter` • `com.waveset.adapter.SiteminderExampleTableResourceAdapter`

If you plan to use the SiteMinder Admin resource adapter, you must set the LIBPATH (or LD_LIBPATH, or SHLIB_PATH, depending on the application server platform) in the application server startup script or environment before starting the application server.

For example, on Solaris, the Web agent is installed in the following directory, which contains a file named `nete_wa_env.sh`:

```
/opt/netegrity/siteminder/webagent
```

For WebLogic, add these lines to start `Weblogic.sh` in `/bea/wlserver6.1/config/mydomain`:

```
# In order to pickup the Siteminder libraries, the Netegrity
# Web agent libs need to be added to LIBPATH,
# LD_LIBRARY_PATH, and SHLIB_PATH
. /opt/netegrity/siteminder/webagent/nete_wa_env.sh
```

These lines set up the appropriate variables for the Java Native Interface methods used by the SiteMinder Admin resource adapter.

When you are finished, restart the Identity Manager application server.

Usage Notes

Before Identity Manager 5.5, the SiteMinder LDAP Active Sync adapter used the **`Process to run with changes`** field to determine which process to launch when a change was detected. The process specified in this field is now specified in the Active Sync Resolve Process rule.

In addition, before Identity Manager 5.5, if the **`Process deletes as updates`** check box was selected, Identity Manager would disable a deleted Identity Manager user as well as all resource accounts and mark the user for later deletion. By default, this check box was selected. In Identity Manager 5.5 and beyond, this functionality is configured by setting the Delete Rule set to None.

If the checkbox was previously deselected, then the Delete Rule will be set to **`ActiveSync has isDeleted set`**.

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses JNDI over SSL to communicate with SiteMinder.

Required Administrative Privileges

None

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes for SiteMinder LDAP and Table. Not applicable for SiteMinder Admin
Rename account	
Pass-through authentication	Yes
Before/after actions	
Data loading methods	Import from resource

Account Attributes

Resource Object Management

Identity Template

$accountId$

Sample Forms

`SiteminderAdminUserForm.xml`

SiteminderExampleTableUserForm.xml

SiteminderLDAPUserForm.xml

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- com.waveset.adapter.SiteminderAdminResourceAdapter
- com.waveset.adapter.SiteminderLDAPResourceAdapter
- com.waveset.adapter.SiteminderExampleTableResourceAdapter

Solaris

The Solaris resource adapter is defined in the `com.waveset.adapter.SolarisResourceAdapter` class.

This adapter supports the following versions of Solaris:

- 8
- 9
- 10

Resource Configuration Notes

If you will be using SSH (Secure Shell) for communication between the resource and Identity Manager, set up SSH on the resource before configuring the adapter.

Identity Manager Installation Notes

No additional installation procedures are required on this resource.

Usage Notes

The Solaris resource adapter primarily provides support for the following Solaris commands:

- `useradd, usermod, userdel`
- `groupadd, groupmod, groupdel`
- `passwd`

For more information about supported attributes and files, refer to the Solaris manual pages for these commands.

When a rename of a user account is executed on a Solaris resource, the group memberships are moved to the new user name. The user's home directory is also renamed if the following conditions are true:

- The original home directory name matched the user name.
- A directory matching the new user name does not already exist.

The Bourne-compliant shell (sh, ksh) must be used as the root shell when connecting to a UNIX resource (AIX, HP-UX, Solaris, or Linux).

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager can use the following connections to communicate with the Solaris adapter:

- Telnet
- SSH (SSH must be installed independently on the resource.)

Required Administrative Privileges

The adapter supports logging in as a standard user, then performing a su command to switch to root (or root-equivalent account) to perform administrative activities. Direct logins as root user are also supported.

The adapter also supports the sudo facility (version 1.6.6 or later), which can be installed on Solaris 9 from a companion CD. sudo allows a system administrator to give certain users (or groups of users) the ability to run some (or all) commands as root or another user.

In addition, if sudo is enabled for a resource, its settings will override those configured on the resource definition page for the root user.

If you are using sudo, you must set the `tty_tickets` parameter to true for the commands enabled for the Identity Manager administrator. Refer to the man page for the sudoers file for more information.

The administrator must be granted privileges to run the following commands with sudo:

User and Group Commands		NIS Commands	Miscellaneous Commands	
• auths	• passwd	• make	• awk	• ls
• groupadd	• profiles	• ypcat	• cat	• mv
• groupdel	• roles	• ypmatch	• chmod	• rm
• groupmod	• useradd	• yppasswd	• chown	• sed
• last	• userdel		• cp	• sleep
• listusers	• usermod		• cut	• sort
• logins			• diff	• tail
			• echo	• touch
			• grep	• which

In addition, the NOPASSWORD option must be specified for each command.

You can use a test connection to test whether

- These commands exist in the administrator user's path
- The administrative user can write to /tmp
- The administrative user have rights to run certain commands

Note A test connection can use different command options than a normal provision run.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Solaris does not natively support Identity Manager enable and disable actions. Identity Manager simulates enabling and disabling accounts by changing the user password. The changed password is exposed on enable actions, but it is not exposed on disable actions. As a result, enable and disable actions are processed as update actions. Any before or after actions that have been configured to operate on updates will execute.
Rename account	Yes
Pass-through authentication	Yes
Before/after actions	Yes
Data loading methods	• Import directly from resource • Reconcile with resource

You can define resource attributes to control the following tasks for all users on this resource:

- Create a home directory when creating the user
- Copy files to the user's home directory when creating the user
- Delete the home directory when deleting the user

Account Attributes

The following tables list the Solaris user account attributes, including options for all versions of Solaris and for Solaris 8 or later.

Notes:

- Attributes are optional unless noted in the description.

- All attributes are Strings.

Options for All Versions of Solaris

Resource User Attribute	useradd Equivalent	Description
accountId	login	Required. The user's login name.
comment	-c *comment*	The user's full name.
dir	-d *directory*	The user's home directory.
expire	-e *expiration date*	Last date the account can be accessed.
group	-g *group*	The user's primary group.
inactive	-f *days*	Number of days the account can be inactive before it is locked
secondary_group	-G *group*	The user's secondary group or groups.
shell	-s /*Path*	The user's login shell.
time_last_login	Obtained from the last command.	The date and time of the last login. This value is read-only.
uid	-u *User ID*	The user ID, in digit form.

Options for Solaris 8 and Later

Resource User Attribute	useradd Equivalent	Description
authorization	-A *authorization*	A comma-separated list of authorizations.
profile	-P *profile*	A comma-separated list of profiles.
role	-R *role*	A comma-separated list of roles.

Resource Object Management

Identity Manager supports the following native Solaris objects:

Resource Object	Features Supported	Attributes Managed
Group	Create, update, delete, rename, save as	groupName, gid, users

Identity Template

$accountId$

Sample Forms

Built-In

- Solaris Group Create Form
- Solaris Group Update Form

Also Available

SolarisUserForm.xml

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.SolarisResourceAdapter`
- `com.waveset.adapter.SVIDResourceAdapter`
- `com.waveset.adapter.ScriptedConnection`

SQL Server

The SQL Server resource adapter has been deprecated. Use the MS SQL Server resource adapter instead.

Sun ONE Identity Server

The Sun ONE Identity Server resource adapter has been deprecated. Use the Sun Java System Access Manager resource adapter instead.

Sample Forms

Support for the following Identity Server sample forms will be continued for this release:

- Sun ONE Identity Server Create Dynamic Subscription Group Form
- Sun ONE Identity Server Create Filtered Group Form
- Sun ONE Identity Server Create Organization Form
- Sun ONE Identity Server Create Role Form
- Sun ONE Identity Server Create Static Subscription Group Form
- Sun ONE Identity Server Update Dynamic Subscription Group Form
- Sun ONE Identity Server Update Filtered Group Form
- Sun ONE Identity Server Update Organization Form
- Sun ONE Identity Server Update Role Form
- Sun ONE Identity Server Update Static Subscription Group Form

The `SunISUserForm.xml` form will also be available.

Sun Java System Access Manager

The Sun Java System Access Manager resource adapter is defined in the `com.waveset.adapter.SunAccessManagerResourceAdapter` class. This adapter supports the following versions:

- Sun ONE Identity Server 6.0
- Sun ONE Identity Server 6.1
- Sun ONE Identity Server 6.2
- Sun™ Java System Identity Server 2004Q2

- Sun™ Java System Access Manager 6 2005Q1
- Sun™ Java System Access Manager 7 2005Q4

Note Sun ONE Identity Server has been renamed to Sun™ Java System Access Manager.

Resource Configuration Notes

This resource adapter works with the following products:

- Sun™ Java System Identity Server
- Sun™ Java System Identity Server Policy Agent 2.1
- Sun Java™ System Access Manager

The Policy Agent is an optional module that you can use to enable single sign-on (SSO). Do not attempt to follow Policy Agent configuration or installation procedures if this product is not being used in your environment.

Note See `http://docs.sun.com/db?p=doc%2F816-6772-10` for more information about Policy Agents.

The following sections describe how to install and configure Sun Java System Access Manager and Policy Agent.

Installing and Configuring Sun Java System Access Manager

If you install Sun Java System Access Manager on the same system as the Identity Manager server, see *Sun Java System Access Manager Resource Adapter* on page 1-327 for information about configuration. If you are using the Policy Agent, go to *Installing and Configuring the Policy Agent* on page 1-326 for additional information.

If Sun Java System Access Manager is installed on a different system than the Identity Manager server, then perform the following steps on the Identity Manager system.

1. Create a directory to place files that will be copied from the Sun Java System Access Manager server. This directory will be called *CfgDir* in this procedure. The location of the Sun Java System Access Manager will be called *AccessMgrHome*.
2. Copy the following files from *AccessMgrHome* to *CfgDir*. Do not copy the directory structure.
   - `lib/*.*`
   - `locale/*.properties`
   - `config/serverconfig.xml`
   - `config/SSOConfig.properties` (Identity Server 2004Q2 and later)
   - `config/ums/ums.xml`
3. On UNIX, it may be necessary to change the permissions of the jar files in the *CfgDir* to allow universal read access. Run the following command to change permissions:

   ```
   chmod a+r CfgDir/*.jar
   ```
4. Append the JAVA classpath with the following:
   - **Windows**: *CfgDir*`;`*CfgDir*`/am_sdk.jar;`*CfgDir*`/am_services.jar;`*CfgDir*`/am_logging.jar`
   - **UNIX**: *CfgDir*`:`*CfgDir*`/am_sdk.jar:`*CfgDir*`/am_services.jar:`*CfgDir*`/am_logging.jar`
5. If you are using version 6.0, set the Java system property to point to your *CfgDir*. Use a command similar to the following:

   ```
   java -Dcom.iplanet.coreservices.configpath=CfgDir
   ```
6. If you are using version 6.1, add or edit the following lines in the *CfgDir*`/AMConfig.properties` file:

   ```
   com.iplanet.services.configpath=CfgDir
   com.iplanet.security.SecureRandomFactoryImpl=com.iplanet.am.util.
   SecureRandomFactoryImpl
   com.iplanet.security.SSLSocketFactoryImpl=netscape.ldap.factory.
   JSSESocketFactory
   com.iplanet.security.encryptor=com.iplanet.services.util.
   JCEEncryption
   ```

   The first line sets the `configpath`. The last three lines change security settings.
7. Copy the *CfgDir*`/am_*.jar` files to `$WSHOME/WEB-INF/lib`. If you are using version 6.0, also copy the `jss311.jar` file to the `$WSHOME/WEB-INF/lib` directory.

8. If Identity Manager is running on Windows and you are using Identity Server 6.0, copy *IdServer*\lib\jss*.dll to *CfgDir* and add *CfgDir* to your system path.

Note In an environment where Identity Manager is installed on a different system from Sun Java System Access Manager check the following error conditions. If an error java.lang.ExceptionInInitializerError, followed by java.lang.NoClassDefFoundError, on subsequent attempts, is returned when attempting to connect to the Sun Java System Access Manager resource, then check for incorrect or missing configuration data.

Check that the *CfgDir* contains all the data outlined in Step 6 and that all the configuration properties have been assigned correctly.

See *Sun Java System Access Manager Resource Adapter* on page 1-327 for more information about preparing Identity Manager for this resource.

Installing and Configuring the Policy Agent

You must install the Identity Server Policy Agent 2.1 must be installed on the Identity Manager server. The Policy Agent can be obtained from the following location:

http://wwws.sun.com/software/download/inter_ecom.html#dirserv

Follow the installation instructions provided with the Policy Agent. Then perform the following tasks.

Edit the AMAgent.properties File

The AMAgent.properties file must be modified so that Identity Manager can be protected. It is located the following directory:

- **Windows**: *AgentInstallDir*\es6\config*_PathInstanceName*\
- **UNIX**: /etc/opt/SUNWam/agents/es6/config/*_PathInstanceName/*

Be sure to use the files located the preceding directories. Do not use the copy located in the *AgentInstallDir*\config directory.

1. Add or edit the following lines:

```
com.sun.am.policy.am.fetchHeaders=true
com.sun.am.policy.am.headerAttributes=entrydn|sois_user
com.sun.am.policy.agents.fqdnDefault = FullyQualifiedIDMgrServer
```

Note There can be values lines defining headerAttributes and fqdnDefault values.

2. Restart the web server so that the changes to the AMAgent.properties files can take effect.

Create a Policy in Sun Java System Access Manager

1. From within the Sun Java System Access Manager application, create a new policy named `IDMGR` (or something similar) with the following rules:

Service Type	Resource Name	Actions
URL Policy Agent	http://*server*:*port*/idm	Allow GET and POST actions
URL Policy Agent	http://*server*:*port*/idm/*	Allow GET and POST actions

2. Assign one or more subjects to the `IDMGR` policy.

Identity Manager Installation Notes

This section provides installation and configuration notes for the Sun Java System Access Manager resource adapter and the Policy Agent.

Sun Java System Access Manager Resource Adapter

If the Sun Java System Access Manager is installed on a different system than the Identity Manager server, then perform the procedure described in *Installing and Configuring Sun Java System Access Manager* on page 1-324.

Otherwise, copy the *AccessMgrHome*`/lib/am_*.jar` files to `$WSHOME/WEB-INF/lib`. If you are using version 6.0, also copy the `jss311.jar` file to the `$WSHOME/WEB-INF/lib` directory.

After the files have been copied, add the Sun Java System Access Manager resource to the Identity Manager resources list, you must add the following value in the Custom Resources section of the Configure Managed Resources page.

```
com.waveset.adapter.SunAccessManagerResourceAdapter
```

Policy Agent

You must modify the administrator and user login modules so that the Sun Java System Access Manager login modules are listed first.

Note A Sun Java System Access Manager resource must be configured before performing this procedure:

1. From the Identity Manager Administrator Interface menu bar, click **Configure**.
2. Click **Login**.

3. Click the **Administrator Interface** link.
4. Select the Sun Java System Access Manager Login Module from the drop-down list.
5. Configure the module as desired and click the **Save** button.
6. Click the check box to the left of the Sun Java System Access Manager Login Module option and click the **Move Up** button.
7. Save your changes and repeat this procedure for the User Interface.

Usage Notes

If you are running Identity Manager under WebLogic, and native changes made in Sun Java System Access Manager do not appear in Identity Manager, add `am_services.jar` in the classpath before `weblogic.jar`.

To set the protocol handler when you have more than one:

```
java.protocol.handler.pkgs=com.iplanet.services.comm|sun.net.
www.protocol
```

Security Notes

This section provides information about supported connections and authorization requirements needed to perform basic tasks.

Supported Connections

Identity Manager uses JNDI over SSL to communicate with this adapter.

Required Administrative Privileges

The user name that connects to the Sun Java System Access Manager must be assigned permissions to add or modify user accounts.

Provisioning Notes

This section contains a table that summarizes the provisioning capabilities of the adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	No
Pass-through authentication	Yes. The Web Proxy Agent is required for single sign-on.
Before/after actions	No
Data loading methods	• Import directly from resource • Reconcile with resource

Account Attributes

The following table lists the Sun Java System Access Manager user account attributes supported by default. All attributes are optional, unless noted in the description.

Resource User Attribute	Resource Attribute Type	Description
cn	String	Required. The user's full name.
dynamicSubscriptionGroups	String	A list of dynamic groups to which the user is subscribed.
employeeNumber	Number	The user's employee number.
givenname	String	The user's first name.
iplanet-am-user-account-life	Date	The date and time the user account expires. The account does not expire if this value is not set.
iplanet-am-user-alias-list	String	A list of aliases that may be applied to the user.
iplanet-am-user-failure-url	String	The URL that the user will be redirected to upon unsuccessful authentication.

Resource User Attribute	Resource Attribute Type	Description
iplanet-am-user-success-url	String	The URL that the user will be redirected to upon successful authentication.
mail	Email	The user's e-mail address.
postalAddress	String	The user's home address.
roles	String	A list of roles assigned to the user.
sn	String	The user's last name.
staticSubscriptionGroups	String	A list of static groups to which the user is subscribed.
telephoneNumber	String	The user's telephone number.
uid	String	Required. A unique user ID for the user.
userPassword	Password	Required. The user's password.

Resource Object Management

Identity Manager supports the following Sun Java System Access Manager objects:

Resource Object	Features Supported	Attributes Managed
Role	List, update, delete	cn, iplanet-am-role-aci-description, iplanet-am-role-description, iplanet-am-role-type, accountMembers
Static subscription group	List, create, update, delete, save as	cn, iplanet-am-group-subscribable, uniqueMember
Filtered group	List, create, update, delete, save as	cn, accountMembers, membershipFilter
Dynamic subscription group	List, create, update, delete, save as	cn, accountMembers, iplanet-am-group-subscribable
Organization	List, create, delete, save as, find	o

Identity Template

The default identity template is

```
uid=$uid$,ou=People,dc=MYDOMAIN,dc=com
```

The default template must be replaced with a valid value.

Sample Forms

This section lists the sample forms that are built-in and available for the Sun Java System Access Manager resource adapter.

Built-In

- Sun Java System Access Manager Update Static Group Form
- Sun Java System Access Manager Update Role Form
- Sun Java System Access Manager Update Organization Form
- Sun Java System Access Manager Update Filtered Group Form
- Sun Java System Access Manager Update Dynamic Group Form
- Sun Java System Access Manager Create Static Group Form
- Sun Java System Access Manager Create Role Form
- Sun Java System Access Manager Create Organization Form
- Sun Java System Access Manager Create Filtered Group Form
- Sun Java System Access Manager Create Dynamic Group Form

Also Available

```
SunAMUserForm.xml
```

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

```
com.waveset.adapter.SunAccessManagerResourceAdapter
```

Sun Java System Communications Services

Identity Manager provides the Sun Java™ System Communications Services resource adapter to support Sun Java™ System Messaging Server (Messaging Server) and the Sun Java™ System Calendar Server (Calendar Server): These systems must be implementing LDAP Schema 2. In addition, Sun Java™ System Directory Server must be used as the user store.

The Sun Java™ System Communications Services resource adapter is defined in the `com.waveset.adapter.SunCommunicationsServicesResourceAdapter` class.

This adapter extends the LDAP resource adapter. See the documentation for the LDAP adapter for information about implementing LDAP-features. the following topics:

The Communications Services adapter provides provisioning services for standard Directory Server installations.It can also read the replication changelog of Directory Server and apply those changes to Identity Manager users or custom workflows.

Resource Configuration Notes

To setup a Sun Java™ System Directory Server resource for use with the Communications Services adapter, you must configure the server to enable the change log and enable tracking of modifier information. This is done from the directory server configuration tab.

1. Click on the Replication folder, then select the “Enable change log” box. For 5.0 and later servers, you must also enable the RetroChangelog Snapin. On the configuration tab go to the plugin object, select the Retro change log plugin and enable it.
2. To verify that the server is configured to maintain special attributes for newly created or modified entries, in the Directory Server console, click Configuration > select the root entry in the navigation tree in the left pane.
3. Click Settings > verify that the Track Entry Modification Times box is checked.

The server adds the following attributes to a newly created or modified entry to determine if an event was initiated from Identity Manager.

- **creatorsName**: The DN of the person who initially created the entry.
- **modifiersName**: The DN of the person who last modified the entry.

Identity Manager Installation Notes

No additional installation procedures are required on this resource.

Usage Notes

Service Accounts

It is recommend that you create an Identity Manager service account to connect to Communications Services, rather than using the administrator account CN=Directory Manager. Use your Directory Server management tool to set permissions via an ACI (access control instructions) at each base context.

Set the permissions in the ACI based on the source. If the adapter is connecting to an authoritative source, then set read, search, and possibly compare permissions only. If the adapter is used to write back, then you will need to set write and possibly delete permissions.

Note If the account will be used for the monitoring the changelog, an ACI should also be created on `cn=changelog`. The permissions should be set to read and search only, because you cannot write or delete changelog entries.

The `sources.`*`ResourceName`*`.hosts` property in the `waveset.properties` file can be used to control which host or hosts in a cluster will be used to execute the synchronization portion of an Active Sync resource adapter. *`ResourceName`* must be replaced with the name of the Resource object.

Before and After Actions

The Sun Communications Services resource adapter does not perform before or after actions. Instead, you may use the **Action Proxy Resource Adapter** field in the Resource Wizard to designate a proxy resource adapter that has been configured to run actions.

The following example script could be run on the proxy resource after creating a user:

```
SET PATH=c:\Sun\Server-Root\lib
SET SYSTEMROOT=c:\winnt
SET CONFIGROOT=C:/Sun/Server-Root/Config
mboxutil -d -P user/%WSUSER_accountId%.*
```

The following example script will delete the user's mailboxes when the user is deleted.

```
SET PATH=c:\Sun\Server-Root\lib
SET SYSTEMROOT=c:\winnt
```

```
SET CONFIGROOT=C:/Sun/Server-Root/Config
mboxutil -d -P user/%WSUSER_accountId%.*
```

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses Java Naming and Directory Interface (JNDI) over TCP/IP or SSL to communicate with the Communications Services adapter.

- If you are using TCP/IP, specify port 389 on the Resource Attributes page.
- If you are using SSL, specify port 636.

Required Administrative Privileges

If the value `cn=Directory Manager` is specified in the User DN resource parameter, then the Identity Manager administrator has the necessary permissions to manage accounts. If a different distinguished name is specified, that user must have the ability to read, write, delete, and add users.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	Yes
Pass-through authentication	Yes
Before/after actions	No, but a proxy resource adapter may be specified.
Data loading methods	• Import directly from resource • Reconcile with resource • Active Sync

Account Attributes

The syntax (or type) of an attribute usually determines whether the attribute is supported. In general, Identity Manager supports boolean, string, and integer syntaxes. Binary strings and similar syntaxes are not supported.

The following table lists the supported LDAP syntaxes. Other LDAP syntaxes might be supported, as long as it is boolean, string, or integer in nature.

LDAP Syntax	Attribute Type	Object ID
Boolean	Boolean	1.3.6.1.4.1.1466.115.121.1.7
Country String	String	1.3.6.1.4.1.1466.115.121.1.11
DN	String	1.3.6.1.4.1.1466.115.121.1.12
Directory String	String	1.3.6.1.4.1.1466.115.121.1.15
Generalized Time	String	1.3.6.1.4.1.1466.115.121.1.24
IA5 String	String	1.3.6.1.4.1.1466.115.121.1.26
Integer	Int	1.3.6.1.4.1.1466.115.121.1.27
Postal Address	String	1.3.6.1.4.1.1466.115.121.1.41
Printable String	String	1.3.6.1.4.1.1466.115.121.1.44
Telephone Number	String	1.3.6.1.4.1.1466.115.121.1.50

Default Account Attributes

The following attributes are displayed on the Account Attributes page for the Communications Services resource adapters.

Note All attributes are of type String unless otherwise noted.

Identity System User Attribute	Resource User Attribute	Description
accountId	uid	User ID
accountId	cn	Required. The user's full name.
password	userPassword	Encrypted

Identity System User Attribute	Resource User Attribute	Description
firstname	givenname	The user's first (given) name.
lastname	sn	Required. The user's last name (surname).
email	mail	The user's fully-qualified email address.
modifyTimeStamp	modifyTimeStamp	Indicates when a user entry was modified. By default, this attribute is displayed for the Sun Communications Services adapter only.
objectClass	objectClass	The object class to monitor for changes.
alternateEmail	mailalternateaddress	Alternate email address of this recipient.
mailDeliveryOption	maildeliveryoption	Specifies delivery options for the mail recipient. One or more values are permitted on a user or group entry, supporting multiple delivery paths for inbound messages. Values will apply differently depending on whether the attribute is used in inetMailGroup or inetMailUser.
mailHost	mailhost	The fully qualified host name of the mail transfer agent (MTA) that is the final destination of messages sent to this recipient.
mailForwardingAddress	mailforwardingaddress	Specifies one or more forwarding addresses for inbound messages.
inetUserStatus	inetuserstatus	the status of a user's account with regard to global server access. The possible values are `active`, `inactive`, or `deleted`.
mailQuota	mailquota	The amount of disk space, in bytes, allowed for the user's mailbox.
mailAutoReplySubject	mailautoreplysubject	Text to be used as the subject of an auto-reply response.

Identity System User Attribute	Resource User Attribute	Description
mailAutoReplyText	mailautoreplytext	Auto-reply text sent to all senders except users in the recipient's domain.
mailAutoReplyTextInternal	mailautoreplytextinternal	Auto-reply text sent to senders from the recipients domain.
vacationStartDate	vacationstartdate	Vacation start date and time, in the format *YYYYMMDDHHMMSS***Z**.
vacationEndDate	vacationenddate	Vacation end date and time, in the format *YYYYMMDDHHMMSS***Z**.
mailAutoReplyMode	mailautoreplymode	The autoreply mode for user mail account. The possible values are `echo` and `reply`.

Default Supported Object Classes

By default, the Sun Java System Communications Services resource adapter uses the following object classes when creating new user objects in the LDAP tree. Other object classes may be added.

- top
- person
- inetUser
- organizationalPerson
- inetOrgPerson
- ipUser
- userPresenceProfile
- iplanet-am-managed-person
- inetMailUser
- inetLocalMailRecipient
- icscalendaruser

top Object Class

The top object class must contain the `objectClass` attribute, which is present as an account attribute by default. The top object class is extended by a number of object classes, including the `person` object class.

person Object Class

The following table lists additional supported attributes that are defined in the LDAP person object class.

Resource User Attribute	LDAP Syntax	Attribute Type	Description
description	Directory string	String	A short informal explanation of special interests of a person
seeAlso	DN	String	A reference to another person.
telephoneNumber	Telephone number	String	Primary telephone number

inetUser Object Class

The inetUser object class represents a user account, or a resource (defined as any object to which services are provided) account, and is used in conjunction with `inetMailUser` and `ipUser` for creating a mail account. When creating user accounts, this object class extends the base entry created by `inetOrgPerson`.

The following table lists additional supported attributes that are defined in the inetUser object class.

Resource User Attribute	LDAP Syntax	Attribute Type	Description
inetUserStatus	Directory string	String	Specifies the status of a user's account with regard to global server access.The possible values are active, inactive, and deleted.

organizationalPerson Object Class

The following table lists additional supported attributes that are defined in the LDAP Organizationalperson object class. This object class can also inherit attributes from the Person object class.

Resource User Attribute	LDAP Syntax	Attribute Type	Description
destinationIndicator	Printable string	String	This attribute is used for the telegram service.
facsimileTelephoneNumber	Facsimile telephone number	String	The primary fax number.
internationaliSDNNumber	Numeric string	String	Specifies an International ISDN number associated with an object.
l	Directory string	String	The name of a locality, such as a city, county or other geographic region
ou	Directory string	String	The name of an organizational unit
physicalDeliveryOfficeName	Directory string	String	The office where deliveries are routed to.
postalAddress	Postal address	String	The office location in the user's place of business.
postalCode	Directory string	String	The postal or zip code for mail delivery.
postOfficeBox	Directory string	String	The P.O. Box number for this object.
preferredDeliveryMethod	Delivery method	String	The preferred way to deliver to addressee
registeredAddress	Postal Address	String	A postal address suitable for reception of telegrams or expedited documents, where it is necessary to have the recipient accept delivery.
st	Directory string	String	State or province name.
street	Directory string	String	The street portion of the postal address.
teletexTerminalIdentifier	Teletex Terminal Identifier	String	The teletex terminal identifier for a teletex terminal associated with an object

Resource User Attribute	LDAP Syntax	Attribute Type	Description
telexNumber	Telex Number	String	The telex number in the international notation
title	Directory string	String	Contains the user's job title. This property is commonly used to indicate the formal job title, such as Senior Programmer, rather than occupational class, such as programmer. It is not typically used for suffix titles such as Esq. or DDS.
x121Address	Numeric string	String	The X.121 address for an object.

inetOrgPerson Object Class

The following table lists additional supported attributes that are defined in the LDAP inetOrgPerson object class. This object class can also inherit attributes from the organizationalPerson object class.

Resource User Attribute	LDAP Syntax	Attribute Type	Description
businessCategory	Directory string	String	The kind of business performed by an organization.
carLicense	Directory string	String	Vehicle license or registration plate
departmentNumber	Directory string	String	Identifies a department within an organization
displayName	Directory string	String	Preferred name of a person to be used when displaying entries
employeeNumber	Directory string	String	Numerically identifies an employee within an organization
employeeType	Directory string	String	Type of employment, such as Employee or Contractor
homePhone	Telephone number	String	The user’s home telephone number.
homePostalAddress	Postal address	String	The user’s home address.

Resource User Attribute	LDAP Syntax	Attribute Type	Description
initials	Directory string	String	Initials for parts of the user's full name
labeledURI	Directory string	String	A Universal Resource Indicator (URI) and optional label associated with the user.
mail	IA5 string	String	One or more email addresses.
manager	DN	String	Directory name of the user's manager.
mobile	Telephone number	String	The user’s cell phone number.
o	Directory string	String	The name of an organization.
pager	Telephone number	String	The user’s pager number.
preferredLanguage	Directory string	String	Preferred written or spoken language for a person.
roomNumber	Directory string	String	The user’s office or room number.
secretary	DN	String	Directory name of the user’s administrative assistant.

The following attributes are not supported:

- audio (octet string)
- jpegPhoto (JPEG)
- photo (Fax)
- userCertificate (certificate)
- userSMIMECertificate (octet string)
- userPKCS12 (octet string)
- x500uniqueIdentifier (bit string)

ipUser

The ipUser object class holds the reference to the personal address book container and the class of service specifier.

The following table lists additional supported attributes that are defined in the ipUser object class.

Resource User Attribute	Syntax	Attribute Type	Description
inetCoS	String, multi-valued	String	Specifies the name of the Class of Service (CoS) template supplying values for attributes in the user entry.
memberOfPAB	String, multi-valued	String	The unique name of the personal address book(s) in which this entry belongs.
maxPabEntries	Integer, single-valued	Integer	The maximum number of personal address book entries users are permitted to have in their personal address book store.
pabURI	String, single valued	String	LDAP URI specifying the container of the personal address book entries for this user.

userPresenceProfile

The userPresenceProfile object class stores the presence information for a user.

This object class may contain the `vacationStartDate` and `vacationEndDate` attribute, which are present as account attributes by default.

iplanet-am-managed-person

The iplanet-am-managed-person object class contains attributes that Sun Java™ System Access Manager needs to manage users.

The following table lists additional supported attributes that are defined in the ipUser object class.

Resource User Attribute	Syntax	Attribute Type	Description
iplanet-am-modifiable-by	DN, multi-valued	String	The role-dn of the administrator who has access rights to modify the user entry.
iplanet-am-role-aci-description	String, multi-valued	String	Description of the ACI that belongs to the role.
iplanet-am-static-group-dn	DN, multi-valued	String	Defines the DNs for the static groups the user belongs to. Example
iplanet-am-user-account-life	Date string, single-valued	String	Specifies the account expiration date in the following format: yyyy/mm/dd hh:mm:ss

inetMailUser

The inetMailUser extends the base entry created by `inetOrgPerson` to define a messaging service user. It represents a mail account and is used in conjunction with `inetUser` and `inetLocalMailRecipient`.

The following table lists additional supported attributes that are defined in the inetMailUser object class.

Resource User Attribute	Syntax	Attribute Type	Description
dataSource	String, single-valued	String	Text field to store a tag or identifier.
mailAllowedServiceAccess	String, single-valued	String	Stores access filters (rules).
mailAntiUBEService	String, multi-valued	String	Instructions for a program that handles unsolicited bulk email.
mailAutoReplyTimeOut	Integer, single-valued	Integer	Duration, in hours, for successive auto-reply responses to any given mail sender.

Resource User Attribute	Syntax	Attribute Type	Description
mailConversionTag	String, multi-valued	String	Method of specifying unique conversion behavior for a user or group entry.
mailDeferProcessing	String, single-valued	String	Controls whether or not address expansion of the current user or group entry is performed immediately, or deferred.
mailEquivalentAddress	String, multi-valued	String	Equivalent to mailAlternateAddress in regard to mail routing, except with this attribute, the header doesn't get rewritten.
mailMessageStore	String, single-valued	String	Specifies the message store partition name for the user.
mailMsgMaxBlocks	Integer, single-valued	Integer	The size in units of MTA blocks of the largest message that can be sent to this user or group.
mailMsgQuota	Integer, single-valued	Integer	Maximum number of messages permitted for a user
mailProgramDeliveryInfo	String, multi-valued	String	Specifies one or more programs used for program delivery.
mailSieveRuleSource	String, multi-valued	String	Contains a SIEVE rule (RFC 3028 compliant) used to create a message filter script for a user entry.
mailSMTPSubmitChannel	String, single-valued	String	This attribute is a factor involved in setting up guaranteed message delivery, or in setting up other special classes of service.
mailUserStatus	String, single-valued	String	Current status of the mail user. Can be one of the following values: active, inactive, deleted, hold, overquota, or removed.
nswmExtendedUserPrefs	String, multi-valued	String	Holds the pairs that define Messenger Express preferences, such as sort order and Mail From address.

inetLocalMailRecipient

The inetLocalMailRecipient object class stores information that provides a way to designate an LDAP entry as one that represents a local email recipient, to specify the recipient's email addresses, and to provide routing information pertinent to the recipient.

The following table lists additional supported attributes that are defined in the inetLocalMailReceipient object class. (All other attributes in this object class are present as account attributes by default.)

Resource User Attribute	LDAP Syntax	Attribute Type	Description
mailRoutingAddress	String, single-valued	String	Used together with `mailHost` to determine whether or not the address should be acted upon at this time or forwarded to another system.

icsCalendarUser

The icsCalendarUser object class defines a Calendar Server user.

The following table lists additional supported attributes that are defined in the icsCalendarUser object class. (All other attributes in this object class are present as account attributes by default.)

Resource User Attribute	LDAP Syntax	Attribute Type	Description
icsAllowedServiceAccess	String, single-valued	String	Disallows calendar services to a user.
icsCalendar	String, single-valued	String	The calendar ID (calid) of the default calendar for a user or resource. Required attribute for Calendar Manager.
icsCalendarOwned	String, multi-valued	String	Calendars owned by this user.

icsDWPHost	String, single-valued	String	Stores a Database Wire Protocol (DWP) host name so that the calendar ID can be resolved to the DWP server that stores the calendar and its data.
icsExtendedUserPrefs	String, multi-valued	String	Extensions for calendar user preferences.
icsFirstDay	String, single-valued	Integer	First day of the week to be displayed on user's calendar.
icsSet	String, multi-valued	String	Defines one group of calendars. The value for this attribute is a six-part string, with each part separated by a dollar sign ($).
icsStatus	String, single-valued	String	This attribute must be set when assigning calendar services to a domain. The possible values are `active`, `inactive`, and `deleted`.
icsSubscribed	String, multi-valued	String	List of calendars to which this user is subscribed.
icsTimezone	String	String	The default time zone for this user or resource calendar if one is not explicitly assigned through their own user preferences.
preferredLanguage	String, single-valued	String	Preferred written or spoken language for a person.

Resource Object Management

Identity Manager supports the following LDAP objects by default. Any string-, integer-, or boolean-based attributes can also be managed.

Resource Object	Object Classes	Features Supported	Attributes Managed
Group	groupOfUniqueNames iplanet-am-managed-group iplanet-am-managed-filtered-group iplanet-am-managed-assignable-group iplanet-am-managed-static-group inetMailGroup inetLocalRecipient	Create, update, delete, rename, saveas, find	cn, description, owner, uniqueMember
Domain	domain organization inetdomainauthinfo sunManagedOrganization' sunNameSpace mailDomain' icsCalendarDomain	find	dc
Organizational Unit	organizationalUnit iplanet-am-managed-people-container	Create, rename, saveas, find	ou
Organization	organizatiion	Create, rename, saveas, find	o

Identity Template

None. You must supply the identity template with a valid value.

Sample Forms

- Sun Java System Communications Services ActiveSync Form
- Sun Java System Communications Services Create Group Form
- Sun Java System Communications Services Create Organizational Unit Form
- Sun Java System Communications Services Create Organization Form
- Sun Java System Communications Services Update Group Form
- Sun Java System Communications Services Update Organizational Unit Form

Troubleshooting

Use the Identity Manager debug pages to set trace options on one or more of the following classes:

- `com.waveset.adapter.SunCommunicationsServicesResource Adapter`
- `com.waveset.adapter.LDAPResourceAdapter`
- `com.waveset.adapter.LDAPResourceAdapterBase`

Sybase

The Sybase resource adapter is defined in the `com.waveset.adapter.SybaseResourceAdapter` class.

This adapter supports the following versions of Sybase Adaptive Server:

- 12.x

Use this adapter to support user accounts for logging into Sybase Adaptive Server. If you have a custom Sybase table, see *Database Table* on page 1-84 for information about using the Resource Adapter Wizard to create a custom Sybase table resource.

Resource Configuration Notes

None

Identity Manager Installation Notes

The Sybase resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. To add this resource to the Identity Manager resources list, you must add the following value in the Custom Resources section of the Configure Managed Resources page.

   ```
   com.waveset.adapter.SybaseResourceAdapter
   ```

2. Copy the `Sybase\jConnect-5_5\classes\jconn2.jar` file to the *`InstallDir`*`\idm\WEB-INF\lib` directory.

Usage Notes

The Sybase resource adapter uses the following system procedures to manage user accounts:

- sp_addlogin, sp_droplogin
- sp_adduser, sp_dropuser
- sp_changegroup
- sp_displayroles
- sp_helpuser
- sp_locklogin
- sp_password

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses JDBC over SSL to communicate with this adapter.

Required Administrative Privileges

The following table lists the permissions needed to execute the system procedures:

System Procedure	Permissions Required
sp_addlogin, sp_droplogin	System Administrator or System Security Officer
sp_adduser, sp_droplogin	Database Owner, System Administrator, or System Security Officer
sp_changegroup	Database Owner, System Administrator, or System Security Officer
sp_displayroles	System Administrator or System Security Officer
sp_helpuser	None
sp_locklogin	System Administrator or System Security Officer
sp_password	Only a System Security Officer can execute sp_password to change another user's password. Any user can execute sp_password to change his or her own password.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	No

Feature	Supported?
Pass-through authentication	Yes
Before/after actions	No
Data loading methods	• Import directly from resource • Reconcile with resource

Account Attributes

The following table provides information about Sybase account attributes.

Resource User Attribute	Description
Sybase Roles	The Sybase roles (system capabilities) assigned to the user.
Sybase Group	The Sybase groups (databases) assigned to the user.

Resource Object Management

None

Identity Template

$accountId$

Sample Forms

None

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.SybaseResourceAdapter`
- `com.waveset.adapter.JdbcResourceAdapter`

Top Secret

The Top Secret resource adapter supports management of user accounts and memberships on an OS/390 mainframe via the IBM Host Access Class Library APIs. The adapter manages Top Secret over a TN3270 emulator session.

The Top Secret resource adapter is defined in the `com.waveset.adapter.TopSecretResourceAdapter` class. The adapter supports the following versions of Top Secret:

- 5.3

Note The Top Secret Active Sync adapter (`com.waveset.adapter.TopSecretResourceAdapter` has been deprecated as of Identity Manager 5.0 SP1. All features in this adapter are now in the Top Secret adapter. Although existing instances of the Top Secret Active Sync adapter will still function, new instances of these can no longer be created.

Resource Configuration Notes

The Top Secret Active Sync adapter works by using FTP to retrieve the output from the TSSAUDIT facility. It then parses the output to look for account creations, modifications, and deletions. This facility generates a report from the data in the Top Secret Recovery file. Therefore, the Recovery File must be enabled and large enough to hold all changes that will occur between the Active Sync poll interval. A job should be scheduled to run the TSSAUDIT utility so that the output will be available before the next Active Sync adapter poll.

An optional Generational Data Group (GDG) can be set-up to contain the results of the TSSAUDIT output. A GDG stores previous versions of the TSSAUDIT output. The Active Sync adapter supports retrieving from a GDG to help avoid missing events if it is not able to run at its normal time. The adapter can be configured to go back multiple generations to pick up any events that it might have missed

The following sample JCL runs the TSSAUDIT batch job:

```
//LITHAUS7  <<<< Supply Valid Jobcard >>>>>>
//* ************************************************************
//* *  THIS JOB RUNS THE TSS AUDIT PROGRAM 'CHANGES'
//* *    & CREATES A GDG MEMBER FOR IDENTITY MANAGER
//* *  You may choose to use standard MVS Delete/Defines or
//* * request a system programmer to establish a small GDG
//* ************************************************************
//AUDIT01  EXEC   PGM=TSSAUDIT,
//          PARM='CHANGES DATE(-01)'
//AUDITOUT DD DSN=auth hlq.LITHAUS.ADMIN.DAILY(+1),
```

```
//          DISP=(NEW,CATLG),UNIT=SYSDA,RECFM=FB,LRECL=133,
//          BLKSIZE=2793,SPACE=(CYL,(2,1),RLSE)
//RECOVERY DD   DSN=your.TSS.recovery.file ,DISP=SHR
//AUDITIN  DD DUMMY
```

Identity Manager Installation Notes

The Top Secret resource adapter is a custom adapter. You must perform the following steps to complete the installation process:

1. To add a Top Secret adapter to the Identity Manager resources list, you must add one of the following values in the Custom Resources section of the Configure Managed Resources page.

   ```
   com.waveset.adapter.TopSecretResourceAdapter
   com.waveset.adapter.TopSecretActiveSyncAdapter
   ```

2. The Identity Manager mainframe adapters use the IBM Host Access Class Library (HACL) to connect to the mainframe. The HACL is available in IBM Websphere Host On-Demand (HOD). The recommended jar containing HACL is `habeans.jar` and is installed with the HOD Toolkit (or Host Access Toolkit) that comes with HOD. The supported versions of HACL are in HOD V7.0, V8.0, and V9.0.

   However, if the toolkit installation is not available, the HOD installation contains the following jars that can be used in place of the `habeans.jar`:

   - `habase.jar`
   - `hacp.jar`
   - `ha3270.jar`
   - `hassl.jar`
   - `hodbase.jar`

   Copy the `habeans.jar` file or all of its substitutes into the `WEB-INF/lib` directory of your Identity Manager installation. See http://www.ibm.com/software/webservers/hostondemand/ for more information.

Usage Notes

This section provides information related to using the Top Secret resource adapter, which is organized into the following sections:

- *Administrators*
- *Resource Actions*
- *Resource Action Context*

- *HostAccess API*
- *Mnemonic Keywords for the SendKeys Method*
- *Sample Resource Actions*
- *SSL Configuration*

Administrators

TSO sessions do not allow multiple, concurrent connections. To achieve concurrency for Identity Manager Top Secret operations, you must create multiple administrators. Thus, if two administrators are created, two Identity Manager Top Secret operations can occur at the same time. We recommend that you create at least two (and preferably three) administrators.

CICS sessions are not limited to one session per admin; however, you can define more than one admin if desired.

If you are running in a clustered environment, you must define an admin for each server in the cluster. This applies even if (as in the case of CICS) it is the same admin. For TSO, there must be a different admin for each server in the cluster.

If clustering is not being used, the server name should be the same for each row (the name of the Identity Manager host machine).

Note Host resource adapters *do not* enforce maximum connections for an affinity administrator across multiple host resources connecting to the same host. Instead, the adapter enforces maximum connections for affinity administrators within each host resource.

If you have multiple host resources managing the same system, and they are currently configured to use the same administrator accounts, you might have to update those resources to ensure that the same administrator is not trying to perform multiple actions on the resource simultaneously.

Resource Actions

The Top Secret adapter requires login and logoff resource actions. The login action negotiates an authenticated session with the mainframe. The logoff action disconnects when that session is no longer required.

A thin client host access 3270 emulator is provided to the context of the resource action by the resource adapter to simplify execution of commands in the scripted session.

Resource Action Context

Several global variables may be expected within the context of the scripted action.

Object	Description	Usage
hostAccess	TN3270 emulator; provides an interface for executing commands and parsing responses from the mainframe; wrapped by com.waveset.object.HostAccess to provide convenience methods	Use to send responses to the mainframe, wait for responses, and parse back results
hostAccessLogin	Implemented by TopSecret RA; declares methods such as getHost(), getPort(), and getRequestTimeout(), which provide connection-specific information	
user	ACID of mainframe user	Use for authentication
password	Encrypted object which stores the password of the mainframe user; use password.decryptToString() to convert to plain text	Use for authentication
system	mainframe system name	

HostAccess API

The following table describes the methods available on the hostAccess object passed to the resource action.

Method Signature	Description
void sendKeys(String input)	Send the string of characters specified by **input**.
void sendKeysAndWait(String input, String msg)	sendKeys(input) waitForInput() Send the string of characters specified by **input**; throw a timeout Exception with **msg** if no response is received.
boolean waitForStringFound(String s)	Returns whether the string of characters specified by **s** is found.
void waitForString(String s)	Wait until the specific String is received. *Not generally recommended except for confirmation, like confirmation of a successful logout.*
void waitForStringAndInput(String s)	waitForString(s) waitForInput() if not found, include search text and text that was received in error message
void waitForString(String s, ArrayList stringsToHide)	waitForString() if not found, include text that was received in error message, excluding any strings listed in **stringsToHide**
void waitForInput()	Wait until the system is ready to receive a new message or until the system-configured timeout.
int getRequiredString(String s)	searchText(s,true) searches forward from the cursor position for the specified string
int getRequiredString(String s, ArrayList stringsToHide)	searchText(s,true) if not found, include text that was received in error message, excluding any strings listed in **stringsToHide**
String hideFields(String screen, ArrayList stringsToHide)	Return a string with any fields that should be hidden blanked out.

Method Signature	Description
int searchText(String s, boolean forward)	Return index of found string or 0 if not found.
void setCursorPos(int pos)	Move cursor to specific position on screen
String getScreen()	Return text currently displayed on screen

Mnemonic Keywords for the SendKeys Method

The following table describes the special functions that may be executed through the 3270 emulator to simulate keying the non-alphanumeric values.

Function	Mnemonic Keyword	Function	Mnemonic Keyword
Attention	[attn]	F1	[pf1]
Backspace	[backspace]	F2	[pf2]
Backtab	[backtab]	F3	[pf3]
Beginning of Field	[bof]	F4	[pf4]
Clear	[clear]	F5	[pf5]
Cursor Down	[down]	F6	[pf6]
Cursor Left	[left]	F7	[pf7]
Cursor Right	[right]	F8	[pf8]
Cursor Select	[cursel]	F9	[pf9]
Cursor Up	[up]	F10	[pf10]
Delete Character	[delete]	F11	[pf11]
DUP Field	[dup]	F12	[pf12]
Enter	[enter]	F13	[pf13]
End of Field	[eof]	F14	[pf14]
Erase EOF	[eraseeof]	F15	[pf15]
Erase Field	[erasefld]	F16	[pf16]

Function	Mnemonic Keyword	Function	Mnemonic Keyword
Erase Input	[erinp]	F17	[pf17]
Field Mark	[fieldmark]	F18	[pf18]
Home	[home]	F19	[pf19]
Insert	[insert]	F20	[pf20]
New Line	[newline]	F21	[pf21]
PA1	[pa1]	F22	[pf22]
PA2	[pa2]	F23	[pf23]
PA3	[pa3]	F24	[pf24]
Page Up	[pageup]		
Page Down	[pagedn]		
Reset	[reset]		
System Request	[sysreq]		
Tab Field	[tab]		

Sample Resource Actions

The following code is a complete sample of login and login resource actions. The sample is tailored to a specific customer's environment. As such, the text of commands, prompt, and command sequences will most likely differ across deployments (for example, Line 32 – "ISPF"). Note that the resource actions wrap Javascript inside of XML.

Login Action

```
1   <?xml version='1.0' encoding='UTF-8'?>
2   <!DOCTYPE Waveset PUBLIC 'waveset.dtd' 'waveset.dtd'>
3      <Waveset>
4         <ResourceAction name='ACME Login Action'>
5            <ResTypeAction restype='TopSecret'>
6               <act>
7                  var TSO_MORE = " ***";
8                  var TSO_PROMPT = " READY";
9                  var TS_PROMPT = " ?";
10                 hostAccess.waitForString("ENTER YOUR APPLICATION NAME");
11                 hostAccess.sendKeys("tso[enter]");
12                 hostAccess.waitForString("ENTER USERID -");
```

```
13                hostAccess.sendKeys(user + "[enter]");
14                hostAccess.waitForString("TSO/E LOGON");
15                hostAccess.sendKeys(password.decryptToString() +
    "[enter]");
16                hostAccess.sendKeys(password.decryptToString());
17                var pos = hostAccess.searchText("  -Nomail", false);
18                if (pos != 0) {
19                   hostAccess.setCursorPos(pos);
20                   hostAccess.sendKeys("S");
21                }
22                pos = hostAccess.searchText("  -Nonotice", false);
23                if (pos != 0) {
24                   hostAccess.setCursorPos(pos);
25                   hostAccess.sendKeys("S");
26                }
27                hostAccess.sendKeys("[enter]");
28                hostAccess.waitForStringAndInput(TSO_MORE);
29                hostAccess.sendKeys("[enter]");
30                hostAccess.waitForStringAndInput(TSO_MORE);
31                hostAccess.sendKeys("[enter]");
32                hostAccess.waitForStringAndInput("ISPF");
33                hostAccess.sendKeys("=x[enter]");
34                hostAccess.waitForString(TSO_PROMPT);
35                var resp =hostAccess.doCmd("PROFILE NOPROMPT MSGID
    NOINTERCOM NOPAUSE NOWTPMSG PLANGUAGE(ENU) SLANGUAGE(ENU)
    NOPREFIX[enter]", TSO_PROMPT, TSO_MORE);
36                hostAccess.waitForStringAndInput("ENTER LOGON:");
37                hostAccess.sendKeys(system + "[enter]");
38                hostAccess.waitForStringAndInput("USER-ID.....");
39                hostAccess.sendKeys(user + "[tab]" +
    password.decryptToString() + "[enter]");
40                var stringsToHide = new java.util.ArrayList();
41                stringsToHide.add(password.decryptToString());
42                hostAccess.waitForString("==>", stringsToHide);
43                hostAccess.waitForInput();
44                hostAccess.sendKeys("[pf6]");
45                hostAccess.waitForInput();
46             </act>
47          </ResTypeAction>
48       </ResourceAction>
```

Logoff Action

```
49       <ResourceAction name='ACME Logoff Action'>
50          <ResTypeAction restype='TopSecret'>
51             <act>
52                var TSO_PROMPT = " READY";
53                hostAccess.sendKeys("[clear]end[enter]");
54                hostAccess.waitForString(TSO_PROMPT);
55                hostAccess.sendKeys("logoff[enter]");
```

```
56            </act>
57          </ResTypeAction>
58        </ResourceAction>
59    </Waveset>
```

SSL Configuration

Connecting the Adapter to a Telnet/TN3270 Server using SSL or TLS.

Use the following steps to connect Top Secret resource adapters to a Telnet/TN3270 server using SSL/TLS.

1. Obtain the Telnet/TN3270 server's certificate in the PKCS #12 file format. Use `hod` as the password for this file. Consult your server's documentation on how to export the server's certificate. The procedure "Generating a PKCS #12 File" below for some general guidelines.
2. Create a `CustomizedCAs.class` file from the PKCS #12 file. If you are using a recent version of HOD, use the following command to do this.

   ```
   ..\hod_jre\jre\bin\java -cp ../lib/ssliteV2.zip;../lib/sm.zip
   com.ibm.eNetwork.HOD.convert.CVT2SSLIGHT CustomizedCAs.p12 hod
   CustomizedCAs.class
   ```
3. Place the `CustomizedCAs.class` file somewhere in the Identity Manager server's classpath, such as `$WSHOME/WEB-INF/classes`.
4. If a resource attribute named **`Session Properties`** does not already exist for the resource, then use the BPE or debug pages to add the attribute to the resource object. Add the following definition in the `<ResourceAttributes>` section:

   ```
   <ResourceAttribute name='Session Properties' displayName='Session
   Properties' description='Session Properties' multi='true'>
   </ResourceAttribute>
   ```
5. Go to the Resource Parameters page for the resource and add the following values to the **`Session Properties`** resource attribute:

   ```
   SESSION_SSL
   true
   ```

Generating a PKCS #12 File

The following procedure provides a general description of generating a PKCS #12 file when using the Host OnDemand (HOD) Redirector using SSL/TLS. Refer to the HOD documentation for detailed information about performing this task.

1. Create a new `HODServerKeyDb.kdb` file using the IBM Certificate Management tool. As part of that file, create a new self-signed certificate as the default private certificate.

   If you get a message that is similar to “error adding key to the certificate database” when you are creating the `HODServerKeyDb.kdb` file, one or more of the Trusted CA certificates may be expired. Check the IBM website to obtain up-to-date certificates.
2. Export that private certificate as Base64 ASCII into a `cert.arm` file.
3. Create a new PKCS #12 file named `CustomizedCAs.p12` with the IBM Certificate Management tool by adding the exported certificate from the `cert.arm` file to the Signer Certificates. Use `hod` as the password for this file.

Troubleshooting

You can enable tracing of the HACL by adding the following to the Session Properties resource attribute:

```
SESSION_TRACE
ECLSession=3 ECLPS=3 ECLCommEvent=3 ECLErr=3 DataStream=3 Transport=3
ECLPSEvent=3
```

Note The trace parameters should be listed without any new line characters. It is acceptable if the parameters wrap in the text box.

The Telnet/TN3270 server should have logs that may help as well.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	Yes
Pass-through authentication	No
Before/after actions	Yes
Data loading methods	• Import directly from resource • Reconciliation • Active Sync

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses TN3270 to communicate with the Top Secret adapter.

Required Administrative Privileges

Administrators must have the following privileges:

- ACID(CREATE) authority, via the TSS ADMIN function, to CREATE ACIDs under their administrative scope
- RESOURCE(OWN) authority, via the TSS ADMIN function, to assign resource ownership to ACIDs within their scope
- MISC1, MISC2 and MISC9 authorities, via the TSS ADMIN function, to assign many of the security attributes

Account Attributes

The following table provides information about the default Top Secret account attributes.

Identity System Attribute Name	Resource Attribute Name	Data Type	Description
Profiles	PROFILE	string	The profile assigned to the user. This attribute is capable of having multiple values.
accountId	ACID	string	Required. Account ID
fullname	NAME	string	The user’s first and last name
Installation Data	INSTDATA	string	Installation data
TSOO Access	TSO_ACCESS	boolean	Indicates whether the user has TSO access
TSOLPROC	TSO.TSOLPROC	string	TSO login procedure

Identity System Attribute Name	Resource Attribute Name	Data Type	Description
OMVS Access	OMVS_ACCESS	boolean	Indicates whether the user has OMVS access
Groups	GROUP	string	A list of groups assigned to the user
Default Group	DFLTGRP	string	The user’s default group
UID	OMVS.UID	string	OMVS User ID
OMVSPGM	OMVS.OMVSPGM	string	The user’s initial OMVS program
HOME	OMVS.HOME	string	The user’s OMVS home directory
Attributes	ATTRIBUTE	string	A list of account attributes

The following table lists account attributes that are suported, but are not listed in the schema map by default. The data type for these attributes is string.

Resource Attribute Name	Description
CICS.OPTIME	Controls the period of time allowed before CICS considers a terminal user to be timed-out.
CICS.OPID	Specifies the CICS operator ID.
DEPT	Specifies the department name.
DIV	Specifies the division name.
ZONE	Specifies the zone name.
FACILITY	Specifies a list of facilities an ACID may or may not access.
DATASET	Specifies a list of datasets for the user.
CORPID	Specifies a list of corporate IDs.
OTRAN	Specifies a list of ownable transactions.
TSOACCT	Specifies a list of TSO acciount numbers.
SOURCE	Specifies a list of source readers or terminal prefixes through which the associated ACID may enter the system.
TSO.TRBA	Specifies the relative block address (RBA) of the user's mail directory entry in the broadcast data set

Resource Attribute Name	Description
TSO.TSOCOMMAND	Provides a default command to be issued at TSO logon.
TSO.TSODEFPRFG	Assigns a default TSO performance group.
TSO.TSODEST	Provides a default destination identifier for TSO generated JCL for TSO users.
TSO.TSOHCLASS	Assigns a default hold class for TSO generated JCL for TSO users.
TSO.TSOJCLASS	Assigns a default job class for TSO generated job cards from TSO users.
TSO.TSOLACCT	Provides a default account number to be used for TSO logon.
TSO.TSOLSIZE	Assigns a default region size (in kilobytes) for TSO.
TSO.TSOMCLASS	Assigns a default message class for TSO generated JCL for TSO users.
TSO.TSOMSIZE	Defines the maximum region size (in kilobytes) that a TSO user may specify at logon.
TSO.TSOOPT	Assigns default options that a TSO user may specify at logon.
TSO.TSOSCLASS	Assigns a default SYSOUT class for TSO generated JCL for TSO users.
TSO.TSOUDATA	Assigns a site-defined data field to a TSO user.
TSO.TSOUNIT	Assigns a default unit name to be used for dynamic allocations under TSO.
TSO.TUPT	Specifies the value of the user profile table.

Contact your services organization for details about supporting other Top Secret resource attributes.

Identity Template

```
$accountId$
```

Sample Forms

Built-In

None

Also Available

`TopSecretUserForm.xml`

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following classes:

- `com.waveset.adapter.HostAccess`
- `com.waveset.adapter.TopSecretResourceAdapter`

The hostAccess object may be traced in Identity Manager. The class to trace via the debug pages is `com.waveset.adapter.HostAccess`. Trace level 3 is sufficient to identify which keystrokes and wait messages were sent to the mainframe; trace level 4 will display the exact message sent and the response from the mainframe.

Note Verify that the Trace File location is meaningful. By default the trace file is placed in the application directory under *`InstallDir`*`/idm/config`. If the application is deployed from a WAR, the path may need to be hardcoded with an absolute directory path. In a clustered environment, it is recommended that the trace file be written to a network share.

In addition to source tracing, it may also be useful to log the screen text before each attempt to send keystrokes. This can be accomplished through a file writer. The sequence of commands is:

1. `var file = new java.io.File('<filename>');`
2. `var writer = new java.io.BufferedWriter(new java.io.FileWriter(file));`
3. `writer.write(hostAccess.getScreen());`
4. `writer.flush();`
5. `hostAccess.sendKeysAndWait(<cmd>,<msg>);`
6. `writer.newLine();`
7. `writer.write(hostAccess.getScreen());`
8. `writer.flush();`

9. `writer.close();`

`<filename>` should reference a the location of a file on the local file system of the application server. The writer will open a handle to that location and write what is stored in it's buffer when the flush() method is invoked. The close() method releases the handle to the file. The getScreen() method is useful to pass to this function to get a dump of the screen contents for debugging purposes. This tracing should, of course, be removed once the screens are successfully navigated and login / logout is performed successfully.

Windows NT

The Windows NT resource adapter is defined in the `com.waveset.adapter.NTResourceAdapter` class. It provides support for the following:

- Full support of users and groups on Microsoft Windows NT 4.0.
- Support for local users and groups on Microsoft Windows 2000 and 2003.

Resource Configuration Notes

This section describes Windows NT provisioning across multiple domains with two-way trusts.

The following constraints apply when managing multiple domains from a single domain.

Note Terms referenced this section are:

- **`Gateway domain`** — Domain that the gateway machine is a member of.
- **`Resource admin account`** — Administrative account defined in the Identity Manager resource.
- **`Service account`** — Account that the gateway service is running as.

These trusts must be established:

- The gateway domain needs to trust each domain in which a resource admin account is defined.
- The gateway does a local login using the resource admin account, so its domain needs to trust the domain that account lives in.
- The gateway domain needs to trust each domain for which you will be doing pass-through authentication.
- The gateway does a local login to authenticate user accounts, so its domain needs to trust the domain for those accounts.
- The resource admin account must be a member of the Account Operators group in each domain that it will be used to manage accounts. Each of these domains must trust the domain that contains the resource admin account.
- You cannot add an account to a local group unless the account's domain is trusted by the local group's domain.
- The domain of the service account must be trusted by the gateway domain.

When the gateway service is started, a local login of the service account is done. If any of the resource admin accounts are different than the service account or you will be doing pass-through authentication for any of the domains, then the service account needs the Act As Operating System and Bypass Travers Checking user rights in the gateway domain. These rights are required for the service account to login as and impersonate another.

If you will be creating home directories, then the resource admin account needs to be able to create directories on the file system on which the directories will be created. If the home directory will be created on a network drive, the resource admin account must have write access to that share.

If you will be running before, after, or resource actions, the resource admin account needs read and write access to the file system in the TEMP or TMP environment variables of the gateway process; or, if not defined, the gateway process' working directory (this is either WINNT or WINNT\system32).

The gateway writes the scripts and script output to one of these directories (the directory is selected in the order they are mentioned).

We recommend that a separate resource adapter be configured for each domain. The same gateway host may be used.

It should be possible to manage multiple domains using a single resource by overriding any domain-specific resource attributes (the domain and possibly the administrator and password) for each user.

Notes:

- Since a domain trusts itself, some of the trust relationships do not need to be made explicit when the two domains in questions are really the same domain.
- You can use the same account for the resource admin account for all managed domains, as well as the service account, if you set up the appropriate trust relationships, group membership, and user rights.

Identity Manager Installation Notes

The Windows NT adapter does not require any additional installation procedures.

Usage Notes

None

Security Notes

This section provides information about supported connections and privilege requirements.

Supported Connections

Identity Manager uses the Sun Identity Manager Gateway to communicate with this adapter.

Required Administrative Privileges

Administrators must have permissions to create and maintain users and groups on the resource.

Provisioning Notes

The following table summarizes the provisioning capabilities of this adapter.

Feature	Supported?
Enable/disable account	Yes
Rename account	Yes
Pass-through authentication	Yes
Before/after actions	Yes
Data loading methods	• Import from resource • Reconciliation

Note The following admininistrative privileges are required to support Active Directory pass-thru authentication for Windows 2003 running in Windows 2000 mode:

- When configuring the Gateway to run as a user, that user must have the Act As Operating SystemUser Right to perform pass-through authentication for the Windows NT and Windows 2000/Active Directory resources. The user must also have the Bypass Traverse Checking User Right, but this right is enabled for all users by default.
- Accounts being authenticated must have the Access This Computer From The Network User Right on the Gateway system.
- When Identity Manager is updating Users Rights, there may be a delay before the security policy is propagated. Once the policy has been propagated, you must restart the Gateway.
- When performing account authentication, use the `LogonUser` function with the `LOGON32_LOGON_NETWORK` logon type and the `LOGON32_PROVIDER_DEFAULT` logon provider. (The `LogonUser` function is provided with the Microsoft Platform Software Development Kit.)

Account Attributes

The following table provides information about Windows NT account attributes.

Resource User Attribute	Tab/NT Field	Attribute Type
AccountLocked	General/Account is locked out	Boolean
description	General/Description	String
fullname	General/Full Name	String
groups	Member Of/Member of	String
HomeDirDrive	Profile/Connect	String
HomeDirectory	Profile/Local Path	String
LoginScript	Profile/Login script	String
PasswordNeverExpires	General/Password never expires	Boolean
Profile	Profile/Profile path	String

Resource User Attribute	Tab/NT Field	Attribute Type
userPassword	Password	Encrypted
WS_PasswordExpired	General/User must change password at next login	Boolean
PasswordAge	Not displayed by default. Indicates the amount of time since the last password change. To implement use the java.util.Date class to convert the value into a human-readable format.	Int

Resource Object Management

Identity Manager supports the following objects:

Resource Object	Features Supported	Attributes Managed
Group	Create, update, delete	description, member, groupType

Identity Template

$accountId$

Sample Forms

Built-In

Windows NT Create Group Form

Windows NT Update Group Form

Also Available

`NTForm.xml`

Troubleshooting

Use the Identity Manager debug pages to set trace options on the following class:

`com.waveset.adapter.NTResourceAdapter`

2 Implementing the AttrParse Object

The AttrParse object encapsulates a grammar used to parse user listings. It is primarily used by mainframe-based resource adapters that receive a screen full of data at a time and must parse out the desired results. (This technique is often called screen scraping.) The Scripted Gateway adapter also uses AttrParse with getUser and getAllUsers actions.

These adapters model the screen as a Java string. An instantiation of an AttrParse object contains one or more tokens. Each token defines a portion of the screen. These tokens are used to tokenize the screen string and allow the adapters to discover the user properties from the user listing.

After parsing a user listing, AttrParse returns a map of user attribute name/value pairs.

Configuration

As with all other Identity Manager objects, the AttrParse objects are serialized to XML for persistent storage. These AttrParse objects can then be configured to support differences in customer environments. For example, the ACF2 mainframe security system is often customized to include additional fields and field lengths. Since AttrParse objects reside in the repository, they can be changed and configured to account for these differences without requiring that a custom adapter be written.

As with all Identity Manager configuration objects, objects that are to be changed should be copied, renamed, and then modified.

1. From the Debug page, select AttrParse from the drop-down menu adjacent to the List Objects button. Click List Objects.
2. From the list of available objects, select the object you want to edit.
3. Copy, edit, and rename the object in your XML editor-of-choice.
4. From the Configure page, select Import Exchange File to import the new file into Identity Manager.
5. In your resource, change the AttrParse resource attribute to the name of the new AttrParse string.

For examples of AttrParse objects that ship with Identity Manager see the `sample\attrparse.xml` file. It lists the default AttrParse objects used by the screen scraping adapters.

AttrParse Element and Tokens

AttrParse Element

The AttrParse element defines the AttrParse object.

Attributes

Attribute	Description
name	Uniquely defines the AttrParse object. This value will be specified on the Resource Parameters page for the adapter.

Data

One or more tokens that parse user listings. There are currently 10 tokens supported by the AttrParse object. These are: t, int, str, eol, multiLine, opt, skip, skipToEol, skipWhitespace, and flag.

Example

The following example reads the first 19 characters of a line, trims extraneous whitespace, and assigns the string as the value to the USERID resource attribute. It then skips forward five spaces and extracts the NAME resource attribute. This attribute has a maximum of 21 characters, and whitespace is trimmed. The sample checks for the string “Phone number: “. A telephone number will be parsed out and assigned to the PHONE resource attribute. The phone number begins after the space in “Phone number: “ and ends at the next space encountered. The trailing space is trimmed.

```
<AttrParse name='Example AttrParse'>
   <str name='USERID' trim='true' len='19'/>
   <skip len='5'/>
   <str name='NAME' trim='true' len='21'/>
   <t offset='-1'>Phone number: </t>
   <str name='PHONE' trim='true' term=' '/>
</AttrParse>
```

The following strings satisfy the Example AttrParse grammar. (The • symbols represent spaces.)

```
gwashington123•••••ABCD•George•Washington••••Phone•number:•123-1234•
alincoln••••••••••XYZ••Abraham•Lincoln••••••Phone•number:•321-4321•
```

In the first case after parsing, the user attribute map would contain:

```
USERID="gwashington123", NAME="George Washington", PHONE="123-1234"
```

Similarly, the second user attribute map would contain:

```
USERID="alincoln", NAME="Abraham Lincoln", PHONE="321-4321"
```

The rest of the text is ignored.

collectCsvHeader Token

The `collectCsvHeader` token reads a line designated as the header of a comma seperated values (CSV) file.

The Scripted Gateway adapter is the only adapter that can use this token. The `collectCsvHeader` and `collectCsvLines` tokens are the only tokens that determine attributes that can be used with this adapter.

Each name in the header must be the same as a resource user attribute on the schema map on the resource adapter. If a string in the header does not match a resource user attribute name, it and the values in the corresponding position in the subsequent data lines will be ignored.

Attributes

Attribute	Description
idHeader	Specifies which value in the header is considered the account ID. This attribute is optional, but recommended. If it is not specified, then the value for the `nameHeader` attribute will be used.
nameHeader	Specifies which value in the header is considered the name for the account. This is often the same value as `idHeader`, and if not specified, the value in `idHeader` is used. This attribute is optional but recommended.
delim	Optional. The string that separates values in the header. The default value is , (comma).

Attribute	Description
minCount	Specifies the minimum number of instances of the string specified in the `delim` attribute that a valid header must have.
trim	Optional. If set to `true`, then if a value has leading or trailing blanks, remove them. The default is `false`.
unQuote	Optional. If set to `true`, then if a value is enclosed in quotes, remove them. The default is `false`.

Data

None

Example

The following example identifies `accountId` as the value to be used for the account ID. Whitespace and quotation marks are removed from values.

```
<collectCsvHeader idHeader='accountId' delim=',' trim='true'
unQuote='true'/>
```

collectCsvLines Token

The `collectCvsLines` token parses a line in a comma seperated values (CSV) file. The `collectCvsHeader` token must have been previously invoked.

The Scripted Gateway adapter is the only adapter that can use this token. The `collectCsvHeader` and `collectCsvLines` tokens are the only tokens that determine attributes that can be used with this adapter.

Attributes

If any of the following attributes are not specified, then the value is inherited from the previously-issued `collectCsvHeader` token.

Attribute	Description
idHeader	Specifies which value is considered the account ID.
nameHeader	Specifies which value is considered the name for the account.
delim	Optional. The string that separates values in the header. The default value is , (comma).
trim	Optional. If set to `true`, then if a value has leading or trailing blanks, remove them. The default is `false`.
unQuote	Optional. If set to `true`, then if a value is enclosed in quotes, remove them. The default is `false`.

Data

None

Example

The following example removes whitespace and quotation marks from values.

```
<collectCsvLines trim='yes' unQuote='yes'/>
```

eol Token

The eol token matches the end of line character (\n). The parse position will be advanced to the first character on the next line.

Attributes

None

Data

None

Example

The following token matches the end of line character.

```
<eol/>
```

flag Token

The `flag` token is often used inside an `opt` token to determine if a flag that defines an account property exists on a user account. This token searches for a specified string. If the text is found, AttrParse assigns the boolean value `true` to the attribute, then adds the entry to the attribute map.

The parse position will be advanced to the first character after the matched text.

Attributes

Attribute	Description
name	The name of the attribute to use in the attribute value map. The name is usually the same as a resource user attribute on the schema map on the resource adapter, but this is not a requirement.
offset	The number of characters to skip before searching for the text for the token. The offset can have the following values: • 1 or higher — Moves the specified number of characters before trying to match the token's text. • 0 — Searches for text at the current parse position. This is the default value. • -1 — Indicates the token's text will be matched at the current parse position, but the parse position will not go past the string specified in the `termToken` attribute, if present.
termToken	A string to use as an indicator that the text being searched for is not present. This string is often the first word or label in the next line on the screen output. The parse position will be the character after the termToken string. The termToken attribute can only be used if the len attribute is negative one (-1).

Data

The text to match.

Examples

1. The following token will match `AUDIT` at the current parse position, and if found, adds `AUDIT_FLAG=true` to the user attribute map.

```
<flag offset='-1' name='AUDIT'>AUDIT_FLAG</flag>
```

2. The following token will match *xxxx*CICS at the current parse position, where *xxxx* are any four characters, including spaces. If this string is found, AttrParse adds CICS=true to the user attribute map.

```
<flag offset='4' name='CICS'>CICS</flag>
```

int Token

The int token captures an account attribute that is an integer. The attribute name and integer value will be added to the account attribute map. The parse position will be advanced to the first character after the integer.

Attributes

Attribute	Description
name	The name of the attribute to use in the attribute value map. The name is usually the same as a resource user attribute on the schema map on the resource adapter, but this is not a requirement.
len	Indicates the exact length of the expected integer. The length can have the following values: • 1 or higher — Captures the specified number of characters and checks to see if the text is an integer value or if it matches the characters specified in the noval attribute. • -1 — The parser will take the longest string of digits starting at the current parse position unless the next characters equal the noval attribute. This is the default value.
noval	Optional. A label on the screen that indicates the attribute does not have an integer value. Essentially, it is a null value indicator. The parse position will be advanced to the first character after the noval string.

Data

None

Examples

1. The following token matches a 6-digit integer and puts integer value of those digits into the attribute value map for the SALARY attribute.

```
<int name='SALARY' len='6'/>
```

If the value 010250 is found, AttrParse adds SALARY=10250 to the value map.

2. The following token matches any number of digits and adds that integer value to the attribute map for the `AGE` attribute.

   ```
   <int name='AGE' len='-1' noval='NOT GIVEN'/>
   ```

   If the value `34` is found, for example, `AGE=34` would be added to the attribute map. For string `NOT GIVEN`, a value will not be added to the attribute map for the AGE attribute.

loop Token

The `loop` token repeatedly executes the elements it contains until the input is exhausted.

Attributes

None

Data

Varies

Example

The following example reads the contents of a CSV file.

```
<loop>
   <skipLinesUntil token=',' minCount='4' />
   <collectCsvHeader idHeader='accountId' />
   <collectCvsLines />
</loop>
```

multiLine Token

The `multiLine` token matches a pattern that recurs on multiple lines. If the next line matches the `multiLine's` internal AttrParse string, the parsed output will be added to the account attribute map at the top level. The parse position will be advanced to the first line that doesn't match the internal AttrParse string.

Attributes

Attribute	Description
opt	Indicates the internal AttrParse string might be optional. Indicates that there might be no lines that match the internal AttrParse string and that parsing should continue with the next token.

Data

Any AttrParse tokens to parse a line of data.

Example

The following `multiLine` token matches multiple group lines that have a `GROUPS[space][space][space]=` tag and a space delimited group list.

```
<multiLine opt='true'>
   <t>GROUPS[space][space][space]=</t>
   <str name='GROUP' multi='true' delim=' ' trim='true'/>
   <skipToEol/>
</multiLine>
```

AttrParse would add GROUPS = {Group1,Group2,Group3,Group4} to the account attribute map, given the following string is read as input:

```
GROUPS[space][space][space]= Group1[space]Group2\n
GROUPS[space][space][space]= Group3[space]Group4\n
Unrelated text...
```

opt Token

The `opt` token parses optional strings that are arbitrarily complex, such as those that are composed of multiple tokens. If the match token is present, then the internal `AttrParse` string is used to parse the next part of the screen. If an optional section is present, the parse position will be advanced to the character after the end of the optional section. Otherwise, the parse position is unchanged.

Attributes

None

Data

Contains the `apMatch` token, followed by an `AttrParse` token.

`apMatch` — Contains the token to match to determine whether the optional section is present. `apMatch` is a subtoken that can be used only within the `opt` token. `apMatch` token always contains the `flag` token as a subtoken.

`AttrParse` — Specifies how to parse the optional part of the screen. This version of the AttrParse element does not use the name argument. It can contain any other token.

Example

The following `opt` token attempts to match a `CONSNAME=` text token. If it is found, then it will parse a string of length 8, trim whitespace, and add the string to the account attribute map for the `NETVIEW.CONSNAME` attribute.

```
<opt>
   <apMatch>
      <t offset='-1'> CONSNAME= </t>
   </apMatch>
   <AttrParse>
      <str name='NETVIEW.CONSNAME' len='8' trim='true' />
   </AttrParse>
</opt>
```

skip Token

The `skip` token tokenizes areas of the screen that can be skipped and that don't contain useful information about the user that should be parsed. The parse position will be advanced to the first character after the skipped characters.

Attributes

Attribute	Description
len	indicates the number of characters to skip on the screen.

Data

None

Examples

In the following examples, the first token skips 17 characters, while the second skips only one character.

```
<skip len='17'/>
<skip len='1'/>
```

skipLinesUntil Token

The `skipLinesUntil` token skips over lines of input until one is found that has at least the specified number of instances of a given string.

Attributes

Attribute	Description
token	The string to search for.
minCount	The minimum number of instances of the string specified in the token attribute that must be present.

Data

None

Example

The following token skips forward to the next line that contains two commas. The parse position will be at the first character of that line.

```
<skipLinesUntil token=',' minCount='2'/>
```

skipToEol Token

The `skipToEol` token skips all characters from the current parse position to the end of the current line. The parse position will be advanced to the first character on the next line.

Attributes

None

Data

None

Example

The following token skips all characters until the end of the current line. The parse position will be at the first character of the next line.

```
<skipToEol/>
```

skipWhitespace Token

The `skipWhitespace` token is used to skip any number of whitespace characters. The system uses Java's definition of whitespace. The parse position will be advanced to the first non-whitespace character.

Attributes

None

Data

None

Example

The following token skips all the whitespace at the current parse position.

```
<skipWhitespace/>
```

str Token

The `str` token captures an account attribute that is a string. The attribute name and string value will be added to the account attribute map. The parse position will be advanced to the first character after the string.

Attributes

Attribute	Description
name	The name of the attribute to use in the attribute value map. The name is usually the same as a resource user attribute on the schema map on the resource adapter, but this is not a requirement.
len	Indicates the exact length of the expected string. The length can have the following values: • 1 or higher — Captures the specified number of characters, unless the characters equal the `noval` attribute. • -1 — Captures all the characters from the current parse position until the next whitespace character, unless the next characters equal the `noval` attribute. This is the default.
term	A string that indicates parsing should stop for this `str` token when any of the characters in the string are reached. If the len argument is 1 or higher, then either the `str` token will end at len, or the term character, whichever comes first.
termToken	A string to use as an indicator that the text being searched for is not present. This string is often the first word or label in the next line on the screen output. The parse position will be the character after the termToken string. The string added to the attribute map will be all the characters before the termToken was found. The termToken attribute can only be used if the len attribute is negative one (-1).
trim	Optional. A `true` or `false` value that indicates whether the returned value or multiple values (if the multi attribute is specified) are trimmed before being added to the account attribute map. The default value is `false`.
noval	A label on the screen that indicates the attribute doesn't have an string value. Essentially, it is a null value indicator. The parse position will be advanced to the first character after the `noval` string.

Attribute	Description
multiLine	A `true` or `false` value that indicates whether the string will span multiple screen lines. This attribute can only be used if a len attribute is supplied and is assigned a value greater than zero. If multiLine is present, end of line characters will be skipped until the number of characters specified in the len attribute have been parsed.
multi	A `true` or `false` value that indicates that the string captured is a multi-valued attribute that must be further parsed to find each sub-value. The multiple values can either be appended together using the `appendSeparator` or can be turned into a list of values.
delim	A delimiter for parsing the multi-valued string. This attribute can only be used if the multi attribute is specified. If this is not specified, then the multi `str` token is assumed to be delimited by spaces.
append	A `true` or `false` value that indicates that the multiple values should be appended together into a string using the `appendSeparator`. If append is not present, the multiple values will be put into a list for the account attribute value map. This attribute is used in conjunction with the multi attribute.
appendSeparator	Indicates the string to separate the multiple values for an append token. This attribute is only valid if the append attribute is set to true. If the `appendSeparator` is not present, the append attribute does not use a separator. Instead, it concatenates the multiple values into the result string.

Data

None

Examples

1. The following token matches a string of length 21 characters and trims whitespace off of the front and back.

   ```
   <str name='NAME' trim='true' len='21'/>
   ```

   Given the string `[space][space]George Washington[space][space]`, AttrParse adds `NAME="George Washington"` to the account attribute map.

2. The following token matches a string of arbitrary length terminated by a) (right parenthesis).

   ```
   <str name='STATISTICS.SEC-VIO' term=')' />
   ```

   Given the string, `2 - Monday, Wednesday - )text`, AttrParse adds `STATISTICS.SEC-VIO="2 - Monday, Wednesday - "` to the account attribute map.

3. The following token matches a list of words delimited by spaces from the current parse position to the end of the current line.

   ```
   <str name='GROUP' multi='true' delim=' ' trim='true'/>
   ```

   Given the string, `Group1 Group2 newGroup lastGroup\n`, AttrParse adds a list of group name strings {`Group1`, `Group2`, `newGroup`, `lastGroup`} to the account attribute map for the GROUP attribute.

4. The following token performs the same function as the previous example, except the account attribute map will contain `GROUP={Group1:Group2:newGroup:lastGroup}`

   ```
   <str name='GROUP' multi='true' delim=' ' trim='true' append='true'
   appendSeperator=':' />
   ```

t Token

The `t` token is used to tokenize text. It is commonly used to recognize labels during screen scraping and provide knowledge of where on the screen you are parsing. The parse position will be advanced to the first character after the matched text. The parser always moves left to right within a line of text.

Attributes

Attribute	Description
offset	The number of characters to skip before searching for the text for the token. The offset can have the following values: • 1 or higher — Moves the specified number of characters before trying to match the token's text. • 0 — Searches for text at the current parse position. This is the default value. • -1 — Indicates the token's text will be matched at the current parse position, but the parse position will not go past the string specified in the termToken attribute, if present.
termToken	A string that indicates parsing should stop for this token. The parse position will be the character after the termToken string. The termToken attribute can only be used if the offset attribute is negative one (-1).

Data

The text to match

Examples

1. The following token matches `Address Line 1:[space]` at the current parse position.

   ```
   <t offset='-1'>Address Line 1: </t>
   ```

2. The following token matches *xx*`Zip Code:[space]` at the current parse position, where *xx* can be any two characters, including spaces.

   ```
   <t offset='2'>Zip Code: </t>
   ```

3. The following token matches `Phone:[space]` at the current parse position. If AttrParse finds the string `Employee ID` first, then it will generate an error.

   ```
   <t offset='-1' termToken='Employee ID'>Phone: </t>
   ```

3 Adding Actions to Resources

This chapter describes how to create and implement actions on UNIX, Windows NT, and Windows Active Directory resources in Sun Java™ System Identity Manager.

What are Actions?

Actions are scripts that run within the context of a managed resource, if native support exists for scripted actions. For example, on a system with a UNIX operating system, actions are sequences of UNIX shell commands. In Microsoft Windows environments, actions are DOS-style console commands that can execute within the CMD console. Actions reside within Identity Manager repository as objects. In mainframe environments, actions are Javascript scripts that are capable of sending and receiving keystrokes and commands to and from the mainframe.

Use actions to perform work that is not performed directly against the resource account object but that is instead performed before or after that resource account is created, updated, or deleted. Resource actions support copying files to a new user's directory, updating the SUDOers file on UNIX for the user after they have been created, or other native activities. You could perform this type of work by using a custom resource adapter. However, it is simpler to deploy a resource adapter with actions than to deploy a custom resource adapter.

Three types of results messages are associated with actions:

- **Success** – Displays an Identity Manager success message.
- **Success with action output** – Displays an Identity Manager success message along with standard error and output information.
- **Failure** – Displays a Identity Manager failure message, along with standard error and output information.

Supported Processes

The following processes support before and after actions:

- `create`
- `update`
- `delete`
- `enable`
- `disable`

Supported Resources

You can add actions to the following resources:

- ACF2
- Domino
- HP-UX
- IBM AIX
- IBM OS/400
- Microsoft Active DIrectory
- Microsoft Exchange
- Microsoft Windows
- Natural
- RACF
- Sun Solaris
- TopSecret

Refer to the individual adapter sections in this book for a list of supported versions.

Defining Actions

To create a new resource action, you must

1. Create the action file.
2. Load the action file into Identity Manager.

Creating the Action File

To create an action, you must create a file with the following text:

```
<ResourceAction name="Name">
   <ResTypeAction restype="ResourceType" timeout="Milliseconds">
      <act>
      ...
      </act>
   </ResTypeAction>
</ResourceAction>
```

where:

- ***Name*** is the name of the resource action.

- ***ResourceType*** is the type of resource (such as AIX or HP-UX).
- ***Milliseconds*** (optional) is the amount of time to wait for the action to complete.

For example, the following XML defines an action for the Solaris resource:

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE Waveset PUBLIC 'waveset.dtd' 'waveset.dtd'>
<Waveset>
   <ResourceAction name='after-create'>
      <ResTypeAction restype='Solaris' timeout='60000'>
         <act>
         #!/bin/ksh
         echo "$WSUSER_accountId says Hello World!"
         # exit $DISPLAY_INFO_CODE if there is not a failure, but you
want
         # the output to be propagated to the UI
         #exit 0
         exit $DISPLAY_INFO_CODE
         </act>
      </ResTypeAction>
   </ResourceAction>
</Waveset>
```

Note The code contained within the `<act>` elements is the same as seen in a UNIX script (`ksh` or `sh`) or a Windows batch script.

Environment variables are exported and available to actions. These comprise any one of the schema-mapped attributes that have values on the user (defined in the resource schema map in the Identity System Resource Attribute column), prefixed by `WSUSER_`. For instance, the preceding example uses the environment variable `WSUSER_AccountId`, formed by preceding the `AccountId` attribute defined in the Solaris resource schema map by `WSUSER_`. These variables should be identified as environment variables within the respective shell, so that in Solaris, the variable name is preceded by `$` (dollar sign).

Notes:

- If you change any variable names in the Identity Manager Resource Attribute column on the schema map, you must change the names in this object as well.
- Because the actions are included in an XML expression, some characters must be escaped. Escape these characters as follows:

  `&` (ampersand): `&`

  `<` (less than): `<`
- On UNIX resources, spaces in attribute names are replaced with _ (underscore). On Windows NT and Active Directory resources, spaces are maintained.

- Multi-valued attributes consist of a comma-separated list, as in:

  ```
  WSUSER_groups=staff,admin,users
  ```

- Gateway-based adapters use a pipe-delimited list for multi-valued attributes. For example:

  ```
  WSUSER_NotesGroups=group1|group2|group3
  ```

- On Windows NT and Active Directory resources, actions are run using the Windows command interpreter `cmd.exe` with extensions enabled.

  Actions that run before a user operation must return a zero value. Otherwise, the operation is aborted.

- On mainframe adapters (ACF2, Natural, RACF, and Top Secret), actions are interpreted as Javascript. The JavaScript should be written assuming the availability of the following global variables:
  - **hostAccess** — an instance of `com.waveset.adapter.HostAccess`. It is used for sending and receiving keystrokes and commands to/from the mainframe.
  - **identity** — (String) Contains the accountId for the user on the resource
  - **userAttrs** — Instance of `java.util.Map` containing values for each of the Resource User Attributes needed by the action
  - **out** — Instance of `java.io.PrintStream`. If the Javascript writes to this stream (for example, `out.print("Hello")`), the contents will be traced, and will be shown in the UI results displayed for resource actions.
  - **err** — This is an instance of `java.io.PrintStream`. If the Javascript writes to this stream (for example,. `err.print("Error")`), the contents will be traced, and will be shown in the UI results displayed for resource actions.
- A Javascript is assumed to have completed successfully unless it throws an exception.

Loading the Action File into Identity Manager

Follow these steps to import the action into Identity Manager:

1. Log in to the Identity Manager Administrator Interface.
2. From the menu bar, select **Configure**, then **Import Exchange File**.
3. Enter or browse for the XML file containing the action, and then click **Import**.

Implementing Actions

After you have defined an action, follow these steps to implement it:

1. Define fields on the Identity Manager user form.
2. Add entries to the schema map for the resources on which you want to invoke the action.

Step 1: Define Identity Manager User Form Fields

Create user form fields to assign an action that will run before or after a user operation:

- **Field name** – Indicates when the action will run and for which operation
- **Field value** – Contains the action name

In this example, the field defines an action named `after-create` that runs after a user create operation:

```
<Field name='global.create after action'>
   <Expansion>
      <s>after-create</s>
   </Expansion>
</Field>
```

The field name is formatted as:

```
{create|update|delete} {before|after} action
```

For detailed information about working with forms in Identity Manager, refer to the chapter titled *Identity Manager Forms*.

Step 2: Add Schema Map Entries

Add an entry to the schema map for the resources on which you want the action to run. To do this:

1. Click **Resources** on the Identity Manager menu bar, and then select a resource.
2. On the Edit Resource page, click **Edit Schema**.
3. On the schema map, click **New Row** to add a row to the schema map.
4. In the Identity System User Attribute column, enter **`create after action`**.
5. Enter **`IGNORE_ATTR`** in the Resource User Attribute column. The **`IGNORE_ATTR`** entry causes the attribute to be ignored during normal account attribute processing.

6. Click **Save**.

Windows NT Examples

This section provides examples of actions that you can run on a Windows NT resource after a resource adapter performs the following operations:

- Creation of a user
- Update or edit of a user account
- Deletion of a user

Example 1: Action that Follows Creation of a User

This procedure shows how to include an action that will run after the creation of a new user on the WIndows NT resource.

1. Enter **`create after action`** in the Identity Manager User Attribute column of the Windows NT schema map.
2. In the Attribute Type column, select string.
3. In the Resource User Attribute column, enter **`IGNORE_ATTR`**. Leave the Required, Audit, Read Only, and Write Only columns unchecked.
4. Add the following code to the user form you are using to create or edit users:

```
<Field name='accounts[NT].create after action'>
   <Expansion>
      <s>AfterCreate</s>
   </Expansion>
</Field>
```

5. Create the following XML file and import it into Identity Manager. (Change the file paths according to your environment.)

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE Waveset PUBLIC 'waveset.dtd' 'waveset.dtd'>
<Waveset>
   <ResourceAction name='AfterCreate'>
      <ResTypeAction restype='Windows NT' timeout='6000'>
         <act>
         echo create >> C:\Temp\%WSUSER_accountId%.txt
         exit
         </act>
      </ResTypeAction>
   </ResourceAction>
</Waveset>
```

Example 2: Action that Follows the Update or Edit of a User Account

This procedure shows how to include an action that will run after the update or edit of a user on a Windows NT resource.

1. Enter **`update after action`** in the Identity Manager User Attribute column of the Windows NT schema map.
2. In the Attribute Type column, select string.
3. In the Resource User Attribute column, enter **`IGNORE_ATTR`**. Leave the Required, Audit, Read Only, and Write Only columns unchecked.
4. Add the following fields to the user form that you are using to create and edit users:

```
<Field name='accounts[NT].update after action'>
   <Expansion>
      <s>AfterUpdate</s>
   </Expansion>
</Field>
```

5. Create the following XML file and import it into Identity Manager. (Change file paths according to your environment.)

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE Waveset PUBLIC 'waveset.dtd' 'waveset.dtd'>
<Waveset>
   <ResourceAction name='AfterUpdate'>
      <ResTypeAction restype='Windows NT' timeout='6000'>
         <act>
         echo update >> C:\Temp\%WSUSER_accountId%.txt
         exit
         </act>
      </ResTypeAction>
   </ResourceAction>
</Waveset>
```

Example 3: Action that Follows the Deletion of a User

This procedure shows how to include an action that will run after the deletion of a user on the Windows NT resource.

1. Enter **`delete after action`** in the Identity Manager User Attribute column of the Windows NT schema map.
2. In the Attribute Type column, select string.
3. In the Resource User Attribute column, enter **`IGNORE_ATTR`**. Leave the Required, Audit, Read Only, and Write Only columns unchecked.
4. Add this to the Deprovision Form user form after the `</Include>` tag:

```
<Field name= 'accounts[NT].attributes.delete after action'>
   <Expansion>
      <s>AfterDelete</s>
   </Expansion>
</Field>
```

5. Create the following XML file and import into Identity Manager. (Change file paths according to your environment.)

```
<?xml version='1.0' encoding='UTF-8'?> <!DOCTYPE Waveset PUBLIC
'waveset.dtd' 'waveset.dtd'>
<Waveset>
   <ResourceAction name='AfterDelete'>
      <ResTypeAction restype='Windows NT' timeout='6000'>
         <act>
         echo delete >> C:\Temp\%WSUSER_accountId%.txt
         exit
         </act>
      </ResTypeAction>
   </ResourceAction>
</Waveset>
```

6. Edit the XML for the NT resource and add information to the “delete after action” schema mapping. Here is an example of a complete schema mapping for this resource with the new additions. (You will be adding the views-related information.)

```
<AccountAttributeType id='12' name='delete after action'
syntax='string' mapName='IGNORE_ATTR' mapType='string'>
   <Views>
      <String>Delete</String>
   </Views>
</AccountAttributeType>
```

Domino Example

Domino resources support before and after actions.

There are currently two supported types of actions: LotusScript and cmd shell. Any operation action can have any number of actions that will be executed.

The following examples demonstrate the use of LotusScript and cmd shell resource actions:

LotusScript Example

```
<ResourceAction name='iterateAttributes' createDate='1083868010032'>
   <ResTypeAction restype='Domino Gateway' actionType='lotusscript'>
      <act>
         Sub Initialize
            Main
         End Sub
         Sub Main
            Dim session As New NotesSession
            Dim doc As NotesDocument
            Set doc = session.DocumentContext
            Forall i In doc.Items
               Dim attrVal As Variant
               attrVal = doc.GetItemValue(i.Name)
            End Forall
         End Sub
      </act>
   </ResTypeAction>
</ResourceAction>
```

cmd shell Example

```
<ResourceAction name='getDirectoryContents'
createDate='1083868010032'>
   <ResTypeAction restype='Domino Gateway'>
      <act>dir</act>
   </ResTypeAction>
</ResourceAction>
```

Note A null `actionType` defaults to `cmd` script type.

Running LotusScript

On Domino, the execution of LotusScript is handled by an agent attached to a database. The Domino adapter will execute LotusScript in any one of the following ways:

Input	Results
agentName	Runs the agent.
agentName and script	Updates the agent with the script and runs the agent.
agentName, agentCreate, and script	Creates an agent with the script and runs the agent.

The following customized account attributes can be used with LotusScript. If any of these attributes are to be used, add the attribute on the Domino Gateway schema map. Specify `IGNORE_ATTR` as the value in the Resource User Attribute column.

- `agentName` - Name of the agent to execute. This attribute must be specified, or an error will be returned.
- `agentServer` - Location of the database where the agent has been installed, and where to run the agent. This attribute defaults to the value specified in the Registration Server Machine resource parameter (REG_SERVER) if not present.
- `agentDBName` - Database name where the agent can be found. This attribute defaults to the value specified in the Names Database resource parameter (NAB) on the resource.
- `agentCreate` - Flag that indicates whether the adapter should create a new agent, if the named agent is not found. This attribute defaults to false. A non-NULL value enables this flag.

Note If you specify agentCreate you must also specify LotusScript to be executed.

Arguments to LotusScript

Agents arguments will be given in a note handle to LotusScript via a special property from the back-end NotesSession class. It can be defined as follows:

```
NotesDocument = NotesSession.DocumentContext
```

The `NotesDocument` can be instantiated by the action script routine and its field values can be read in as parameters to the LotusScript subroutine.

The following is a Lotus script example that gets the name a value of any arguments defined in the document.

```
Dim session As New NotesSession
```

```
Dim doc As NotesDocument
Set doc = session.DocumentContext

Forall i In doc.Items
   Dim attrVal As Variant
   attrVal = doc.GetItemValue(i.Name)
   Print(" Attribute Name: " + i.Name + " Value: " + attrVal(0))
End Forall
```

All of the attributes defined during the action call will be put into the `NotesDocument` prefixed with `WSUSER_`, just as in the case of the NT actions.

Running cmd Shell

Actions are run using the Windows command interpreter cmd.exe with extensions enabled. Actions that run before a user operation must return a zero value. Otherwise, the operation is aborted.

Arguments to the cmd Shell

As with NT/ADSI `cmd` actions, the environment variables are exported and available to actions. These comprise any one of the schema-mapped attributes that have values on the user (defined in the resource schema map in the Identity Manager User Attribute column), prefixed by WSUSER_.

Multi-valued attributes consist of a pipe-separated list, as in:

```
WSUSER_groups=staff|admin|users
```

Extending Views

You can add additional attributes to a view. All attributes must be registered.

The user attributes that are available to the different provisioning activities in Identity Manager are limited to those necessary to complete the action. For example, when editing a user, all possible user attributes are retrieved form the assigned resources and available for update. In contrast, the Change Password process needs only a subset of attributes to perform the request.

Attribute Registration

Attributes can be registered in one of two locations:

Location	Register Attributes Here If ...
`AccountAttributeType` definition in the resource	... the attributes you want to update are specific to a particular resource, rather than to all resources of that type.
System Configuration Object	...you want to make global registrations for all resources of a particular type. These registrations must be done in XML format.

You can register different attributes for different views. For example, you can register the `lock` attribute for the Password view and the `firstname` attribute for the Rename view or the resource action for the Enable, Disable, or Delete view.

Note In the case of before or after actions, you must extend the view for any process except the create or update user process. For information on extending a view, see *Identity Manager Views*.

Global Registration

To make global registrations, add an attribute in the System Configuration object with this path:

```
updatableAttributes.ViewName.ResourceTypeName
```

where ***ViewName*** is one of Password, Reset, Enable, Disable, Rename, or Delete, and ResourceTypeName is the name of the resource type. The type name `all` is reserved for registrations that apply to all resources.

The value of this attribute must be a List of <String>s. The strings are names of the attributes you want to update. The following example registers the attribute named delete before action in the Delete view for all resources.

```
<Attribute name='updatableAttributes'>
   <Object>
      <Attribute name='Delete'>
         <Object>
            <Attribute name='all'>
               <List>
                  <String>delete before action</String>
               </List>
            </Attribute>
         </Object>
      </Attribute>
      <Attribute name='Enable'>
         <Object>
            <Attribute name='all'>
               <List>
                  <String>enable before action</String>
               </List>
            </Attribute>
         </Object>
      </Attribute>
   </Object>
</Attribute>
```

Resource-Specific Registration

To make resource-specific registrations, modify the resource object from the Identity Manager Debug page and insert a <Views> sub-element in the AccountAttributeType element. <Views> must contain a list of strings whose values are the names of the views in which this attribute can be updated.

```
<AccountAttributeType name='lastname' mapName='sn' mapType='string'>
   <Views>
      <String>Rename</String>
   </Views>
</AccountAttributeType>
```

In the view, attributes you want to modify are placed within this object:

`resourceAccounts.currentResourceAccounts[`***ResourceTypeName***`].attributes`

Example:

```
<Field name=
'resourceAccounts.currentResourceAccounts[OS400ResourceName].attribute
s.delete before action' hidden='true'>
   <Expansion>
      <s>os400BeforeDeleteAction</s>
   </Expansion>
</Field>
```

4 Synchronizing LDAP Passwords

This chapter describes the Identity Manager product enhancements to support password synchronization from the Sun Java™ System Directory Server (formerly known as Sun ONE Directory Server and iPlanet Directory Server) to the Identity Manager system.

Overview

Directory Server allows password changes to be processed by third parties through its public plugin API. A custom plugin, Password Capture plugin, was developed to capture password changes in Directory Server.

The responsibilities of the Password Capture plugin include:

- Intercepting password changes during LDAP ADD and MODIFY operations.
- Encrypting the new password value with a shared secret.
- Augmenting the original LDAP operation with a special attribute/value pair, `idmpasswd`, where the value is the encrypted password value.

The Directory Server Retro Changelog plugin must be installed on the directory server before the Password Capture plugin can be implemented. The Retro Changelog plugin records changes to the `idmpasswd` attribute in the changelog database after the operation is executed by the Directory Server core.

The LDAP resource adapter with Active Sync enabled polls the changelog database at regular intervals, parses relevant changes, and feeds these changes into Identity Manager. The LDAP adapter parses the `idmpasswd` attribute, decrypts the password using the shared secret, and makes the real password available to the rest of the system.

Password Capturing Process

The Password Capture plugin is invoked by the Directory Server core each time the server is about to process an LDAP ADD or an LDAP MODIFY operation. The plugin inspects the changes, and if there is a password change, it inserts the `idmpasswd` attribute/value pair, where the value is the encrypted password.

Passwords captured by the Password Capture plugin are encrypted using a shared key. (The same shared key is used by the configured LDAP Resource Adapter to decrypt the password).

If the change is accepted by the server, then the Retro Changelog plugin logs the changes, including the new value for the `idmpasswd` attribute, into the Retro-Changelog database. The LDAP resource adapter processes the change to the `idmpasswd` attribute and makes the value available to other components inside Identity Manager in the form of an encrypted string.

The `idmpasswd` attribute does not appear in the Directory Server's regular database when the user changes password.

Passwords in the Retro-Changelog Database

The encrypted password is recorded in the Retro-Changelog database. The Retro Changelog plugin can be configured to remove entries from the Retro Changelog database periodically. The correct setting of the database trimming depends on the target environment. Too frequent trimming may not allow room for small network outages, or other service disruptions and the LDAP resource adapter may miss certain changes. On the other hand, allowing the database to grow too large may increase the security risk associated with having encrypted passwords in the database.

Access to the contents of the Retro Changelog Database suffix (`cn=changelog`) should be limited. It is therefore recommended to allow read access to the LDAP resource adapter only.

Schema Changes

The `idmpasswd` attribute is defined as an operational attribute. Operational attributes do not require any changes to the objectclass definitions of the target entry. As a result, existing or new users in Directory Server do not need to be modified to use the password synchronization feature.

The `idmpasswd` attribute is defined in the schema as follows

```
attributeTypes: ( idmpasswd-oid NAME 'idmpasswd' DESC 'IdM Password'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.40{128} USAGE directoryOperation X-
ORIGIN 'Identity Manager' )
```

Configuring Identity Manager for LDAP Password Synchronization

Before an LDAP adapter can be used to synchronize LDAP passwords, you must perform the following tasks:

- Configure the LDAP resource adapter.
- Enable the password synchronization features.

Step 1: Configure the LDAP Resource Adapter

Use the following steps to configure the LDAP resource adapter to support password synchronization.

1. Import the LDAP Password ActiveSync Form into Identity Manager. This form is defined in `$WSHOME/sample/forms/LDAPPasswordActiveSyncForm.xml`.
2. In the Active Sync wizard for the resource, set the input form to LDAP Password ActiveSync Form.

Step 2: Enable Password Synchronization Features

To enable password synchronization in the LDAP resource adapter, Identity Manager provides a custom JSP page that allows the administrator to

- Enable password synchronization in any LDAP resource adapter
- Generate a configuration LDIF file (required for the installation of the Password Capture plugin)
- Re-generate the password encryption key and salt, if desired. This is an optional feature.

The LDIF file contains 3 entries:

- Schema change — Updates the Directory Server schema to allow the use of the `idmpasswd` operational attribute
- Plugin definition — Registers the plugin with the Directory Server and enables the plugin
- Plugin configuration — Provides basic configuration of the plugin. For example, the obfuscated password encryption key is in the configuration entry.

Use the following steps to implement these features.

1. Open the Identity Manager Configure Password Synchronization page, which is located at `http://`*`PathToIdentityManager`*`/configure/passwordsync.jsp`.
2. Select the LDAP resource that will be used to synchronize passwords from the **Resource** menu.
3. Select **Enable Password Synchronization** from the **Action** menu.
4. Click **OK**. The page refreshes to display a new item in the **Action** menu.
5. Select **Download plugin configuration LDIF** from the **Action** menu.
6. Click **OK**. The page refreshes to display several new options.
7. Select the resource's operating system from the **Operating System Type** menu.
8. In the **Plugin Installation Directory** field, enter the directory on the host where the plugin will be installed.
9. Click **OK** to generate and download the LDIF file. If necessary, you may now regenerate an encryption key.
10. Select **Regenerate encryption key** from the **Action** menu.
11. Click OK. The encryption parameters are updated.

After password synchronization is enabled, the following attributes on the Resource Specific Settings page on Active Sync wizard parameters page of the resource will be displayed.

- **Enable password synchronization**
- **Password encryption key**
- **Password encryption salt**

Only the **Enable password synchronization** field should be editable. The encryption attributes should only be updated using the JSP page.

Installing the Password Capture Plugin

Before starting the plugin installation, make sure you completed the resource configuration. See *Configuring Identity Manager for LDAP Password Synchronization* on page 4-3 for more information.

Note If the Directory Server instances are set up in a multi-master replicated environment, then the plugin must be installed on each master replica. For example, iPlanet Directory Server 5.1 allows up to two master replicas, while Sun ONE Directory Server 5.2 and later allows four master replicas to be defined.

To install the Password Capture plugin, you must perform the following general steps. See the product documentation for detailed information about performing these tasks.

1. Upload the configuration LDIF file into the target Directory Server. You can use the LDAP command line utilities bundled with the Directory Server. For example,

   ```
   /opt/iPlanet/shared/bin/ldapmodify -p 1389 -D "cn=directory
   manager" -w secret -c -f /tmp/pluginconfig.ldif
   ```

2. Place the plugin binary (`idm-plugin.so`) on the host where the Directory Server is running. In this example, `/opt/SUNWidm/plugin`. Make sure that the user running the directory server is able to read the plugin library. Otherwise, the Directory Server will fail to start.
3. Restart the Directory Server. (For example, `/opt/iPlanet/slapd-examplehost/restart-slapd`). The Password Capture plugin is not loaded after Directory Server is restarted.

Note In a multi-master replicated environment, new plugin configuration must be generated for each installation (unless the operating system type and the plugin installation directory are the same on each host). In this type of environment, repeat the procedure described in *Step 2: Enable Password Synchronization Features* on each installation.

5 Active Directory Synchronization Failover

This document describes how to handle an Active Directory synchronization failover. Implementing this customization can help limit the number of repeated events that occur when you switch to a new domain controller.

The Active Directory synchronization failover uses a task to periodically collect and maintain a history of the `HighestCommittedUSN` from a configurable set of domain controllers to which it can fail over. If the ActiveSync domain controller goes down, another task can be run that will change the configuration of the Active Directory resource to point to one of the failover domain controllers. Because changes made in Active Directory can take a while to replicate to all domain controllers, Active Directory ActiveSync cannot just start processing only new changes on the failover domain controller. Instead, it must also look at older changes made on the failover domain controller that might not have been replicated to the domain controller before it went down. To this end, it will use a saved `HighestCommittedUSN` for the failover domain controller that is far enough in the past to account for any replication delay. This prevents ActiveSync from missing events, but some changes will likely be processed twice.

Architectural Components

This procedure involves the following components:

- The Active Directory Synchronization Failure Process, which is defined on the Active Directory resource by the On Synchronization Failure Process Active Directory resource attribute
- Active Directory Recovery Collector Task
- Active Directory Failover Task

On Synchronization Failure Process Resource Attribute

The On Synchronization Failure Process Active Directory active synchronization resource attribute specifies the name of a process to be executed on a synchronization failure. By default, the value of this resource attribute is empty.

This attribute gives Identity Manager administrators the ability to execute a process when Active Directory synchronization failures occur.

Active Directory On Failure Process

The process specified by the resource attribute is launched by the resource on failure. Sun recommends invoking a process that sends email to the Active Directory administrator that alerts them to a synchronization failure. The body of the email might contain the error messages that were returned from the adapter poll method.

You can also design a business process that, when a specified error occurs, automatically calls the Synchronization Failover task after an approval by an administrator is given.

Process Context

The following arguments are available to the native process.

Argument	Description
`resourceName`	Identifies the resource where the failure occurred
`resultErrors`	Lists strings that represent the errors returned by the poll method
`failureTimestamp`	Indicates when the failure occurred

Active Directory Recovery Collector Task

You can schedule and launch the Active Directory Recovery Collector task from the Task Schedule pages of the Identity Manager Administrator interface. This process uses the resource object interface to contact each domain controller's `rootDSE` object. The task's schedule determines the frequency at which the data is collected from the domain controllers.

This task collects and stores resource recovery information in a Configuration object named `ADSyncRecovery_resourceName`. The extension to this configuration object is a GenericObject that stores a list of `HighestCommittedUSN` and the timestamp (milliseconds) that was collected for each domain controller.

During each execution, the task prunes old values for `HighestCommittedUSN` from the recovery data. You can configure the length of time to store this data through the `daysToKeepUSNS` argument.

Arguments

Argument	Description
`resourceName`	Specifies the Active Directory resource for which Identity Manager collects backup data.
`backupDCs`	Lists the fully qualified domain controller hostnames that should be contacted for recovery data. This can and should include the original host, which permits Identity Manager to include the source resource host if Identity Manager must fail over to the resource.
`daysToKeepUSNS`	Specifies the number of days for which Identity Manager stores the data (default is 7 days).

Active Directory Failover Task

This task reconfigures the failed resource and the IAPI Object to use an alternate domain controller and `usnChanged` starting point. The task input form displays the available `usn-changed` times for a given host from the stored failover data.

Certain errors can identify conditions where failover is appropriate. One example of the potential difficulty of automatically calling the failover task is the `java.net.UnknownHostException` error message. The failure indicated by this message can occur for at least two reasons:

1. The host cannot be reached from the gateway machine due to a temporary routing issue.
2. The host cannot be reached and will be down for the next eight hours due to a planned outage.

Failover Modes

You can take one of two approaches towards implementing Active Directory failover resolution:

- **Manual mode**. When a problem occurs, the administrator specifies which backup domain controller and USN to use. This is the only mode available when running tasks from the Identity Manager interface.

- **Semi-auto mode**. Semi-auto mode permits you to semi-automate the fail-over resolution process. In semi-auto mode, the task uses the collected data to identify the best backup domain controller and USN to use. It computes this by looking for a collection point that is closest to a derived `TargetTimestamp` without exceeding this value

  where `TargetTimestamp = (FailureTimestamp - MaxReplicationTime)`

  Semi-auto mode is not available from the Identity Manager Administrator interface.

Arguments

If you have determined that launching semi-auto failover is appropriate for a particular error, set the following tasks arguments. Setting these arguments reconfigures the failed resource and the IAPI Object to use an alternate domain controller and `usnChanged` starting point.

Argument	Description
`resourceName`	Identifies (by name or resource ID) where the failure has occurred.
`autoFailover`	Specifies whether auto failure is set. Must be set to `true`.
`failureTimestamp`	Indicates when the failure occurred. This value is derived from the onSync failure process.
`maxReplicationTime`	Specifies the maximum time in hours for data to replicate across an Active Directory environment.

To manually specify which domain controller to fail over to and which saved `HighestCommittedUSN` number to start from, set the following arguments.

Argument	Description
resourceName	Specifies the name or ID of the resource where the failure has occurred.
backupDC	Specifies the name of the host with which to begin the synchronization process.
usnDate	The timestamp to use that correlates to a collected HighestCommittedUSN changed value from the collected data. This would be computed just as targetTime was computed in the semi-auto mode.
restartActiveSync	Specifies whether to start ActiveSync after the switch to the new domain controller is complete.

Resource Object Changes

The Active Directory Recovery Collector task updates either the LDAPHostname or the GlobalCatalog resource attribute value (depending on which value is in use). If the search subdomains resource attribute is set to true, and the global catalog attribute value is not empty, the global catalog server attribute is changed. Otherwise, the LDAPHostname is changed to the name of the backup domain controller.

IAPI Object Changes

The Active Directory Recovery Collector task also updates the IAPI object so that the Active Directory resource adapter knows which changes to look for the next time it runs. The task updates the HighCommitedUSN value for both lastUpdated and lastDeleted attribute values.

Setting Up Active Directory Synchronization Failover

Step One: Configure the Active Directory Synchronization Recovery Collector Task

- Configure the maximum number of hours to retain data. The default value is seven days. This value controls how far back the HighestCommittedUSN values are kept.

  You must configure one workflow per Active Sync resource that must be configured.

- Schedule this task from the Task page of the Identity Manager Administrator interface. The polling interval, which establishes how often to contact each host for their `HighestCommittedUSN` value, is set by the task schedule.

  When this task is executed, it calls out to the Active Directory adapter to retrieve the `HighestCommittedUSN` number from each domain controller's `rootDSE`. It then stores this value in an Identity Manager configuration object. The task generates one configuration object per defined Active Sync resource to store alternate domain controller `HighestCommittedUSN` values.

Step Two: Define the Active Directory On-Error Process Active Sync Attribute

On each Active Directory Active Sync resource, Identity Manager defines an `onError` process that is called when a failure occurs during the synchronization of a resource. If an Active Directory resource defines an on-error process, this process is called if there errors occur when the poll method is called on the resource during active synchronization. This process checks the result from the IAPI objects, and if an error occurs, calls the defined process.

Configure this process to notify an administrator through email when an error occurs. Include the error text in the email body so that the administrator can determine if the error warrants that Identity Manager fails over to another domain controller.

Using the error text, the administrator is alerted to a potentially lengthy outage or an outrage due to a temporary, quickly resolved issue (such as a temporary routing issue that is resolved by the next poll attempt).

Step Three: Run Active Directory Synchronization Failover Task for the Failed Resource

If the domain controller returns an error that warrants failing over to another domain controller, run the Active Directory Synchronization Failover task from the Task page.

For manual fail-over mode, the fail-over task requests

- The name of the downed domain controller or resource
- The name of the DC hostname to fail over
- The timestamp of the collected `HighestCommittedUSN` value to use

You also must choose whether to restart ActiveSync after the switch to a new domain controller is complete.

How the Task Works

When executed, the Active Directory Synchronization Failover task

1. Stops the Active Sync process on the failed resource
2. Reads in the fail over configuration object
3. Changes necessary resource attribute values
4. Optionally restarts the Active Sync process.

Example of Synchronization Failure Workflow

You can configure the following example workflow as the On Synchronization Failure Process resource attribute of an Active Directory resource. The workflow looks for the `java.net.UnknownHostException` error message. If it finds this message, the workflow launches a notification email to the administrator.

```
<TaskDefinition name='Sample AD Sync On Error Workflow'
  executor='com.waveset.workflow.WorkflowExecutor'
  syncControlAllowed='true' execMode='sync'
  taskType='Workflow'>

   <Extension>
     <WFProcess title='Example AD Sync OnError Workflow'>
        <Variable name='resultErrors' input='true'>
          <Comments>Errors returned from the resource.
          </Comments>
        </Variable>

        <Variable name='resourceName' input='true'>
          <Comments>Name of the AD resource that returned the errors.
           </Comments>
        </Variable>

        <Variable name='failureTimestamp' input='true'>
          <Comments>Failure timestamp, when it occurred.
          </Comments>
        </Variable>

        <Activity name='start'>
          <Transition to='checkErrors'/>
        </Activity>
        <Activity name='checkErrors'>
          <Variable name='criticalError'>
           <Comments>Local variable to hold if we need to notify
           </Comments>
          </Variable>

        <Action name='iterateMessage'>
          <dolist name='msg'>
            <ref>resultErrors</ref>
```

```
              <cond>
                <match>
              <ref>msg</ref>
                <s>java.net.UnknownHostException</s>
                </match>
                  <set name='criticalError'>
                   <s>true</s>
                  </set>
              </cond>
            </dolist>
          </Action>

          <Transition to='notify'>
            <notnull>
              <ref>criticalError</ref>
            </notnull>
          </Transition>
          <Transition to='end'/>
          </Activity>

          <Activity name='notify'>
            <Action application='notify'>
              <Argument name='template'
    value='#ID#EmailTemplate:ADSyncFailoverSample'/>
              <Argument name='resultErrors' value='$(resultErrors)'/>
            </Action>
          <Transition to='end'/>
          </Activity>

          <Activity name='end'/>
       </WFProcess>
    </Extension>
  </TaskDefinition>
```

Index

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

R

S

T

U

V

W

X