

CA Identity Manager

Connectors Guide

r12 CR1



This documentation and any related computer software help programs (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by CA at any time.

This Documentation may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA. This Documentation is confidential and proprietary information of CA and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of the Documentation for their own internal use, and may make one copy of the related software as reasonably required for back-up and disaster recovery purposes, provided that all CA copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the provisions of the license for the Product are permitted to have access to such copies.

The right to print copies of the Documentation and to make a copy of the related software is limited to the period during which the applicable license for the Product remains in full force and effect. Should the license terminate for any reason, it shall be the user's responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

EXCEPT AS OTHERWISE STATED IN THE APPLICABLE LICENSE AGREEMENT, TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO THE END USER OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED OF SUCH LOSS OR DAMAGE.

The use of any product referenced in the Documentation is governed by the end user's applicable license agreement.

The manufacturer of this Documentation is CA.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

Copyright © 2008 CA. All rights reserved.

CA Product References

This document references the following CA products:

- CA Identity Manager
- eTrust Directory

Contact CA

Contact Technical Support

For online technical assistance and a complete list of locations, primary service hours, and telephone numbers, contact Technical Support at <http://ca.com/support>.

Provide Feedback

If you have comments or questions about CA product documentation, you can send a message to techpubs@ca.com.

If you would like to provide feedback about CA product documentation, please complete our short [customer survey](#), which is also available on the CA Support website.

Contents

Chapter 1: Connectors	13
Connector Overview	13
Connector Types	15
Flat Connectors	16
Hierarchical Connectors	16
Static Connectors	16
Dynamic Connectors	17
Common Connector Operations	17
Add Operation	18
Modify Operation	18
Delete Operation	18
Rename Operation	18
Search Operation	18
Finding Capability and Initial Attributes	19
 Chapter 2: Installation Requirements and Scenarios	 21
Connector and Agent Installation	21
Scenario for Connector and Agent Installation	22
Connectors Installed Using the Connector and Agent Installation	22
Connector and C++ Connector Server Installation	23
Connectors Installed Using the Connector and C++ Connector Server Installation	24
Connector Only Installation	24
Connectors Installed Using the Connector Only Installation	25
Java Connector Server Installation	26
Connectors Using the Java Connector Server Installation	27
C++ to Java Connectors Migration	27
 Chapter 3: Managing the Connectors	 29
The Provisioning Manager	29
Endpoints	29
Accounts	30
Provisioning Roles and Account Templates	30
Groups	30
DYN GUI	31

Chapter 4: Database Connectors

33

DB2 UDB for z/OS Connector	34
DB2 Installation Prerequisites	34
DB2 Limitation	34
DB2 z/OS Support for FIPS and IPv6	34
Connector-Specific Features	35
IBM DB2 UDB Connector	40
DB2 UDB Installation	40
Connector-Specific Features	41
MS SQL Server Connector	45
MS SQL Configuration	45
MS SQL Migration Steps	46
MS SQL Support for FIPS and IPv6	46
Connector-Specific Features	47
Ingres Connector	48
Administer the Ingres Server	49
Ingres Database Limitations	50
Ingres Support for FIPS and IPv6	50
Connector-Specific Features	51
Oracle Connector	53
Oracle Configuration	54
Oracle Migration Steps	55
Oracle Support for FIPS and IPv6	55
Connector-Specific Features	55
Oracle Applications Connector	58
Oracle Applications Installation and Configurations	59
Oracle Applications Support for FIPS and IPv6	62
Connector-Specific Features	62

Chapter 5: Distributed Connector

67

Windows NT Connector	67
Configuring	68
Upgrading the Provisioning Server	68
Installing the Provisioning Windows NT Remote Agent with msixexec	68
Installing the Provisioning Agent for Windows Local Users and Groups with setup.exe	69
Configure the CAM and CAFT Service for Windows NT	69
Windows NT Support for FIPS and IPv6	74
Connector-Specific Features	74

Chapter 6: Mainframe Connectors

83

CA ACF2 Connector	83
How to Configure Your CA ACF2 System	84
Securing Provisioning Server Communication to the CA LDAP Server	85
ACF2 Support for FIPS and IPv6	85
Connector-Specific Features	85
Acquire a CA ACF2 System	86
ACF2 Provisioning Roles and Account Templates	88
ACF2 Accounts	88
Access and Resource Rules and Rule Keys	89
Change Control	89
Rule Lines	90
Secondary Authids	90
TSO Alias Support	94
ACF2 Program Exits	95
Proxy Configuration	95
Proxy Administration Support	96
Conventions	97
LDAP Directory Services (LDS)	97
CA Top Secret Connector	101
Install and Configure the TSS Connector	101
Top Secret Support for FIPS and IPv6	102
Connector-Specific Features	103
Best Practices	118
RACF Connector	126
How to Configure Your RACF System	127
Securing Provisioning Server Communication to the CA LDAP Server	127
RACF Support for FIPS and IPv6	128
Connector-Specific Features	128

Chapter 7: UNIX Connector

137

UNIX Connector	137
Product Interchange Format (PIF) Installation Procedures	138
Install and Configure the CAM Service for Irix and NCR	150
Manage the CAM Service	152
How to Restrict CAFT Commands	156
Install the CAM and CAFT Encryption Key	157
UNIX Support for FIPS and IPv6	158
Connector-Specific Features	159
Using Identity Manager Exit Commands	161
Managing Passwords	163

Chapter 8: Directory Services Connectors 175

Active Directory Services Connector	176
Configure Your Windows Servers Using SSL	177
ADS Defaults	185
ADS Support for FIPS and IPv6	195
Connector-Specific Features	196
Generic LDAP Connector	216
Connector-Specific Features	216
Extending Custom Attributes	221
Extending the Generic LDAP Connector	222
NDS Connector	238
Additional NDS Install Requirements	238
NDS Support for FIPS and IPv6	238
Connector-Specific Features	239

Chapter 9: Non-Managed System Connector 243

Universal Provisioning Connector	243
UPO Installation Requirements	243
Install UPO	243
How to Configure UPO	244
UPO Support for FIPS and IPv6	245
Connector-Specific Features	246
UPO Operational Modes	246
Data Passed to Program Exits	247
Exits	247
Service Level Agreement (SLA)	248
Program Exit Objects	248
How to Add the Exit Reference to the UPO Endpoint	248
How to Explore the UPO Endpoint	252
Assigning Account Templates to Provisioning Roles	253
Managing Exits	253
Non-Managed Mode Operations	259

Chapter 10: Groupware Connectors 265

Microsoft Exchange 200x Connector	265
How to Manage Mailboxes	266
Configuring the Exchange 200x Remote Agent Component	267
Activating CAM and CAFT Encryption	270
Managing the CAM and CAFT Service for Microsoft Exchange 200x	272
E2K Support for FIPS and IPv6	273

Connector-Specific Features	273
Lotus Notes/Domino Connector	280
LND Installation	280
Configure the Lotus Notes/Domino Connector	286
Set Up the Domino Server	290
Administration Process (Adminp)	292
Etautil Script Considerations	293
LND Support for FIPS and IPv6	293
Connector-Specific Features	294

Chapter 11: Host Access Management Connectors 313

CA Access Control Connector	314
Recommended Patch Levels	314
ACC Connector Multi-Threading Support	315
Runtime Environment Settings	315
Connector-Specific Features	320
Password Synchronization	330
CA SSO Connector for Advanced Policy Server	337
Configuring the CA Single Sign-On Server	338
Using Failover	340
Enable Application Password Propagation	341
PLS Support for FIPS and IPv6	342
Connector-Specific Features	342
Kerberos Connector	345
Kerberos Connector Limitations	345
Kerberos Installation and Deployment	348
Connector-Specific Features	360

Chapter 12: Midrange Connectors 393

OS/400 Connector	393
OS/400 Installation	394
OS/400 Support for FIPS and IPv6	406
Connector-Specific Features	407
OpenVMS Connector	418
OpenVMS Installation and Configuration	418
OpenVMS Support for FIPS and IPv6	427
Connector-Specific Features	427
HP NonStop SAFEGUARD Connector	431
Installing the HP NonStop SAFEGUARD Agent	431
Common Communication Interface (CCI) Installation for HP NonStop Safeguard	435
NSK Support for FIPS and IPv6	436

Connector-Specific Features	436
Chapter 13: Middleware Connectors	439
Embedded Entitlements Manager Connector	439
EEM Installation	440
EEM Support for FIPS and IPv6	440
Connector-Specific Features	440
Acquire an EEM Server Machine	441
EEM Account Templates	442
EEM Accounts	442
EEM Access Policies	442
EEM Calendars	443
EEM Groups	443
EEM Resource Classes	443
Chapter 14: Business Application Connectors	445
SAP Connector	446
SAP Installation	446
Connector-Specific Features	451
Siebel Connector	461
Siebel Installation	462
Connector-Specific Features	462
Chapter 15: Remote Access Connector	471
The RSA ACE (SecurID) Connector	471
RSA Installation	472
Connector-Specific Features	473
Chapter 16: Public Key Infrastructure Connector	481
Entrust PKI Connector	481
Entrust PKI Installation	481
Connector-Specific Features	484
Chapter 17: Feed Connector	493
PeopleSoft Feed	493
Install the PeopleSoft Feed	493
How the PeopleSoft Feed Works	494
Supported Use Cases	496
Using the XSLT Template	498

Using PeopleSoft Feed	498
PeopleSoft Feed Localization	499
Authenticating to the Identity Manager Server	499
Configuring the PeopleSoft Feed	501
PeopleSoft Feed Error and Response Handling	503
 Appendix A: Endpoint Schema and Structure	 505
Endpoint Schema	505
SchemaAbridged.txt File	506
SchemaUnabridged.txt File	506
File Formats	507
Endpoint Structure	510
Distinguished Names	510
 Appendix B: Procedures	 577
Endpoint Procedures	577
Registering Endpoints	577
Modifying or Viewing Endpoints	578
Account Template Procedures	578
Creating Account Templates	578
Modifying or Viewing Account Templates	579
Using the Default Account Template	579
Account Procedures	579
Creating Accounts	579
Adding Group Memberships to Existing Accounts	580
Modifying or Viewing Account Properties	581
Checking Account Synchronization	581
Deleting Accounts	582
Group Procedures	582
Creating Groups	582
Deleting Groups	583
 Index	 585

Chapter 1: Connectors

This section contains the following topics:

[Connector Overview](#) (see page 13)

[Connector Types](#) (see page 15)

[Common Connector Operations](#) (see page 17)

[Finding Capability and Initial Attributes](#) (see page 19)

Connector Overview

Identity Manager Connectors run as part of the wider Provisioning Server architecture and communicate with the systems managed in your environment. A connector acts as a gateway to a native endpoint type system technology. For example, to manage computers running Active Directory Services (ADS) you must install the ADS connector on a Connector Server with which the Provisioning Server can communicate. Connectors manage the objects that reside on the systems. Managed objects include accounts, groups, and optionally, endpoint type specific objects.

Connectors are installed on the Connector Server and some components are installed on the Provisioning Server (for example, server plug-in) or Provisioning Manager (user interface plug-ins).

Some connectors require an agent on the systems they manage in order to complete the communication cycle; in which case, they can be installed using the Provisioning Installer. There are two categories of agents:

- **Remote Agents**—Installed on the managed endpoint systems.
- **Environment Agents**—Installed on systems, such as CA ACF2, CA Top Secret, and RACF.

Certain components now work on UNIX and Windows, including these C++ Connector Server-based options:

- UNIX (ETC, NIS)
- Access Control (ACC)

Note: The UNIX ACC connector can manage only UNIX ACC endpoints. The Windows ACC connector is required to manage the Windows ACC endpoints but can also manage UNIX ACC endpoints.

- OpenVMS
- CA-ACF2
- RACF
- CA-Top Secret

You can access the other C++ Connector server-based connectors from the Solaris Provisioning Server by using on the Connector Server Framework (CSF). The CSF allows a Provisioning Server on Solaris to communicate with connectors running on Windows.

Note: The CSF must run on Windows to use these connectors.

Connector Types

The following types of connectors are available:

- **Flat Connectors**—A flat connector has a single set of containers (folders) beneath an endpoint object that each contain all objects of the same type (for example, all accounts or all groups).

The majority of connectors are considered flat connectors. Examples include:

- Windows NT
- UNIX
- Oracle

- **Hierarchical Connectors**—Hierarchical connectors manage large enterprise solutions. Their endpoints have a series of containers beneath where their managed objects are stored. These containers act logically as a series of nested flat endpoint containers. The endpoint itself is also a container.

Some examples of hierarchical connectors include:

- Active Directory Services
- NDS
- LDAP

- **Static Connectors**—Static connectors are designed to work in a fixed way and their behavior is largely dictated by their implementation and frozen at the time of release, rather than by the user of the connector.

Examples of static connectors include:

- ADS
- Windows NT
- RSA

- **Dynamic Connectors**—You can configure dynamic connectors using Identity Manager Connector Xpress. You define their behavior and the way they are mapped to the target endpoint system.

Dynamic connectors are flexible and configurable. You do not have to have significant coding knowledge to define the behavior of a dynamic connector.

Some examples of dynamic connectors include:

- JDBC versions of Oracle
- JNDI/LDAP

Note: For more information on Dynamic Connectors, see the *Connector Xpress Guide* and help system.

Flat Connectors

A flat connector has a single set of containers (folders) beneath an endpoint object, that each contain all objects of the same type (for example, all accounts or all groups).

The majority of connectors are considered flat connectors. Examples include:

- Windows NT
- UNIX
- Oracle

Hierarchical Connectors

Hierarchical connectors manage large enterprise solutions. Their endpoints have a series of containers beneath where their managed objects are stored. These containers act logically as a series of nested flat endpoint containers. The endpoint itself is also a container.

Some examples of hierarchical connectors include:

- Active Directory Services
- NDS
- LDAP

Static Connectors

Static connectors are designed to work in a fixed way and their behavior is largely dictated by their implementation and frozen at the time of release, rather than by the user of the connector.

Examples of static connectors include:

- ADS
- Windows NT
- RSA

Dynamic Connectors

Dynamic connectors can be configured using Identity Manager Connector Xpress. You define their behavior and the way they are mapped to the target endpoint system.

Dynamic connectors are flexible and configurable. You do not have to have significant coding knowledge to define the behavior of a dynamic connector.

Some examples of dynamic connectors include:

- JDBC versions of Oracle
- JNDI/LDAP

Note: For more information on Dynamic Connectors, see the *Connector Xpress User Guide* and help system.

Common Connector Operations

Once a connector has been installed on a Connector Server, the Provisioning Server can manage accounts on the target endpoint system provided:

- The Provisioning Server has been configured to allow communication with the Connector Server.
- Any required software components have also been installed on the Provisioning Server.
- An endpoint has been created for the target endpoint system.

With a few exceptions, all of the managed objects for an endpoint system, such as accounts, groups, and endpoint-type specific objects, can have the same operations performed on them. The operations are listed below:

- **Add**—Lets you add managed objects such as accounts and groups to an endpoint system.
- **Modify**—Lets you change the value of attributes on an endpoint system's managed objects, including modifying associations between them (for example, changing which accounts belong to a group).
- **Delete**—Lets you delete an endpoint system's managed objects.
- **Rename**—Lets you rename an endpoint system's managed objects.
- **Search**—Lets you query the value of attributes stored for an endpoint system or the managed objects contained by it.

Examples of exceptions are:

- Objects for which renaming is not supported
- Objects that are managed using native tools on the endpoint system rather than directly by the connector and therefore, cannot be added or deleted.

Add Operation

The Add operation lets you add managed objects such as accounts and groups to an endpoint system.

Modify Operation

The Modify operation lets you change the value of attributes on an endpoint system's managed objects, including modifying associations between them (for example, changing which accounts belong to a group).

Delete Operation

The Delete operation lets you delete an endpoint system's managed objects.

Rename Operation

The Rename operation lets you rename an endpoint system's managed objects.

Search Operation

The Search operation lets you query the value of attributes stored for an endpoint system or the managed objects contained by it.

Finding Capability and Initial Attributes

To find out which attributes are defined as capabilities and which are initial, you need to generate the eTACapability.txt file. Enter the following command from a Windows Command prompt:

```
PS_HOME\dumpptt.exe -c > eTACapability.txt
```

PS_Home

Specifies

Value: C:\Program Files\CA\Identity Manager\Provisioning Server\bin

A version of the file is generated for all of the connectors that you have installed.

Chapter 2: Installation Requirements and Scenarios

This section contains the following topics:

[Connector and Agent Installation](#) (see page 21)

[Connector and C++ Connector Server Installation](#) (see page 23)

[Connector Only Installation](#) (see page 24)

[Java Connector Server Installation](#) (see page 26)

[C++ to Java Connectors Migration](#) (see page 27)

This chapter describes the administrative requirements and installation scenarios for connectors.

To install your Identity Manager connectors, locate the Identity Manager Installation Program and follow the onscreen instructions to complete your installation. For detailed connector installation and configuration information, see the installation section of each connector.

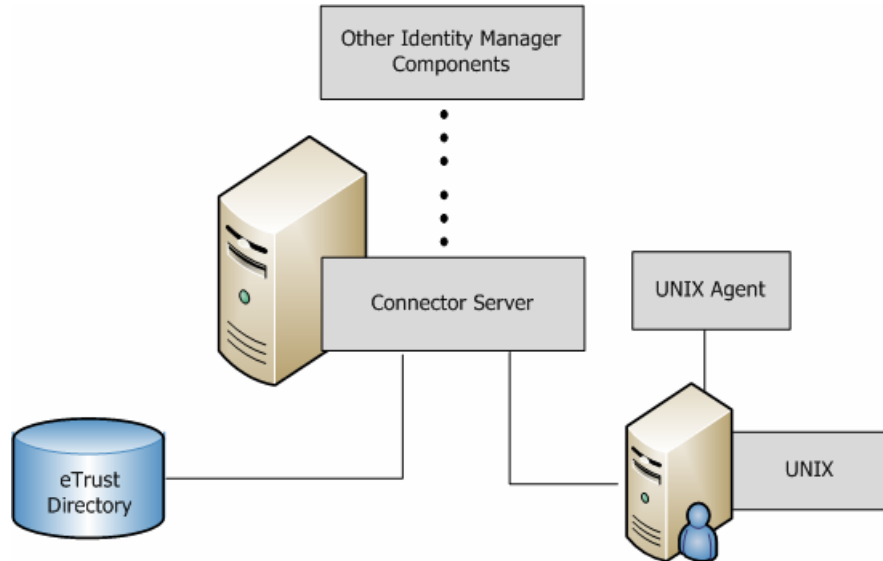
Connector and Agent Installation

Some installations require that only the connector and agent are installed. For those connectors, the following is required:

- You must install the connector on the Connector Server.
- You must install the agent on all endpoints that you want to administer.

Scenario for Connector and Agent Installation

The following diagram shows the connector and agent installation:



Connectors Installed Using the Connector and Agent Installation

You can install the following connectors using this scenario:

- UNIX
- Windows NT
- RSA ACE (SecurID)
- HP NonStop SAFEGUARD
- Microsoft Exchange 200x

More information:

[UNIX Connector](#) (see page 137)

[Windows NT Connector](#) (see page 67)

[The RSA ACE \(SecurID\) Connector](#) (see page 471)

[HP NonStop SAFEGUARD Connector](#) (see page 431)

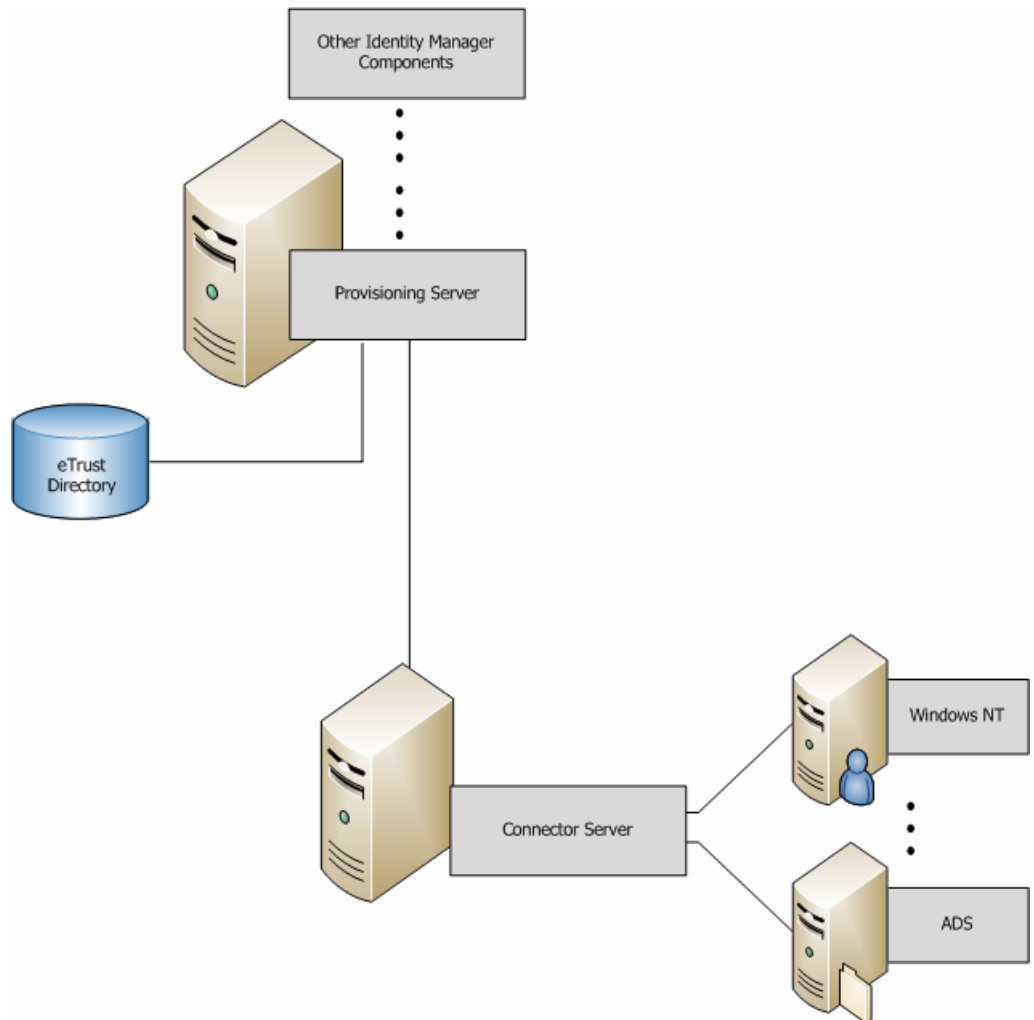
[Microsoft Exchange 200x Connector](#) (see page 265)

Connector and C++ Connector Server Installation

You can install several connectors along with a C++ Connector Server. The administrative requirements for these connectors are as follows:

- You must install the connector on the Provisioning Server computer.
- You must install the Connector Agent plug-in on the computer where you install the C++ Connector Server.
- You must install the Connector Provisioning Manager plug-in on the computer where you installed the Provisioning Manager.

The diagram below shows the connector and C++ Connector Server installation scenario:



Connectors Installed Using the Connector and C++ Connector Server Installation

You can install the following connectors using this scenario:

- [Active Directory](#) (see page 176)
- [CA Access Control](#) (see page 314)
- CA Web Access Control
- [CA SSO for Advanced Policy Server](#) (see page 337)
- [DB2 UDB](#) (see page 40)
- [DB2 for z/OS](#) (see page 34)
- [Embedded Entitlements Manager \(EEM\), formally EIAM](#) (see page 439)
- [HP NonStop SAFEGUARD Connector](#) (see page 431)
- [Ingres](#) (see page 48)
- [Microsoft Exchange 200x](#) (see page 265)
- [NDS](#) (see page 238)
- [Oracle Applications](#) (see page 59)
- [RSA ACE \(SecurID\)](#) (see page 471)
- [Siebel](#) (see page 461)
- [UNIX](#) (see page 137)
- [Universal Provisioning](#) (see page 243)
- [Windows NT](#) (see page 67)

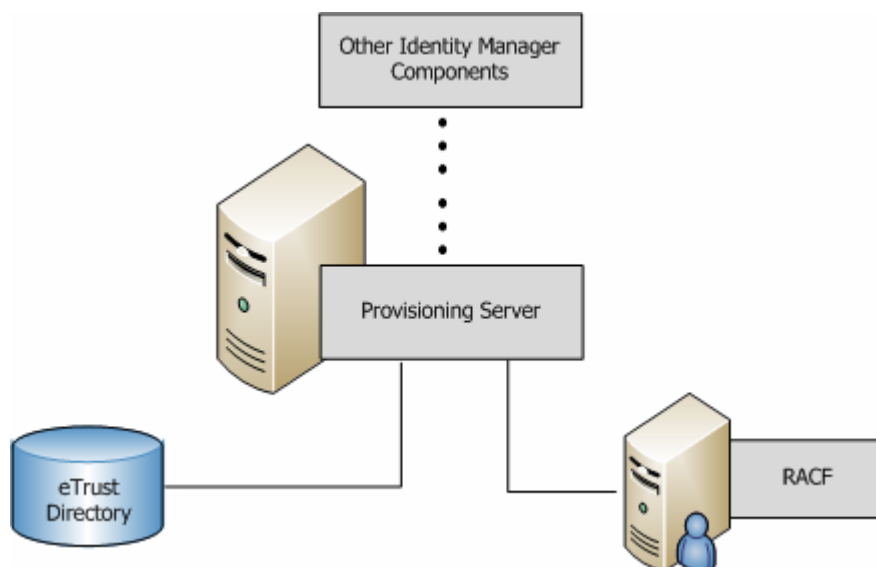
The following connectors require that you install their Administrative Client on the computer where you installed the C++ Connector Server:

- DB2
- [Ingres](#) (see page 48)
- [Oracle Applications](#) (see page 59)
- [NDS](#) (see page 238)

Connector Only Installation

Some connectors require only the installation of the connector on the Provisioning Server machine.

The diagram below shows the connector only installation scenario:



Connectors Installed Using the Connector Only Installation

You can install the following connectors using this scenario:

- CA-ACF2
- CA-Top Secret
- RACF
- OpenVMS

More Information:

[CA ACF2 Connector](#) (see page 83)

[CA Top Secret Connector](#) (see page 101)

[RACF Connector](#) (see page 126)

[OpenVMS Connector](#) (see page 418)

Java Connector Server Installation

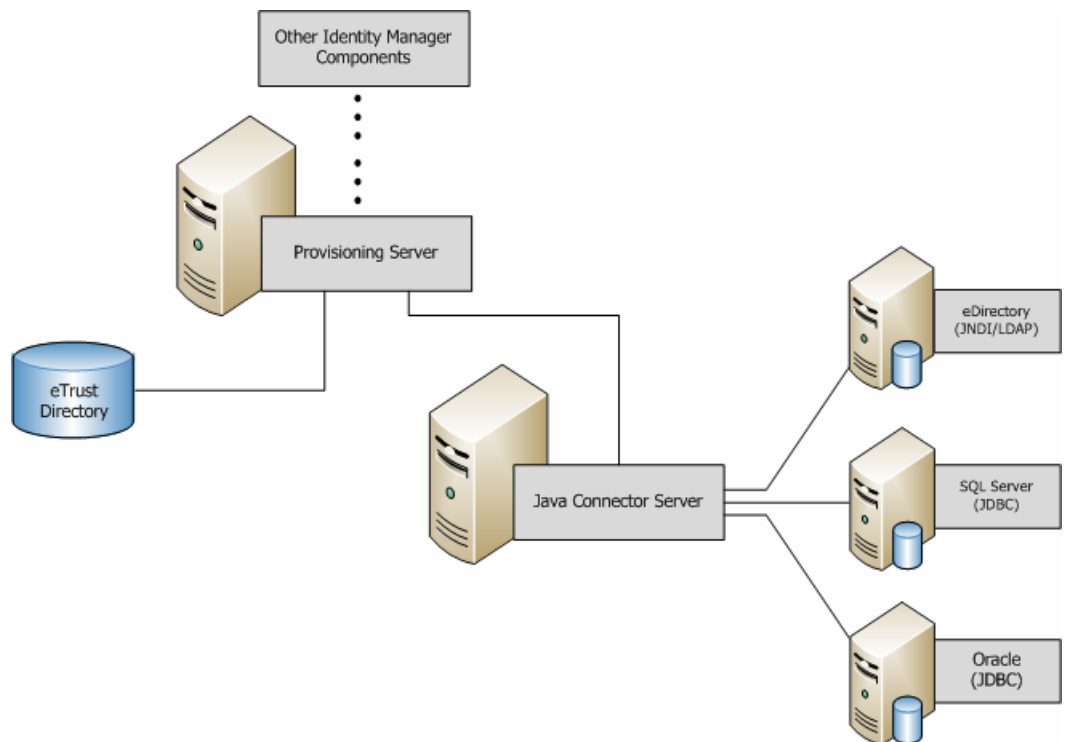
Some connectors can be installed on the Java Connector Server. The following are the administrative requirements for those connectors:

- You must install the connector on the Provisioning Server computer
- You must install and register the Java Connector Server on the Provisioning Server where you installed the connector.
- You must install the Connector Provisioning Manager plug-in on the computer where the Java Connector Server is installed.

Note: For more information on the Java Connector Server, see the *Java Connector Server Implementation Guide*.

The diagram below shows the Java Connector Server installation scenario:

Note: You cannot use both the Java Connector Server and the C++ Connector Server to manage the same endpoint type.



Connectors Using the Java Connector Server Installation

Connectors that can be installed using this scenario include the following:

- Oracle
- OS400
- MS SQL
- SAP
- Lotus Notes/Domino
- Kerberos
- Dynamic endpoints

More information:

[Oracle Connector](#) (see page 53)

[OS/400 Connector](#) (see page 393)

[MS SQL Server Connector](#) (see page 45)

[SAP Connector](#) (see page 446)

[Kerberos Connector](#) (see page 345)

[Lotus Notes/Domino Connector](#) (see page 280)

C++ to Java Connectors Migration

For this release, we are releasing several former C++ Connectors as Java Connectors. When migrating from the C++ version to the Java version, there are several migration considerations and steps to keep in mind.

Click the migration link under More Information to view the considerations and migration steps for your Connector(s).

More Information:

[MS SQL Migration Steps](#) (see page 46)

[Oracle Migration Steps](#) (see page 55)

[LND Java Implementation Considerations](#) (see page 281)

[LND Migration](#) (see page 284)

[OS/400 Migration Steps](#) (see page 394)

[SAP Migration Steps](#) (see page 450)

Chapter 3: Managing the Connectors

This section contains the following topics:

[The Provisioning Manager](#) (see page 29)

The Provisioning Manager

You can use the Provisioning Manager to perform all provisioning-related administrative tasks on your managed endpoints like viewing and manipulating objects and their relationships. The most common tasks that you will perform with the Provisioning Manager are discussed in this section.

Note: For detailed information about provisioning components, functions, and tools, see the *Provisioning Guide*.

Endpoints

An endpoint is a specific installation of a platform or application, such as Active Directory or Windows NT, that communicates with the Provisioning Server to synchronize information and manage users, logins, and other resource objects.

The following topics are an overview of the acquire, explore, and correlate features.

Note: For detailed information regarding your connector, see the individual connector topic sections in this book.

Acquire Endpoints

In order to manager the accounts, users, groups, and other resource objects in your environment, you must acquire endpoints. Each connector has its own requirements for acquiring endpoints. See the individual connector sections for connector-specific instructions.

Explore and Correlate Endpoints

After you have acquired an endpoint, you must explore the endpoint so that the provisioning manager can manage your objects, and correlate the objects to global users.

Accounts

Accounts give users access to the resources on an endpoint. You can manage accounts from the Endpoint task frame using the Account property sheet for your connector.

Provisioning Roles and Account Templates

Provisioning roles define the job functions in your organization using one or more account templates. You can use account templates to control the generation or removal of accounts, initialization of new accounts, and global synchronization of certain account characteristics. Therefore, provisioning roles and account templates let you manage endpoints without having to keep track of the individual user accounts.

Groups

You can create and maintain endpoint groups using the Endpoint tasks frame. Use the Group property sheet for your connector when managing your groups.

DYN GUI

The DYN GUI is a data driven GUI in the Provisioning Manager. It uses Connector Xpress generated metadata describing a JDBC or JNDI connector to display the following for accounts, account templates, group, containers, and endpoints:

- Fields
- Labels
- Selection controls (for example, a date picker for date fields)

It also:

- Performs validation of user supplied values
- Populates fields with default values
- Splits fields across tabs

The DYN GUI in the Provisioning Manager allows you to manipulate arbitrary endpoint objects with a single Provisioning Manager plug-in.

The DYN GUI leverages the general controls of that application with dynamic property sheets defined at run-time by a metadata XML stream, known as the model metadata.

For example, when you map a field in Connector Xpress, an item is placed in the metadata to represents that field. Whenever you examine any object in that connector, the DYN GUI uses the metadata to display the appropriate fields.

Account Template Creation with etautil

If you use etautil to create a JDBC or JNDI account template, the account template will not work unless the correct value is explicitly provided for the eTAccountContainer attribute.

For JDBC, the value is always:

```
eTDYNAccountContainerName=Accounts
```

For JNDI, the value varies depending on which container you use to store accounts. For example, the following setting would result in new accounts being placed in container cont1 that is nested inside the top-level container, cont:

```
eTDYNContainerName=cont1,eTDYNContainerName=cont0
```


Chapter 4: Database Connectors

This section contains the following topics:

[DB2 UDB for z/OS Connector](#) (see page 34)

[IBM DB2 UDB Connector](#) (see page 40)

[MS SQL Server Connector](#) (see page 45)

[Ingres Connector](#) (see page 48)

[Oracle Connector](#) (see page 53)

[Oracle Applications Connector](#) (see page 58)

The following connectors are considered to be database connectors:

- DB2 UDB for z/OS
- IBM DB2 UDB
- MS SQL
- Ingres Server
- Oracle
- Oracle Applications

The information in this chapter describes how to install and manage these connectors.

DB2 UDB for z/OS Connector

The CA Identity Manager DB2 UDB for z/OS Connector (DBZ) lets you manage user authorization and privileges of a DB2 UDB on z/OS instance and database on a z/OS mainframe.

Using this connector, you can do the following:

- Create, modify, or delete DBZ Endpoint Types, endpoints, users, and account templates in Identity Manager
- Create, modify, and remove users in the DBZ database on z/OS
- Manage user identifiers, authorizations, and privileges that exist in the DBZ authorization and privileges tables.

This connector is managed using the Connector and C++ Server installation process.

Note: For more information and requirements, see *Connector and C++ Connector Server Installation* (see page 23).

You should review the information in the *Provisioning Guide* and install the Provisioning Server before you proceed with the procedures in this document.

DBZ Installation Prerequisites

You must install the IBM DBZ Connect Enterprise Edition 8.2 for Win on the machine where you installed the C++ Connector Server.

DB2 Limitation

You cannot map stored functions.

DB2 z/OS Support for FIPS and IPv6

For release r12 of Identity Manager, the DBZ Connector does not support FIPs or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

DBZ Endpoint

The DBZ endpoint registers a Windows System ODBC Data Source Name (DSN) for the database and saves the necessary information to establish a connection and execute SQL statements with the database.

Acquire a DBZ Database

To acquire a DBZ database, you must do the following:

From the Endpoint Type task view

1. Register the database as an endpoint in Identity Manager.

Use the DBZ Endpoint property sheet to register a DB2 z/OS database. During the registration process, Identity Manager identifies the DBZ database you want to administer and gathers information about it.

2. Explore the objects that exist on the endpoint.

After registering the database in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all DBZ database authorization names that exist in the database authorization tables. You can correlate the authorization names of the User type (DBZ Users) with global users at this time, or you can wait to correlate them.

3. Correlate the explored DBZ users with global users.

When you correlate DBZ users, Identity Manager creates or links the DBZ users to an endpoint with global users, as follows:

- a. Identity Manager attempts to match the DBZ user name with each existing global user name. If a match is found, Identity Manager associates the DBZ user name with the global user. If a match is not found, Identity Manager performs the following step.
- b. If the Create Global Users as Needed button is checked, Identity Manager creates a new global user and then associates the DBZ account with the global user. If the Create Global Users as Needed button is unchecked, Identity Manager performs the next step.
- c. Identity Manager associates the DBZ user with the [default user] object.

Acquire or Remove a New Endpoint

When the DBZ connector receives an 'Add new endpoint' or 'Remove an endpoint' request, the following steps are taken:

On the machine running the C++ Connector Server

1. Catalog or un-catalog a database entry for a database within the DBZ instance.
2. Register or un-register an ODBC system data source.

DBZ Account Templates

The DBZ Default Policy, provided with your connector, gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

Synchronize an Account from an Account Template

There are several rules for account synchronization from an account template in the DBZ Connector.

During the account synchronization process

1. When there are multiple account templates associated with a DBZ account, the DBZ Connector merges those account templates to generate an intermediate effective account template. During the merge, if there are conflicting settings with the same authority, database privilege, or object privilege among the different account templates, the DBZ Connector selects the setting with the highest restriction.

For example, if Account Template One grants DBADM and Account Template Two does not, the effective account template does not grant DBADM. Another example: If Account Template One grants CONTROL and SELECT with GRANT option on view SYSCAT.ATTRIBUTES, but Account Template Two revokes CONTROL from and grants SELECT on view SYSCAT.ATTRIBUTES, the effective account template grants only SELECT on view SYSCAT.ATTRIBUTES and revokes CONTROL from SYSCAT.ATTRIBUTES.

2. If one of the merged account templates is set to use strong synchronization, the DBZ Connector applies the effective account template to the account using strong synchronization. If not, the effective account template uses weak synchronization.
3. For strong synchronization, the DBZ Connector replaces the account's authorities and privilege settings with that of the effective account template.
4. For weak synchronization, if there is a difference between the account settings and the effective account template, the DBZ Connector uses the setting that has the higher restriction.

For example, if an account is granted DBADM, and the effective account template does not grant DBADM, the account will not be granted DBADM. If an account is not granted DBADM and the effective account template grants DBADM, the account will still not be granted DBADM.

Another example: If an account is granted CONTROL and SELECT with GRANT option on view SYSCAT.ATTRIBUTES, but the effective account template revokes CONTROL from and grants SELECT on view SYSCAT.ATTRIBUTES, the account is granted only SELECT on view SYSCAT.ATTRIBUTES and CONTROL is revoked from SYSCAT.ATTRIBUTES.

When checking account or account template synchronization, the same process of generating effective account template applies, as do the rules of comparison. If you are going to synchronize account settings with the effective account template, and the account's authority and privilege settings do not change, the DBZ Connector considers the account synchronized with its associated account templates.

DBZ Accounts

The DBZ Account represents the authentication and privileges of the DBZ users of the DBZ instance and database on a z/OS mainframe.

The DBZ Connector does not manage user accounts and groups of the operating system. The DB2 Users that are managed by the DB2 z/OS Connector are the user identifiers, authorizations, and privileges that exist in the DB2 authorization and privileges tables.

Create DBZ Accounts

Identity Manager lets you manage accounts from the Endpoint Type task view. Use the DBZ User property sheet when managing your accounts

To create DBZ Accounts

1. Click the Endpoint Type task button and select DBZ Endpoint from the drop-down list box.
2. Search for the endpoint on which you want to create an account.
3. Right-click on the endpoint in the list view and choose Content from the pop-up menu.
4. Select Accounts in the Container Tree box and click New.

The DBZ User Property Sheet appears.

5. Complete the DBZ User Property Sheet and click OK.

A new DBZ account is now created.

DBZ User Property Sheet

The DBZ User Property Sheet consists of 16 property pages with the following 14 pages specific to the DBZ Connector that show specific authorization and property information:

- Database
- Subsystem
- Table
- View
- Buffer Pool
- Storage Group
- Collection
- Package
- Plan
- Table Space
- Procedure
- User Defined Function
- Schema
- User Defined Type

IBM DB2 UDB Connector

Along with the Identity Manager Connector for the underlying operating system, the DB2 UDB Connector lets you administer accounts and groups on DB2 UDB databases and provides a single point for all user administration by letting you:

- Register DB2 UDB endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage DB2 UDB database authorization names (users and groups) using DB2 UDB-specific account templates
- Synchronize global users with their provisioning roles or synchronize global users' accounts with their account templates
- Assign a DB2 UDB account template to each of your DB2 UDB endpoints
- Use the default endpoint type account template to create DB2 UDB users with the minimum security level needed to access a DB2 UDB endpoint
- Create and manage DB2 UDB groups (Windows only)

DB2 UDB Installation

This connector is managed using the Connector and C++ Server installation process.

Note: For more information and requirements, see *Connector and C++ Connector Server Installation (see page 23)*.

Installation Requirements for Windows

The following connector and agent are necessary to administer the DB2 Universal Database:

- **DB2 UDB Connector** must be installed.
- To administer DB2 UDB authentication, an appropriate Identity Manager Connector for the underlying operating system of DB2 UDB Server installation must be installed on the Provisioning Server. Such options include, but are not limited to the NT Connector, ETC Connector, NIS Connector and the ADS Connector.
- **DB2 UDB** Administration Client must already be installed where the DB2 UDB Connector will be installed.
- **TCP/IP** must be one of the supported communication protocols of the DB2 UDB installation when DB2 UDB server is at a remote location.
- **TCP/IP Communication** must be set up for the DB2 UDB Instance on DB2 UDB Server using Control Center and have either a TCP/IP Service Name or Port Number assigned (default to 50000) when the DB2 UDB server is at a remote location.
- **Database Manager Instance** should be started on the DB2 UDB Server.

Note: The DB2 UDB Connector supports any DB2 UDB server installations that the DB2 UDB Administrative Client for Window supports, but tests have been done only with DB2 UDB server installations on Windows 2000 and AIX.

DB2 UDB Support for FIPS and IPv6

For release r12 of Identity Manager, the DB2 UDB Connector supports IPv6, but not FIPS.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire a DB2 UDB Database

You must acquire the DB2 UDB database before you can administer it with Identity Manager.

From the Endpoint type task view

1. Register the database as an endpoint in Identity Manager.

Use the DB2 UDB Endpoint property sheet to register a DB2 UDB database. During the registration process, Identity Manager identifies the DB2 UDB database you want to administer and gathers information about it.

2. Explore the objects that exist on the endpoint.

After registering the database in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all DB2 UDB database authorization names that exist in the database authorization tables. You can correlate the authorization names of the User type (DB2 UDB Users) with global users at this time, or you can wait to correlate them.

3. Correlate the explored DB2 UDB users with global users.

When you correlate DB2 UDB users, Identity Manager creates or links the DB2 UDB users to an endpoint with global users, as follows:

- a. Identity Manager attempts to match the DB2 UDB user name with each existing global user name. If a match is found, Identity Manager associates the DB2 UDB user name with the global user. If a match is not found, Identity Manager performs the following step.
- b. If the Create Global Users as Needed button is checked, Identity Manager creates a new global user and then associates the eTrust DB2 UDB account with the global user. If the Create Global Users as Needed button is unchecked, Identity Manager performs the next step.
- c. Identity Manager associates the DB2 UDB user with the [default user] object.

DB2 Provisioning Roles and Account Templates

By defining account templates for the underlying operating system to a provisioning role, you can manage the operating system accounts and groups while managing the authorization name of the DB2 UDB database. Therefore, provisioning roles and account templates let you manage all the aspects of the DB2 UDB database security.

The DB2 UDB Default Policy, provided with the DB2 UDB Connector, gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

Create Account Templates

The Default Account Template, provided with each connector, gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

To create an account template

1. Click the Provisioning Roles task button, select the connector's Account Template in the Object Type drop-down list box and click New.

The Account Template Property Sheet for the specified connector appears.

2. Complete the Account Template Property Sheet by:
 - a. Selecting an endpoint to populate the drop-down and group selection lists.
 - b. Selecting group memberships and other account settings.
 - c. Clicking OK.

A new account template is created for your connector.

In Identity Manager DB2 UDB Users give users access to the resources on an endpoint. Identity Manager lets you manage all DB2 UDB database authorization names of the type User from the Endpoint type task view. Use the DB2 UDB User property sheet when managing your users.

DB2 UDB Groups

Identity Manager lets you create and maintain DB2 UDB authorization names of the type Group using the Endpoint type task view. Use the DB2 UDB Group property sheet when managing your groups.

Add New Endpoint Request

When the DB2 Connector receives an 'Add new endpoint' request, it:

1. Catalogs a new DB2 Local or TCP/IP node for the instance.
2. Catalogs a new DB2 Database entry for the database.
3. Configures an ODBC system data source for the database.

How to Synchronize an Account from an Account Template

These are the rules for account synchronization from an account template in the DB2 Connector.

1. During the account synchronization process, when there are multiple account templates associated with a DB2 account, the DB2 connector merges those account templates to generate an intermediate effective account template. During the merge, if there are conflicting settings with the same authority, database privilege, or object privilege among the different account templates, the DB2 Connector selects the setting with the highest restriction.

For example, if Account Template One grants DBADM and Account Template Two does not, the effective account template does not grant DBADM. Another example: If Account Template One grants CONTROL and SELECT with GRANT option on view SYSCAT.ATTRIBUTES, but Account Template Two revokes CONTROL from and grants SELECT on view SYSCAT.ATTRIBUTES, the effective account template grants only SELECT on view SYSCAT.ATTRIBUTES and revokes CONTROL from SYSCAT.ATTRIBUTES.

2. If one of the merged account templates is set to use strong synchronization, the DB2 Connector applies the effective account template to the account using strong synchronization. If not, the effective account template uses weak synchronization.
3. For strong synchronization, the DB2 Connector replaces the account's authorities and privilege settings with that of the effective account template.
4. For weak synchronization, if there is a difference between the account settings and the effective account template, the DB2 Connector uses the setting that has the higher restriction.

For example, if an account is granted DBADM, and the effective account template does not grant DBADM, the account will not be granted DBADM. If an account is not granted DBADM and the effective account template grants DBADM, the account will still not be granted DBADM.

Another example: If an account is granted CONTROL and SELECT with GRANT option on view SYSCAT.ATTRIBUTES, but the effective account template revokes CONTROL from and grants SELECT on view SYSCAT.ATTRIBUTES, the account is granted only SELECT on view SYSCAT.ATTRIBUTES and CONTROL is revoked from SYSCAT.ATTRIBUTES.

When checking account or account template synchronization, the same process of generating effective account template applies, as do the rules of comparison. If you are going to synchronize account settings with the effective account template, and the account's authority and privilege settings do not change, the DB2 Connector considers the account synchronized with its associated account templates.

MS SQL Server Connector

The MS SQL Server Connector provides a single point for all user administration by letting you perform the following:

- Manage logins on MS SQL server platforms
- Register endpoints, explore them for objects to manage, and correlate their logins with global users
- Create and manage MS SQL server logins using MS SQL server-specific account templates
- Change login passwords and activations in one place
- Synchronize users with their provisioning roles or synchronize users' logins with their account templates
- Assign a MS SQL server account template to each of your MS SQL server endpoints
- Use the SQL Default Policy to create logins with the minimum level of security needed to access an MS SQL server endpoint
- Generate and print reports about MS SQL server logins and hosts

MS SQL Configuration

The MS SQL connector must be managed with the Java Connector Server installation process. For more information on this installation process, click [here](#) (see page 26).

To administer MS SQL server machines with Identity Manager, the MS SQL server connector must be installed on each Provisioning server.

The following sections detail the configurations that are needed in order for the MS SQL connector to work correctly.

Configure the JDBC URL

Communication between the Provisioning server and the MS SQL server relies on a JDBC connection. A URL specifies connection details to each server, as illustrated in the following examples:

Basic URL

```
jdbc:sqlserver://serverHost
```

Integrated Security URL

```
jdbc:sqlserver://serverHost;integratedSecurity=true
```

Named instance on port 1433 URL

```
jdbc:sqlserver://serverHost:1433;instanceName=instance1
```

Connecting with IPv6

```
jdbc:sqlserver://;serverName=<IPv6 address here>
```

```
jdbc:sqlserver://;serverName=<IPv6 address>;port=CA  
Portal;databaseName=<DB>
```

Note: For more details see [Building the Connection URL](#) on MSDN.

Configure the Windows Service

If you want to use Windows NT authentication, you must ensure that the system account running the JCS service also exists as an account on the server running MS SQL and has administrative rights. The JCS service by default runs as the LocalSystem account. You will need to change this to an account on the same domain or system as the MS SQL servers you wish to manage using the 'Services' dialog in the 'Control Panel'.

Note: Windows NT trusted authentication is only supported on Windows platforms.

MS SQL Migration Steps

To migrate from the C++ MS SQL connector to the Java MS SQL connector, you must do the following:

- Install the MS SQL Java connector using the JCS installation
- You can remove your DSN if it is not being used for another other purpose
- Add the URL as defined in [Configure the JDBC URL](#) (see page 45) to each existing MS SQL endpoint. To do this, right click on each endpoint and select "Custom... -> Change Admin Password" and supply the URL in the JDBC URL field.

Once this has been done, all types of operations can be executed against the existing MS SQL endpoints seamlessly.

MS SQL Support for FIPS and IPv6

For release r12 of Identity Manager, the MS SQL Connector supports IPv6, but not FIPS.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire an MS SQL Server

You must acquire each MS SQL server before you can administer it with Identity Manager. To acquire an MS SQL server, you must do the following:

From the Endpoint Type task view

1. Register the machine as an endpoint using the Provisioning Manager.

Use the MS SQL server Endpoint property sheet to register an MS SQL server instance. During the registration process, the Provisioning Server identifies the MS SQL server that you want to administer, and gathers information about it.

2. Explore the objects that exist in the endpoint.

After registering the machine with the Provisioning Manager, you can explore its contents, using the Explore and Correlate Endpoint dialog. The Exploration process finds all MS SQL server objects. You can correlate the logins with global users at this time, or later.

3. Correlate the explored logins with global users.

When you correlate logins, the Provisioning Server creates or links the logins on an endpoint with global users, as follows:

- a. The Provisioning Server attempts to match the login name with each existing global user name. If a match is found, the Provisioning Server associates each MS SQL server login with the respective global user. If a match is not found, the Provisioning Server performs the next step.
- b. If the Create Global Users as Needed button is checked, the Provisioning Server creates a new global user and then associates the login with the global user. If the Create Global Users as Needed is unchecked, the Provisioning Server performs the next step.
- c. The Provisioning Server associates the login with the *default user* object.

The MS SQL Connector uses Logins in place of accounts. Use the MS SQL Server Login property page to manage MS SQL Logins.

Database Users

The database users have administrative power in the system.

Database Roles

You can list the database roles, and include (or exclude) users from the database roles.

For more information, see the *Provisioning Guide*.

MS SQL Conventions

Use the following MS SQL Server conventions in your etacutil commands:

- The endpoint type name (eTNamespaceName) is MS SQL Server
- The endpoint type prefix is SQL. Therefore, the MS SQL Server class names are the following:
 - eSQLDirectory for an endpoint
 - eSQLPolicyContainer for an account template container
 - eSQLPolicy for an account template

More information:

[Distinguished Names](#) (see page 510)

For more information about the etacutil command, see the *Reference* help and *Provisioning Guide*.

Ingres Connector

The Ingres[®] Connector lets you administer the users of Ingres Server machines.

The Ingres Connector provides a single point for all user administration by letting you do the following:

- Display, create, modify, or delete a user
- Alter expiration date for a user
- Add or alter the profile name for a user
- Add or alter the default group name for a user
- Add or alter the set of groups assigned to a user
- Grant or revoke user roles

Administer the Ingres Server

Administering the Ingres Server requires you to:

1. Install the Ingres Client and the Ingres Connector.
2. Configure the Virtual Node (vnode).
3. Configure ODBC.

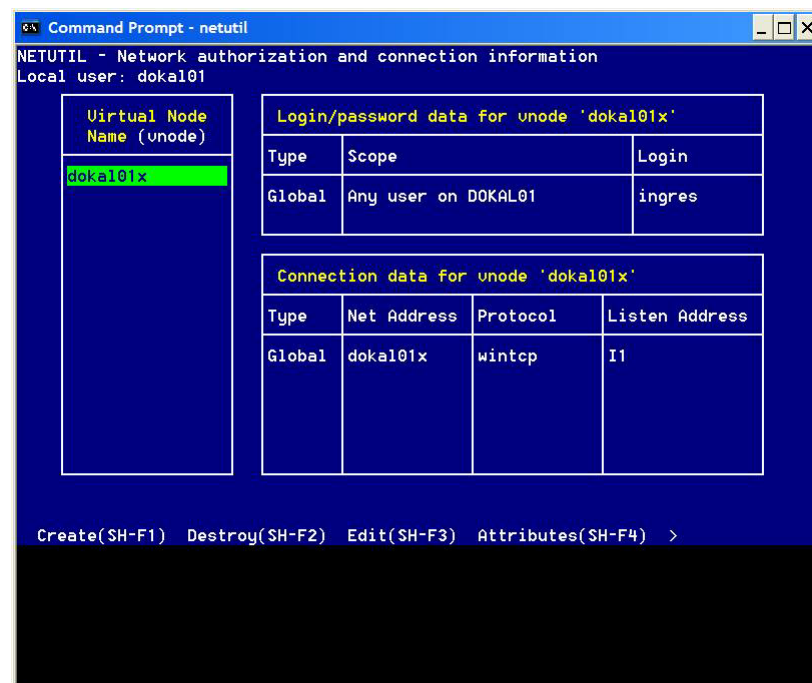
Install the Ingres Client and Ingres Connector

This connector is managed using the Connector and C++ Server installation process.

Note: For more information and requirements, see *Connector and C++ Connector Server Installation* (see page 23).

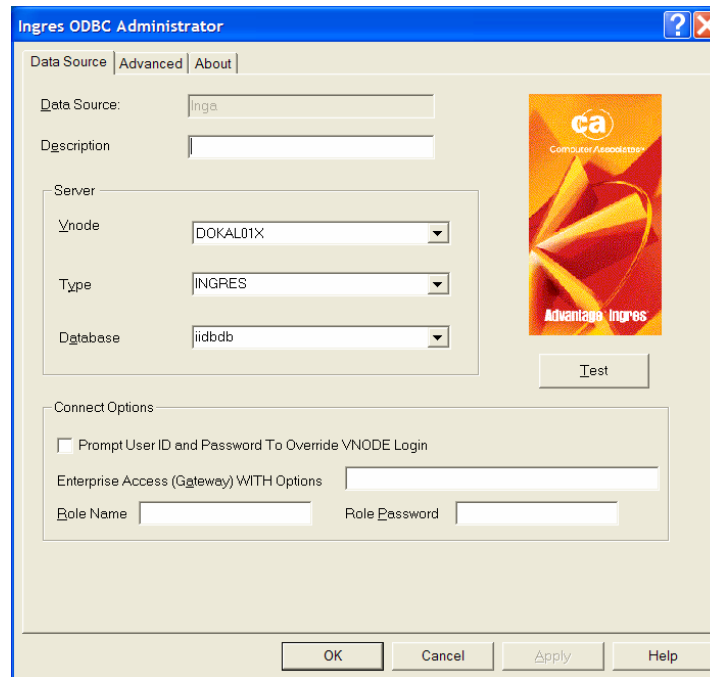
Configure the Virtual Node

Use the Netutil command to configure the Virtual Node (vnode) to get access to the Ingres server machine.



Configure ODBC

ODBC should be configured to provide system DSN pointing to the target system. The iidbdb database should be used as the database to connect to.



Note: To administer Ingres Server authentication, an appropriate Identity Manager Connector for the underlying operating system of the Ingres Server installation must be installed on the Provisioning Server machine. See the Connector FAQ section of the Provisioning Manager for more information.

Ingres Database Limitations

The following are Ingres database limitations:

- You cannot map the Bigint type.
- You cannot use the result of stored procedures.

Ingres Support for FIPS and IPv6

For release r12 of Identity Manager, the Ingres Connector does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire an Ingres Database

You must acquire the Ingres database before you can administer it with Identity Manager. To acquire an Ingres database, you must do the following:

From the Endpoint Type task view

1. Register the database as an endpoint in Identity Manager.

Use the Ingres Endpoint property sheet to register an Ingres database. During the registration process, Identity Manager checks that the Ingres database you want to administer is accessible.

2. Explore the objects that exist on the database.

After registering the database in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all necessary Ingres objects. You can correlate the accounts with global users at this time, or you can wait to correlate them.

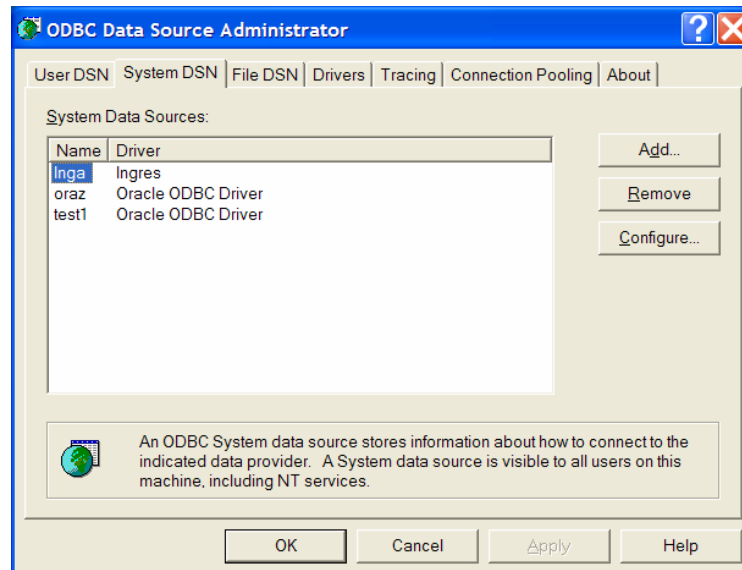
3. Correlate the explored accounts with global users.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the username with each existing global user name. If a match is found, Identity Manager associates the Ingres account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the account name with each existing global user's full name. If a match is found, Identity Manager associates the Ingres account with the global user. If a match is not found, Identity Manager performs the next step.
- c. If the Create Global Users as Needed button is checked, Identity Manager creates a new global user and associates the Ingres account with the global user. If the Create Global Users as Needed button is unchecked, then Identity Manager performs the next step.
- d. Identity Manager associates the Ingres account with the [default user] object.

Ingres Directory Tab Authentication Attributes

On the Endpoint Tab of the Ingres Endpoint Property Sheet you can specify the authentication attributes such as the Data Source Name (DSN). This name should coincide with one of the names in the ODBC Data Source Administrator on the System DSN page on the machine where Provisioning Server is installed. This DSN should be properly configured to point to the desirable target system.



Ingres Accounts

If, when creating an Ingres account, you set a default group, you must add that group to the set of groups that are assigned to this account. Otherwise, the account creation fails.

Ingres Profiles

The Profiles container holds all Ingres profiles that are available on the target system at the moment of the last explore.

This container provides read-only access. To add a new object to this container, use native tools to add the object to the target system, and then re-explore the container. Likewise, to delete an object from this container, use native tools to delete the object from the target system, and then re-explore the container.

Ingres Groups

The Groups container holds all Ingres groups that are available on the target system at the moment of the last explore.

This container provides read-only access. To add a new object to this container, use native tools to add the object to the target system, and then re-explore the container. Likewise, to delete an object from this container, use native tools to delete the object from the target system, and then re-explore the container.

Ingres Roles

The Roles container holds all Ingres roles that are available on the target system at the moment of the last explore.

This container provides read-only access. To add a new object to this container, use native tools to add the object to the target system, and then re-explore the container. Likewise, to delete an object from this container, use native tools to delete the object from the target system, and then re-explore the container.

Ingres Etautil Conventions

Use the following Ingres conventions in your etautil commands:

- The endpoint type name (eTNamespaceName) is Ingres
- The endpoint type prefix is ING. Therefore, the Ingres class names are:
 - eTINGDirectoryName for an endpoint
 - eTINGPolicyContainerName for an account template container
 - eTINGPolicy for an account template

Oracle Connector

The Oracle Connector lets you administer accounts and groups on Oracle systems and provides a single point for all user administration by letting you do the following:

- Register endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage Oracle accounts using Oracle-specific account templates
- Change account passwords and account activations in one place
- Synchronize global users with their provisioning roles or synchronize global users' accounts with their account templates

- Assign an Oracle account template to each of your Oracle endpoints
- Use the default Oracle Policy to create accounts with the minimum security level needed to access an Oracle endpoint
- Create and manage Oracle profiles and roles
- Generate and print reports about Oracle accounts
- Assign Oracle packages and procedures to Oracle accounts

Oracle Configuration

The Oracle connector must be managed with the Java Connector Server installation process. For more information on this installation process, click [here](#) (see page 26).

To administer Oracle machines with Identity Manager, the MS SQL server connector must be installed on each Provisioning server.

The Oracle GUI plug-in must be installed on the machine where the Provisioning Manager is installed. No additional Oracle agent, client, or driver installation is required.

Communication between the Provisioning Server and the Oracle server relies on a JDBC connection. A URL specifies connection details to each server, as illustrated in the following example:

```
jdbc:oracle:thin:@hostname:port:servicename
```

Where the field definitions are:

hostname

The hostname or IP address of the Oracle Server

port

The numeric port number of the Oracle service. The default value for this is 1521.

servicename

Oracle Service Name to connect to.

For example, the following URL would connect to an Oracle instance named ORACLE running on the default port on server oracle_server_host:

```
jdbc:oracle:thin:@oracle_server_host:1521:ORACLE
```

For more information, see the Oracle JDBC site:

http://www.oracle.com/technology/tech/java/sqlj_jdbc/htdocs/jdbc_faq.htm

Oracle Migration Steps

To migrate from the C++ Oracle connector to the Java Oracle connector, you must do the following:

- Install the Oracle Java connector using the JCS installation
- Add the URL as defined in [Oracle Configuration](#) (see page 54) to each existing Oracle endpoint. To do this, edit the endpoint and supply the URL in the JDBC URL field.
- You can remove your DSN if it is not being used for another other purpose

Once this has been done, all types of operations can be executed against the existing Oracle endpoints seamlessly.

Oracle Support for FIPS and IPv6

For release r12 of Identity Manager, the Oracle Connector does not support FIPs or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Oracle Limitations

The Oracle connector cannot be used to perform the following operation. The Oracle database administrator must perform this operation using native Oracle administrative tools.

- Manage system privileges or object privileges that apply to Oracle accounts or Oracle roles

Required Oracle Administrator Account Privileges

The Oracle administrator account that you use with Identity Manager is the account name that you enter in the System Logon field of the Endpoint tab of the Oracle Endpoint property sheet. This account requires the following minimum privileges to take full advantage of the Oracle Connector:

- The following system privileges:
 - Alter Profile
 - Alter Any Role
 - Alter User
 - Create Profile
 - Create Role
 - Create Session
 - Create User
 - Drop Profile
 - Drop User
 - Drop Any Role
 - Grant Any Privilege
 - Grant Any Role
- The SELECT object privilege on the following views in the SYS schema:
 - DBA_OBJECTS
 - DBA_PROFILES
 - DBA_ROLES
 - DBA_ROLE_PRIVS
 - DBA_TABLESPACES
 - DBA_TAB_PRIVS
 - DBA_TS_QUOTAS
 - DBA_USERS

To grant privileges to Oracle accounts for packages and procedures, ensure that the account name specified in the Logon field of the Oracle Endpoint property sheet is either:

- The owner of these packages and procedures
- Has execute privileges with "Admin Option" for these packages and procedures

Acquire an Oracle System

You must acquire the Oracle system before you can administer it with Identity Manager. To acquire an Oracle system, you must do the following:

From the Endpoint Type task view

1. Register the Oracle server as an endpoint using the Provisioning Manager.

Use the Oracle Endpoint property sheet to register an Oracle system. During the registration process, the Provisioning Server identifies the Oracle system you want to administer and gathers information about it.

Note: Use the native Oracle tools to verify that the Oracle system can be accessed using the given system login ID and password.

2. Explore the objects that exist on the endpoint.

After registering the server with the Provisioning Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all Oracle accounts, profiles, and roles. You can correlate the accounts with global users at this time or you can correlate them later.

3. Correlate the explored accounts with global users.

When you correlate accounts, the Provisioning Server creates or links the accounts on a endpoint with global users, as follows:

- a. The Provisioning Server attempts to match the Oracle name with each existing global user name. If a match is found, the Provisioning Server associates the Oracle account with the global user. If a match is not found, the Provisioning Server performs the next step.
- b. If the Create Global Users as Needed button is checked, the Provisioning Server creates a new global user and then associates the Oracle account with the global user. If the Create Global Users as Needed button is unchecked, the Provisioning Server performs the next step.
- c. The Provisioning Server associates the Oracle account with the [default user] object.

Oracle Account Templates

The Oracle Default Policy automatically sets the user name and password to the global user account ID and the authentication type to LOCAL.

Oracle Accounts

Use the Oracle Account property sheet when managing your accounts.

Oracle Packages

The Oracle Packages container holds all stored Oracle packages on your system.

Oracle Procedures

The Oracle Procedures container holds all stored Oracle procedures on your system.

Oracle Roles

The Provisioning Manager lets you create and maintain Oracle roles using the Endpoint Type task view. Use the Oracle Role property sheet when managing your Oracle roles.

Oracle Profiles

An Oracle profile lets you specify the values of Oracle account attributes such as connect time, CPU time, and password rules.

Oracle Etatil Conventions

Use the following Oracle conventions in your etatil commands:

- The endpoint type name (eTNamespaceName) is Oracle Server
- The endpoint type prefix is ORA. Therefore, the Oracle class names are:
 - eTORADirectory for an endpoint
 - eTORAPolicyContainerName for an account template container
 - eTORAPolicy for an account template

Oracle Applications Connector

The Oracle Applications Connector lets you administer users of Oracle E-Business Suite applications and provides a single point for all user administration by letting you do the following:

- Register endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage Oracle Applications users by using Oracle-specific account templates
- Manually manage an Oracle Applications user responsibility list or automatically manage a group of users based on provisioning roles and account templates
- Change account passwords and account activations in one place
- Synchronize global users with their provisioning roles or synchronize global users' accounts with their account templates
- Assign an Oracle Applications account template to each of your Oracle Applications directories
- Generate and print reports about Oracle Applications users

Oracle Applications Installation and Configurations

This connector is managed using the Connector and C++ Server installation process.

Note: For more information and requirements, see *Connector and C++ Connector Server Installation* (see page 23).

The following sections provide installation and configuration information for this connector.

How to Configure the Oracle Applications Connector

Before installing the Oracle Application Connector, install the Oracle Client on the same machine that the Oracle Application Connector will be installed on.

After installing your Oracle Administrative Client from the Oracle Client CD, do the following to configure it:

1. Create a service for your Oracle client.
2. Configure ODBC on your Oracle client.

Creating a Service for Your Oracle Client

Create a service for your Oracle client using the Oracle Net Configuration Assistant for Oracle Client Release 9i or 10g.

From the Oracle Configuration and Migration Tools program group

1. Start Oracle Net Configuration Assistant.
The Oracle Net Configuration Assistant wizard appears.
2. Select Local Net Service Name Configuration.
3. Select Add New Service.
4. Enter the Service Name.
5. Select TCP/IP (Internet Protocol).
6. Enter the host name for the computer where the database is located.
7. Change the port number to match your Oracle server port number.
 - For Windows systems, the default port number on Oracle systems is 1521.
 - For UNIX systems, the default port number on Oracle systems is 1526.
8. Select Yes to perform a connection test.
9. Enter a name for the net service name.
10. Click Finish to save the information.

You can view configured services by scanning the list of names on the Service Naming node of the Oracle Net Manager.

Configure ODBC on Your Oracle Client

To configure ODBC on your Oracle client, use this procedure.

From the Control Panel

1. Select ODBC Manager/Data Sources, DSN tab, Add.
The Create New Data Source wizard appears.
2. Select the Oracle ODBC Driver, and click Finish.
The Oracle ODBC Driver Setup dialog appears.
3. Enter the data source name for the Oracle server in the Data Source Name text box.
4. Enter the service name that you created in Creating A Service For Your Oracle client
5. Enter the Oracle administrator's ID in the UserID text box.
6. Click OK.

After configuring the Oracle client, you are ready to install the Oracle Applications Connector. See the *Provisioning Guide* for installation instructions.

Oracle Applications Prerequisite

You must set the NLS_LANG variable to NLS_LANG=UTF8 on the machine where the Provisioning Server is installed if you want to manage Oracle Applications as an endpoint.

Oracle Applications Limitations

The Oracle Applications Connector cannot be used to manage responsibilities that can be assigned to Oracle Applications users. The Oracle Applications System Administrator must perform these operations using native Oracle Applications administrative tools (JInitiator).

Required Oracle Administrator Account Privileges

The Oracle Applications Connector needs the database account of the Database Administrator to manipulate the Oracle Applications database (which is usually APPS) and an application user with System Administrator responsibility. You must first create an Oracle Applications user, with the name "ETAAPP", to be used when managing Oracle Applications. This user needs to have System Admin Standard Responsibility.

Oracle Applications Support for FIPS and IPv6

For release r12 of Identity Manager, the Oracle Applications Connector does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire an Oracle Application System

You must acquire the Oracle Applications system before you can administer it with Identity Manager.

From the Endpoint type task view

1. Register the Oracle Applications system as an endpoint in Provisioning Manager.

Use the Oracle Applications Endpoint property sheet to register an Oracle Applications system. During the registration process, Identity Manager identifies the Oracle Applications system you want to administer and gathers information about it.

Note: Use the native Oracle tools to verify that the Oracle Applications system can be accessed using the given system login ID and password.

2. Explore the objects that exist on the endpoint.

After registering the server in Identity Manager, you can explore its contents, using the Explore and Correlate Endpoint dialog. The Exploration process finds all Oracle Applications users. You can correlate the accounts with global users at this time or you can correlate them later. The topic "Explore and Correlate Endpoint Dialog" in Provisioning Manager help provides a complete explanation of this dialog.

3. Correlate the explored accounts with global users.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the Oracle Applications user name with each existing global user name. If a match is found, Identity Manager associates the Oracle Applications user with the global user. If a match is not found, Identity Manager performs the next step.
- b. If the Create Global Users as Needed button is checked, Identity Manager creates a new global user and then associates the Oracle Applications user with the global user. If the Create Global Users as Needed button is unchecked, Identity Manager performs the next step.
- c. Identity Manager associates the Oracle Applications user with the [default user] object.

Oracle Applications User Provisioning Roles and Account Templates

The Oracle Applications Default Account Template, provided with the Oracle Applications Connector, does not give a user the minimum security level needed to access an endpoint. One or more responsibilities need to be assigned for the account to be active. The list of responsibilities depends on which Oracle Applications are installed. However, you can use it to as a model to create new account templates.

Note: The Oracle Applications Default Account Template automatically sets the user name and password to the global user account ID so that the user can access Oracle Applications.

Note: An endpoint must first be included (associated) with an account template before responsibilities can be added to it.

Create Account Templates

The Default Account Template, provided with each connector, gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

To create an account template

1. Click the Provisioning Roles task button, select the connector's Account Template in the Object Type drop-down list box and click New.

The Account Template Property Sheet for the specified connector appears.

2. Complete the Account Template Property Sheet by:
 - a. Selecting an endpoint to populate the drop-down and group selection lists.
 - b. Selecting group memberships and other account settings.
 - c. Clicking OK.

A new account template is created for your connector.

Manage Oracle Applications User Accounts

To manage FND accounts, use this procedure:

From the Endpoint Type task view

1. Select an endpoint and click Search.
A list of endpoints for the selected endpoint type is presented.
2. Right-click an endpoint, and select Content from the context menu.
The Endpoint Content appears.
3. Select the Users container from the container tree and enter the search criteria in Search For Content, and then click Search.
4. Click Done.
A list of the user accounts in that endpoint appears.

Chapter 5: Distributed Connector

This section contains the following topics:

[Windows NT Connector](#) (see page 67)

The following connector is considered to be a distributed connector: Windows NT.

The information in this chapter describes how to install and manage this connector.

Windows NT Connector

The Windows NT option provides a single point for all user administration by letting you do the following:

- Register endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage Windows NT accounts using Windows NT-specific policies
- Change account passwords and account activations in one place
- Synchronize global users with their roles or synchronize global users' accounts with their policies
- Assign a Windows NT account template to each of your Windows NT endpoints
- Manage Windows NT Trust relationship between your Windows domains
- Use the default Endpoint Type account template to create accounts with the minimum level of security needed to access a Windows NT endpoint
- Create and manage Windows NT user groups
- Create and manage Windows NT shared folders
- Generate and print reports about Windows NT accounts, groups, and hosts

This connector is managed using the Connector and agent installation process. For more information and requirements, click [here](#) (see page 21).

This connector can also be managed using the [Connector and C++ Server](#) (see page 23) installation process as well.

Configuring

The NT Windows Connector is always installed when you install the NT Connector on the Provisioning Server on Windows. If you plan to acquire that Provisioning Server system as an endpoint, you must install the Remote Agent.

Note: After installing the Remote Agent, add the local machine to the Caft host list.

Upgrading the Provisioning Server

After upgrading the Provisioning Server to r12, you must install the Provisioning Agent for Windows Local Users and Groups. After installing the Agent, you may need to re-authorize the Provisioning Server with caft host by issuing the following command from the SharedComponents\CAM\bin directory:

```
caft host -a hostname/IP address
```

Installing the Provisioning Windows NT Remote Agent with msixec

To install the Provisioning Agent for Windows Local Users and Groups, perform the following steps:

1. Copy the contents of the folder ~\RemoteAgent\Windows200x from the CD to your local machine. For example, C:\temp\RN16.
2. Open a Command Prompt and navigate to the directory where you copied the folder.
3. Issue the following commands:

```
Instmsiw.exe
```

Note: This executable installs or updates the Windows Installer release 2 infrastructure. On some recent Windows Platforms, such as Windows XP and Windows Server 2003, Windows Installer release 2 is installed by default so the Instmsiw.exe command is not required.

```
msiexec /I ISScript11.Msi /qn
```

```
msiexec /I "CA Identity Manager - Provisioning Agent for Windows Local Users and Groups.msi"
```

Follow the prompts on the graphical installer to complete the installation.

To perform a silent install, add in the property=value line found at the end of the EULA. The EULA can be viewed in graphical mode.

```
msiexec /I "CA Identity Manager - Provisioning Agent for Windows Local Users and Groups.msi" /qn  
PROPERTY=VALUE
```

Installing the Provisioning Agent for Windows Local Users and Groups with setup.exe

In this example, we install the Provisioning Agent for Windows Local Users and Groups by using the setup.exe command. Perform the following steps:

1. Copy the contents of the folder ~\RemoteAgent\Windows200x from the CD to your local machine. For example, C:\temp\RN16.
2. Open a Command Prompt and navigate to the directory where you copied the folder.
3. Issue the following command:

```
setup.exe
```

The graphical installer will launch and the Remote Agent can be installed by following the prompts.

4. (Optional) To perform a silent install, add the qn argument and the property=value line found at the bottom of the EULA. (Read the EULA in graphical mode first):

```
setup.exe /s /v"/qn PROPERTY=VALUE"
```

Configure the CAM and CAFT Service for Windows NT

The CAM/CAFT service is used to communicate between Identity Manager C++ Connector Server and the Windows NT targets.

Install the CAM and CAFT Service for Windows NT

You must install the Windows NT Remote Agent and configure the CAM and CAFT Service on any Windows NT machine that you want to administer.

Important! For installing both the Identity Manager Windows NT Remote Agent **and** the Identity Manager Microsoft Exchange Agent on the same machine, use the CAM and CAFT configuration steps for the Microsoft Exchange Agent in the Groupware Connectors section. Be sure to update the CAM and CAFT service logon account, as described in that section.

How to Configure the CAM and CAFT Service for Windows NT

There are two ways to configure the CAM and CAFT service.

To configure the CAM and CAFT Service using the command prompt

1. Log on to your Windows NT machine as the domain administrator or log on to your Windows NT Workgroup machines as the local administrator.
2. Issue the following command from a command window:

```
CAFTHOST -a NT_node_name
```

NT_node_name

Name of the Provisioning Server or the C++ Connector Server if used.

Note: If the Provisioning Server is networked using DHCP or you do not use DNS for name resolution, the network name will not be recognized. Under these conditions, use the TCP/IP address for the Windows NT node name or add a Windows NT node entry in the local hosts file on your Windows NT machine.

3. Verify the previous command by issuing the following command:

```
CAFTHOST -l
```

Note: Firewalls may need to be configured to allow communications using the CAM/CAFT service.

To configure the CAM and CAFT Service using the Host to Caft Definition dialog

1. Log on to your Windows NT machine as the domain administrator or log on to your Windows NT Workgroup machines as the local administrator.
2. Run Host to Caft Definition located in the default Identity Manager Start program group.

Start > Programs > CA > Identity Manager > Host to Caft Definition

3. In the Enter a server name field, enter the name of the Provisioning Server or the C++ Connector Server if used. Click Add.

Note: The same conditions regarding DHCP and DNS listed in the previous section also applies here.

4. Verify that the server name added is listed in the Permitted managing servers list. Click OK.

Note: Firewalls may also need to be configured to allow communications using the CAM/CAFT service.

Activate the CAM and CAFT Encryption for Windows NT

If your Identity Manager installation is using the CAM/CAFT encryption, ask your Identity Manager administrator for a copy of the Public Key keyfile and password in use.

If this is an initial installation of Provisioning Server, Provisioning Manager or Identity Manager Agent, and you want to activate CAM/CAFT encryption for the communication between the Provisioning Server and other Identity Manager servers or system endpoints, you must generate a Public Key file by entering the following command at the command prompt:

```
>caftkey -g keyfile password
```

keyfile

Defines the name that you assign to the key file.

password

Defines the password that you assign to the key file.

To activate the CAM and CAFT encryption

1. Install your Public Key on both CAFT Agent and CAFT Identity Manager boxes using the previously-generated key file (see above) by entering the following command at the command prompt:

```
>caftkey -policy_setting keyfile password
```

- keyfile and password must have the values that you specified while generating the Public Key file.
- policy_setting must be -i, -m, or blank.

The policy_setting governs the communication between this computer (the local computer) and other computers that have the CAM and CAFT service installed, but may or may not have the CAM and CAFT encryption certificates installed.

- Policy -1 (caftkey -i keyfile password)

The -i option specifies Policy -1. This policy lets computers running previous versions of the CAM and CAFT service execute commands on this computer and lets this computer execute commands on those computers. Policy -1 encrypts messages if the other computer has these certificates installed. This policy does not encrypt messages if the other computer does not have these certificates installed.

- Policy 1 (caftkey -m keyfile password)

The -m option specifies Policy 1. This policy prohibits other computers from executing commands on this computer if they are running previous versions of the CAM and CAFT service without the encryption certificates. This policy also prohibits this computer from executing commands on those computers.

If both computers have the CAM and CAFT encryption certificates installed, but have different Public Key Files installed when Policy 1 is set, the command requests between the two computers always fails.

■ Blank Option

The blank option specifies Policy 0. This policy is set if no Public Key File is installed, the CAM and CAFT encryption certificates were not installed properly, or if you do not specify a policy setting when you enter the caftkey command. Policy 0 specifies no encryption.

2. Recycle the CAM Service on each box where you install the new Key as follows:

```
prompt> cam close           //stop Cam/Caft service and processes
```

```
prompt> cam start          //start CAM service and process
```

3. After recycling the CAM service, recycle the CAFT service by issuing the following statement:

```
prompt> caft
```

4. Check the log produced by the CAFT service, and confirm the policy setting by issuing the following statement:

```
prompt> type "%CAI_MSQ%\ftlogs\dg000"
```

The output will be similar to the following example:

```
D:\> type "%CAI_MSQ%\ftlogs\dg000"
```

```
Thu Feb 16 09:05 Starting CAFT version 1.12 (Build 28)
```

```
Thu Feb 16 09:05 Encryption Policy -1
```

```
Thu Feb 16 09:05 ----- CAFT initialize complete -----
```

Check the Policy Setting

To see what mode the machine is operating in, look in the following file:

```
%CAI_MSQ%\ftlogs\dg000
```

The log is as it was lastly generated by the CAFT command. After you change the configuration, you must initiate a new CAFT command so that the log will reflect the latest configuration. You can do this by issuing the following command:

```
Prompt> caft
```


Manage the CAM and CAFT Service for Windows NT

Note: The CAM and CAFT Service allows encryption through certificates.

The CAM and CAFT Service is a daemon process. You can control this process using the Services panel on your Control Panel. To view the Services panel, click the Services icon. The CAM and CAFT Service is called CA Message Queuing.

View the CAM and CAFT Service for Windows NT

Perform the following procedure to view the CAM and CAFT service.

To view the CAM and CAFT service

1. Open the Windows Task Manager.
2. Click the Processes tab on the Windows Task Manager.

The CAM and CAFT daemon processes appear. The following is a sample of these processes:

Image Name	User Name	CPU	CPU Time	Mem Usage
Caftf.exe	Administratortor	00	0:00:16	1,600 K
Cam.exe	SYSTEM	00	0:00:08	704 K

Start the CAM and CAFT Service for Windows NT

Although the CAM and CAFT Service starts automatically, there may be times when you have to manually start it.

To start the CAM and CAFT Service

1. Double-click the Services icon on the Control Panel.
The Services dialog appears.
2. Select Unicenter Message Queuing from the Service window, and click the Start button.
3. Click Close.

Stop the CAM and CAFT Service for Windows NT

Perform the following procedure to stop the CAM and CAFT service.

To stop the CAM and CAFT Service

1. Double-click the Services icon on the Control Panel.

The Services dialog appears.

2. Select Unicenter Message Queuing from the Service window, and click the Stop button.
3. Click Close.

Note: After you stop the CAM and CAFT Service, the service must be restarted so Identity Manager can communicate with the Windows NT Remote Agent.

Windows NT Support for FIPS and IPv6

For release r12 of Identity Manager, the Windows NT Connector supports both FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

How to Acquire the Windows NT Machine

You must acquire the Windows NT machine before you can administer it with Identity Manager. Perform the following procedure to acquire a Windows NT machine.

From the Endpoint Type task view

1. Register the machine as an endpoint in Identity Manager.

Use the Windows NT Endpoint property sheet to register a Windows NT machine. During the registration process, Identity Manager identifies the Windows NT machine you want to administer and gathers information about it.

2. Explore the objects that exist on the endpoint.

After registering the machine in Identity Manager, you must explore its contents by means of the Explore and Correlate Endpoint dialog. The Exploration process finds all the Windows NT objects.

3. Correlate the explored accounts with global users.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the username with each existing global user name. If a match is found, Identity Manager associates the Windows NT account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the full name with each existing global user's full name. If a match is found, Identity Manager associates the Windows account with the global user. If a match is not found, Identity Manager performs the next step.
- c. If the Create Global Users as Needed button is checked, Identity Manager creates a new global user and associates the Windows account with the global user. If the Create Global Users as Needed button is unchecked, then Identity Manager performs the next step. Identity Manager associates the Windows account with the [default user] object.

Note: You can correlate the accounts with global users at this time, or you can do it later.

Significant Fields in the Endpoint Tab of the Endpoint Property Sheet

The Endpoint Tab of the Endpoint Property Sheet includes the following significant fields:

Explore timeout:

Specifies, in seconds, the maximum duration time a request takes for the explore operation (one level and sub-tree searches). Default is 7200 seconds (2 hours).

Operations timeout:

Specifies, in seconds, the maximum time any CAM/CAFT request can take, except the explore option (based searches). Default is 300 seconds (5 minutes).

Changing password

Removes expired flag field. When this field is checked and a password is changed by an administrator, the expired password flag is removed (W2K and Win2003 behavior). If the field is not checked, the expired flag remains (NT behavior), and you must change your password on the first logon.

Terminal Server Attributes Management for Accounts

On the Windows NT Node Property Tab (Windows NT Endpoint Property Sheet), the Terminal Server field is used to identify the Terminal Services family machines. If there are no Terminal Server machines, the field is blank.

For each account in the Terminal Services systems, you can see and manage the attribute from the Environment and Sessions Tab and Terminal Services Profile Tab (Windows NT Account Template or Windows NT Account Property Sheets).

The values associated with the fields on these tabs are the same as those that are provided in the NT native tools, as listed below:

Starting program

When checked, the program in the Program file name field from the directory in the Start in field is launched.

Client devices

When checked, each box causes the action it describes to be performed at account login.

Sessions

Lets you specify actions to be taken in case of long time idle sessions or disconnected sessions.

Terminal Services Profile

Lets you specify the user profile, home directory and login to the terminal server.

Important!--It is strongly recommended not to use the @ symbol in an NT account name if you are managing NT systems (NT4, 2000, 2003, XP) with the terminal services option.

Synchronize BDC Systems

If a Backup Domain Controller (BDC) has been promoted to a PDC (Primary Domain Controller) using NT native tools, you can synchronize BDC promotions.

To synchronize BDC systems

1. Right-click the endpoint and select Custom, Synchronize BDC Promotion.
The NT4 Synchronize with BDC promotion dialog appears.
2. If the selected machine is a BDC, that has been promoted to PDC using NT native tools, fill in the dialog and click Start.

When the operation has run, the BDC is flagged as being the action PDC.

Note: Once the Start button has been clicked, the action cannot be stopped.

Rename Accounts

You have the ability to rename accounts.

To rename an account

1. Right-click the required account, and select Rename from the menu.
The Windows NT account renaming dialog appears.
2. Enter a new name into the New name field and click OK.

At the end of the action, the old name is deleted and the new name is added.

Note: If the name is empty or longer than 20 characters, an error message is displayed.

Windows NT Groups

You can create and maintain Windows NT groups using the Endpoint Type task view. Use the Windows NT Group property sheet to manage your groups.

Trust Relationships

You create and maintain Windows NT trust relationships using the Endpoint Type task view. Use the Windows NT Endpoint property sheet to manage your trust relationships. The endpoint containing the trust relationships must be a PDC.

In managed NT4 PDC properties, you can create or delete inclusions between objects by clicking the Group Settings or Account Settings buttons in the Trust Relationship page.

Search filters for the local groups and for the global objects, where you can specify the attribute and corresponding value, enable you to restrict lists to see only a portion of the available objects.

Shared Folders

You can create and maintain shared folders on Windows NT machines from the Endpoint Type task view. Use the Windows NT Shared Folder property sheet to manage your shared folders.

Size Limit Exceeded

When result size limits are exceeded, every panel only returns as many items as possible. The following are particularly affected:

- Endpoint screens where inclusions are made for trust relationships
- Local Group tab for global group inclusions

For more information, see the following:

- The *Administrator Guide*
- The Working with Endpoints, Windows NT topic in the *Procedures* help

Exit Commands

The following native program exits are supported for the NT connector:

- Pre-Exit: The Windows NT agent executes a user command *before* it performs its own operation.
- Post-Exit: The Windows NT agent executes a user command *after* it performs its own operation.

Resources to write program exits comes with the Provisioning SDK. For more information, see the *Programming Guide for Provisioning*.

Configuration File

Currently, Identity Manager implements the pre-exits and post-exits in the NT Domain. Therefore, to trigger the user add-on commands, you must define them in the ExitSetup.ini file that is installed by Identity Manager. By default, this file does not activate any specific command.

The ExitSetup file is located in the following directory:

Agent Home Dir\Config\ExitSetup.ini

For example:

C:\Program Files\CA\Identity Manager\Provisioning Agent for Windows Local Users and Groups\data\ExitSetup.ini

The following table describes the typical contents of the configuration file:

Headers and Variables	Value	Description
[Pre-exit]		Pre-exit section header
Command=	Provided by the user	User command specified with an absolute path
Stop on error=	Yes/No	Yes-indicates that the agent command is not launched if the pre-exit fails No-indicates that the agent command is launched even if the pre-exit fails. This is the default value.
[Post-exit]		Post-exit section header
Command=	Provided by the user	User command specified with an absolute path

Requirements for the User Commands

The requirements, that are needed to implement the pre-exit and post-exit commands, are as follows:

- For Identity Manager, the execution of a command is successful when its return code (RC) is equal to 0; any other value indicates that the execution failed. This is important because the values that are retrieved by Identity Manager are processed according to this value.
- The argument values, which are sent to Identity Manager agents, are also sent to the user program.
- The pre-exit and post-exit user commands are logged in the Identity Manager log files.
- The pre-exit and post-exit commands are executed each time the agent is executed.

Chapter 6: Mainframe Connectors

This section contains the following topics:

[CA ACF2 Connector](#) (see page 83)

[CA Top Secret Connector](#) (see page 101)

[RACF Connector](#) (see page 126)

The following connectors are considered to be mainframe connectors:

- CA-ACF2 Connector
- CA-Top Secret Connector
- RACF Connector

The information in this chapter describes how to install and manage these connectors.

These connectors are managed using the Connector only installation process. For more information and requirements, click [here](#) (see page 24).

CA ACF2 Connector

The CA ACF2 Connector lets you administer accounts and resources on CA ACF2 systems and provides a single point for all user administration by letting you do the following:

- Register endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage accounts using account templates specific to CA ACF2
- Change account passwords and account activations in one place
- Synchronize global users with their roles or synchronize global users' accounts with their account templates
- Assign account templates to each of your CA ACF2 endpoints
- Use the default Endpoint Type account template to create accounts with the minimum level of security needed to access an CA ACF2 directory
- Create and delete data set and resource rule lines
- Generate and print reports about CA ACF2 accounts

How to Configure Your CA ACF2 System

Once the CA ACF2 Connector has been installed with the Provisioning Server, you must configure your CA ACF2 system to communicate with the connector.

To configure your CA ACF2 system

1. Install the CA LDAP Server for z/OS on your CA ACF2 system.
2. Review the CA LDAP Server for z/OS configuration options.

Step 1. Install the CA LDAP Server

The CA LDAP Server for z/OS provides the communication mechanism for this Identity Manager Connector. This product is a free offering from CA and can be downloaded from support.ca.com. Once downloaded, refer to the *CA LDAP Server for z/OS Installation Guide* for information and instructions on how to install it.

Step 2. Review the CA LDAP Server for z/OS Configuration Options

Once all CA LDAP Server installation steps have been completed and your CA LDAP Server is started, it will be ready to support administration for this Connector. Some clients may need or want to setup additional configuration options for the CA LDAP Server in order to provide additional functionality for the this Connector. Some examples of this additional functionality are the `enable_refresh` option (instructs the CA LDAP Server to refresh User Profile data whenever it is changed on a Logon ID using the CA LDAP Server), or the `enable_secauth` option (provides Secondary Authid maintenance on Logon IDs using the CA LDAP Server). For more information on all available configuration options, see the chapter titled, "CAACF2_DN Backend" in the *CA LDAP Server for z/OS Administrator Guide*.

Securing Provisioning Server Communication to the CA LDAP Server

All communication between the Provisioning Server and the CA LDAP Server for z/OS can be encrypted using SSL (Secure Socket Layers).

To establish communication

- Setup your CA LDAP Server for z/OS to use the Server Mode for SSL connections. For information on how to configure this, see the chapter titled, "CA LDAP Server Using Digital Certificates" in the *CA LDAP Server for z/OS Administrator Guide*.
- Turn on SSL support within the Provisioning Server for your ACF2 endpoint. To do this, bring up the properties of your ACF2 endpoint using the Provisioning Manager. In the section entitled 'Mainframe LDAP Server Information', enable the check box entitled 'Use Server-side SSL' and click Apply. Now, all communication to the configured CA LDAP Server will attempt to use an SSL connection, and will fail and provide an appropriate error message if SSL cannot be established.

ACF2 Support for FIPS and IPv6

For release r12 of Identity Manager, the ACF2 Connector and the Password Synchronization Agent for ACF2 support IPv6, but not FIPS.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire a CA ACF2 System

You must acquire the CA ACF2 system before you can administer it with Identity Manager. When acquiring a CA ACF2 system, perform the following steps from the Endpoint Type task view:

1. Register the server as an endpoint in Identity Manager. This phase is performed by adding a new endpoint under the CA ACF2 Endpoint Type in Identity Manager.

Use the CA ACF2 Endpoint property sheet to view or customize a CA ACF2 system. During the registration process, Identity Manager identifies the CA ACF2 system you want to administer and gathers information about it.

2. Explore the objects that exist on the endpoint.

After registering the server in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all CA ACF2 accounts and objects. You can correlate the accounts with global users at this time or you can correlate them later.

3. Correlate the explored accounts with global users.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the logon ID with each existing global user name. If a match is found, Identity Manager associates the CA ACF2 account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the logon ID name with each existing global user's full name. If a match is found, Identity Manager associates the CA ACF2 account with the global user. If a match is not found, Identity Manager performs the next step.
- c. If the Create Global Users as Needed button is selected, Identity Manager creates a new global user and then associates the CA ACF2 account with the global user. If the Create Global Users as Needed button is cleared, Identity Manager performs the next step.
- d. Identity Manager associates the CA ACF2 account with the [default user] object.

Register ACF2 Endpoints on a Windows System

If you have a Windows system, you can register CA ACF2 endpoints using the Provisioning Manager.

From the Endpoint Type task view

1. Select ACF2 Endpoint from Object Type
2. Click the New button and enter the following information:

Endpoint Name

Specifies a name to refer to the new CA ACF2 endpoint.

Mainframe LDAP IP Address/Machine Name

Specifies the IP address of the CA ACF2 managed system where the CA LDAP Server is configured and running.

Mainframe LDAP Port

Specifies the port number that you specified during the CA LDAP Server for z/OS install. If you are not sure of the Mainframe LDAP Port, see the section Checking Your CA LDAP Server for z/OS Configuration Information.

Mainframe LDAP Suffix

Click the 'Get Suffixes' button to retrieve a list of valid suffixes configured for this CA LDAP Server operations in im naming mode. (See the chapter titled, "CAACF2_DN Backend" in the *CA LDAP Server for z/OS Administrator Guide* for more information on naming mode.)

Admin User

Specifies the CA ACF2 LID of an administrator used from within Identity Manager to manage the CA ACF2 system.

Password

Specifies the password of the CA ACF2 LID above.

Confirm Password

Specifies the same password as above, for confirmation.

Note: After you click OK, the Provisioning Server attempts to establish a connection with the CA LDAP Server for z/OS at the IP address and port supplied, as well as validating the Admin user and Password values supplied. An appropriate error message is displayed if this connection fails. A Global User is inserted in the Provisioning Server with Domain Administrator authority using the ID and password supplied. This Global User is used to administer this endpoint.

Register ACF2 Endpoints on a Solaris System

To register CA ACF2 endpoints, use the batch utility `etutil` to define a CA ACF2 endpoint by specifying a `directoryName`, Mainframe LDAP IP Address, Port, and Suffix. For example:

```
etutil -u USERID -p PASSWORD add 'eTNamespaceName=CA-ACF2,dc=DOMAIN,dc=eta' eTACFDirectory  
name='DIRECTORYNAME' eTZOSLDAPIPAddress=IPADDRESS eTZOSLDAPPort=PORT  
eTZOSLDAPSuffix=SUFFIX
```

where,

DIRECTORYNAME

Specifies the name you desire for this endpoint.

IPADDRESS

Specifies the IP Address or Machine name of the CA-ACF2 system where your CA LDAP Server for z/OS is running.

PORT

Specifies the port number the CA LDAP Server is using.

SUFFIX

Specifies a valid suffix configured for this CA LDAP Server operating in `im` naming mode (For more information on the `naming_mode` option, see the chapter titled, "CAACF2_DN Backend" in the *CA LDAP Server for z/OS Administrator Guide*.)

Check Your CA LDAP Server for z/OS Configuration Information

To view all pertinent information regarding your CA LDAP Server and its configuration, issue a `STATUS` command from the mainframe console where your CA LDAP Server is running. The `STATUS` command provides information such as, version information, port number, configured databases, and suffixes. See the chapter titled, "Startup Options" in the *CA LDAP Server for z/OS Administrator Guide* for more information on the `STATUS` command.

ACF2 Provisioning Roles and Account Templates

The CA-ACF2 Default Policy, provided with the C-ACF2 connector gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

ACF2 Accounts

When synchronizing CA ACF2 accounts, strong synchronization is always the default.

Access and Resource Rules and Rule Keys

Identity Manager lets you maintain existing access and resource rules created in CA ACF2.

Identity Manager protects all resources and data sets by default. Since resources and data sets are evaluated in the same way, Identity Manager provides a consistent approach to security regardless of the physical characteristics of the protected resource or data set.

Access rules and resource rules are defined with different rule types. When you expand the rule types, rule keys appear. Access and resource rules are defined as rule keys.

An individual resource or data set exists in Identity Manager only as a specific member in a rule key. A rule key can represent one resource or data set, or a group of them.

A rule key protects resources and data sets from access by users other than the owner. When a user tries to access a rule key, Identity Manager checks for a rule permission allowing the access. The rule permissions associated with that user determine which resources and data sets that user is authorized to access and the conditions under which access can occur. A user who is not the owner of the resource can access the resource only if the owner or the security administrator authorizes access in a rule permission.

In CA-ACF2, resources and rule keys are only accessed by the following:

- A user who has been granted specific access by the rule permission
- An owner of the rule key
- A user with special override privileges specified in the user's logon ID

Change Control

This identifies the logon IDs or UIDs that can change the rule set. This is called a *change permission*. The change control is defined on the Rule Key Property page under - Change or Restricted Change.

Note: You can only view existing Change Control in the list boxes. You cannot create or delete them.

Rule Lines

Rule lines identify asset type, any qualifiers of the rule key name, or a mask that applies to the rule key. In Identity Manager, rule lines appear under the Rule Lines Tab.

You must identify the logon IDs or UIDs that have access to the resource or data set. This is called a *rule permission*. When a user attempts to access an asset, CA ACF2 checks for a rule permission allowing the access.

UID strings appear under rule lines. These UID strings identify a user or a group of users in a rule permission. The rule permission describes the environment for accessing the resource and determines whether the access is permitted for the user or group of users. Access is restricted by day and time, source, and access path (such as a particular program or library).

Note: You can create and delete rule lines; however, you cannot modify them after you have added them.

Secondary Authids

The ACF2 Connector provides support for administration of DB2 Secondary Authids (ACF2 Cross-Reference or XREF records) on ACF2 Accounts and Policies. DB2 Secondary Authids specify the access rights ACF2 Accounts are permitted when accessing DB2 resources.

Configuration for ACF2

The GSO OPTS record for the current sysid needs to be changed to specify a value in the ALTSYS field. The value should be the same list of sysids (minus the current sysid) as provided on the enable_secauth parameter in the CA LDAP Server for z/OS configuration file (slapd.conf) on the mainframe.

Administer Secondary Authids

To administer Secondary Authids, there is a tab on ACF2 Accounts and Account Templates called Secondary Authids. This tab contains a list control containing the Secondary Authids assigned to the Account or Account Template. There are four buttons (three for Account) on this screen that provide the following functionality:

- Add Record(s) is always activated and provides a dialog screen for searching existing Secondary Authid records and assigning them to the Account or Account Template.
- Manual Add is always activated and provides a dialog screen to manually define a Secondary Authid to an Account or Account Template. The dialog contains a Sysid field and Secondary Authid field for an Account, and an additional Children field for an Account Template.
- Delete Record is activated when a record is selected in the list control, and will remove a Secondary Authid record from an Account or Account Template.
- Add/Del Children is only visible on an Account Template, and is activated when a record is selected in the list control. This button displays a dialog box that lets you maintain the list of Child Secondary Authids assigned to a Parent.

For account templates, Secondary Authids are classified as Parent and Child Secondary Authids. A Parent Secondary Authid is an XREF group record. The Secondary Authids that belong to this group are known as its Children, or Child Secondary Authids, which are XREF source records. When a Parent Secondary Authid is specified with a list of Children, any Account created from this Account Template will be assigned to one of the Child Secondary Authids, not the Parent. The Child Secondary Authid assigned to the Account is determined by the following algorithm:

The Child Secondary Authids are traversed in the order they are listed in the Account Template:

1. If the Child Secondary Authid has room for another entry, the Account will be added to that record.
2. If the record is full, then the next Child Secondary Authid is examined and if it has room, it is added to that one. This continues until a record is found that has room or does not exist.
3. If a Child Secondary Authid is encountered that does not exist, then it will be created with the Account added to it, and it will be added to the Parent Secondary Authid as a Child.

If a Parent Secondary Authid is added to an Account Template with no Children specified, then that Parent record is treated as a Source record and the Account will be added to that record.

In this case, no Child Secondary Authids should ever exist.

Search Screen Notes

When searching for existing Parent records to add to the Account Template, clients should take care that the chosen Parent record does not contain any additional Parent records (XREF Group records). This would occur in a multi-layered Source Group record structure. In this case, the Child Secondary Authids that are XREF Group records on that Parent should be manually removed using the Add/Del Children button, and only Child Secondary Authids (XREF Source records) and un-created records should be listed as Children of that Parent.

Account Screen Specifications

When listing Secondary Authids on an Account, only the Child Secondary Authid records that the Account actually belongs to will be displayed and not the Parent-Child relationship displayed in an Account Template. If an ACF2 Account is deleted, then that Account will be removed from all Secondary Authids it is currently assigned to.

Policy Synchronization

Secondary Authids are synchronized when an Account/Global User is synchronized to its Account Templates/Provisioning Roles. For Account Templates that use the Parent-Child relationship method, an Account is considered synchronized to a particular Secondary Authid entry if the Account is assigned to one of the Child records specified for that entry in the Account Template. Likewise, if the Account is in possession of a Secondary Authid entry that does not exist on the Account Template as either a stand-alone Parent record or as a Child record in a Parent-Child entry, then that record is removed from the Account.

Automated Administration of Secondary Authids

The ACF2 Agent performs the following several automated tasks to maintain the Secondary Authids assigned to an Account or Policy:

- **Record Creation** - The ACF2 Agent will create a Secondary Authid (Parent or Child) if it does not exist. It will add the specified Account to the new Secondary Authid if it is a Child record or stand-alone Parent record, and it will add a new Child record to a new Parent record if the specified Parent does not exist.
- **Record Modification** - If a new Child record (specified on the Account Template) needs to be created, then it will be created and added to the Parent record.
- **Record Deletion** - If an ACF2 Account is removed from a Secondary Authid (either through an Account modification or deletion), and the Account is the last Logonid on the Secondary Authid, then that record will be deleted and removed from any Parent records it belongs to. If the Secondary Authid is also the last record on the Parent record, then the Parent record will be deleted also.

Errors

If an error occurs on an add or modify Account transaction related to Secondary Authids, the LDAP server will return an error code of `PARTIAL_SUCCESS`. This tells Identity Manager that an error occurred during processing, but processing continued and the transaction is to be considered a success. The reason for this is that any processing that had already occurred against the Account (creation, modification, and deletion) will have occurred successfully before Secondary Authid maintenance was performed. The Provisioning Manager will report the error, but proceed as a successful transaction. The error can then be reviewed, and appropriate action can be determined and taken against Secondary Authids.

For Account delete transactions, if an error occurs during Secondary Authid maintenance, this error will be returned as an error and the Account will not be deleted. This is because the eTA LDAP Server does not handle partial success situations with Account deletes; therefore the message returned must be a success or failure. Review the error returned and take appropriate action to enable the transaction to proceed properly.

The following error messages can be returned when an error occurs with Secondary Authids:

LDP0143E Manual maintenance required for XREF SGP record(s): recordlist,...

An error occurred during maintenance of at least one Secondary Authid record and the user will need to examine this list of records and provide appropriate maintenance based upon the type of transaction that occurred.

LDP0144E F ACF2,NEWXREF failed and manual maintenance required for XREF SGP record(s): recordlist,...

This is similar to the message above, but the refresh command failed also. Take appropriate action against the list of Secondary Authids and issue a `F ACF2,NEWXREF` command when completed.

LDP0145E F ACF2,NEWXREF failed.

The `F ACF2,NEWXREF` command failed and needs to be issued manually.

TSO Alias Support

The ACF2 Connector also supports the creation of a TSO Alias when a user is created or modified with TSO access being granted. The alias value is always the value of the ACF2 Account being created or modified. To enable this support, see the chapter titled, "CAACF2 DN Backend" in the *CA LDAP Server for z/OS Administrator Guide*, for more information.

ACF2 Program Exits

The CA ACF2 Connector supports the use of Program Exits which are incorporated as Common Exits. Program Exits provide you with the capability to perform certain actions before or after an account is created, modified or deleted from Identity Manager. These exits are referenced either on the Endpoint property page to execute custom code on a single endpoint, or on the Account Template property page to execute custom code on multiple endpoints. Actions might include native CA DSI (CA Distributed Security Integration) or CA LDAP Server for z/OS commands in order to modify account privileges or access to resources on the CA ACF2 system.

To see a sample program exit, refer to *PS_HOME\Templates\OS390Exits*.

For more detailed information about how to write program exits, see the *Programming Guide for Provisioning*.

Proxy Configuration

The CA ACF2 Endpoint page contains a section where clients can configure a Proxy administrative ID and password to be used for user password changes from the SAWI. When configured, this ID and password is used to issue the password change request for the Self-Service user to change their password. This is helpful and needed if a Self-Service user cannot supply a password (for example, the password is forgotten) or their password is expired on ACF2 and they cannot be authenticated. When using a proxy administrative ID, standard ACF2 security rules apply (for example, scoping) and password syntax checking specified in the GSO Pswd Record is enforced. However, the ACF2 mindays value for the user is not enforced since the password change is done through an administrator.

Note: The check boxes on the Endpoints Setting tab are for legacy purposes only. With Identity Manager r12, you can perform proxy configuration and administration support from the Identity Manager Self-Service interface.

Proxy Administration Support

You can now configure a proxy ID for all tasks accomplished within Identity Manager. Previously, a proxy ID could only be configured for use with requests generated from the SAWI interface. This proxy ID is maintained on the main Endpoint page in the Proxy Administration Configuration section. The proxy ID can be used for any type of Identity Manager request against supported objects, and for any Identity Manager Administrator that is logged on.

Note: The enhancement is only recommended to use after careful consideration (and preparation) of the following consequences:

1. Any Global User (with the proper privileges provided within Identity Manager) is able to administer ACF2 Logonids and Rules under the configured proxy ID. Any mainframe security product scoping is lost; only the scoping of the proxy ID is enforced.
2. As mentioned above, Identity Manager security settings are now the only point of enforcement against a Global User manipulating mainframe security data.
3. Any reports or auditing methods against administration of your mainframe security data that originate from the mainframe is now compromised; the only ID that shows up for any administration that occurred from Identity Manager is the configured proxy ID.
4. If the proxy ID's password changes on the mainframe, the password must be changed on every Endpoint Page within Identity Manager that it is configured for.

By default, the Connector operates in the same mode as in past releases; the logged-on Global User and their password are used for submitting any requests destined to the mainframe security product. The common directory page entitled Endpoint Settings provides two checkbox controls under the description Administrator Credentials that control the three possible settings:

Use logged-in Administrator's credentials

Default setting. Indicates that the logged-in Administrator (Global User) is used as the credentials for ALL requests, even from the SAWI.

Use proxy for SAWI changes

Indicates that the logged-in Administrator (Global User) is used as the credentials for all requests EXCEPT for requests from the SAWI interface. The proxy ID credentials (if available) are used for requests coming from the SAWI interface.

Use proxy for ALL requests

When no checkbox is checked, this indicates that the proxy ID credentials (if available) are to be used for ALL requests.

When any request occurs from Identity Manager, these settings are checked against the endpoint where the request is targeted. If, based on the endpoint settings and the type of request (SAWI or otherwise), proxy credentials are to be used, the credentials that are defined for that endpoint are retrieved and used for the request. In the case where endpoint credentials are supposed to be used, but no credentials exist (either Proxy ID or password contains no value), the proxy credentials are not used for the request and the request proceeds using the logged-in Administrator (Global User) credentials.

Note: The check boxes on this tab are for legacy purposes only. With Identity Manager r12, you can perform proxy configuration and administration support from the Identity Manager Self-Service interface.

Conventions

Use the following CA-ACF2 conventions in your etauil commands:

- The endpoint type name (eTNamespaceName) is CA-ACF2
- The endpoint type prefix is ACF. Therefore, the CA-ACF2 class names are:
 - eTACFDirectory for an endpoint
 - eTACFPolicyContainer for an account template container
 - eTACFPolicy for an account template

LDAP Directory Services (LDS)

CA-ACF2 provides the ability to synchronize z/OS security information management with LDAP compliant directory managed servers. The LDS component of the CA-ACF2 system sends requests through LDAP commands to the Provisioning service and is directed to the LDS backend that is located on a Windows based PC.

The module that handles LDS processing within Identity Manager is named `back_lds.dll` (dynamic link library) and is intended to augment the functionality provided within Identity Manager and the CA ACF2 Security product. The Identity Manager LDS backend allows the user of the CA ACF2 for z/OS the ability to interface directly with the Identity Manager database.

When a command is issued on the CA ACF2 system to add a user, a Global User is created in the local Identity Manager database with the specified password. Additionally, if the user is to be associated with a Provisioning Role, the Identity Manager inclusion will be generated to associate this user with the desired provisioning role(s).

When a command is issued to the CA ACF2 system to change the password, or any other mapped field of a user, a change results in the local Identity Manager database and optionally is propagated to all necessary platforms.

When a command is issued to the CA ACF2 system to delete a user, the Global User is deleted from the local Identity Manager database and any associated inclusions are removed. Depending upon the platform and Identity Manager settings, this may also result in the deletion of accounts on other platforms.

LDS Features

The LDS backend within Identity Manager provides special processing when any of the following LDAP attributes are mapped to respective CA ACF2 fields in the LDS XREF field mapping:

- If LDAP attribute eTS390Role is specified as part of the command, the user is attached or removed from the specified Provisioning Role value. This may result in the addition or removal of accounts on any platform depending upon the Identity Manager Account Templates attached to that Provisioning Role. Typically, this attribute is mapped to a single-valued, user-defined field containing a comma-separated list of Provisioning Roles of which the user is a member.
- If the CA ACF2 password is mapped to the eTS390Password attribute, the password is applied to the Identity Manager Global User and automatically propagated to all accounts associated with that Global User.

Note: This attribute should only be used by clients that do **NOT** explore/correlate their ACF accounts into Identity Manager. Clients that do run explore/correlates should follow the instructions in the Password Synchronization feature bulletin.

- Password Synchronization - Password changes to ACF2 Loginids can be propagated into Identity Manager to change a Global User's password and all accounts associated with the Global User. To accomplish this, an LDS record must be manually created (not from the LDS wizard) using the instructions from the Configuration and Usage section with the following changes:
 - Step 3: Set the USERDNS (user dns) as follows:
`eTACFLidName=%L,eTACFLidContainerName=Accounts,`
`eTACFDirectoryName=www,`
`eTNamespaceName=CA-ACF2,dc=XXX,dc=eta`
where eTACFEndpointName=www equals the Identity Manager CA ACF2 Endpoint name and dc=XXX is the name of the Identity Manager domain for this LDAP node in UPPERCASE
 - Step 4: Set the OBJCLASS (object class) to eTACFLid
 - Step 6: In the XREF section, add the entry: Password/eTSyncPassword
- If the CA ACF2 password is mapped to the eTPassword attribute, this causes the password to only be applied to the Global User with no propagation to other accounts.
- If LDAP attribute eTS390Profile is specified as part of the command, the user is attached or removed from the specified Role value. This may result in the addition or removal of accounts on any platform depending upon the Identity Manager Account Templates attached to that Provisioning Role.

Note: Values specified for both eTS390Profile and eTS390Role must exist within Identity Manager as valid Provisioning Role names.

Configuration and Usage

In order to configure your CA ACF2 system to drive requests through LDS to your Provisioning service, we recommend that you use the LDS Wizard provided within Identity Manager on the Endpoint property page, to create or modify the LDS record on your ACF2 system. This wizard is only active and usable if the appropriate version of ACF2 is running, which supports LDS.

If you choose not to use the wizard, then you must perform the following steps on the LDS record to invoke the LDS backend. For detailed information on LDS and setting up the LDS record, see the CA ACF2 Administrator Guide.

1. Sign on to the mainframe CA ACF2 system and create an LDS LDAP control record with the INSERT command. Set the ADMDN (admin dn) as follows:

```
eTGlobalUserName=<user>,eTGlobalUserContainerName=Global Users,  
eTNamespaceName=CommonObjects,dc=XXX,dc=eta
```

where eTGlobalUserName is the name of an Identity Manager global user that has full authority to the domain (DomainAdministrator). dc=XXX is the name of the Identity Manager domain for this LDAP node in UPPERCASE.

2. Set the ADMPSWD (admin password) to the correct password for the Identity Manager global user.
3. Set the USERDNS (user dns) as follows:

```
eTACFLidName=%L,eTACFLidContainerName=Accounts,  
eTACFDirectoryName-www,  
eTNamespaceName=CA-ACF2,dc=XXX,dc=eta
```

where eTACFEndpointName=www equals the CA ACF2 Endpoint name and dc=XXX is the name of the Identity Manager domain for this LDAP node in UPPERCASE.

4. Set the OBJCLASS (object class) to eTGlobalUser.
5. Set the URL (uniform resource locator) to the machine name or IP address that is running the Provisioning service. Make sure that this URL contains the correct port. 20389 is used in the following example:

```
LDAP://machine.ca.com:20389
```

6. Add the appropriate XREF mappings between ACF2 fields and LDAP attributes as required.
7. Refresh LDS by issuing the following commands:

```
F ACF2,REFRESH(LDAP),TYPE(LDS)
```

8. Confirm that LDS is started on the ACF2 system.

CA Top Secret Connector

The Top Secret Connector lets you administer accounts and resources on CA Top Secret systems. The Top Secret Connector provides a single point for all user administration by letting you do the following:

- Register endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage CA Top Secret accounts using account templates specific to CA Top Secret
- Change account passwords and account activations in one place
- Synchronize global users with their roles or synchronize global users' accounts with their account templates
- Assign an CA Top Secret policy to each of your CA Top Secret endpoints
- Use the default Endpoint Type policy to create accounts with the minimum level of security needed to access an CA Top Secret directory
- Generate and print reports about CA Top Secret accounts

Install and Configure the TSS Connector

When installing the CA Top Secret Connector, perform the following steps:

1. Install the Provisioning Server with the CA CA-Top Secret Connector
2. Configure your CA CA-Top Secret system.

You should review the information in the *Provisioning Guide* and install the Provisioning Server before you continue with this chapter.

How to Configure Your CA Top Secret System

Once the CA Top Secret Connector has been installed with the Provisioning Server, you must configure your CA Top Secret system to communicate with the connector.

To configure your CA Top Secret system

1. Install the CA LDAP Server for z/OS on your CA Top Secret system.
2. Review the CA LDAP Server for z/OS configuration options.

Step 1. Install the CA LDAP Server

The CA LDAP Server for z/OS provides the communication mechanism for this Identity Manager Connector. This product is a free offering from CA and can be downloaded from support.ca.com. Once downloaded, refer to the *CA LDAP Server for z/OS Installation Guide* for information and instructions on how to install it.

Step 2. Review the CA LDAP Server for z/OS Configuration Options

Once all CA LDAP Server installation steps have been completed and your CA LDAP Server is started, it will be ready to support administration for this Connector. Some clients may need or want to setup additional configuration options for the CA LDAP Server in order to provide additional functionality for the this Connector. An example of this additional functionality is the `enable_refresh` option (instructs the CA LDAP Server to activate UID or GID fields for an ACID when changed using the CA LDAP Server). For more information on all available configuration options, see the chapter titled, "CATSS_DN Backend" in the *CA LDAP Server for z/OS Administrator Guide*.

Securing Provisioning Server Communication to the CA LDAP Server

All communication between the Provisioning Server and the CA LDAP Server for z/OS can be encrypted using SSL (Secure Socket Layers).

To establish communication

- Setup your CA LDAP Server for z/OS to use the Server Mode for SSL connections. For information on how to configure this, see the chapter titled "CA LDAP Server Using Digital Certificates" in the *CA LDAP Server for z/OS Administrator Guide*.
- Turn on SSL support within the Provisioning Server for your TSS endpoint. To do this, bring up the properties of your TSS endpoint using the Provisioning Manager. In the section entitled 'Mainframe LDAP Server Information', enable the check box entitled 'Use Server-side SSL' and click Apply. Now, all communication to the configured CA LDAP Server will attempt to use an SSL connection, and will fail and provide an appropriate error message if SSL cannot be established.

Top Secret Support for FIPS and IPv6

For release r12 of Identity Manager, the TSS Connector and the Password Synchronization Agent for TSS support IPv6 but not FIPS.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire a CA Top Secret System

You must acquire the CA Top Secret system before you can administer it with Identity Manager. When acquiring a CA Top Secret system, perform the following steps from the Endpoint Type task view:

1. Register the server as an endpoint in Identity Manager. This phase is performed by adding a new endpoint under the CA Top Secret Endpoint Type in Identity Manager.

Use the CA Top Secret Endpoint property sheet to view or customize a CA Top Secret system. During the registration process, Identity Manager identifies the CA Top Secret system you want to administer and gathers information about it.

2. Explore the objects that exist on the endpoint.

After registering the server in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all CA Top Secret accounts and objects. You can correlate the accounts with global users at this time or you can correlate them later.

3. Correlate the explored accounts with global users.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the ACID with each existing global user name. If a match is found, Identity Manager associates the CA CA-Top Secret account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the ACID name with each existing global user's full name. If a match is found, Identity Manager associates the CA Top Secret account with the global user. If a match is not found, Identity Manager performs the next step.
- c. If the Create Global Users as Needed button is checked, Identity Manager creates a new global user and then associates the CA Top Secret account with the global user. If the Create Global Users as Needed button is unchecked, Identity Manager performs the next step.
- d. Identity Manager associates the CA Top Secret account with the [default user] object.

Note: If you are exploring and correlating a very large database, we recommend that you run the Explore, Correlate, and Update steps individually and use the `dxtunedb` utility to tune the database after each step. This significantly reduces your explore and correlate time.

To accomplish this, follow these steps:

1. Stop the Provisioning Service and the eTrust Directory services for `etrustadmin`.
2. Run the `dxtunedb` command to tune the Identity Manager database.
3. Restart the Provisioning Service and eTrust Directory services for `etrustadmin`.
4. Run each step.

Register CA Top Secret Endpoints on Windows

If you have a Windows system, you can register CA-Top Secret endpoints using the Provisioning Manager.

From the Endpoint Type task view

1. Select CA Top Secret Endpoints from Object Type.
2. Click the New button. You are required to fill in the following information:

Endpoint Name

Specifies a name to refer to the new CA Top Secret endpoint.

Mainframe LDAP IP Address/Machine Name

Specifies the IP address of the CA-Top Secret managed system where the CA LDAP Server is configured and running.

Mainframe LDAP Port

Specifies the port number that you specified during the CA LDAP Server for z/OS install. If you are not sure of the Mainframe LDAP Port, see the section Checking Your CA LDAP Server for z/OS Configuration Information.

Mainframe LDAP Suffix

Click the 'Get Suffixes' button to retrieve a list of valid suffixes configured for this CA LDAP Server operations in im naming mode. (See the chapter titled, "CATSS_DN Backend" in the *CA LDAP Server for z/OS Administrator Guide* for more information on naming mode.)

Admin User

Specifies the CA Top Secret ACID of an administrator to be used from within Identity Manager to manage the Top Secret system.

Password

Specifies the password of the Top Secret ACID above.

Confirm Password

Specifies the same password as above, for confirmation.

Note: After you click OK, the Provisioning Server attempts to establish a connection with the CA LDAP Server for z/OS at the IP address and port supplied, as well as validating the Admin user and Password values supplied. An appropriate error message is displayed if this connection fails. A Global User is inserted in the Provisioning Server with Domain Administrator authority using the ID and password supplied. This Global User is used to administer this CA Top Secret endpoint.

Register CA Top Secret Endpoints on Solaris

To register CA-Top Secret endpoints, use the batch utility `etaultil` to define a TSS endpoint by specifying a `directoryName`, Mainframe LDAP IP Address, Port, and Suffix. For example:

```
etaultil -u USERID -p PASSWORD add 'eTNamespaceName=TSS,dc=DOMAIN,dc=eta' eTTSSDirectory  
name='DIRECTORYNAME' eTZOSLDAPIPAddress=IPADDRESS eTZOSLDAPPort=PORT  
eTZOSLDAPSuffix=SUFFIX
```

where,

DIRECTORYNAME

Specifies the name you desire for this endpoint.

IPADDRESS

Specifies the IP Address or Machine name of the TSS system where your CA LDAP Server for z/OS is running.

PORT

Specifies the port number the CA LDAP Server is using.

SUFFIX

Specifies a valid suffix configured for this CA LDAP Server operating in `im` naming mode (For more information on the `naming_mode` option, see the chapter titled, "CATSS_DN Backend" in the *CA LDAP Server for z/OS Administrator Guide*.)

Check Your CA LDAP Server for z/OS Configuration Information

To view all pertinent information regarding your CA LDAP Server and its configuration, issue a `STATUS` command from the mainframe console where your CA LDAP Server is running. The `STATUS` command provides information such as, version information, port number, configured databases, and suffixes. See the chapter titled, "Startup Options" in the *CA LDAP Server for z/OS Administrator Guide* for more information on the `STATUS` command.

TSS Provisioning Roles and Account Templates

The CA-Top Secret Default Policy, provided with the TSS connector gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

TSS Accounts

ACIDs are called accounts in Identity Manager. Use the CA Top Secret Account property sheet when managing your accounts. Only the ACID types of USER, DCA, VCA, ZCA, LSCA and SCA are populated under the accounts container when you explore a CA Top Secret endpoint.

Note: The containers described below may or may not get populated depending on the authority of the administrator and the security implementation of the CA Top Secret system.

Departments

Only the ACID types of DEPARTMENT are populated under the departments container when you explore a CA Top Secret endpoint.

Divisions

Only the ACID types of DIVISION are populated under the divisions container when you explore a CA Top Secret endpoint..

Zones

Only the ACID types of ZONE are populated under the zones container when you explore a CA Top Secret endpoint.

Groups

Only the ACID types of GROUP are populated under the accounts container when you explore a CA Top Secret endpoint.

Note: When synchronizing CA Top Secret accounts, strong synchronization is the default.

Profiles

Profiles let you group a collection of similar resource access authorizations into one ACID. This profile ACID is then added to all users who require those particular access rights. Should the access requirements change, only the profile needs to be changed; the associated ACIDs are automatically updated.

Identity Manager lets you create and maintain CA Top Secret profiles using the Endpoint Type task view. Use the CA Top Secret Profile property sheet when managing your profiles.

Facilities

Facilities identify specific facilities under which the user must access the resource. Identity Manager lets you create and maintain CA Top Secret facilities using the Endpoint Type task view. Use the CA Top Secret Facility property sheet when managing your facilities.

Ownerships

Ownerships identify the resources that are protected by CA Top Secret. Identity Manager lets you create and maintain CA Top Secret ownerships using the Endpoint Type task view. Use the CA Top Secret Ownership property sheet when managing your ownerships.

Permissions

Permissions identify the conditions under which the ACID can access a resource. Identity Manager lets you create and maintain CA Top Secret permissions using the Endpoint Type task view. Use the CA Top Secret Permission Name property sheet when managing your permissions.

Profile Lists

Profile lists identify common resource access authorizations and are usually classified by task. Identity Manager lets you create and maintain CA Top Secret profile lists using the Endpoint Type task view. Use the CA Top Secret Profile List property sheet when managing your profile lists.

TSO Alias Support

The CA Top Secret connector also supports the creation of a TSO Alias when a user is created or modified with TSO access being granted. The alias value is always the value of the TSS Account being created or modified. To enable this support, see the chapter titled, "CATSS_DN Backend" in the *CA LDAP Server for z/OS Administrator Guide* for more information.

TSS Program Exits

The CA Top Secret connector supports the use of Program Exits which are incorporated as 'Common Exits'. Program Exits provide you with the capability to perform certain actions before or after an account is created, modified or deleted from Identity Manager. These exits can be referenced either on the Endpoint property page to execute custom code on a single endpoint, or on the Account Template property page to execute custom code on multiple endpoints. Actions might include native CA DSI (CA Distributed Security Integration) or CA LDAP Server for z/OS commands in order to modify account privileges or access to resources on the CA Top Secret system.

To see a sample program exit, refer to the OS390 subdirectory under the Identity Manager Templates directory.

For more detailed information about how to write program exits, see the *Programming Guide for Provisioning*.

Proxy Configuration

The TSS Endpoint page contains a section where clients can configure a Proxy administrative ID and password to be used for user password changes from the SAWI interface. When configured, this ID and password is used to issue the password change request for the SAWI user to change their password. This is helpful and needed if a SAWI user cannot supply a password (for example, the password is forgotten) or their password is expired on Top Secret and they cannot be authenticated. The following is an explanation of the algorithm followed by the TSS Agent when a password change is instantiated through the SAWI using a proxy administrator:

1. Under the authority of the proxy ID, an administrative reset is done to the SAWI users' password to an eight digit random number. This is needed in order to enforce password syntax rules specified by the NEWPW Control Option. The password change must occur under authority of the SAWI user. Since the SAWI user can instantiate a password change without supplying their current password, this administrative reset is necessary to set the password to a known value for the TSS Agent. This password change is immediately expired, so in the event of any kind of failure, this ID cannot access the system.
2. After the administrative reset, the TSS Agent can then issue a password change under the authority of the SAWI user, using the random number password to authenticate. This password change request is now subject to password syntax rules specified by the NEWPW Control Option, and the SAWI user sees an appropriate message if the new password does not comply.

When using a proxy administrative ID, standard TSS security rules apply (for example, scoping) and password syntax checking specified in the NEWPW Control Option is enforced. However, the NEWPW control Option Mindays value for the user is not enforced since the administrative reset password change is done through an administrator and is set to immediately expire. If the administrative password reset was not set to immediately expire, the Mindays Control Option would be enforced on the subsequent password change through the Self-Service user, and would likely fail.

Note: The check boxes on the Endpoints Setting tab are for legacy purposes only. With Identity Manager r12, you can perform proxy configuration and administrative support from the Identity Manager Self-Service interface.

Proxy Administration Support

You can now configure a proxy ID for all tasks accomplished within Identity Manager. Previously, a proxy ID could only be configured for use with requests generated from the SAWI interface. This proxy ID is maintained on the main Endpoint page in the Proxy Administration Configuration section. The proxy ID can be used for any type of Identity Manager request against supported objects, and for any Identity Manager Administrator that is logged on.

Note: The enhancement is only recommended to use after careful consideration (and preparation) of the following consequences:

1. Any Global User (with the proper privileges provided within Identity Manager) is able to administer Top Secret Acids and their access privileges, (for example, facilities and permissions) under the configured proxy ID. Any mainframe security product scoping is lost; only the scoping of the proxy ID is enforced.
2. As mentioned above, Identity Manager security settings are now the only point of enforcement against a Global User manipulating mainframe security data.
3. Any reports or auditing methods against administration of your mainframe security data that originate from the mainframe is now compromised; the only ID that shows up for any administration that occurred from Identity Manager is the configured proxy ID.
4. If the proxy ID's password changes on the mainframe, the password must be changed on every Endpoint Page within Identity Manager that it is configured for.

By default, the Connector operates in the same mode as in past releases; the logged-on Global User and their password are used for submitting any requests destined to the mainframe security product. The common directory page entitled Endpoint Settings provides two checkbox controls under the description Administrator Credentials that control the three possible settings:

Use logged-in Administrator's credentials

Default setting. Indicates that the logged-in Administrator (Global User) is used as the credentials for ALL requests, even from the SAWI.

Use proxy for SAWI changes

Indicates that the logged-in Administrator (Global User) is used as the credentials for all requests EXCEPT for requests from the SAWI interface. The proxy ID credentials (if available) are used for requests coming from the SAWI interface.

Use proxy for ALL requests

When no checkbox is checked, this indicates that the proxy ID credentials (if available) are to be used for ALL requests.

When any request occurs from Identity Manager, these settings are checked against the endpoint where the request is targeted. If, based on the endpoint settings and the type of request (SAWI or otherwise), proxy credentials are to be used, the credentials that are defined for that endpoint are retrieved and used for the request. In the case where endpoint credentials are supposed to be used, but no credentials exist (either Proxy ID or password contains no value), the proxy credentials are not used for the request and the request proceeds using the logged-in Administrator (Global User) credentials.

Note: The check boxes on this tab are for legacy purposes only. With Identity Manager r12, you can perform proxy configuration and administration support from the Identity Manager Self-Service interface.

TSS Conventions

Use the following CA-Top Secret conventions in your etutil commands:

- The endpoint type name is CA-Top Secret
- The endpoint type prefix is TSS. Therefore, the CA-Top Secret class names are:
 - eTTSSDirectory for an endpoint
 - eTTSSPolicyContainerName for an account template container
 - eTTSSPolicy for an account template

LDAP Directory Services (LDS)

CA Top Secret provides the ability to synchronize z/OS security information management with LDAP compliant directory managed servers. The LDS component of the CA Top Secret system sends requests via LDAP commands through the Provisioning service and is directed to the LDS backend that is located on a Windows-based PC.

The module that handles LDS processing within Identity Manager is named `back_lds.dll` (dynamic link library) and is intended to augment the functionality provided within Identity Manager and the CA Top Secret Security product. The Identity Manager LDS backend allows the user of the CA Top Secret for z/OS the ability to interface directly with the Identity Manager database.

When a command is issued on the CA Top Secret system, to add a user, a Global User is created in the local Identity Manager database with the specified password. Additionally, if the user is to be associated with an Identity Manager Role, the Identity Manager inclusion will be generated to associate this user with the desired role or roles.

When a command is issued to the CA Top Secret system to change the password, or any other "mapped" field of a user, a change results in the local Identity Manager database and optionally is propagated to all necessary platforms.

When a command is issued to the CA Top Secret system to delete a user, the Global User is deleted from the local Identity Manager database and any associated inclusions are removed. Depending upon the platform, and Identity Manager settings, this may also result in the deletion of accounts on other platforms.

Features

The LDS backend within Identity Manager provides special processing when any of the following LDAP attributes are mapped to respective CA Top Secret fields in the NDT XREF record:

- If LDAP attribute eTS390Role is specified as part of the command, the user is attached or removed from the specified Provisioning Role value. This may result in the addition or removal of accounts on any platform depending upon the Identity Manager Account Templates attached to that Provisioning Role. Typically, this attribute is mapped to a single-valued user defined field containing a comma-separated list of Provisioning Roles that the user is a member of.
- If the CA Top Secret password is mapped to the eTS390Password attribute, the password is applied to the Identity Manager Global User and automatically propagated to all accounts associated with that Global User.

Note: This attribute should only be used by clients that do **NOT** explore/correlate their TSS accounts into Identity Manager. Clients that do run explore/correlate should follow the instructions in the Password Synchronization feature bulletin.

- Password Synchronization - Password changes to TSS User Acids (Acid types User, DCA, VCA, ZCA, LSCA, SCA) can be propagated into Identity Manager to change a Global User's password and all accounts associated with the Global User. To accomplish this, an LDS record must be manually created (not from the LDS Wizard) using the instructions from the Configuration and Usage section with the following changes:
 - Step 3. Set the USERDNS (user dns) as follows:

```
eTTSSAcidName=%L,eTTSSAcidContainerName=Accounts,
eTTSSDirectoryName=www,
eTNamespaceName=CA-Top Secret,dc=XXX,dc=eta
```

where eTTSSEndpointName=www equals the Identity Manager CA Top Secret Endpoint name and dc=XXX is the name of the Identity Manager domain for this LDAP node in UPPERCASE.
 - Step 4. Set the OBJCLASS (object class) to eTTSSAcid.
 - Step 6. In the XREF section, add the entry:
Password,eTSSyncPassword.
- If the CA Top Secret password is mapped to the eTPassword attribute, this causes the password to only be applied to the Global User with no propagation to other accounts.

- If LDAP attribute eTS390Profile is specified as part of the command, the user is attached or removed from the specified Provisioning Role value. This may result in the addition or removal of accounts on any platform depending upon the Identity Manager Account Templates attached to that Provisioning Role. Typically, this attribute would be mapped to the Profiles field within TSS, because 'membership' to a TSS Profile can directly translate to Provisioning Role/Account Template concept within Identity Manager.

Note: Values specified for both eTS390Profile and eTS390Role must exist within Identity Manager as valid Provisioning Role names.

Configuration and Usage

In order to configure your CA Top Secret system to drive requests through LDS to your Provisioning service, we recommend that you use the LDS Wizard provided within Identity Manager on the Endpoint property page, to create or modify the NDT record on your CA Top Secret system. This wizard is only active and usable if the appropriate version of CA Top Secret is running, which supports LDS.

If you choose not to use the wizard, then you must perform the following steps on the LDS record to invoke the LDS backend. For detailed information on LDS and setting up the LDS record, please refer to Chapter 13 of the CA Top Secret 5.3 Administrator Guide under LDAP Directory Services.

1. Sign on to the mainframe CA Top Secret system and create an LDAP node definition with the TSS ADD(NDT) LDAPNODE(XXXXXX) command. Set the ADMDN (admin dn) as follows:

```
eTGlobalUserName=<user>,eTGlobalUserContainerName=Global Users,
eTNamespaceName=CommonObjects,dc=XXX,dc=eta
```

where eTGlobalUserName is the name of an Identity Manager global user that has full authority to the domain (DomainAdministrator). dc=XXX is the name of the Identity Manager domain for this LDAP node in UPPERCASE.

2. Set the ADMPSWD (admin password) to the correct password for the Identity Manager global user.
3. Set the USERDNS (user dns) as follows:

```
eTGlobalUserName=%L,eTGlobalUserContainerName=Global Users,
eTNamespaceName=CommonObjects,dc=XXX,dc=lds
```

where dc=XXX is the name of the Identity Manager domain for this LDAP node in UPPERCASE.

4. Set the OBJCLASS (object class) to eTGlobalUser.
5. Set the URL (uniform resource locator) to the machine name or IP address that is running the Provisioning service. Make sure that this URL contains the correct port. 20389 is used in the example below:

```
LDAP://machine.ca.com:20389
```

6. Add the appropriate XREF mappings between CA Top Secret fields and LDAP attributes as required.
7. Recycle LDS and refresh the NDT by issuing the following commands:

```
TSS MODIFY(LDS(OFF))
TSS REP(NDT) ACTIVE(YES)
TSS MODIFY(LDS(ON))
```

Best Practices

This section describes the recommended practices to use for the implementation of CA Top Secret security administration.

Using z/OS Security Administration

There is a distinction between a security officer and a security administrator. The security officer for z/OS is a senior CA Top Secret administrator and is typically an infrequent user of Identity Manager. However, the security officer is responsible for the z/OS security policies established and used with Identity Manager.

On the other hand, a designated security administrator manages a subset of the security management function. This subset depends on the management policy of the organization and may include the creation of new user accounts on various systems, the resetting of passwords, and so on. The capabilities of these security administrators (whether through Identity Manager or some other means) is guided by the security officer.

Because policies control the capabilities that a user has on a CA Top Secret security system, it is important that they be set up correctly to enforce the existing policies in your organization. The following sections discuss recommendations for using Identity Manager to create and manage your policies.

However, Identity Manager is not intended to be the primary interface for the experienced CA Top Secret administrator. An experienced CA Top Secret security administrator manages CA Top Secret better by using commands issued under TSO rather than working in the Identity Manager framework. Many functions are managed directly, and certain capabilities are *only* available through direct CA Top Secret commands. For those instances, we recommend that you implement the LDAP Directory Synchronization (LDS) option for CA Top Secret security (available in CA-Top Secret Version 5.3 and above). This ensures that the information that is added into CA Top Secret outside of Identity Manager coincides with Identity Manager.

Understanding the Explore and Correlate Function

The Identity Manager explore and correlate function gathers ACID, profile, and group information from the CA Top Secret security file. This information is added into Identity Manager at the appropriate level in the endpoint tree. Because the information that is gathered is live data from CA Top Secret, performing the explore and correlate function on a stable, non-changing CA Top Secret data store is recommended. To ensure the information recorded into the Provisioning repository is current and up-to-date, start this process in the evening when there is no activity occurring in the CA Top Secret data store.

With the potential for many thousands of records, the explore and correlate function could run for many hours. As a result, the function for adding CA Top Secret objects runs as a separate task in Identity Manager. This enables you to continue working in Identity Manager while the exploration and correlation of the CA Top Secret security information continues.

The following are three different options for the Explore and Correlate function of a CA Top Secret endpoint:

- No Check - No Action
- Gray check with gray background - explores, correlates, and updates the database section by section
- Black check with white background - explores, correlates, and updates all of the sections of a database as one database in entirety.

The result section and message log reports information consistent with each choice. The difference is most evident with the initial explore and correlate of a CA Top Secret Endpoint.

How Exploration and Correlation Works

It is important to understand the steps of the explore and correlate function so that you can determine how and when to use it in your organization. There are three steps to performing the explore and correlate function. The first step gathers the information from the CA Top Secret security file. The length of time this step takes is a constant that is based on the number of ACIDs, profiles, and groups on your z/OS system. Because the number of changes that have been made to the CA Top Secret data store outside of Identity Manager cannot be determined, a complete list of objects is always generated.

The second step compares the information in the CA Top Secret security data store to the information in the Identity Manager repository and merges the results. This consists of adding any new account information into the Identity Manager repository and removing accounts that were deleted outside of Identity Manager. The initial exploration takes a long time. However, in each subsequent execution of the exploration, the length of time it takes is directly proportional to the amount of activity and changes made outside of Identity Manager.

The third step correlates the information gathered from the CA Top Secret security file into the Provisioning repository. This step is only performed if the Correlate accounts with global users check box has been selected in the Explore and Correlate Endpoint dialog. We recommend that you also select the Create global users check box at the same time since these two tasks are closely related. The length of time that this step takes, like the previous step, increases with the number of the new objects being added, modified, or deleted.

When to Perform Exploration and Correlation

Two factors dictate when and how often to run the explore and correlate function. The first factor is based on the amount of activity that takes place in the CA Top Secret security data store outside of Identity Manager. Usually, this can follow a predetermined schedule established in your organization. Periodically, the security officer may request certain Identity Manager administrators to perform this function because they have added a key CA Top Secret profile that may be needed for immediate use, or because they are aware of a high volume of changes that may have occurred.

The second factor is whether your organization has implemented the LDAP Directory Services (LDS) option for CA Top Secret security (available in Version 5.3 and above). If you have implemented this option, the requirement to run the explore and correlate function is dramatically decreased because each modification on the z/OS host is automatically propagated into the eTrust Admin repository, if configured on your mainframe. For those organizations that continue to be z/OS-centric, we recommend the implementation and use of the LDS option.

Note: If running an explore and correlate on multiple TSS directories, you should have RDT2BYTE activated or deactivated on all directories. Your user defined fields on the system with RDT2BYTE inactive will be incorrect if all are not specified.

Setting Up Role-based Administration

The basic principle of user administration is allowing the user to access only the resources they need to perform their job. You should also base access rights on a user's responsibility or job role, rather than on the individual user ID. By using these simple rules, you ensure that you are building a structure that does not require constant administrative attention.

CA Top Secret security lets you meet this objective by building profile records that reflect the many job roles and responsibilities in your organization. To start your security implementation, map job roles and responsibilities to CA Top Secret security profiles, and then map those profiles to Identity Manager account templates. You can then implement an Identity Manager provisioning role with the appropriate account template that maps into your CA Top Secret profile. This results in an ACID that has the appropriate CA Top Secret permissions for that job function.

If you spend time building a good profile structure before you start protecting resources, you will have a well-designed security system and simplified administration. Failing to build a strong profile structure results in custom administration based on individual access rights. The load on your security administrators increases as you protect more and more resources. As a result, users have access to the resources that they need to do their job function in addition to other resources that reflect previous job functions and responsibilities. Consequently, the administration tasks continue to increase over time.

Understanding Account Templates

An account template is used to control the capabilities and attributes of an account on any endpoint. The CA Top Secret policy contains over 150 attributes that you can set. These attributes control everything from what department a user is in, the capabilities the user has when logged on to TSO, and the user's CA Top Secret profiles. Basically, most attributes that you can define for an ACID can be established through an account template.

Account Templates are used to define the attributes to use when creating new accounts and also to ensure that existing accounts conform to established account templates. This is accomplished when you synchronize an account template to the accounts assigned to the account template. Since this concept is vital when implementing security, it is important to fully understand how this functionality works.

Understanding Synchronization

Synchronization can be either weak or strong. You can set the type of synchronization you want on the policy. For the CA Top Secret Connector, the default action is set to strong synchronization. Strong synchronization automatically enforces the attributes on the account template to the account. Any attributes that are enabled at the account level and disabled at the account template level are disabled at the account level after synchronization completes. For more information, see *Propagating Changes to Accounts Automatically* in the *Administrator Guide*.

When weak synchronization is performed, the attribute in the account and the attribute in the account template are compared. For example, for Boolean attributes, if an attribute is turned on in the account template and off in the account, it is turned on at the account level. If the attribute is turned off (a value of 0) at the account template level and on (a value of 1) at the account level, the attribute remains on after synchronization occurs. Consequently, the greater value of the attributes, whether in the account template or in the account, takes precedence and is applied.

This same rule applies for string values. For example, if the value for TSOPROC is equal to PROC111 in the account template, and the value that an account has is equal to PROC999, when synchronization occurs, the value on the account remains the same because PROC111 is less than PROC999.

Assigning multiple policies to a role increases the complexity of the merging algorithm, even when using strong synchronization. If multiple policies are assigned to a role, the same logic is used to determine the value applied to the account. For example, if account template1 has a value of CAPROC with strong synchronization selected, account template2 has a value of DAPROC with strong synchronization selected, and the account already has ACPROC, when synchronization completes, the account is updated to DAPROC (the greatest of the three values). This algorithm also applies when a global user is assigned to multiple provisioning roles and each provisioning role has an assigned account template.

Capability Attributes

An account template can contain two types of attributes: initial or capability. An initial attribute is used only when a CA Top Secret CREATE is done. A capability attribute affects processing when a CA Top Secret CREATE is done and when synchronization is performed. The CA Top Secret security capability attributes appear as boldface characters in the account template property sheet.

In Identity Manager, when a CREATE is done and multiple account templates are assigned to the global user, the product performs a CREATE using the values of the first account template and issues successive MODIFY statements to implement the values of the subsequent account templates. This becomes important with CA Top Secret because, at a minimum, each ACID must be assigned a department. We recommend that you enter a department value in the global user attributes and that each account template use the %UDEPT% rule string to obtain the value from the global user record. This ensures uniformity during the CREATE and also ensures that the ACID does not get moved between departments during subsequent MODIFY statements.

The Department field does not appear in the account template as a required field but may be required for a successful CREATE. You should always verify the use of your account templates under as many conditions as will exist in your Identity Manager implementation.

Using Account Templates

The effective use of account templates in Identity Manager is critical to establishing a secured environment. We recommend that you use the same values in a CA Top Secret account template that were used in the CA Top Secret profile record (or multiple profile records). This way, the security officer makes a change in one place—the CA Top Secret profile record. The change is effective immediately for every user assigned to use that profile, synchronization does not need to be performed in Identity Manager, and CA Top Secret security is only updated once.

Using Profiles with Account Templates

When using CA Top Secret profiles in an Identity Manager account template, there are additional issues to consider. First, because the order of profiles on an ACID are critical to the permissions of that ACID and because LDAP does not provide a guarantee of that order, we recommend entering all profiles on a single account template when the order is imperative. Second, when Identity Manager processes account template changes, the current profiles from the ACID record are removed and the profiles from the account template are added. If you have modified the list of profiles attached to the ACID outside of Identity Manager, these modifications are lost. Finally, the Profile drop-down list is generated based on the exploration of the z/OS host. If changes are made through CA Top Secret or by adding an ACID of TYPE=PROFILE in Identity Manager, you must perform an explore at the profile level to ensure that the information in Identity Manager and CA Top Secret coincides.

There may be instances where profiles do not exist, or it may not be advantageous to establish one for small variations in access authorities, or you may want to set values that are not supported in a profile record. If individual attributes are needed or are going to be used on an Identity Manager account template, we recommend that you establish policies based on the different segments in CA Top Secret. That is, an account template can control TSO information, another may control OMVS, and another may set MISC or PROFILE values. Combined, these policies completely define an ACID's attributes.

The advantage of using this methodology is that it provides a flexible interface to update certain key information in multiple ACIDs with the least amount of overhead. For example, one account template has values that control TSO authorities and is called TSouser. Another is called TSOProgrammer and has different, expanded TSO authorities. If your provisioning role has the TSouser account template assigned to it, but you want to provide the users with increased capabilities, you can change the provisioning role to point to the TSOProgrammer account template, and then synchronize the provisioning role. This grants the authorities for TSO based on the TSOProgrammer account template to the users assigned to that provisioning role. If this is a temporary change to last only the length of a project, change the provisioning role back to point to the TSouser account template when the project is complete and synchronize again. This results in removing those attributes and reapplying the original values. A couple of mouse clicks and you have potentially updated numerous ACIDs with a standard set of attributes and the only attributes affected were those of TSO.

As a general rule, it is always a good practice to run a Check Sync command before doing the actual synchronization. This allows you to validate that the changes you made are the only changes to send to CA Top Secret and you are not affected by a change made to an account template that was not synchronized at an earlier date.

Using Model ACIDs

User administration also includes decisions about what applications users need access to. Some of these access rights can be defined through the profile definitions, but others have to be defined through the ACID definitions. CA Top Secret allows you to define user definitions to use as a model when you add a new user to a provisioning role in Identity Manager. These are called Model ACIDs. Their use is another effective way to reduce both network and security system overhead. By specifying a Model ACID on an account template in the Using field, you can effectively reduce the magnitude and complexity of the commands issued to CA Top Secret when creating new ACIDs.

Authorizing Account Template Administration

Identity Manager provides *admin profiles* that give administrators certain types of access and privileges to manage objects in a domain. These profiles ensure that delegated administrators have the authority to change only what they are authorized to change. We recommend that only the administrators involved in account template management be allowed to modify Identity Manager account templates. This can be established for all account templates or for only CA Top Secret account templates depending on your needs. You can also use admin profiles to distribute account template management based on platform.

RACF Connector

The Resource Access Control Facility (RACF) Connector lets you administer accounts, groups, and resources on RACF systems and provides a single point for all user administration by letting you do the following:

- Register endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage RACF accounts using RACF-specific account templates
- Change account passwords and account activations in one place
- Synchronize global users with their provisioning roles or synchronize global users' accounts with their account templates
- Assign RACF account templates to each of your RACF endpoints
- Use the default endpoint type account template to create accounts with the minimum level of security needed to access a RACF endpoint
- Create and manage RACF groups and permissions
- Generate and print reports about RACF accounts

How to Configure Your RACF System

Once the RACF connector has been installed with the Provisioning Server, you must configure your system to communicate with the connector.

To configure your RACF system

1. Install the CA LDAP Server for z/OS on your RACF system.
2. Review the CA LDAP Server for z/OS configuration options.

Step 1. Install the CA LDAP Server for z/OS

The CA LDAP Server for z/OS provides the communication mechanism for the RACF connector. This product is a free offering from CA and can be downloaded from <http://support.ca.com>. Once downloaded, refer to the *CA LDAP Server for z/OS Installation Guide* for information and instructions on how to install it.

Step 2. Review the CA LDAP Server

Once all CA LDAP Server installation steps have been completed and your CA LDAP Server is started, it will be ready to support administration for the Identity Manager RACF Connector. However, you may need or want to setup additional configuration options for the CA LDAP Server in order to provide additional functionality for the RACF Connector. For more information on all available configuration options, see the chapter titled "RACF_ETA Backend" of the *CA LDAP Server for z/OS Administrator Guide*.

Securing Provisioning Server Communication to the CA LDAP Server

All communication between the Provisioning Server and the CA LDAP Server for z/OS can be encrypted using SSL (Secure Socket Layers).

To establish communication

- Setup your CA LDAP Server for z/OS to use the Server Mode for SSL connections. For information on how to configure this, see the chapter titled, "CA LDAP Server Using Digital Certificates", in the *CA LDAP Server for z/OS Administrator Guide*.
- Turn on SSL support within the Provisioning Server for your RACF endpoint. To do this, bring up the properties of your RACF endpoint using the Provisioning Manager. In the section entitled 'Mainframe LDAP Server Information', enable the check box entitled 'Use Server-side SSL' and click Apply. Now, all communication to the configured CA LDAP Server will attempt to use an SSL connection, and will fail and provide an appropriate error message if SSL cannot be established.

RACF Support for FIPS and IPv6

For this release of Identity Manager, the RACF Connector supports IPv6 but not FIPS.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire a RACF Endpoint

Acquiring an endpoint is the first task you must perform before you can administer a RACF system. When acquiring an endpoint, perform this procedure.

From the Endpoint type task view

1. Register the server as an endpoint in Identity Manager. This phase is performed by adding a new endpoint under the RACF Endpoint type in Identity Manager.

Use the RACF Endpoint property sheet to view or customize a RACF system. During the registration process, Identity Manager identifies the RACF system you want to administer and gathers information about it.

2. Explore the objects that exist on the endpoint.

After registering the server in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all RACF accounts, groups, and permissions. You can correlate the accounts with global users at this time or you can correlate them later.

3. Correlate the explored accounts with global users.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the user ID with each existing global user name. If a match is found, Identity Manager associates the RACF account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the user ID with each existing global user's full name. If a match is found, Identity Manager associates the RACF account with the global user. If a match is not found, Identity Manager performs the next step.
- c. If the Create Global Users as Needed button is checked, Identity Manager creates a new global user and then associates the RACF account with the global user. If the Create Global Users as Needed button is unchecked, Identity Manager performs the next step.
- d. Identity Manager associates the RACF account with the [default user] object.

Register RACF Endpoints on Windows Machines

To register RACF endpoints on a Windows system, perform this procedure.

From the Endpoint task view

1. Select RACF Endpoint from Object Type
2. Click the New button. You will be required to fill in the following information:
 - **Endpoint Name** specifies a name to refer to the new RACF endpoint.
 - **Mainframe LDAP IP Address/Machine Name** specifies the IP address or machine name of the RACF managed system where the CA LDAP Server is configured and running.
 - **Mainframe LDAP Port** specifies the port number that you specified during the CA LDAP Server for z/OS install. If you are not sure of the Mainframe LDAP Port, see the section "Checking your CA LDAP Server for z/OS Configuration Information".
 - **Mainframe LDAP Suffix** specifies the suffix to use for this endpoint. This combo box is automatically populated with all valid and available suffixes when you click the "Get Suffixes" button. Suffixes can be retrieved once valid values have been provided for the Mainframe IP Address/Machine Name and Mainframe LDAP Port fields.

Note: If, after clicking "Get Suffixes", there are no suffixes showing, you are either pointing to the wrong CA LDAP Server or the CA LDAP Server is not configured for use by Identity Manager.
 - **Admin User** specifies the RACF Userid of an administrator to be used from within Identity Manager to manage the RACF system.
 - **Password** specifies the password of the RACF Userid above.
 - **Confirm Password** specifies the same password as above for confirmation.

Note: After you click OK, the Provisioning Server attempts to establish a connection with the CA LDAP Server for z/OS at the IP address and port supplied, as well as, validating the Admin user and Password values supplied. An appropriate error message is displayed if this connection fails. A Global User is inserted in the Provisioning Server with Domain Administrator authority using the ID and password supplied. This Global User should now be used to administer this RACF endpoint.

Register RACF Endpoints on Solaris Machines

To register IBM RACF endpoints on a Solaris machine, use the batch utility, `etautl`, to define a RACF endpoint by specifying a `directoryName`, Mainframe LDAP IP Address, Port, and Suffix. For example,

```
etautl -u USERID -p PASSWORD add 'eTNamespaceName=RACF,dc=DOMAIN,dc=eta' eTRACDirectory  
name='DIRECTORYNAME' eTZOSLDAPIPAddress=IPADDRESS eTZOSLDAPPort=PORT  
eTZOSLDAPSuffix=SUFFIX
```

where,

DIRECTORYNAME

Specifies the name you desire for this endpoint.

IPADDRESS

Specifies the IP Address or Machine name of the RACF system where your CA LDAP Server for z/OS is running.

PORT

Specifies the port number the CA LDAP Server is using,

SUFFIX

Specifies a valid suffix configured for this CA LDAP Server operating in `im` naming mode. (See, "Chapter 3: Configuration Options", in the *CA LDAP Server for z/OS Administrator's Guide* for information on the `naming_mode` option.)

Checking Your CA LDAP Server for z/OS Configuration Information

To view all pertinent information regarding your CA LDAP Server and its configuration, issue a `STATUS` command from the mainframe console where your CA LDAP Server is running. The `STATUS` command provides information, such as version information, port number, configured databases, and suffixes. For more information on the `STATUS` command, see "Chapter 2: Startup Options" section, "Console Interface" in the *CA LDAP Server for z/OS Administrator Guide*.

RACF Provisioning Roles and Account Templates

The RACF Default Policy, provided with the RACF Connector, gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

RACF Accounts

When synchronizing RACF accounts, strong synchronization is always the default.

RACF Groups

Groups allow users to share common access to the resources on an endpoint.

Like the UNIX file, `unix/etc/group`, a RACF user can belong to many groups on the RACF system but is defined to one primary group called the Default Group. In Identity Manager, all groups, including the Default Group, appear as provisioning roles in the browser. In Identity Manager, all groups, including the Default Group, appear as provisioning roles in the browser.

By default, RACF systems use a root group called SYS1. This root is defined as the user's default group if no other group is specified. The user's account appears under the role (default group), but the user also appears under each group that the user is defined to.

Identity Manager lets you create and maintain RACF groups using the Endpoint type task view. Use the RACF Group property sheet when managing your groups.

TSO Alias Support

The RACF connector also supports the creation of a TSO Alias when a user is created or modified with TSO access being granted. The alias value is always the value of the RACF Account being created or modified. To enable this support, see the chapter titled, "RACF_ETA Backend" in the *CA LDAP Server for z/OS Administrator Guide* for more information.

Permissions

Permissions are *resources* in RACF. You can view permissions from any RACF account or endpoint. Identity Manager lets you create and maintain RACF permissions using the Endpoint type task view. Use the RACF Permission property sheet when managing your permissions.

Note: The following permission classes are **not** supported by the RACF connector: DIGTCERT, DIGTRING, and DIGTNMAP.

RACF Program Exits

The RACF connector supports the use of Program Exits which are incorporated as 'Common Exits'. Program Exits provide you with the capability to perform certain actions before or after an account is created, modified or deleted from Identity Manager. These exits can be referenced either on the Endpoint property page to execute custom code on a single endpoint, or on the Account Template property page to execute custom code on multiple endpoints. Actions might include native CA DSI Server for z/OS (CA Distributed Security Integration) or CA LDAP Server for z/OS commands in order to modify account privileges or access to resources on the RACF system.

To see a sample program exit, refer to the OS390 subdirectory under the Identity Manager Templates directory.

For more detailed information about how to write program exits, see the Programming Guide for Provisioning.

Note: If you are upgrading from a previous release and use exits, those exits should be reviewed. There is the potential that by using CA LDAP Server for z/OS-specific configuration options, you can remove existing exits. Even if the exits cannot be removed completely, the number of operations they need to perform, might be reduced. A detailed review of the exits will let you determine if they can or should be recoded to use the CA LDAP Server instead of the CA DSI Server. While this is not required, if possible, exits should be migrated to use the CA LDAP Server.

RACF Conventions

Use the following RACF conventions in your etauil commands:

- The endpoint type name (eTNamespaceName) is RAC endpoint type
- The endpoint type prefix is RAC. Therefore, the RACF class names are:
 - eTRACDirectory for an endpoint
 - eTRACPolicyContainerName for an account template container
 - eTRACPolicy for an account template

Proxy Configuration

The RACF Endpoint page contains a section where clients can configure a Proxy administrative ID and password to be used for user password changes from the SAWI. When configured, this ID and password is used to issue the password change request for the SAWI user to change their password. This is helpful and needed if a SAWI user cannot supply a password (for example, the password is forgotten) or their password is expired and RACF and they cannot be authenticated. When using a proxy administrative ID, standard RACF security rules apply (for example, scoping) and password syntax checking specified in the RACF System Options Password field is enforced.

Note: The check boxes on the Endpoints Setting tab are for legacy purposes only. With Identity Manager r12, you can perform proxy configuration and administration support from the Identity Manager Self-Service interface.

Proxy Administration Support

You can configure a proxy ID for all tasks accomplished within Identity Manager. Previously, a proxy ID could only be configured for use with requests generated from the SAWI interface. This proxy ID is maintained on the main Endpoint page in the Proxy Administration Configuration section. The proxy ID can be used for any type of Identity Manager request against supported objects, and for any Identity Manager Administrator that is logged on.

Note: The enhancement is only recommended to use after careful consideration (and preparation) of the following consequences:

1. Any Global User (with the proper privileges provided within Identity Manager) is able to administer RACF Userids, Groups, and Permissions under the configured proxy ID. Any mainframe security product scoping is lost; only the scoping of the proxy ID is enforced.
2. As mentioned above, Identity Manager security settings are now the only point of enforcement against a Global User manipulating mainframe security data.
3. Any reports or auditing methods against administration of your mainframe security data that originate from the mainframe is now compromised; the only ID that shows up for any administration that occurred from Identity Manager is the configured proxy ID.

4. If the proxy ID's password changes on the mainframe, the password must be changed on every Endpoint Page within the Provisioning Manager that it is configured for.

By default, the Connector operates in the same mode as in past releases; the logged-on Global User and their password are used for submitting any requests destined to the mainframe security product. The common endpoint page entitled Endpoint Settings provides two checkbox controls under the description Administrator Credentials that control the three possible settings:

Use logged-in Administrator's credentials

Default setting. Indicates that the logged-in Administrator (Global User) is used as the credentials for ALL requests, even from the SAWI.

Use proxy for SAWI changes

Indicates that the logged-in Administrator (Global User) is used as the credentials for all requests EXCEPT for requests from the SAWI interface. The proxy ID credentials (if available) are used for requests coming from the SAWI interface.

Use proxy for ALL requests

When no checkbox is checked, this indicates that the proxy ID credentials (if available) are to be used for ALL requests.

When any request occurs from Identity Manager, these settings are checked against the endpoint where the request is targeted. If, based on the endpoint settings and the type of request (SAWI or otherwise), proxy credentials are to be used, the credentials that are defined for that endpoint are retrieved and used for the request. In the case where endpoint credentials are supposed to be used, but no credentials exist (either Proxy ID or password contains no value), the proxy credentials are not used for the request and the request proceeds using the logged-in Administrator (Global User) credentials.

Note: Proxy IDs must be what are referred to as 'logon-able' user IDs. That means they must be 7 bytes in length or less. 8 byte user IDs are no longer valid.

Note: The check boxes on this tab are for legacy purposes only. With Identity Manager r12, you can perform proxy configuration and administration support from the Identity Manager Self-Service interface.

Logging

To view logging information regarding credential lookups, enable logging for the endpoint on the Logging endpoint page and enable the eTrust log Connector, specifying all logging levels. This provides all endpoint credential lookup information (successes and failures) into the standard eTrust log.

Chapter 7: UNIX Connector

This section contains the following topics:

[UNIX Connector](#) (see page 137)

The following connector is considered to be a UNIX connector: UNIX.

The information in this chapter describes how to install and manage this connector.

UNIX Connector

The UNIX option provides a single point for all user administration by letting you do the following:

- Register endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage UNIX accounts using UNIX-specific account templates
- Change account passwords and account activations in one place
- Synchronize global users with their roles or synchronize global users' accounts with their account templates
- Assign a UNIX policy to each of your UNIX endpoints
- Use the default Endpoint Type policy to create accounts with the minimum level of security needed to access a UNIX directory
- Create and manage UNIX groups
- Generate and print reports about UNIX accounts and groups

This connector is managed using the Connector and agent installation process. For more information and requirements, click [here](#) (see page 21).

This connector can also be managed using the [Connector and C++ Server](#) (see page 23) installation process as well.

Product Interchange Format (PIF) Installation Procedures

After installing the UNIX Connector, you must install and configure the UNIX agents and the CAM Service.

Note: Each package can be installed using the `lsm` command using an included `@pif` file. If the `lsm` program is not installed or available, the script installation (`.sh`) method must be used. The `.sh` script includes the `lsm` program, and, when invoked, will first install `lsm`, and then install the product.

The UNIX agents and the CAM service can be installed by one of the following installation methods:

- Interactive installation
- Silent installation

Interactive Installation

Interactive installation includes the following steps:

1. Mounting the CD-ROM.
2. Selecting the required installation script.
3. Starting the setup wizard

Mount the CD-ROM

You must perform the following procedure to mount the CD-ROM.

From your CD-ROM driver

1. Insert the Identity Manager distribution CD.

Note: If your UNIX computer does not have a CD-ROM drive attached, you can insert the CD into a computer and use ftp, sftp, scp, Unicenter Data Transport, or Samba, depending on your network environment, to transfer the installation files to a temporary directory on your UNIX computer.

2. Login to your UNIX computer as 'root'.

Important! The Identity Manager UNIX Remote Agent must be installed under the 'root' user. To collect information from database files, and update UNIX Security database files, a certain level of access is required. By default, the 'root' user is the only user that has the requisite privileges.

3. Some UNIX systems, for example, Solaris and Linux, are configured to mount the CD-ROM disks automatically when they are inserted in the drive. If the CD-ROM is not mounted automatically, you must mount the CD-ROM disk before CDs can be accessed.

If a mount point to the CD-ROM disk does not already exist, create one by entering the following command:

```
# mkdir /cdrom
```

On a Solaris system that does not have the Volume Manager active, you cannot create /cdrom. Make sure that you do not try to mount a CD-ROM disk that has already been mounted automatically. If you do, you will get an error message.

The format of the CD-ROM is ISO 9660 RockRidge extension.

Note: You can insert the CD into any Windows CD-ROM drive and copy the installation files to the target computer using FTP. When doing so, make sure that the "binary" mode is used and the file names are kept unchanged, that is, case sensitivity and blanks must persist.

To mount the CD-ROM, enter the mounting command for your operating system according to the following table:

Platform	Mounting Command
AIX 5.2 and later	Automatic mounting by default. When disabled, use <code>cdmount</code>
AIX 5.1 and previous	<code># mount -r -v cdrfs device_path /cdrom</code> Example: <code>device path /dev/cd0</code>

Platform	Mounting Command
HP-UX	Start the service: <code>/usr/sbin/pfs_mounted</code> and then start the service: <code>/usr/sbin/pfsd</code> and <code>/usr/sbin/pfs_mount -x unix -t rrip device_path /cdrom</code> Example: <code>device_path /dev/dsk/c1t2d0</code>
Solaris	Automatic mounting by enabled Volume Manager (vold). If not mounted automatically, use the following command: <code># mount -F hsfs -r/dev/dsk/c0t6d0s2/cdrom</code>
Linux	The following command must be used for CDs that have not been auto-mounted already: <code># mount -t iso9660 device_path /cdrom</code> Example: <code>device_path /dev/cdrom</code>
Tru64 4.0	<code># mount -t cdfs -o rrip device_path cdrom</code> Example: <code>device_path /dev/rz4c</code>
Tru64 5.0	<code># mount -t cdfs -o noversion device_path /cdrom</code> Example: <code>device_path /dev/disk/cdrom0c</code>

Select the Installation Script

The installation packages for the various platforms can be found on the CD-ROM in their own subdirectory. Each package provides an lsm installation method (using a `.@pif` file) and a script installation method (using a `.sh` script), in case lsm is not available. For example, when no other CA products are installed on the system.

Platforms	CD-ROM Folder
AIX	<code>/cdrom/RemoteAgent/UNIX/AIX</code>
HP-UX	<code>/cdrom/RemoteAgent/UNIX/HPUX</code>
Solaris Sparc	<code>/cdrom/RemoteAgent/UNIX/Solaris</code>
Solaris Intel	<code>/cdrom/RemoteAgent/UNIX/SolarisIntel</code>
Linux ix86	<code>/cdrom/RemoteAgent/UNIX/Linux</code>
Linux s390	<code>/cdrom/RemoteAgent/UNIX/LinuxS390</code>
Tru64	<code>/cdrom/RemoteAgent/UNIX/TRU64</code>

If you want to copy the Installation files from another machine to the UNIX computer, select the relevant folder for the platform and copy all its contents.

Start the Installation Wizard

Perform the following procedure to start the installation wizard.

To start the installation wizard

1. Switch to the directory where the installation files are located.

Example for AIX:

```
# cd /cdrom/UNIX/AIX
```

2. Depending on the installation method that you want to use, enter either of the following commands to start the setup wizard:

```
sh IdentityManager.platform_name.sh
```

```
lsm -i IdentityManager.platform_name.@pif
```

Examples for AIX:

```
# sh IdentityManager.AIX.sh
```

```
# lsm -i IdentityManager.AIX.@pif
```

lsm provides a variety of installation options that can be viewed by typing lsm -? in the command line.

More Information

[Installation Options](#) (see page 141)

Installation Options

The installation options are described in the following table:

Installation Option	Description
lsm -I product file [-r response file] [-F]	Installs the product. A response file can be added to customize unattended installation. The switch '-F' performs a forced installation and prevents the backup of the previous version of the product. Example: lsm -i test-product.Any.@pif

Installation Option	Description
lsm -e <i>product name</i> [-s]	Removes the installed product. Switch '-s' runs the uninstallation in unattended mode. Example: lsm -e test-product
lsm -l [-S]	Lists all installed products or shared components (-S).
lsm -A <i>product name</i> -d <i>product file</i> [-o]	Creates a backup from the installed product. Switch '-o' overwrites an existing product file.
lsm -c <i>product name</i>	Checks the installed products consistency.
lsm -q <i>product name</i> [-l]	Shows the content of the product file. Switch '-l' shows a long list including all product files.
lsm -Q <i>product file</i> [-l]	Shows the content of the product file. Switch '-l' shows a long list including all product files.
lsm -a <i>product file</i> -r <i>response file</i>	Runs the installation dialogs and creates a response file with the entered values. The product is not installed.
lsm -v	Prints the version of the Installer being used.

Silent Installation

In some cases, for example, Unicenter Software Delivery, it is important to have a software product that installs automatically without any user interaction. The `lsm -i` command can be executed with the option `-r response file`, and additional options, to install the UNIX Remote Agent without any questions being asked. You must provide the full path to a response file after the `-r` option.

Important! You can also perform a silent installation using a shell script if you are installing or upgrading on a computer where `lsm` is not installed.

The following example shows a typical response file:

```
PATHeTrustAdmin=/opt/CA/IdentityManager/ProvisioningUnixAgent
IM_INSTALL=1
OWNERroot=root
GROUPsys=sys
```

The following example shows how a response file is created using `lsm`:

```
lsm -a IdentityManager.Linux.@pif -r IdentityManager.Linux.rsp
```

The following example shows how such a response file is used:

```
lsm -i IdentityManager.linux.@pif -r IdentityManager.linux.rsp
```

The following example shows how a response file is created using a shell script:

```
sh IdentityManager.Linux.sh -r IdentityManager.Linux.rsp
```

More Information

[Installation Options](#) (see page 141)

Silent Installation Notes

The following is a list of important notes:

- The current installation default path is /opt/CA/IdentityManager (the previous path was /opt/CA/eta).
- If the same CAM version and build level is already installed on the target machine, CAM will not be re-installed.
- If a previous CAM version and build level is already installed on the target machine, CAM will be upgraded using the installation path of the current installation, which is stored in the following file:

`/etc/catngcampath`

If a previous Identity Manager Remote Agent is already installed on the target machine, the Remote Agent will be upgraded using the installation path of the current installation, which is stored in the following file:

`/etc/catngdmopath.tng`

- If, on the target machine, the DISPLAY variable is set and a JAVA VM is installed, the installation will run in graphical mode.
- In VT100 mode, the terminal must provide a resolution of 80 (columns) x 24 (rows) or higher.
- On a UNIX machine with double-byte characters, CAM must be started with a shell having the "locale" set to C/Posix:

```
`cat /etc/catngcampath`/bin/camclose
LANG=C
export LANG
`cat /etc/catngcampath`/scripts/rc
```

- If you install the UNIX agent using Telnet, make sure that the environmental variable TERM is set to VT100.

Install the Identity Manager UNIX Remote Agent

Perform the following procedure to install the Identity Manager UNIX remote agent.

To install the Identity Manager UNIX remote agent

1. Locate the Identity Manager Provisioning Components download or other media.
2. Start the Product Explorer.
3. Select Install Products, Endpoint Agent Software
4. Choose Agent for UNIX
5. Follow the onscreen instructions to complete the installation.
6. The Welcome dialog that shows the UNIX Remote Agent version appears. View the dialogue.
7. Click Next. The Select Installation directory dialog appears. Enter a valid installation directory.

The product is installed under the specified installation directory.

`/opt/CA/IdentityManager/ProvisioningUnixAgent`

This is the name of the actual directory where you want to install the Identity Manager UNIX Remote Agent. All files are placed in this directory or its subdirectories. You can change the name of the installation directory or it will be created if it does not already exist.

Note: If you run this procedure on a computer on which an older version of the Identity Manager UNIX Remote Agent is installed, the old installation path is read from the `/etc/catngdmopath.tng` marker file and set as the Installation directory.

During an update installation, the product installation directory cannot be modified.

8. Click Next to continue.

The Summary dialog appears. Check the following installation parameters:

```
PATHIdentity Manager=/opt/CA/IdentityManager/ProvisioningUnixAgent
IM_INSTALL=1
OWNERroot=root
GROUPsys=sys
```

9. Click Install product to run the installation.

10. View the installation log.

The following lines are logged by the installation:

Installing Dependency - CA Installer [1/2]...

Installing Dependency - CA Installer [2/2]...

Preparing next Installer phase

Executing post interview phase

Checking package dependencies

Checking disk space

Installation product "ca-dsm-sd-installer", version "4.3.x.x"

=====

++ Call script "scripts/preinstall_installer.sh"

++ Script executed successfully

++ Installation component "preinstall"

++ Component "preinstall" installed successfully

++ Installation component "base"

++ Component "base" installed successfully

++ Installation component "base_root"

++ Component "base_root" installed successfully

++ Installation component "base_shared"

++ Component "base_shared" installed successfully

++ Installation component "conf"

++ Component "conf" installed successfully

++ Installation component "man, ENU"

++ Component "man, ENU" installed successfully

++ Call script "scripts/postinstall_installer.sh"

++ Script executed successfully

Job executed successfully

Installing Dependency - CAM...

Installing Dependency - ETPKI...

Preparing next Installer phase

Executing post interview phase
Checking package dependencies
Checking disk space

Backup product "ca-cs-utils", version "11.0.x.x"

=====
=====

++ Backup component "preinit_csutils"
++ Component "preinit_csutils" saved successfully
++ Backup component "csutils"
++ Component "csutils" saved successfully
++ Backup component "csutils_platform_files"
++ Component "csutils_platform_files" saved successfully
++ Backup component "csutils_libv2"
++ Component "csutils_libv2" saved successfully
Job executed successfully

Reinstallation product "ca-cs-utils", version "11.0.x.x"

=====
=====

++ Call script "/bin/sh csutils/scripts/prein_csutils.sh"
++ Script executed successfully
++ Reinstallation component "preinit_csutils"
++ Component "preinit_csutils" installed successfully
++ Reinstallation component "csutils"
++ Component "csutils" installed successfully
++ Reinstallation component "csutils_platform_files"
++ Component "csutils_platform_files" installed successfully
++ Reinstallation component "csutils_libv2"
++ Component "csutils_libv2" installed successfully
++ Call script "/bin/sh csutils/scripts/install.csutils"
++ Script executed successfully
Job executed successfully

Installation product "ca-cs-etpki", version "3.2.x.x"

=====

```
++ Call script "/bin/sh pifscripts/prein.etpki"
++ Script executed successfully
++ Installation component "preinit_etpki"
++ Component "preinit_etpki" installed successfully
++ Installation component "cs-etpki-base"
++ Component "cs-etpki-base" installed successfully
++ Installation component "cs-etpki-lib"
++ Component "cs-etpki-lib" installed successfully
++ Call script "/bin/sh pifscripts/postin.etpki"
++ Script executed successfully
Job executed successfully
```

Installing Identity Manager

Preparing next Installer phase

Executing post interview phase

Checking package dependencies

Checking disk space

Installation product "IdentityManager", version "12.0.x.x"

=====

```
++ Installation component "im"
++ Call script "scripts/imscript.sh preinstall"
++ Script executed successfully
+++ Call component script "scripts/imscript.sh postinstall"
+++ Script executed successfully
++ Component "im" installed successfully
++ Installation component "preinstall, ENU"
++ Component "preinstall, ENU" installed successfully
Job executed successfully
```

Note: All prerequisite components are installed after the Identity Manager UNIX Agent installer has been executed. This applies to both upgrade and new installations despite the "Cancel" option being selected during the installation process

Grant Access to the Provisioning Server Host

To grant access to the Provisioning Server host on this machine, run the following command:

```
`cat /etc/catngcampath/bin/cafhhost -a hostname
```

where *hostname* is the name or the IP address of the machine hosting the Provisioning Server.

Example for any platform:

```
# `cat /etc/catngcampath/bin/cafhhost -a etradmsrv01
```

Install and Configure the CAM Service for Irix and NCR

The CAM, also known as CAM/CAFT or Unicenter Message Queuing Server, service is used by the connector to transfer information from and to the remote UNIX machines that are managed by Identity Manager.

Installing and configuring the UNIX agent and CAM service on Irix and NCR is done after installing the UNIX Connector, and involves the following steps:

1. Transfer the necessary files
2. Install the CAM service
3. Configure the CAM service
4. Install the UNIX agent
5. Verify the Installation

Step 1. Transfer the Necessary Files

Transfer the necessary files under the /RemoteAgent/Unix/Others directory of your installation CD.

The /RemoteAgent/Unix/Others directory contains the following compressed tar format files:

Tar Format File	UNIX Server Type and Version
IRIX_V1.07_20051107_Build220_16.tar.Z	SGI/IRIX Version 6.5
NCR_V1.07_20051107_Build220_16.tar.Z	NCR/mpras SVR4 Version 3.02

Transfer the agent.ins, agent.tar.Z, and the compressed tar format CAM file from the installation CD to your UNIX server in a temporary directory, for example, "/tmp".

Note: If you use an ftp utility to transfer these files, set the binary transfer option.

Step 2. Install the CAM Service

You must install the CAM Service on your UNIX systems.

To install the CAM Service

1. Log on to your UNIX server as root by using the Telnet or SSH client.
2. Create and change to the directory where you want to install the CAM service, for example /opt/CA:

```
mkdir /opt/CA
cd /opt/CA
```

3. Issue the UNIX tar command to install the CAM Service:

```
zcat /tmp/compressed_tar_format_CAM_file | tar xvf -
```

where *compressed_tar_format_CAM_file* is the name of the compressed tar format file that you transferred.

4. Change to the cam/scripts directory:

```
cd ./cam/scripts
```

5. Issue the following UNIX shell script:

```
./install
```

Note: If the AGENTWORKS_DIR environment variable is set and the \$AGENTWORKS_DIR/services directory exists, the CAM “install” scripts copy the CAM files within this \$AGENTWORKS_DIR/services directory.

Step 3. Configure the CAM Service

To configure the CAM service, perform the following procedure.

To configure the CAM service

1. Log on to your UNIX machine as root by using the Telnet or SSH client.
2. Change to the cam scripts directory:

```
cd `cat /etc/catngcampath`/scripts
```

3. Issue the following shell script:

```
./envset
```

Note: This command must have a space between the two dots.

4. Change to the cam bin directory:

```
cd ../bin
```

5. Issue the following UNIX command:

```
./catthost -a hostname
```

where *hostname* is the name or the IP address of the machine hosting the C++ Connector.

More Information

[Manage the CAM Service](#) (see page 152)

Step 4. Install the UNIX Agent

Perform the following procedure to install the UNIX agent.

To install the UNIX agent

1. Log on to your UNIX machine as root by using the Telnet or SSH client.
2. Change to the directory where you transferred the files, for example, /tmp.
3. Issue the following UNIX command:

```
chmod +x agent.ins
```

4. Issue the following UNIX command:

```
./agent.ins
```

Step 5. Verify the Installation

Issue the following command to ensure that the UNIX Agent is installed on your system:

```
uxsautil -IT host all
```

The results contain information about your UNIX system.

Manage the CAM Service

The CAM Service is a daemon process that you can view, stop, or start on your UNIX server. Typically, the superuser or the system's root user starts the CAM Service.

View the CAM Service

You can perform the following procedure to find out who started the service.

To view the CAM service

1. Log on to your UNIX machine as root by using the Telnet or SSH client.
2. Issue the following UNIX command:

```
ps -ef | grep cam
```

A display similar to the following one appears:

```
root 13822    1 11 11:30:12 ?    0:00 cam
```

```
root 13843 13753  3 11:56:31 pts/5  0:00 grep cam
```

Note: If the system's root user does not start the services, they will appear started, but you will be unable to use them. Identity Manager issues the following message: "Permission denied: user must be root".

Stop the CAM Service

You can stop the CAM service by performing the following procedure.

To stop the CAM Service

1. Log on to your UNIX machine as root by using the Telnet or SSH client.
2. Change to the cam scripts directory:

```
cd `cat /etc/catngcampath/scripts
```

3. Issue the following UNIX command:

```
./envset
```

Note: This command must have a space between the two dots.

4. Change to the cam bin directory:

```
cd ../bin
```

5. Issue the following UNIX command:

```
./camclose
```

Note: This command stops the CAM Service. After stopping this service, you must restart it so Identity Manager can communicate with your UNIX server.

Restart the CAM Service

You can restart the CAM service by performing the following procedure.

To restart the CAM Service

1. Log on to your UNIX machine as root by using the Telnet or SSH client.
2. Change to the cam scripts directory:

```
cd `cat /etc/catngcampath`/scripts
```

3. Issue the following UNIX command:

```
./envset
```

Note: This command must have a space between the two dots.

4. Issue the following command to restart the CAM Service:

```
./rc
```

How to Restart Automatically the CAM Service

If you want to automatically start the CAM Service after rebooting a machine, you can use the init or rc utilities.

To start the CAM Service automatically after rebooting a UNIX server, verify the following:

- Unicenter runtime environment is known to the CAM Service
- Unicenter BIN directory appears in the PATH variable

For example, a typical Start shell script appears as follows:

```
#!/bin/sh
# @(#)install 3.24 10:15:49 98/05/29
# Date Created: Tue Jul 20 11:57:34 WET DST 2004
.
.
.
.
# Start CA Message Queuing Server
su $AGENT_OWNER -c /export/home/cam/cam/scripts/rc
If you add the commands above, the Start shell script appears as follows:
#!/bin/sh
# @(#)install 3.24 10:15:49 04/05/29
# Date Created: Tue Jul 20 11:57:34 WET DST 2004
.
.
.
.
PATH=$PATH:$CAIGLBL0000/bin
export PATH
# Start CA Message Queuing Server
su $AGENT_OWNER -c /export/home/cam/cam/scripts/rc
```

Using the Init Utility

To start the CAM Service using the init utility, add the following line to the end of the `/etc/inittab` file:

```
cam::once:cat /etc/catngcampath/scripts/rc
```

By adding this line, the shell script created when you installed the CAM Service is executed. After it executes, verify that you can view the daemon process.

Using the RC Utility

To start the CAM Service using the rc utility, perform the following steps:

1. Copy the start shell script to the `init.d` directory.
2. Create a shell script under the `rc2.d` sequencer directory by following the rc syntax.

Notes:

- The rc utility is not applicable on IBM-AIX platforms.
- The location of the directories shown previously may be different on each UNIX platform; the directories are normally located under either the `/bin` or the `/etc` directory. For more information, see the documentation for your specific UNIX system.

How to Restrict CAFT Commands

By default, CAFT allows any command to be executed from an authorized host. As the UNIX Connector only needs to run the `uxsautil` command, the CAFT `caftexec` script can be customized to filter commands and to allow only the `uxsautil` binary.

An example of such a script and its configuration file are provided in the

``cat /etc/catngdmopath.tng`/scripts` folder, and can be copied to the ``cat /etc/catngcampath`` folder:

```
# cd `cat /etc/catngcampath`
```

```
# mv caftexec caftexec.back
```

```
# cp -p `cat /etc/catngdmopath.tng`/scripts/caftexec* .
```

Install the CAM and CAFT Encryption Key

Encryption is supported for Win32, AIX, HP-UX, Solaris, and Linux x86 applications. The default and only available encryption algorithm is Triple-DES (168 bits key) with CBC mode.

To install the encryption key

1. Enter the following command at the command prompt to generate your key file:

```
#PATH=`cat /etc/catngcampath`/bin:$PATH
#export PATH
#caftkey -g keyfile password
```

keyfile

Name that you assign to the key file.

password

Password that you assign to the key file.

Note: The caftkey command and attributes are the same for Win32 platforms.

2. Install your Public Key on both CAFT Agent and CAFT Admin boxes using the previously-generated key file by entering the following command at the command prompt:

```
#PATH=`cat /etc/catngcampath`/bin:$PATH
#export PATH
#caftkey -policy_setting keyfile password
```

keyfile* and *password

The values that you specified in Step 1.

-Policy_setting

-i, -m, or blank.

More information:

[policy_setting Options](#) (see page 158)

3. Recycle the CAM Service on each box where you installed the new Key by stopping the CAM service, and starting it again:

```
prompt> camclose      //stop Cam/Caft service and processes
prompt> cam start      //start CAM service and process
```

policy_setting Options

Policy_setting governs the communication between this computer (the local computer) and other computers that have the CAM and CAFT service installed, but may or may not have the CAM and CAFT encryption certificates installed.

The options are as follows:

caftkey -i keyfile password

The -i option specifies Policy -1. This policy lets computers running previous versions of the CAM and CAFT service execute commands on this computer and lets this computer execute commands on those computers.

Policy -1 encrypts messages if the other computer has these certificates installed. This policy does not encrypt messages if the other computer does not have these certificates installed.

caftkey -m keyfile password

The -m option specifies Policy 1. This policy prohibits other computers from executing commands on this computer if they are running previous versions of the CAM and CAFT service without the encryption certificates. This policy also prohibits this computer from executing commands on those computers.

If both computers have the CAM and CAFT encryption certificates installed, but have different Public Key Files installed when Policy 1 is set, the command requests between the two computers fails.

caftkey keyfile password

The blank option specifies Policy 0. This policy is set if no Public Key File is installed, the CAM and CAFT encryption certificates were not installed properly, or if you do not specify a policy setting when you enter the caftkey command. Policy 0 specifies no encryption.

Note: The CAM and CAFT service must already be installed on the computer in your network.

Check the Policy Setting

To see the operational mode of the machine, check the following file:

```
%CAI_MSQ%/ftlogs/dg000
```

UNIX Support for FIPS and IPv6

For release r12 of Identity Manager, the UNIX Connector supports IPv6. FIPS is supported only on Solaris Sparc, Linux x86, HP-UX, and AIX.

UNIX PAM supports IPv6 only.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire a UNIX Server

You must acquire the UNIX server before you can administer it with Identity Manager. In order to acquire a UNIX server, perform the following steps from the Endpoint Type task view:

1. Register the machine as a stand-alone server or as an NIS or NIS+ server.
 - To register a UNIX endpoint as a stand-alone server, select UNIX ETC Endpoint.
 - To register a UNIX endpoint as an NIS or NIS+ server, select NISEndpoint from the Object Type, and then click New.

Use the UNIX Endpoint property sheet to register a UNIX machine. During the registration process, Identity Manager identifies the UNIX machine you want to administer and gathers information about it.

2. Explore the objects that exist on the endpoint.

After registering the server in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all UNIX accounts and groups. You can correlate the accounts with global users at this time or later.

3. Correlate the explored accounts with global users.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the UNIX account name with each existing global user name. If a match is found, Identity Manager associates the UNIX account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the UNIX account description field with each existing global user's full name. If a match is found, Identity Manager associates the UNIX account with the global user. If a match is not found, Identity Manager performs the next step.
- c. Identity Manager associates the UNIX account with the *default user* global user object.

Default Primary Group on Endpoint Property Sheet

A field called Default Primary Group is available to let you select a default Primary Group at the Endpoint level.

Selecting the Character Set on the Directory Property Sheet

When checked, UTF-8 Character Set encoding will be used for values passed on between the Provisioning Server and the UNIX Remote Agent instead of the one used by the Provisioning Server. A combo list box is enabled so that you can select the character set used on the end-point system.

UNIX Groups

The Provisioning Manager lets you create and maintain UNIX groups using the Endpoint Type task view. Use the ETC or NIS Group property sheet to manage your groups.

NIS Netgroups

Netgroups are logical groups of hosts, accounts or both. Use the UNIX NIS Netgroup property sheet when managing netgroups and user netgroup activity.

For more information, see the following:

- *Identity Manager Provisioning Guide*
- The Working with Endpoints, Connector Procedures in the *Provisioning* help

Mandatory Properties for etautil Add Command

If you use the etautil add command to register UNIX endpoints, you must specify one of the following properties:

- For stand-alone UNIX servers, specify the eTETCHost eTETCUnicenterSec, and eTETCUnicenterUser properties. For example:

```
etautil ... add
'eTNamespaceName=UNIX - etc' eTETCDirectory eTETCDirectoryName=exdevsrv
eTETCHost=exdevsrv eTETCUnicenterSec=0 eTETCUnicenterUser=0
```

- For UNIX servers running NIS or NIS+, specify the eTNISHost , eTNISDomainName, eTNISUnicenterSec, and eTNISUnicenterUser properties. For example:

```
etautil ... -u etaadmin -p etaadmin add
'eTNamespaceName=UNIX - NIS-NIS plus Domains'
eTNISDirectory eTNISDirectoryname='nisdomain.com[exdevsrv]'
eTNISHost=exdevsrv
eTNISDomainName=nisdomain.com eTNISUnicenterSec=0 eTNISUnicenterUser=0
```


UNIX Etautil Conventions

Use the following UNIX conventions in your etautil commands:

- The Endpoint Type name (eTNamespaceName) is UNIX - etc for stand-alone servers and UNIX - NIS-NIS plus Domains for NIS and NIS+ servers
- The Endpoint Type prefix is ETC. Therefore, the UNIX class names are:
 - eTETCDirectory for an endpoint on a stand-alone server
 - eTETCPolicyContainer for an account template container on a stand-alone server
 - eTETCPolicy for an account template on a stand-alone server
 - eTNISDirectory for an endpoint on an NIS or NIS+ server
 - eTNISPolicyContainer for an account template container on an NIS or NIS+ server
 - eTNISPolicy for an account template object class on an NIS or NIS+ server

For more information about the etautil command, see the *Reference* help and the *Identity Manager Provisioning Guide*.

More Information

[Distinguished Names](#) (see page 510)

Using Identity Manager Exit Commands

Identity Manager supports the following exit types:

- **Pre Exit** means that the agent executes a user command before it performs its own operation.
- **Post Exit** means that the agent executes a user command after it performs its own operation.

Configuration File

Identity Manager implements the pre-exits and post-exits on the UNIX agents. To trigger user add-on commands, you must define them in the ExitSetup.ini file that is installed by Identity Manager. By default, this file does not activate any specific command.

The ExitSetup.ini file is located in the following directory:

`/opt/CA/IdentityManager/ProvisioningUnixAgent/etc/`

Note: The exact location is specified in the `/etc/catngdmopath.tng` file.

The following table describes the typical contents of the configuration file:

Headers and Variables		Value	Description
[Pre-exit]			Pre-exit section header
Command=		Provided by the user	User command specified with the absolute path
Stop on error=		Yes/No	Yes-specifies that the Identity Manager agent command is not launched if the pre-exit fails No-specifies that the Identity Manager agent command is launched even if the pre-exit fails. This is the default value.
[Post-exit]			Post-exit section header
Command=		Provided by the user	User command specified with the absolute path

Conditions for the execution of the EXIT Commands

The conditions for the execution of the exit commands are as follows:

- For Identity Manager, the execution of a command is successful if its return code (RC) is equal to 0; any other code indicates that the execution failed. This is important because the values that are caught by Identity Manager are processed according to the value of the return code.
- The argument values, which are sent to Identity Manager agents, are also sent to the user program.
- The pre-exit and post-exit user commands are logged in the Identity Manager log files. You can also write messages in the log files, using the `PrintMessage` function, which is defined in the source template and delivered with the product.
- The pre-exit and post-exit commands are executed each time the Identity Manager agent is executed.

Managing Passwords

Identity Manager can intercept an account's password change on a UNIX or Linux system, and propagate it to all other accounts associated with its Global User. Identity Manager Pluggable Authentication Module (PAM) lets Identity Manager authenticate passwords against external security systems so that global users can use their existing system passwords to log on to Identity Manager.

For more information, see Identity Manager *Administrator's Guide*.

UNIX Password Synchronization

Identity Manager provides a password synchronization module that detects password change events through the UNIX PAM framework. The UNIX Password Synchronization module notifies the Provisioning Server of a password change. The Provisioning Server finds the associated Global User, and propagates changes to other related accounts automatically.

The UNIX operating systems that support the PAM framework include:

- AIX v5.3 on Power platform with PAM enabled
- HP-UX v11.00 on a PA-RISC platform, and Itanium® 2 platforms
- Solaris v2.6 and higher on Sparc and Intel platforms
- Linux with glibc v2.2 and higher on s390 or Intel i386 platform

How UNIX PAM Works

The following process describes the UNIX PAM feature's functions:

1. A UNIX user's password is to be changed for one of the following reasons:
 - Decision of the user.
 - The user is forced to change the password by system settings or manual intervention.
 - The user's password is changed by an administrator.
2. The new password is submitted to the PAM framework password service.
3. The PAM framework's password service invokes the PAM library to update the local UNIX security files.
4. The PAM framework's password service invokes the UNIX password synchronization module (pam_CA_eta) to notify the Provisioning Server of the password change.
5. The Provisioning Server updates the password of the associated Global User and all accounts associated with the Global User.

Requirements for Using UNIX Password Synchronization

The requirements for using the UNIX Password Synchronization feature are the following:

- The UNIX Password Synchronization agent must be installed on the UNIX system on which you want to detect password changes.
- The UNIX Remote agent and CAM must be installed on the UNIX system on which the UNIX Password Synchronization agent resides.
- The system must be managed as an acquired Identity Manager endpoint. The Password Synchronization agent is installed check box must be selected on the acquired endpoint's properties.
- The accounts on the managed systems must be explored and correlated to Identity Manager global users.
- The Enable Password Synchronization Agent flag is selected on the Global User property sheet's Password tab.

Install the UNIX PAM Feature

Perform the following procedure to install UNIX PAM.

To install the UNIX PAM feature

1. Select the package file that corresponds to your UNIX platform:

UNIX Operating System	Package File Name
HP-UX v11 PA-RISC	pam_CA_eta-1.1.HPUX.tar.Z

HP-UX Itanium2	pam_CA_eta1.1HPUX-IA64.tar.Z
AIX v5.3 Power	pam_CA_eta-1.1.AIX.tar.Z
Solaris Sparc	pam_CA_eta-1.1.Solaris.tar.Z
Solaris Intel	pam_CA_eta-1.1.SolarisIntel.tar.Z
Linux x86	pam_CA_eta-1.1.Linux.tar.gz
Linux s390	pam_CA_eta-1.1.LinuxS390.tar.gz

- Transfer the chosen package file to a temporary folder (/tmp) on the UNIX server using FTP in binary mode, or any other file transfer tool that supports binary files. A sample transfer session might appear as follows:

```

W:\Pam>ftp user01
Connected to user01.company.com.
220 user01 FTP server (Version 1.2.3.4) ready.
User (user01.company.com:(none)): root
331 Password required for root.
Password:
230 User root logged in.
ftp> cd /tmp
250 CWD command successful.
ftp> bin
200 Type set to I.
ftp> put pam_CA_eta-1.1.HPUX.tar.Z
200 PORT command successful.
150 Opening BINARY mode data connection for pam_CA_eta-1.1.HPUX.tar.Z.
226 Transfer complete.
ftp: 117562 bytes sent in 0.09Seconds 1306.24Kbytes/sec.
ftp> quit

```

- Logon as the root user on the UNIX server and extract the package file:

```

# cd /tmp
# zcat pam_CA_eta-1.1.<platform>.tar.Z | tar -xf -

```

On Linux, use the command:

```

# tar -xzf pam_CA_eta-1.1.<platform-hardware>.tar.gz

```

- Copy the configuration and TLS files to the default configuration folder:

```

# cd pam_CA_eta-1.1
# mv pam_CA_eta/etc

```

5. Copy the pam_CA_eta module to the Security libraries folder:

On AIX, use the command:

```
# cp -p pam_CA_eta.o /usr/lib/security/
```

On HP-UX, use the command:

```
# cp -p libpam_CA_eta.1 /usr/lib/security/
```

On HP-UX Itanium2, use the command:

```
# cp -p libpam_CA_eta.1 /usr/lib/security/hpux32
```

On Linux i386 or s390, use the command:

```
# cp -p pam_CA_eta.so /lib/security/
```

On Solaris Sparc or Intel, use the command:

```
# cp -p pam_CA_eta.so /usr/lib/security/
```

6. (Optional) Copy the Testing programs:

```
# cp -p test_* /etc/pam_CA_eta
```

```
# cp -p pam_test* (/usr/lib/security/
```

More Information

[Troubleshooting UNIX Password Synchronization](#) (see page 172)

Configuring the UNIX Password Synchronization Feature

Configuration the UNIX Password Synchronization feature involves setting parameters in the following files:

- /etc/pam_CA_eta/pam_CA_eta.conf
- /etc/pam.conf

Important! Because the password of a highly-privileged Identity Manager user is stored in the pam_CA_eta.conf configuration file, that file must be readable only by the root account. Note that the file settings in the package file include owner=root and mode=500 and that the -p switch of the cp command preserves them during installation.

Configure the pam_CA_eta.conf File

Perform the following procedure to configure the pam_CA_eta.conf file.

To configure the pam_CA_eta.conf file

1. Navigate to the /etc/pam_CA_eta folder.
2. Edit the pam_CA_eta.conf file. This configuration file contains its own documentation.

```
#
#   CA - Identity Manager
#
#   pam_CA_eta.conf
#
#   Configuration file for the Unix PAM password module "pam_CA_eta"
#

# keyword: server
# description: the Identity Manager LDAP server primary and optional alternate server hostname
# value: a valid hostname and an optional server
# default: no default
server ETA_SERVER ALT_SERVER

#

# keyword: port
# description: the numeric TCP/IP port number of the Identity Manager LDAP server
# value: a valid TCP/IP port number
# default: 20390
# port 20390

# keyword: use-tls
# description: does it use the secured LDAP over TLS protocol ?
# value: yes or no
# default: yes
# use-tls yes
```

keyword: time-limit
description: the maximum time in seconds to wait for the end of an LDAP operation.
value: a numeric value of seconds
default: 300
time-limit 300

keyword: size-limit
description: the maximum number of entries returned by the Identity Manager server
value: a numeric value
default: 100
size-limit 100

keyword: root
description: the root DN of the Identity Manager server
value: a valid DN string
default: dc=eta
root dc=eta

keyword: domain
description: the name of the Identity Manager domain
value: a string
default: no default
domain DOMAIN

keyword: user
description: the Identity Manager Global User name used to bind to the Identity Manager server
value: a valid Global User name string
default: etaadmin
user etaadmin

keyword: password
description: the clear-text password of the "binding" Identity Manager Global User
value: the password of the above Global User
default: no default
password SECRET

keyword: directory-type
description: the Identity Manager Unix Endpoint type of this Unix server
value: ETC or NIS
default: ETC
endpoint-type ETC

keyword: endpoint-name
description: the Identity Manager Unix Endpoint name of this Unix server
value: a valid Unix Endpoint name string


```
# default:
# ETC: the result of the "hostname" command (ie: gethostname() system call)
# NIS: "domain [hostname]" where "domain" is the result of the "domainname" command
# (ie: getdomainname() system call) and "hostname" the result of the "hostname"
# command (ie: gethostname() system call)
# endpoint-name dimame
```

```
# keyword: tls-cacert-file
# description: the name of the Identity Manager CA certificate file
# value: a valid full path file name
# default: /etc/pam_CA_eta/eta2_cacert.pem
# tls-cacert-file /etc/pam_CA_eta/eta2_cacert.pem
```

```
# keyword: tls-cert-file
# description: the name of the Identity Manager client certificate file
# value: a valid full path file name
# default: /etc/pam_CA_eta/eta2_clientcert.pem
# tls-cert-file /etc/pam_CA_eta/eta2_clientcert.pem
```

```
# keyword: tls-key-file
# description: the name of the Identity Manager client private key file
# value: a valid full path file name
# default: /etc/pam_CA_eta/eta2_clientkey.pem
# tls-key-file /etc/pam_CA_eta/eta2_clientkey.pem
```

```
# keyword: tls-random-file
# description: the name of the "pseudo random number generator" seed file
# value: a valid full path file name
# default: /etc/pam_CA_eta/pmg_seed
# tls-random-file /etc/pam_CA_eta/pmg_seed
```

```
# keyword: use-status
# description: this module will exit with a non-zero status code in case of failure.
# value: yes or no
# default: no
# use-status no
```

```
# keyword: verbose
# description: this module will display informational or error messages to the user.
# value: yes or no
# default: yes
# verbose yes
```

Note: The server, domain and password parameters do not have a default value and need to be updated.

Configure the pam.conf File

The `/etc/pam.conf` file is the main PAM configuration file. You must edit the file to insert a line in the password service stack. On some Linux systems, the `pam.conf` file is replaced with `/etc/pam.d`, so you will need to edit the `/etc/pam.d/system-auth` file.

To configure the pam.conf file

1. Navigate to the `/etc` directory, or `/etc/pam.d` directory if you are configuring the PAM module on an appropriate Linux system.
2. Edit the `pam.conf` file to insert a Password Synchronization line in the password service stack. For platform-specific configurations, see the examples that follow:

```
passwd password required /usr/lib/security/pam_unix.so
```

```
passwd password optional /usr/lib/security/pam_CA_eta.so
```

3. (Optional) You can add the following optional parameters on the pam_CA_eta module line:

config=/path/file

Indicates the location of an alternate configuration file.

syslog

Sends error and informational messages to the local syslog service.

trace

Generates a trace file for each password update operation. The trace files are named /tmp/pam_CA_eta-trace.<nnnn> where <nnnn> is the PID number of the password process.

4. Implement the following platform-specific configuration changes:

For AIX systems, add the following lines at the bottom of the /etc/pam.conf file:

```
#
# Identity Manager Unix Password Synchronization
#
login password optional /usr/lib/security/pam_CA_eta.so syslog
passwd password optional /usr/lib/security/pam_CA_eta.so syslog
rlogin password optional /usr/lib/security/pam_CA_eta.so syslog
su password optional /usr/lib/security/pam_CA_eta.so syslog
telnet password optional /usr/lib/security/pam_CA_eta.so syslog
sshd password optional /usr/lib/security/pam_CA_eta.so syslog
OTHER password optional /usr/lib/security/pam_CA_eta.so syslog
```

For HP-UX systems, add the following lines at the bottom of the /etc/pam.conf file:

```
#
# Identity Manager Unix Password Synchronization
#
login password optional /usr/lib/security/libpam_CA_eta.1 syslog
passwd password optional /usr/lib/security/libpam_CA_eta.1 syslog
dtlogin password optional /usr/lib/security/libpam_CA_eta.1 syslog
dtaction password optional /usr/lib/security/libpam_CA_eta.1 syslog
OTHER password optional /usr/lib/security/libpam_CA_eta.1 syslog
```

For HP-UX Itanium2, add the following lines at the bottom of the /etc/pam.conf file:

```
#
# Identity Manager Unix Password Synchronization
#
```

```
login password optional /usr/lib/security/$ISA/libpam_CA_eta.1 syslog
passwd password optional /usr/lib/security/$ISA/libpam_CA_eta.1 syslog
dtlogin password optional /usr/lib/security/$ISA/libpam_CA_eta.1 syslog
dtaction password optional /usr/lib/security/$ISA/libpam_CA_eta.1 syslog
OTHER password optional /usr/lib/security/$ISA/libpam_CA_eta.1 syslog
```

For Sun Solaris systems, add the pam_CA_eta line after the existing pam_unix line:

```
#
# Password management
#
other password required /usr/lib/security/pam_unix.so.1
other password optional /usr/lib/security/pam_CA_eta.so syslog
```

For Linux systems, add the pam_CA_eta line between the existing pam_cracklib and pam_unix lines:

```
password required /lib/security/pam_cracklib.so retry=3 type=
password optional /lib/security/pam_CA_eta.so syslog
password sufficient /lib/security/pam_unix.so nullok use_authok md5 shadow
password required /lib/security/pam_deny.so
```

5. For AIX systems, edit the /etc/security/login.cfg file to set auth_type = PAM_AUTH. This enables the PAM framework, which is not enabled by default. This is a run-time setting so you do not have to reboot the system for it to take effect.

Troubleshooting UNIX Password Synchronization

You can troubleshoot the UNIX PAM feature using syslog and trace messages, and by testing the configuration, LDAP/TLS connection, the password synchronization, and the PAM framework.

More Information

[Activating Syslog Messages](#) (see page 172)

[Activating Trace Messages](#) (see page 173)

Activating Syslog Messages

Add the syslog parameter to the pam_CA_eta line in the /etc/pam.conf file to let the pam_CA_eta module generate informational and error messages. When the logging option is in use, the UNIX administrator sees information messages in the syslog files each time a UNIX account changes its password. These messages should provide enough information to diagnose basic problems.

You could set this option permanently on production systems as it does not require many more resources than when running in silent service.

Activating Trace Messages

If the syslog messages do not provide enough information, the trace mode can provide more details. For each password update operation, the trace module generates a file named `/tmp/pam_CA_eta-trace.<nnnn>` (where `<nnnn>` is the PID of the `passwd` process) with an entry for most of the function calls used by the module and the data used or returned by those functions.

Even though the trace files are only readable by the root account, they will contain the clear-text new passwords. For this reason, this parameter should not be used permanently on a production system.

Testing the Configuration File

You can use the `test_config` tool, located in the `/etc/pam_CA_eta` directory, to verify the configuration file. A sample command line entry follows:

```
/etc/pam_CA_eta/test_config [config=/path/to/config_file]
```

An example session follows:

```
./test_config [config=/path/to/config_file]
# ./test_config
./test_config: succeeded
Trace file is /tmp/test_config-trace.1274
```

As the command output shows, a trace file was generated which contains all the details of the configuration file parsing.

Testing the LDAP/TLS Connection

You can use the `test_ldap` tool, located in the `/etc/pam_CA_eta` directory, to verify the connection to the Provisioning Server (using the configuration file parameters). A sample command line entry follows:

```
/etc/pam_CA_eta/test_ldap [config=/path/to/config_file]
```

An example session follows:

```
./test_ldap [config=/path/to/config_file]
# ./test_ldap: succeeded
Trace file is /tmp/test_ldap-trace.1277
```

As the command output shows, a trace file was generated which contains all the details of the configuration file parsing and the connection to the Provisioning Server.

Testing the Password Synchronization

You can use the `test_sync` tool, located in the `/etc/pam_CA_eta` folder, to verify that the password update of a local account is effectively propagated by the Provisioning Server. A sample command line entry follows:

```
/etc/pam_CA_eta/test_sync <user> <password> [config=/path/to/config_file]
```

An example session follows:

```
# /etc/pam_CA_eta/test_sync pam002 newpass1234
Identity Manager password synchronization started.
:ETA_S_0245<MGU>, Global User 'pam002' and associated account passwords updated successfully: (accounts
updated: 2, unchanged: 0, failures: 0)
Identity Manager password synchronization succeeded.
/etc/pam_CA_eta/test_sync: succeeded
Trace file is /tmp/test_sync-trace.2244
```

As the command output shows, a trace file was generated which contains all of the details of the configuration file parsing, the connection to the Provisioning Server, and the update of the account.

When using the verbose mode (by using the default verbose yes parameter in the configuration file), the command provides informational and potential error messages about the password propagation.

Test the PAM Framework

A PAM test library is available to verify that the password changes are correctly detected by the PAM framework.

To test the PAM framework

1. Copy the `pam_test` file to the `/usr/lib/security(/hpux32)` folder.
2. Add a password class line for the `pam_test` library with no parameters.

An example for Solaris follows:

```
other password optional /usr/lib/security/pam_test
```

3. Issue a `passwd` command on a test user and then search for the `pam_test[<pid>]` tagged line in the syslog file.

The command output shows the name of the generated trace file, for example:

```
pam_test[1417]: Succeeded, trace file is /tmp/pam_test-trace.1417
```

Chapter 8: Directory Services Connectors

This section contains the following topics:

[Active Directory Services Connector](#) (see page 176)

[Generic LDAP Connector](#) (see page 216)

[NDS Connector](#) (see page 238)

The following connectors are considered to be directory services connectors:

- Active Directory Services
- Generic LDAP
- NDS

The information in this chapter describes how to install and manage these connectors.

Active Directory Services Connector

The Active Directory Services Connector lets you administer accounts, groups, and containers, printers, computers, and shared folders on Active Directory Services servers and provides a single point for all user administration by letting you do the following:

- Register endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage Active Directory Services accounts using account templates specific to Active Directory Services
- Synchronize global users with their roles or synchronize global users' accounts with their account templates
- Change account passwords and account activations in one place
- Assign an Active Directory Services account template to each of your Active Directory Services endpoints
- Use the default Endpoint Type account template to create accounts with the minimum level of security needed to access an Active Directory Services directory
- Create and manage Active Directory Services groups, containers, printers, shared folders, and computers
- Create and secure Microsoft Exchange 2000 mailboxes
- Generate and print reports about Active Directory Services accounts

This connector is managed using the Connector and C++ Server installation process.

Note: For more information and requirements, see *Connector and C++ Connector Server Installation* (see page 23).

Configure Your Windows Servers Using SSL

Perform the following steps if you are using Secure Socket Layer (SSL) communication:

1. Install the High Encryption Pack on the Active Directory Services (ADS) server you want to manage.
2. Install and configure a Certificate Authority (CA).
3. Set up ADS, the Certificate Authority, and Identity Manager on a single system, on a dual computer system, or on multiple computers.
4. Verify the C++ Connector Server service logon ID.

Note: SSL communication is not mandatory when using the Active Directory Services Connector. However, if you do not use SSL communication, you are not able to use the password management features of the Active Directory Services Connector.

The Active Directory Services Connector must be installed on a Windows 2000 or Windows 2003 system.

Step 1. Install the High Encryption Pack

You should have the High Encryption Pack for Active Directory Services installed. To verify this, from the Internet Explorer menu, select Help, About Internet Explorer. The Cipher Strength should be listed as 128-bit. If it is not, you need to install the High Encryption Pack.

If necessary, download the High Encryption Pack for Active Directory Services from the [Microsoft](#) web site (the pack is free of charge).

Install the High Encryption Pack on the Active Directory Services servers that you want to manage with Identity Manager.

Step 2. Install and Configure a Certificate Authority

Important! If you plan to use the CertServ web page to request and import certificates, the Microsoft IIS World Wide Web Server component must already be installed before you install and configure a Certificate Authority. Therefore, if you have not already installed the IIS World Wide Web Server component on the Certificate Authority server, do so now, before continuing with the steps that follow. For instructions, consult your Microsoft IIS World Wide Web Server component documentation.

On each Active Directory Services server that you want to manage, you will need to configure a public key Certificate Authority.

Microsoft Windows 2000 [Advanced | Datacenter] Server includes a Public Key Certificate Authority. There are two types of Certificate Authority you can install, each of which can be either the root or a child. This document addresses the use of one of the types of Root CAs, either an Enterprise Root CA or a Standalone Root CA.

For information on installing one of these CAs, please refer to Microsoft Knowledge Base Article #231881.

Step 2a. Log on Anonymously (Optional)

If you plan to use the Certsrv web page, you can configure your CA server to allow anonymous access to the CertSrv web page.

From the task bar

1. Click Start, Programs, Administrative Tools, Internet Services Manager.
The Internet Information Services dialog appears.
2. Open the machine name/Default Web Site.
3. Right-click CertSrv.
4. Select Properties.
The CertSrv Properties dialog appears.
5. Select the Directory Security tab.
6. Click the Edit button for Enable Anonymous access, and edit the authentication methods for this resource.
7. Configure the logon information (anonymous or otherwise) for your machine.

Step 3. Set Up ADS, Certificate Authority, and Identity Manager

You can set up ADS, the Certificate Authority, and Identity Manager on a single system, dual computer system, or on multiple computers.

Step 3a. Set Up on a Single Computer

When Identity Manager and Active Directory Services are installed on a single machine, it should not be necessary to do anything with certificates, as the machine inherently trusts itself.

Step 3b. Set Up on a Dual Computer System

When Identity Manager and the CA are on one machine, and Active Directory Services is on a different machine, follow the steps to Request, Issue, and Import the Server Authentication Certificate as outlined in the Multiple System Instructions (later in this chapter). It should not be necessary for the Identity Manager machine to import the root certificate from the CA, as they reside on the same machine.

When Identity Manager is installed on one machine, and the CA and Active Directory Services are installed on another machine, import the root certificate and ensure that the ADS machine is providing it to the Identity Manager machine. These instructions assume the CA was installed with the web enrollment pages functioning.

When the Provisioning Server is installed on one machine, the Identity Manager distributed C++ Connector Server on another, and the CA and Active Directory Services are installed elsewhere, import the root certificate on the machine hosting the Identity Manager C++ Connector Server, and ensure that the ADS machine is providing it to the Identity Manager machine. These instructions assume the CA was installed with the web enrollment pages functioning.

Import the Root Certification Authority

Use the following procedure to import the root certification authority.

On the machine hosting the C++ Connector Server

1. Open a web browser and access the URL, http://CA_FQDN/certsrv.
2. Select Retrieve the CA certificate or certificate revocation list. (On some platforms, this option may appear as Download a CA certificate, certificate chain, or CRL.)
3. Select Install this CA certification path. (On some platforms, this option may appear as Install this CA certificate chain.)

Enable the Use of SSL to Manage ADS

Note: The following steps for Windows 2003 only and All Platforms only apply to enterprise, not standalone systems.

Follow these steps on the Active Directory/Certificate Authority machine:

For Windows 2003 Only:

The following are required to install the CA as a Trusted Certificate Authority in the policy **before** running the Automatic Certificate Request wizard on Enterprise Root systems (not Standalone):

You have to export the CA manually as a Trusted Certificate Authority first in order to import a copy. To do this, follow these steps:

1. Select Tools, Internet Options.
2. Select the Content tab and in the Certificates section, click the Certificates button.
3. Select the Trusted Root Certification Authorities, Certificate Authority, and click the Export button.

A wizard appears and lets you save a copy of the CA certificate which can then be imported.

4. From the Group Policy Editor, select Public Key Policies, Trusted Root Certification Authorities, and import a copy of the Certificate Authority.

All Platforms:

1. Open the Group Policy Editor in the Active Directory Users and Computers Control Panel by right-clicking the domain name.
2. Select Properties.
3. Select the Group Policy Editor tab.
4. Complete the Automatic Certificate Request wizard as described in your operating system documentation for Security Concepts.

Step 3c. Set Up Multiple Systems

The following directions apply to establishing the proper trust relationships where Active Directory Services is installed on a machine (or multiple machines); the Certificate Authority is on a different machine, and the Provisioning Server is on yet another machine. Furthermore, they assume that the CA was installed with the web enrollment pages functioning (this requires having Microsoft IIS installed prior to installing Certificate Services).

Request the Certificate

Perform the following procedure to request the certificate.

From the Active Directory server

1. Open Internet Explorer and browse to `http://CA_FQDN/certsrv` (where *CA_FQDN* is the machine name of the Certificate Authority (CA) server).
2. Select the option to Request a Certificate.
3. Select the option Advanced Request.
4. Select the option Submit a Certificate Request to this CA using a form (on some platforms, this may appear as Create and Submit a Request to this CA).
5. Populate the form. The range of fields may vary depending on your platform and level. If these fields are present, supply the following values:

Name or Friendly Name

Set this value to the `<local FQDN>`.

Intended Purpose

Select Server Authentication.

Certificate Key Options/CSP

Set to Microsoft RSA SChannel Cryptographic Provider *or* Microsoft Enhanced Cryptographic Provider.

Key Options/Key Size

Set to a value of 1024.

Key Options/Local Machine Store or Store Certificate in the Local Computer Certificate Store

Select this option.

Leave all other fields with their default values.

6. Click Submit to issue the request to the CA.
An administrator of the CA must now approve this request.

Issue the Certificate

Perform the following procedure to issue the certificate.

On the machine hosting the Identity Manager C++ Connector Server

1. Open the Certification Authority control panel (click Start, Settings, Control Panel, Administrative Tools or Programs, Administrative Tools).
2. Select Pending Requests.
3. Having identified the proper certificate request, right-click the request.
4. Select All Tasks, Issue.

Import the Certificate

Importing the certificate can be done in one of the following two ways.

1. Using Certsrv web page from the Active Directory Server
2. Manually exporting and importing

1. Use the Certsrv Web Page from the Active Directory Server

Use the following procedure to import the certificate using the web page.

From Internet Explorer

1. Browse to http://CA_FQDN/certsrv.
2. Select Check on Pending.
3. Select the Request you issued earlier.
4. Click Install this Certificate.

2. Manually Export and Import

Use the following procedure to export from the Certificate Authority Server manually.

From the Certification Authority control panel

1. Select Issued Certificates.
2. Double-click the certificate to open it.
3. Select the Details panel.
4. Select Copy to File and leave all defaults (except file name).

Use the following procedure to import the CA on the Active Directory Server manually.

From the Active Directory Server

1. Run the Microsoft Management Console by clicking Start, Run mmc.
2. Select Console, Add/Remove Snap-in and click Add.
3. Select Certificates and click Add.
4. Select Computer Account and click Finish.
5. Click Close.
6. Click OK to exit the Add/Remove Snap-in dialog.
7. Expand the Certificates (Local Computer) folder and then the Personal folder.
8. Right-click the Certificates folder.
9. Select All Tasks, Import. Supply the file created previously and retain the default setting otherwise.
10. Reboot the Active Directory Server before continuing.

Import the Root Certification Authority

Use the following procedure to import the root certification authority.

On the machine hosting the C++ Connector Server

1. Open a web browser and access the URL, http://CA_FQDN/certsrv.
2. Select Retrieve the CA certificate or certificate revocation list. (On some platforms, this option may appear as Download a CA certificate, certificate chain, or CRL.)
3. Select Install this CA certification path. (On some platforms, this option may appear as Install this CA certificate chain.)

Step 4. Verify the C++ Connector Server Service Logon ID

In the previous step, you set up a trust relationship. Normally, the account used to start the C++ Connector Server is Local System account. To manage ADS however, this account should be the same account that acquired the Root Certification Authority in Step 3. Use the following procedure to ensure that the service is logged on properly.

From the Control Panel

1. Select Administrative Tools, Services.
2. Double-click the C++ Connector Server entry.
3. Verify that the account (a local administrator or a domain administrator) being used to run the service is the same account that was used to install the Root Certification Authority.
4. Verify that the account password is correct.
5. If you have changed either the account or password, restart the C++ Connector Server service.

Connect to a Child-Domain

If you are using SSL, and want to use Identity Manager to manage a child-domain, you must establish permissions within Active Directory so that the child-domain can refer to the Certificate Authority that is defined on the parent domain. This is required if you are using an Enterprise Certificate Authority.

Please refer to the following Microsoft articles for further instructions on this:

- Q281271 - Win2000 Certification Authority Configuration to Publish Certificates in Active Directory of Trusted Domain
- Q271861 - Windows Cannot Find a Certificate Authority that Processes the Request

Note: Confirm that your DNS configuration is correct. From both the parent and the child, you should be able to ping the other and receive back the correct IP address. Likewise, you should be able to run an 'nslookup' command on the IP address and receive back the correct fully-resolved name of the other.

If you are using SSL, and experience Identity Manager errors when you attempt to manage a child domain, you can use the standalone ADLDAPDiag utility to connect to the child domain.

Note: ADLDAPDiag should be run on the same machine as the C++ Connector Server. If ADLDAPDiag fails, this indicates that the Identity Manager-errors are due to an SSL problem with the child domain (the syntax of ADLDAPDiag is: ADLDAPDiag *fully_qualified_name_of_the_ADS_server*).

Important! If your Certificate Authority is installed on a Windows 2003 server, auto-enrollment for the child domain needs to be working properly before a proper trust relationship can be established between the parent and child domains.

ADS Defaults

Once the Active Directory Services Connector has been installed and configured, you have everything that you need to run and use the connector. The following sections describe the basic ADS Connector settings and their defaults.

Failover

As the default, Failover is not enabled. To enable Failover, set the environment variable ADS_FAILOVER to 1 and restart the C++ Connector Server and the Provisioning Manager.

Note: If the Provisioning Manager and the C++ Connector Server run on different machines, you must set the ADS_FAILOVER variable on both machines.

After failover is enabled, you can turn on failover support for individual directories by going to the Endpoint Property Sheet and checking the failover checkbox on the Failover tab.

Understanding Failover

Prior to Windows 2000, Windows NT supported multiple domain controllers: Primary (PDC) and Backup (BDC). You could query any controller for information, but changes could only be made to the PDC. Active Directory, introduced as part of the Windows 2000 Server, goes a step further. It allows all controllers to be primaries, and a change to any one controller is automatically propagated to the other controllers.

This allows Identity Manager, which is used to manage an installation, to have failover support. For example, ADS is communicating with a single domain controller and it goes down. ADS then automatically connects to an alternate domain controller and retries the failed operation. Thereafter, all communications happen with the alternate controller.

For technical reasons, it is advantageous to establish an order in which the controllers are to be used. This can be done from the Failover page on the Endpoint property sheet. This page automatically displays the alternate controllers (as retrieved from DNS) and allows the user to prioritize them.

In the background, ADS periodically attempts to reconnect to any failed controllers. When ADS detects that a failed controller of a higher priority than the current controller is back online, it automatically reroutes the next request to the restored controller.

Using Failover

IMPORTANT! By default, ADS Failover logic is disabled. If you decide to use Failover, you must be aware of the following Failover behaviors. Otherwise, you will encounter unexpected Failover behaviors. Please review this section carefully.

- If you wish to activate Failover, you must set the environment variable `ADS_FAILOVER` to 1.

Note: `ADS_FAILOVER` should be set on the machine where the C++ Connector Server and ADS connector run, and on the machine where the Provisioning Manager runs.
- The specified server name (the name of the Active Directory server specified on the Endpoint page) should match its fully qualified DNS server name.
- You should consider enabling logging when using Failover. (For more information, see the Endpoint Logging page on the Endpoint property sheet.) With logging enabled, in the event of unexpected behaviors, the logs provide a record of what transpired.
- The Provisioning Server attempts to retrieve the complete list of backup domain controllers from DNS, or you may elect to supply this list manually.

If DNS is not available or you want to bypass DNS, you may supply a configuration file `PS_HOME\data\ads\directory-name.DNS`. (A sample DNS-configuration file is provided in the distribution of the file `PS_HOME\data\ads\endpoint-name.dns`.)

where `endpoint-name` is the name of the endpoint that you specified in the Name field on the ADS Server tab of the ADS Endpoint property sheet.

In order to view the list of domain controllers, select the Failover tab in the Endpoint property sheet. This page provides the list of domain controllers that ADS uses for failover. If there is only one item in the list (the primary server being managed), or failover was not enabled using the `ADS_FAILOVER` environment variable, then this page is disabled and failover is not operational.

Whether Identity Manager can determine the list of backup controllers automatically from DNS is heavily dependent on your environment. If this attempt fails, try one of the following suggestions:

- Run the Provisioning Server in the same domain as the ADS Server.
- Set the preferred DNS server field on the Provisioning Server correctly.
- Run the Provisioning server under a domain-administrator account.

If all of your domain controllers in your enterprise are not listed on the Failover tab, then failover was unable to retrieve the list from DNS. You must manually provide the `.dns` config file.

You can run the `ADSLISTsites servername` diagnostic utility to determine what information DNS is returning. If a list of sites or servers is returned, then automatic failover is operational. If `ADSLISTsites` is not configured for automatic failover, you will have to manually supply the list of domain controllers.

- If you are using SSL, Provisioning Server must be able to connect to all domain controllers listed on the Failover tab of the Endpoint page using SSL. You must configure each of these servers to present a valid acceptable certificate to the Provisioning Server.

If SSL is used, all the domain controllers associated with a single endpoint must be able to communicate with Identity Manager using SSL. If an SSL connection cannot be established with any one of the domain controllers, then you should not use SSL, or you should omit that domain controller in the `.dns` configuration file.

- Active Directory guarantees that all changes made on any one domain controller are propagated to all other domain controllers. The time that the propagation occurs is installation-defined.
- You must be aware of the effects of propagation delays.

For example, if the Provisioning server makes a change to a controller that subsequently goes down, and Identity Manager automatically connects to a backup controller, any changes made earlier may not yet be reflected on the backup because of propagation delays. This can have adverse results.

That is, if Provisioning Server is communicating with the primary server and encounters a failure, it immediately switches to the secondary server. If the user subsequently creates accounts on the secondary server, and the primary comes back up, ADS reconnects to the primary.

If Active Directory has not propagated the new accounts from the secondary to the primary controller, you receive a not-found error when you attempt to view the new accounts from Identity Manager. This occurs because the account does not yet exist on the primary server.

Even more serious is the case in which you proceed to do an Explore (executed on the primary, now that the connection is restored). When Explore fails to find the newly created accounts, it assumes they have been deleted from the target system. Consequently, Identity Manager then deletes the accounts from the repository.

Note: You may also encounter a situation wherein conflicting changes made on different controllers could cause one of them to be lost.

- There are separate timeouts for the agent and for the Provisioning Manager.

For example, if the agent timeout value is one minute, and the Provisioning Manager timeout is 90 seconds, when Identity Manager attempts to communicate with a particular controller, it takes a full minute before the agent gives up. If the secondary controller is also down, another minute lapses before the agent declares the backup as down and attempts the tertiary controller. In the meantime, the Provisioning Manager times out and the initial operation fails, although the (tertiary) controller was actually available.

- When you change the order of the controllers on the Failover tab, the changes are only in effect for subsequent connections to the server (for example, a new user logs in, or the original primary controller goes down). However, existing valid connections continue to be used until the background process runs again and attempts a better connection.
- Although you can order the controllers in any desired sequence on the Failover tab, the server always arranges the list so that the primary server (that is, the one used on the Endpoint page) is always first. This prevents an inadvertent connection to an alternate controller during a restart.
- As mentioned previously, a background thread runs periodically to attempt to reestablish existing connections to the more preferred domain-controller. By default, this thread runs every 15 minutes. However, you can change this setting by setting the environment variable `ADS_RETRY` to the desired number of minutes.

Note: If, after performing a migration to r12, you notice that the Failover tab on the Endpoint property sheet is completely disabled and Failover is not available, run the MigrateADS utility to correct the problem.

Failover Retry Interval

The default for Failover Retry Interval is 15 minutes. When failover is enabled and the domain controller(s) are down, the ADS connector periodically checks the downed domain controller(s). To increase or decrease the interval time (in minutes), set the `ADS_RETRY` environment variable and restart the C++ Connector Server. If the value is set to less than one minute, the value is ignored and a one minute interval is used.

ADSI Option

Important! ADSI is not fully supported in this release and, by default, is disabled. Contact technical support to enable this option.

ADS lets you use a non-SSL connection to the Active Directory Server so you can use ADS in a test environment when enabling SSL is not feasible. The non-SSL connection lets you connect in one of the following two ways:

- Use ADSI for passwords only and non-secure LDAP for everything else.
- Use non-secure LDAP for all communications. This option silently ignores all password change requests.

Note: Both options use a normal authentication with LDAP that sends the user's credentials over the network. This practice can be a serious security risk. Neither of these options should be used in a production environment.

ADSI provides a way to manage accounts and passwords with SSL. This option uses ADSI to set passwords while all other operations use a non-secure LDAP connection.

ADSI does not work in all environments, particularly in cross-domain networks. Only use this option when you do not want to use the non-SSL option.

If ADSI does not work, try the following:

- Install the Windows 2000 Support Tools on the Active Directory Services server you want to manage.
- Run the Provisioning Server in the same domain as the Active Directory server.
- Confirm that the Preferred DNS server field is set correctly on the Provisioning Server.
- Start the Provisioning service with an ADS-domain -administrator account.

WARNING! Do not use the ADSI or non-secure LDAP options in production environments.

ADS_MANAGE_GROUPS

For an account template marked as strong sync policy, previously for account sync operation (that is, Synchronize Account with Account Template, or Check Account Sync) the ADS option may fail to find remote Universal Group that the account belongs to. For example, if an account on domain D1 is a member of a Universal Group on domain D2, a sync operation may not notice that the account belongs to that remote Universal Group.

Identity Manager supports a mode where the user can specify whether to search the global catalog to find Remote Universal Groups that the account may be a member of, when performing a sync operation.

In some environments, not all domains of an Active Directory forest are managed by Identity Manager. For example, a hypothetical AD forest has three domains, D1, D2 and D3. You have two Identity Manager-managed domains D1 and D2 (that is, you acquire D1 and D2). You can specify whether the new global catalog search feature manages Universal Groups from all domains (D1, D2, and D3), or just the Identity Manager-managed domains (D1 and D2). If you choose to have the new global catalog search feature only deal with Identity Manager-managed domains, then Identity Manager will not deal with groups on domain D3, even if the account belongs to a group that resides on domain D3. For example, if the account's policy indicates that it should not belong to any group, and your account belongs to a Universal Group on domain D3, a check account sync operation will not show that the account is out-of-sync, if you chose to deal only with Identity Manager-managed domains. If you chose to deal with all domains, then the account will be considered out-of-sync (even when domain D3 is not managed by Identity Manager).

By default the sync feature is off.

To run this global catalog search feature, you have to set the environment variable ADS_MANAGE_GROUPS.

ADS_MANAGE_GROUPS can be set to xy as defined in the following paragraphs.

The first digit x - can be 0 or 1:

- 0 - You get the current behavior (default).
- 1 - Optional behavior to query using the global catalog as well.

The second digit y - can be 0 or 1:

- 0 - Deal with groups in all domains (whether they are managed by Identity Manager or not).
- 1 - Deal with groups in Identity Manager-managed domains only.

Note: The x value must be set to 1 in order for the y value to have any affect.

Once this environment variable is set, you must restart the C++ Connector Server for the variable to take effect.

Force Logging

The default for force logging is 0 for no force log. To enable force logging, even if the endpoint has logging turned off, set the environment variable ADS_FORCELOG to 1 and restart the C++ Connector Server.

Ignore Group Insufficient Rights Error

The default for ignore group insufficient rights error is set to false, (do not ignore group insufficient rights error).

When you perform a delete operation to delete a user from a remote group and get back a permissions-error (insufficient rights), you can set the environment variable ADS_NOGROUP_PERMS to 1 to ignore this error and consider the operation a success.

Extend the ADS Schema

The ADS connector lets you manage additional attributes that are used by your Active Directory implementation including, the extended ADS schema you may have implemented on your Active Directory system. If you want to have Identity Manager manage these extended attributes, create a flat file called *PS_HOME\data\ADS\schema.ext*. This file should contain a list of the extended attributes that you want to manage.

Note: Not every attribute is manageable through Identity Manager as the Active Directory does try to protect certain sensitive ones.

Each attribute should be listed on a single line by itself and have the same name as the LDAP display name of the attribute on the target ADS system. For example, if the LDAP display name of the attribute on the target system is *extendedAttribute*, the attribute name in the *schema.ext* file needs to be *extendedAttribute*. The LDAP display name can be found under the Name column of the Active Directory Schema\Attributes or the attribute name when you use the JXplorer to connect to the Active Directory and browse a user account.

With this file in place, (you may have to recycle the Provisioning Server), the Provisioning Manager will then display an additional property page called Custom for both account templates and accounts. This page provides a list of all the extended attributes and their values.

Notes:

- The ADS schema should already be extended on the target machine in order to see the extended attributes.
- Identity Manager assumes that these extensions are in effect for the entire enterprise.

Once the extended ADS schema has been configured in Identity Manager, the extended ADS attributes can be mapped to global user's attributes/custom fields by using rule strings in ADS account templates. For more information on how to create custom fields for Global User objects and how to use rule strings, see the *Administrator Guide*.

Modify the schema.ext File

You can modify the schema.ext file (for example, add or remove attributes) and have the changes picked up by existing objects by restarting the C++ Connector Server and establishing a connection to the target system after making your changes. For example, a connection to the target system can be established by opening the Explore/Correlate window or opening properties of an account from the Provisioning Manager.

Any new attributes that are added to the schema.ext file can be found in the list of extended attributes on the Custom tab on the ADS Account or ADS Account Template property sheet. Attributes that are removed from the schema.ext file are handled in one of two ways:

- From the ADS Account property sheet, the attributes will be automatically removed from the list of extended attributes on the Custom tab.
- From the ADS Account Template property sheet, under the Valid column, the attributes will be marked as invalid (N). The attribute can then be removed and deleted from the provisioning repository.

Correlate ADS Extended Attributes

Extended Active Directory schema attributes that are set for a particular account are stored together with their values in the account's attribute called 'eTADSpayload' (user-friendly name 'payload') in the following format:

```
<extendedAttributeName1>:<reservedValue>:<valueLength>=<value>;<extendedAttributeName2>:<valueN>
```

Note: <reservedValue> is a value reserved for future use. It is currently always set to 01.

Attribute mapping can be set from the managed ADS endpoint by specifying a mapping function substring with an offset and length. For more detailed information, see the section Explore and Correlate Parameters in the *Administrator Guide*.

```
GUAttrName[=Endpoint Type:AccountAttrName[:Offset,Length]]
```

The following is an example of mapping the extended attributes to a global user's custom attributes:

```
eTADSpayload
```

```
extendedAttribute1:01:0006=value1;extendedAttribute2:01:0007=value10;extendedAttribute2:01:0008=value100
```

```
eTCustomField01=eTADSpayload:SUB(28,6)
```

```
eTCustomField02=eTADSpayload:SUB(62,7)
```

```
eTCustomField03=eTADSpayload:SUB(97,8)
```

We can see that the attribute mapping mechanism is using substring (SUB) and specifying the offset and the length of the value.

Important! The mapping extended ADS attributes mechanism has limited functionality and is not intended to support the full functionality of built-in ADS attributes. The mechanism assumes that all of the following conditions are true:

- Extended attributes that are defined in the attribute map must be set for all managed accounts.
- The values of the extended attributes that are defined in the attribute map must have a fixed length.

ADS Support for FIPS and IPv6

For release r12 of Identity Manager, the ADS Connector does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire an ADS Server

You must acquire the Active Directory Services server before you can administer it with Identity Manager. When acquiring an Active Directory Services server, perform the following steps from the Endpoint Type task view:

1. Register the server as an endpoint in Identity Manager.

Use the Active Directory Services Endpoint property sheet to register an Active Directory Services server. During the registration process, Identity Manager identifies the Active Directory Services server you want to administer and gathers information about it.

2. Explore the objects that exist on the endpoint.

After registering the server in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all Active Directory Services accounts and groups. You can correlate the accounts with global users at this time or you can correlate them later.

3. Correlate the explored accounts with global users.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the logon name with each existing global user name. If a match is found, Identity Manager associates the Active Directory Services account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the display name with each existing global user's full name. If a match is found, Identity Manager associates the Active Directory Services account with the global user. If a match is not found, Identity Manager performs the next step.
- c. If the Create Global Users as Needed button is selected, Identity Manager creates a new global user and then associates the Active Directory Services account with the global user. If the Create Global Users as Needed button is cleared, Identity Manager performs the next step.
- d. Identity Manager associates the Active Directory Services account with the [default user] object.

Re-Initialize an ADS Endpoint

Note: Certain Active Directory settings rarely change and for performance gain, the ADS connector only reads the data once when the endpoint is initialized and stores the value internally instead of reading the value on each operation. So, if these settings change, the ADS connector needs to be re-initialized to review the new values.

Some events require that the C++ Connector Server to be restarted include the following:

- Group account template changes are made on the ADS system, for example, changing password account template. You must also restart the Provisioning Manager after making these changes.
- Failover seems to be configured, but is not working.
- Either the administrative user (used to acquire an ADS endpoint) or the administrative users' password is updated.

To restart the C++ Connector Server, follow these steps:

1. Select Start, Settings, Control Panel, Administrative Tools, Services.
2. Right-click the C++ Connector Server entry and select Restart.
3. Click Yes when prompted to restart the Identity Manager Provisioning service.

ADS Default Account Template

The Active Directory Services DefaultPolicy, provided with the Active Directory Services connector, gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

Note: You can create ADS account templates that are associated with multiple endpoints. These account templates can only be used to grant privileges to existing accounts.

Create an ADS Account Template

To create an ADS account template, perform the following steps:

1. Click the Provisioning Roles task button.
2. Select Active Directory Account Template in the Object Type drop-down list box.
3. Click New.
4. Complete the Active Directory Account Template property sheet.
5. Click OK.

Note: When creating or editing Active Directory account templates, it is possible to select any Domain Local or Universal Group from any domain in the endpoint's corresponding ADS Forest. This feature is only supported on single-endpoint account templates.

Delete an ADS Account Template

To delete an ADS account template, perform the following steps:

1. Click the Provisioning Roles task button.
2. Select Active Directory Account Template in the Object Type drop-down list box.
3. Click Search.
4. Right-click on the Active Directory account template and select delete.
5. Click OK.

The Default ADS Account Template

The default ADS account template has been modified to include default values for the Password field. The default value (%P%) of the Password field sets the password on all accounts created with the default account template. If your account template was created prior to Cumulative Patch Revision 4, you must follow these steps to apply this feature:

1. Delete the current ADS DefaultPolicy object.
2. Run the following command from *PS_HOME\bin*:

```
adspop
```

Or, if you have already made changes to this account template, you can manually set the update by adding %P% to the field.

Special Characters for ADS

ADS objects cannot contain an equal sign (=) and comma (,) in the same object common name.

New ADS Accounts

Use the New Object - Account wizard to create new accounts.

Account Management - Map a Certificate to an Account Object

To map a certificate or certificates to an account object, use the Security Identity Mapping or User Certificates Tabs from an ADS Account Property Sheet.

Relocate Accounts

Using the Relocate Accounts function in the Provisioning Manager, you can specify that an account belonging to an account template be moved to the container that is specified by the container rule within the account template. You can also request that all accounts belonging to an account template be moved to their respective containers.

In ADS, the container rule selects a container based on global user attribute values that allow one account template to prescribe different containers for different accounts. The Relocate Accounts function re-evaluates the rule using current global user attribute values and moves each account to the prescribed container.

For accounts, a list of assigned account templates for the account is retrieved and if there is only one account template, an LDAP operation for the account is issued using that account template value. If there is more than one account template assigned to an account, a list of account templates to choose from is retrieved. If there are no account templates assigned to the account, an error message is displayed.

For account templates, child Relocate Accounts operations on each account currently assigned to the account template are initiated. These child operations can succeed or fail individually and the completion message for the account template Relocate Accounts operations contains statistics for the following categories:

- Updated - Account moved to new container
- Unchanged - Account already in correct container
- Failure - Problem determining container or moving account

Relocate operations can also be executed with the Batch Utility (etaultil).

- For Account Template:

```
etaultil -d <eTADomain> -u <eTAUser> p <password> update eTADSPolicyContainerName=Active Directory  
Policies,eTNamespaceName=CommonObjects,dc=<eTADomain> eTADSPPolicy  
eTADSPolicyName=<policyName> to eTRelocateAccounts=1
```

- For Account:

```
etaultil -d <eTADomain> -u <eTAUser> p <password> update  
'eTADSContainerName=Users,eTADSDirectoryName=<directoryName>,eTNamespaceName=ActiveDirectory'  
eTADSAccount eTADSAccountName=<accountName> to eTRelocateAccounts=1  
eTSyncPolicyDN='eTADSPolicyName=<policyName>,eTADSPolicyContainerName=Active Directory  
Policies,eTNamespaceName=CommonObjects,dc=<eTADomain>'
```

User Groups

Identity Manager lets you create and maintain user groups using the Endpoint Type task view. Use the Active Directory Services User Group property sheet when managing your groups.

Group Management - Changing Group Scope

The ADS connector lets you change the scope and type of groups in native mode only.

The following sections list the scopes that can be changed along with their rules for changing.

Domain Local to Universal

Domain Local groups can be converted to Universal groups, provided that the following is true:

- The domain local group is not already a member of another domain local group.
- The domain local group does not contain any other domain local groups.

Global to Universal

Global groups can be changed to Universal groups.

Universal to Domain Local

Universal groups can be changed to Domain Local groups only if the change is written to the Global Catalog (GC).

Note: If the original Universal group has any members that cannot be a member of a Domain Local group, the request to convert fails.

Universal to Global

Universal groups can be changed to Global groups.

Note: If the original Universal group has any members that cannot be a member of a Global group, the request to convert fails.

Group Management - Changing Group Type

The ADS connector lets you change the scope and type of groups in native mode only.

The following sections list the group types that can be changed along with their rules for changing.

Distribution to Security

Distribution type groups can be changed to security type groups.

Security to Distribution

Security type groups can be changed to distribution type groups, but the original group objects access privileges are lost.

Note: A warning indicates that changing a group from security type to distribution type may cause a loss of access control for the members of that group.

Microsoft Best Practices for Group Memberships

Using Microsoft guidelines are recommended when designing how group memberships should be used in ADS, especially where more than one ADS domain is involved. For more information, refer to Microsoft Windows Server 2003 Techcenter, and in particular, the following topics:

- Group Scope; and
- Global Catalog Replication

Terminal Services

The Terminal Services Tab on the Active Directory AccountTemplate and Account Property Sheets lets you configure the Terminal Services user profile, startup environment, and remote control settings and set the Terminal Services timeout and re-connection settings.

Search (read) of Terminal Services is now done in parallel to improve performance.

Connecting to the Nearest Domain Controller

Since an Active Directory domain consists of multiple domain controllers, the question to which domain controller should ADS commands be sent is now extremely important.

The ADS connector lets you choose from the following options, which domain controller to target:

- Always use the primary domain controller
- Direct the commands to the closest domain controller
- Allow the caller to manually specify the intended target domain controller

See the ADS section of the Provisioning Manager online help for more information on setting your preferences for connecting to the nearest domain controller.

Resource Management

The ADS connector lets you manage various resources, including computers, printers, and shared folders. New property pages have been created to support these resources.

Computers

The Computer property sheet lets you add and delete computers. A list of associated printers is also displayed. For creation, you can only enter the computer name and pre-Windows 2000 computer name.

A button to launch the Windows MMC Computer Management tool is also available. Since this tool supports numerous functions, the ability to perform any given function depends on the credentials of the user making the request. Clicking this button launches an additional dialog giving you three different options for selecting credential. They are as follows:

1. Use the Global-User Administrator's credentials
2. Use the Window's credentials of the logged in user
3. Use an alternate set of credentials that the user must manually enter

Note: Options 1 and 3 are only supported on Windows 2000 (or Windows XP) platforms. When running on an NT platform, these options are disabled.

Printers

The Printer property sheet lets you add and delete printers. For creation you can only enter the server and printer name. A button to launch the Windows Print Queue is also available for printer and print job management.

Shared Folders

The Shared Folders property sheet lets you add and delete shared folders. For creation you can only enter the server and share name. A button to launch the Windows MMC Computer Management tool is also available for shared folder management.

Paged Searches

Active Directory normally restricts the number of objects that can be returned in a single search operation. To ensure successful management of containers with a large number of objects (that is where the number of objects exceeds the maximum), ADS implements a paged-search operation.

Note: A page is defined as the number of objects that can be returned in a single search.

If too many objects are returned in a single page, ADS queries the Active Directory server for one page at a time. It continues to query Active Directory, until it retrieves the entire set of objects. This process is automatically handled by the ADS agent, so you do not need to control the paging operation.

Although this is normally not necessary, you can adjust the page size. To do this, you must set the environment variable `ADS_SIZELIMIT`. However, you should never set this value larger than the limit on the Active Directory server. If you set the value too large, it may negatively impact performance on the Active Directory server. (To change the value on the server, see the section, [Changing the Active Directory Search Limit](#)).

Note: The `ADS_SIZELIMIT` variable should be set on the machine where the C++ Connector Server and ADS connector run.

Change the Active Directory Search Limit

For servers with an excessive number of accounts or groups in a single endpoint, ADS automatically does a paged search to retrieve all objects, thus it should not be necessary to increase the search limits.

However, if you choose to increase this limit, you should modify the following parameters for Active Directory:

- `MaxPageSize`, the maximum page size that is supported for LDAP responses. The default is 1000 records.
- `MaxResultSetSize`, the maximum size of the LDAP result set. The default is 262144 bytes.

By default, these objects are located at:

```
CN=Default Query Policy,CN=Query-Policies,CN=Active Directory Service,CN=Windows  
NT,CN=Services,CN=Configuration
```

Increase the values of these parameters to meet your needs. If necessary, consult your ADS system administrator for recommended values.

You can use the Windows 2000 NTDSUTIL.EXE utility to modify these parameters. Start the utility and select the LDAP Policies option from the prompt. For detailed instructions to use this utility, see article Q315071 on the Microsoft web site.

Move Functions

There are several ways to move accounts and groups in ADS. The following sections describe using the Provisioning Manager, LDAPMODRDN command, and Relocate to Sync with Account Template.

Provisioning Manager

If you want to move an account or group from one Organizational Unit to another, follow these steps:

1. List the account or group
2. Right-click on the account or group to be moved
3. Select the Move menu item.

A dialog appears and you can select the Organization Unit that you want to move to.

Relocate to Sync with Account Template

Another Move task you can perform from the Provisioning Manager is the Relocate to Sync with Account Template. If your accounts have been moved around and you want to move them back to the Organizational Unit that is specified in their account template, follow these steps:

1. From the Provisioning Roles Task Frame, list the account template.
2. Right-click the account and select Relocate.

The accounts are moved into the correct Organization Unit as specified in their account template.

LDAPMODRDN Command

You can also use the following `ldapmodrdn` command to specify the new Organization Unit for an ADS account or group:

```
ldapmodrdn -h localhost -p 20389 -D "eTGlobalUserName=etaadmin,eTGlobalUserContainerName=Global Users,eTNamespaceName=CommonObjects,dc=<your_eta_domain>,dc=eta" -w <etaadmin_password> -s "eTADSOrgUnitName=<original_ou_name>,eTADSDirectoryName=<your_ads_name>,eTNamespaceName=ActiveDirectory,dc=<your_eta_domain>,dc=eta" eTADSAccountName=<account_name>,eTADSOrgUnitName=<target_ou_name>,eTADSDirectoryName=<your_ads_name>,eTNamespaceName=ActiveDirectory,dc=<your_eta_domain>,dc=eta" "eTADSAccountName=<account_name>"
```

ETAUTIL Tool

You can use the ETAUTIL tool to sync with the account template. To relocate an account to the Organizational Unit according to the account template by running the following command:

```
etautil -d <domain> -u <user> -p <pass> UPDATE '<ADS Policies Base DN> eTADSPolicy eTADSPolicyName=<policyName>' to eTRelocateAccounts='1'
```

For example:

```
etautil -d sup154 -u etaadmin -p adminadmin UPDATE 'eTADSPolicyContainerName=Active Directory Policies,eTNamespaceName=CommonObjects' eTADSPolicy eTADSPolicyName='ADSAccountPolicy1' to eTRelocateAccounts='1'
```

Move Using ETAUTIL

Important! Using the ETAUTIL tool to move accounts or groups is not recommended.

ETAUTIL does not actually move the account or group, only simulates the move by first deleting the account and then re-adding the account. The newly added account is not exactly the same as the deleted account since there is some internal data that is different in the new account. For ADS, this can be a problem so ETAUTIL is not recommended to move accounts or groups.

ADS Etautil Conventions

Use the following Active Directory Services conventions in your etautil commands:

- The endpoint type name (eTNamespaceName) is ActiveDirectory
- The endpoint type prefix is ADS. Therefore, the Active Directory Services class names are:
 - eTADSDirectory for an endpoint
 - eTADSPolicyContainer for an account template container
 - eTADSPolicy for an account template

Program Exits

Program Exits (Common or Native) let you write software that executes during certain actions that Identity Manager carries out. Program exits extend the framework of Identity Manager and allow for additional functionality that can change or augment the standard Identity Manager behaviors.

With Identity Manager Program Exits, you can write the Common Program Exits and configure them to work with ADS. In addition, ADS has its own separate program exits called The Dynamic Link Library (DLL) Interface and Command Line Interface which are described below.

Using Program Exits When Creating ADS Objects

The ADS connector can be modified to execute custom or third-party software each time an ADS object is created. You can use one of the following two methods to accomplish this task:

- Dynamic Link Library (DLL) interface
- Command line interface

The Dynamic Link Library (DLL) Interface

If you create your own Windows DLL with the name `ADSOptsExits.dll` (case-insensitive) and replace this DLL in the `%PSHOME%\Bin` directory, the ADS connector will call entry points into the DLL to execute your custom code each time an ADS object is created.

The following two optional entry points (exits) into `ADSOptsExits.dll` are called from the ADS connector:

- The `PreAdd` entry point
- The `PostAdd` entry point

The `PreAdd` entry point is called immediately before an ADS object is created and the `PostAdd` entry point is called immediately after an ADS object is created. By implementing these entry points in your version of `ADSOptsExits.dll`, your custom code is executed each time an ADS object is created by the ADS Connector. From the custom code it is possible to read the attribute values of the new ADS object.

Note: The ADS connector is a Windows DLL that is called from the C++ Connector Server process. After `ADSOptsExits.dll` is added to `%PSHOME%\Bin`, the C++ Connector Server process must be restarted.

Sample .CPP file

The following sample .CPP file demonstrates how to construct this DLL. The sample source code assumes the 32-bit edition of Visual C++. Different compilers may require different calling conventions. If Microsoft Visual Studio is used to create the DLL, the Visual Studio project should be created as a Win32 Dynamic Link Library. Use the following defaults for the project settings.

```
#include <stdio.h>
#include <windows.h>
#include <winldap.h>

BOOL APIENTRY DllMain( HANDLE hModule,
                      DWORD ul_reason_for_call,
                      LPVOID lpReserved
                      )
{
    switch (ul_reason_for_call)
    {
        case DLL_PROCESS_ATTACH:
        case DLL_THREAD_ATTACH:
        case DLL_THREAD_DETACH:
        case DLL_PROCESS_DETACH:
            break;
    }
    return TRUE;
}

#ifdef __cplusplus
#define ADS_C_PROTO      extern "C"
#else
#define ADS_C_PROTO
#endif

// this exit is called immediately prior to the creation of the ADS object
// logfile is NULL or if ADS logging is enabled
// a handle to PS_HOME\Log\ADS\TrustDirectoryName.log
ADS_C_PROTO __declspec(dllexport) int __cdecl PreAdd(const PWCHAR pszDN, LDAPModW *ppmods[], FILE
*logfile)
{
    // example to print fully distinguished name of ADS object
    wprintf(L"In PreAdd %s\n", pszDN);
    if (logfile != NULL)
        fwprintf(logfile, L"PreAdd %s\n", pszDN);
    return 0;
    // return 0 to let the ADS option create the object.
    // return -1 to stop creation of the object
}

// this exit is called immediately after the creation of the ADS object
ADS_C_PROTO __declspec(dllexport) int __cdecl PostAdd(const PWCHAR pszDN, LDAPModW *ppmods[],
FILE *logfile)
{
    // example to print fully distinguished name of ADS object
    // and list the string based attribute values used to create the object
    wprintf(L"In PostAdd %s\n", pszDN);
```

```
for (int i=0; pmods[i] != NULL; i++)
{
    if (!(pmods[i]->mod_op & LDAP_MOD_BVALUES))
    {
        // string data
        wprintf(L"Attribute name: %s\n", pmods[i]->mod_type);
        for (int j=0; pmods[i]->mod_vals.modv_strvals[j] != NULL; j++)
            wprintf(L"  Attribute value %d)\n", j, pmods[i]-
>mod_vals.modv_strvals[j]);
    }
}
return 0;
// return 0 to indicate success
// return -1 to indicate an error
}
```

The Command Line Interface

The command line interface executes Windows commands as ADS accounts are created. Unlike the DLL interface, the command line interface is not invoked during the creation of all ADS objects. This interface is only invoked when ADS objects with object class user are created.

Passing Account Attributes

Immediately before each ADS account is created, the command line interface will load the optional file %PSHOME%\bin\ADSExitUsrPreAdd.txt. Each line in the file is inspected and anything that has the form *%LDAP_attribute_name%* is replaced with the actual attribute value of that attribute. (Where *LDAP_attribute_name* is the LDAP name of the attribute on the new ADS account object.)

For example, the ADS connector is creating an ADS account with the givenName of YourName and ADSExitUsrPreAdd.txt contains the line:

```
test.bat "%givenName%.txt"
```

The command line interface replaces this line with the following and then invokes the batch file:

```
test.bat "YourName.txt"
```

Test.bat must not expect user input from the Provisioning Manager since it is running as part of the C++ Connector Server. Test.bat should be placed in the %PSHOME%\bin directory.

Similar to the processing of ADSExitUsrPreAdd.txt, the optional file %PSHOME%\bin\ADSExitUsrPostAdd.txt is processed immediately after an ADS account is created.

If a multi-valued attribute is specified by *%LDAP_attribute_name%*, the attribute values are appended together with a semicolon (;) as a delimiter between each value. If the actual data contains semicolons (;), the command line interface cannot distinguish between the data and the delimiter.

The command line interface does not support the following:

- Binary attributes
- Commands native to the Windows command line interpreter (cmd.exe) in the text file. These commands include: if, call or dir.
- These commands should be wrapped in your own .bat or .cmd file or used with the cmd /c command.

Note: The command line interface uses the Windows API CreateProcess on each line in the text file.

Passing Global User Attributes

When ADS accounts are created through an ADS account template, attributes from the global user associated with the account template can be passed to the Windows command line commands in ADSExitUsrPreAdd.txt and ADSExitUsrPostAdd.txt.

For each global user attribute that is to be passed to the ADS command line interface, a custom ADS attribute must be created. (See the section on Creating Custom ADS Attributes, for more information.) Once a custom ADS attribute is created, the attribute appears in the ADS Account Template property sheet on the Custom page. The custom attributes do not appear on account templates that do not have an ADS endpoint assigned on the Endpoint property page of the account template. A new ADS account template will not display the custom attributes until an ADS endpoint is added on the Endpoint property page and the new ADS account template is saved.

The ADS custom attributes can be assigned to any rule string representing attribute values on the global user. (For more information, see the chapter "The Provisioning Directory" in the *Provisioning Guide*.) During ADS account creation, the value of the ADS custom attribute contains the global user attribute value that is specified in the rule string. This value is accessible in ADSExitUsrPreAdd.txt and ADSExitUsrPostAdd.txt by specifying the ADS custom attribute name surrounded with the percent sign (%) delimiter. The following is an example:

```
myfile.bat %eTADSExitOnlyDiskSize%
```

Creating Custom ADS Attributes

The file *PS_HOME\data\ADS\schema.ext* is used to create custom ADS attributes.

Note: You must first create this file.

Custom ADS attributes are placed one per line in the text file *PS_HOME\data\ADS\schema.ext*. This file is used to specify any custom ADS account attributes that have been added to the ADS schema and any ADS account attributes that do not actually exist in the ADS schema but are used in *ADSExitUsrPreAdd.txt* and *ADSExitUsrPostAdd.txt*. For more information on extending the ADS schema, see the topic *Extending the ADS Schema* in the *ADS Defaults* section.

To let the ADS connector know that the custom ADS account attribute being added is to be used for exit processing and does not exist in ADS, the attribute name in *schema.ext* must begin with *eTADSExitOnly*. This prefix lets the ADS connector know that the custom attribute can be passed to the ADS connector command line interface. Since the attribute does not exist in the ADS schema, the syntax and the single or multi-value indicator must be included in the *schema.ext*. These two values follow the attribute name and are delimited with a colon (:). An example of a line in *schema.ext* follows:

```
eTADSExitOnlyDiskSize:2.2.5.12:T
```

The syntax of the attribute is 2.2.5.12. These syntaxes are defined at the following website:

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

The following characters indicate the value of the attribute:

- **T** indicates that the attribute is single-valued.
- **F** indicates that the attribute is multivalued.

Note: Any changes to the *schema.ext* require a restart of the Identity Manager Provisioning service and the Provisioning Manager.

By default, any attribute listed is assumed to exist for both contacts and accounts. To indicate that a given attribute is to be defined for accounts or contacts only, add an optional prefix (account) or (contact) to the attribute name. Optionally, you can use the prefix (both) to indicate an attribute is valid for both accounts and contacts.

For example, your installation added three new attributes to the schema. *BirthDate* is valid for accounts only. *SSN* is valid for accounts and contacts. *DoNotCall* is valid for contacts only. You can add these entries into your *schema.ext* file as follows:

- (account)BirthDate
- (both)SSN
- (contact)DoNotCall

Example Files

In this example we are creating ADS accounts through an account template. We would like to obtain the value of the homepage and office from the global user and pass it to the command line interface, but we do not want to send that information to the ADS server when the account is created.

This example requires the following files to be in the *PS_HOME\bin* directory:

- ADSExitUsrPreAdd.txt

```
precopy.bat "%CN%" "%eTADSExitOnlyHomePage%" "%eTADSExitOnlyOffice%"
```
- ADSExitUsrPostAdd.txt

```
postcopy.bat "%givenName%" "%eTADSExitOnlyHomePage%" "%eTADSExitOnlyOffice%"
```
- precopy.bat

```
copy ADSExitUsrPreAdd.txt PreADSExit.txt
echo object name is %1 >> PreADSExit.txt
echo home page is %2 >> PreADSExit.txt
echo office is %3 >> PreADSExit.txt
```
- postcopy.bat

```
copy ADSExitUsrPostAdd.txt PostADSExit.txt
echo first name is %1 >> PostADSExit.txt
echo home page is %2 >> PostADSExit.txt
echo office is %3 >> PostADSExit.txt
```
- schema.ext

```
eTADSExitOnlyHomePage:2.2.5.12:T
eTADSExitOnlyOffice:2.2.5.12:T
```

When the Identity Manager Provisioning service and Provisioning Manager are restarted using the schema.ext file shown, the attributes listed in this file will appear on the Custom property sheet of the ADS account template. The rule string %UHP% must be placed in the value field of the eTADSExitOnlyHomePage attribute and the rule string %UO% must be placed in the value field of the eTADSExitOnlyOffice attribute. When this account template is used to create new ADS accounts, the homepage and office values of the associated global user are passed to the command line interface.

Generic LDAP Connector

The Generic LDAP Connector provides a single point for all user administration by letting you do the following:

- Register endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage LDAP user entries using LDAP-specific account templates
- Change account passwords in one place
- Synchronize global users with their roles or synchronize global users' accounts with their account templates
- Assign an LDAP account template to each of your LDAP endpoints
- Use the default Endpoint Type account template to create accounts
- Create and manage LDAP groups and containers
- Generate and print reports about LDAP user entries and groups
- Administer standard user entries on LDAP servers
- Discover existing LDAP group entries
- Change group membership and ownership

This connector is managed using the Connector and C++ Server installation process.

Note: For more information and requirements, see *Connector and C++ Connector Server Installation* (see page 23).

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

LDAP Support for FIPS and IPv6

For release r12 of Identity Manager, the Generic LDAP Connector supports IPv6, but not FIPS.

LDAP Objects

LDAP organizes objects in a hierarchical directory tree. Objects are used to represent its resources. Each object is comprised of properties, and each property can contain values.

Within the LDAP object schema, the following object classes are relevant to the Identity Manager Generic LDAP connector:

Container Objects

Include objects that have subordinate objects. Container objects include, but are not limited to, objects such as Country, Organization, OrganizationalUnit, and DomainComponent.

User Entry Objects

Represent the LDAP individual accounts. The Generic LDAP connector detects any entry that subclasses the Person object as a user entry. In particular, it is tailored to manipulate and create the properties associated with the OrganizationalPerson and InetOrgPerson object classes.

Group Objects

Represent the LDAP groups. Group entries are any entries that have GroupOfUniqueNames or GroupOfNames as their object class. The Generic LDAP connector allows the manipulation and creation of either group type.

When your LDAP objects populate the Provisioning repository, your LDAP trees appear under the Endpoints resource. User entries and groups are located under their respective containers.

If an LDAP system administrator modifies the directory hierarchy of the LDAP server by adding or deleting an organization or organizational unit and objects under their respective container, Identity Manager captures these changes when you perform a new Exploration.

The position of user objects within the directory tree is specified by its context. The complete context or path from an object to the country of the directory tree identifies and forms the object's hierarchical name, for example, CN=xx, OU=yy, O=zz.

Note: All hierarchical names must be unique within the directory tree. In addition, all container and object names must be unique within that container.

Error when Changing Password on Native LDAP System

When the password on the directory endpoint is changed using native tools, then the Provisioning Server's record of the password will be out of sync until the new password is entered using the Admin Manager.

Also, the following fields on the General tab are cleared because the passwords are out of sync. Complete the following fields before submitting the new password:

- LDAP Version
- Base DN
- Use TLS

Creating LDAP User Entries

To create a person in an LDAP server, you must register him or her. When you register a user, you create a Person entry in the remote LDAP server.

In Identity Manager, you register users by dragging them onto a provisioning role which has an LDAP account template defined and an LDAP endpoint associated with the account template.

Alternately, you can create a user under any container by using the Endpoint Content dialog and clicking New.

How to Acquire an LDAP Server

Acquiring an endpoint is the first task you must perform before you can administer an LDAP server. When acquiring an endpoint, you perform the following steps from the Endpoint Type task view:

1. Register the server as an endpoint in Identity Manager.

Use the LDAP Endpoint property sheet to register an LDAP server. During the registration process, Identity Manager identifies the LDAP server you want to administer and gathers information about it.

2. Explore the objects that exist on the endpoint.

After registering the server in Identity Manager, you can explore its contents, using the Explore and Correlate Endpoint dialog. The Exploration process finds all LDAP accounts, groups, and containers. You can correlate the accounts with global users at this time or you can correlate them later.

The LDAP Connector scans for all users with a subclass of Person and makes their properties in the OrganizationalPerson and InetOrgPerson objects available for administration through the Provisioning Manager interfaces. For example, for a user named CAPerson with a subclass of Person, the user's properties that have the same name as orgPerson or inetOrgPerson are the only properties available for administration.

3. Correlate the explored accounts with global users.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the LDAP user ID with each existing global user name. If a match is found, Identity Manager associates the LDAP account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the LDAP display name with each existing global user's full name. If a match is found, Identity Manager associates the LDAP account with the global user. If a match is not found, Identity Manager performs the next step.
- c. If the Create Global Users as Needed button is checked, Identity Manager creates a new global user and then associates the LDAP account with the global user. If the Create Global Users as Needed button is unchecked, Identity Manager performs the next step.
- d. Identity Manager associates the LDAP account with the [default user] object.

Enhanced Filtering

The Generic LDAP connector supports an enhanced filtering facility when exploring and correlating accounts. This facility enables the explore process to return a subset of the full accounts. Repeating the explore process with a sequence of different filters processes all the accounts. This facility can be useful for systems with limited amounts of memory.

Currently the filter is limited to the use of the eTLDAAccountName attribute. The agent creates a filter from the content of this attribute to match entries named by either uid or cn.

An example of an explore request using a filter to limit the search result to only account names that start with the letter "A":

```
ldapsearch -h %eta_server% -p 20389 -D "" -w ""  
-b "eTLDAContainerName=Administration,  
eTLDAContainerName=%eta_container%,eTLDADirectoryName=%eta_dirname%, eTNamespaceName=LDAP  
Namespace, dc=%eta_domain%, dc=eta"  
-s one "(eTLDAAccountName=A*)"  
eTExploreUpdateEtrust
```

The algorithm used by the agent is as follows:

1. The account handling module parses the filter passed down from the C++ Connector Server and looks for the first occurrence of eTLDAAccountName.

Example of a filter received by the account handling module:

```
(&(&(objectClass=eTLDAAccount))(eTLDAAccountName=<filter>))
```

2. If the attribute is found, the account handling module extracts the filter from the right hand side of the '=' until the first ')'.
3. The account handling module then uses this filter value to add the following filter to the default one:

```
((uid=<filter>)(cn=<filter>))
```

The default filter is:

```
((objectclass=person)(objectclass=residentialPerson)(objectclass=organizationalPerson)(objectclass=inetOrgPerson))
```

If an enhanced filter request is present, the updated filter would look like this:

```
(&(((objectclass=person)(objectclass=residentialPerson)(objectclass=organizationalPerson)(objectclass=inetOrgPerson))((cn=<filter>)(uid=<filter>))))
```

LDAP Default Account Templates

LDAP Default Account Templates are provided with the LDAP connector. You can use them as a model to create new account templates. These default account templates are as follows:

- LDAPDefaultAccountPolicy-NamedByCN
- LDAPDefaultAccountPolicy-NamedByUID
- LDAPDefaultAccountPolicy-OrgPerson

The first two account templates are identical, except that the first names new entries by their Common Name, and the second names them by their User ID. The Common Name entries are mapped to the Full Name of the global user, and the User ID entries are mapped to the Account Name of the global user. The third policy names new entries by their Common Name, which is mapped to the Account Name of the global user. The User ID entry is not enabled because it does not exist in an Organizational Person entry.

Identity Manager lets you manage LDAP account templates from the Provisioning Roles task view.

LDAP Accounts

For the LDAP connector, the Endpoint Type task view lets you manage user entries on an LDAP endpoint. User entries are mapped to LDAP accounts within Identity Manager.

LDAP Groups

The Endpoint Type task view lets you directly create and maintain LDAP groups. Use the LDAP Group property sheet to manage your groups. You can also use LDAP account templates to maintain groups automatically.

LDAP Containers

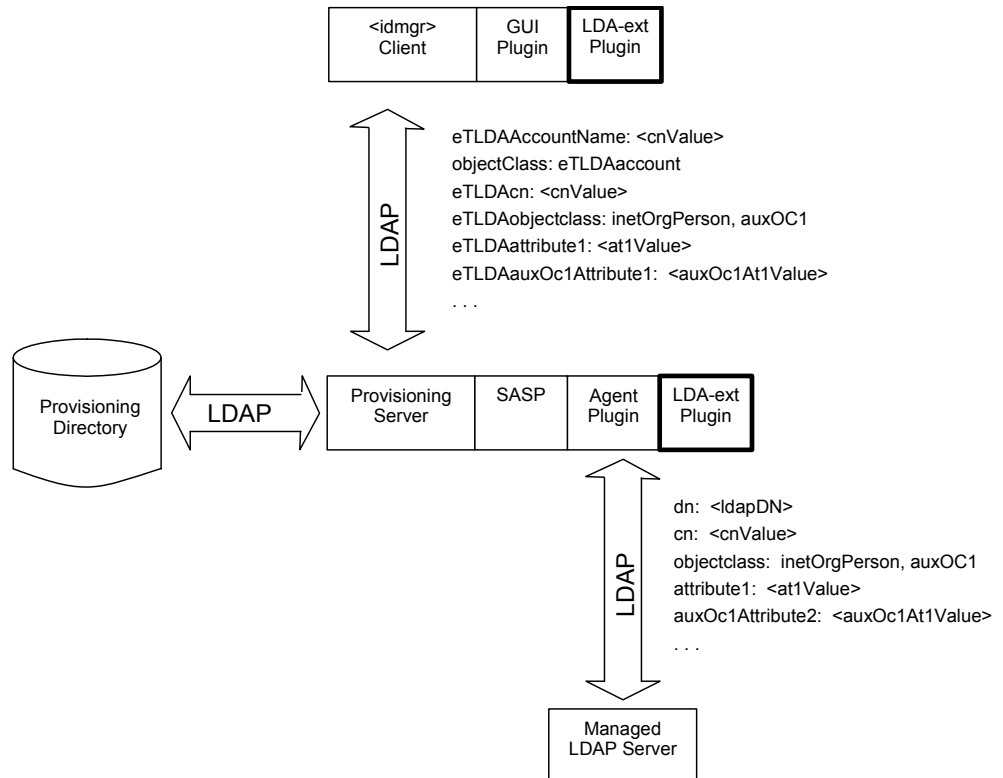
To create and maintain LDAP containers, use the Endpoint Type task view. Use the LDAP Container property sheet when managing your containers. Containers are created with the `organizationalUnit` object class. The organizational unit name is used as the naming attribute. This is the only object class available.

Extending Custom Attributes

For more information on extending custom attributes, see the "Extending Custom Attributes on Endpoints" section in Chapter 3 of the *Provisioning Guide*.

Extending the Generic LDAP Connector

You can extend the Generic LDAP connector so it can manage additional auxiliary object classes. If you want to extend it, you must write an extension definition for each auxiliary object class, build a new set of connector plug-ins, and update the schema. The connector plug-ins are runtime loadable DLLs that contain the extensions to be supported on LDAP accounts. One plug-in is used by the client through the Provisioning Manager plug-in and the other is used by the Provisioning Server through the Agent plug-in.



Important! `internal_LDAP_attribute_name` must contain only alphanumeric characters and dash (-). All other characters are invalid.

For more information, see *the Programming Guide for Provisioning*.

How to Extend the Generic LDAP Connector

The major steps to extend the Generic LDAP connector are:

1. Create an extension definition file for each auxiliary object class.
2. Build the Extended LDAP connector plug-ins.
3. Add the Extended LDAP connector plug-in objects to the Provisioning Directory.
4. Update the schema to include the extended attributes.
5. Restart Services.

Step 1. Create Extension Definition Files

Each new auxiliary object class requires a definition file that describes the object class and each of its attributes.

The following is an example of a definition file for a fictitious auxiliary object class, `companyPerson`. This file should be saved in `PSSDKHOME\Samples\LDAP-SDK\companyPerson.txt`.

```
#####
# companyPerson
#
# LDAP definition:
#
# objectclass companyPerson
#     oid companyPerson-oid
#     superior inetOrgPerson
#     allows
#         middleInitial,
#         department,
#         departmentGroup,
#         subDepartment,
#         facilityLocation
#####
#####
# 1. EXTENSION FILE HEADER
#####

# 1.1 EAO2 Object Type
objectType:         account

# 1.2 Auxiliary Object Class Name
AuxiliaryObjectClass: companyPerson
```

#####

2. ATTRIBUTES

#####

attribute: middleInitial
syntax: dirString
description: Middle initial
guiLabel: Middle initial
guiFlags: minLen=0, maxLen=1
guiControl: edit

attribute: department
syntax: dirString
description: Department
guiLabel: Department
guiControl: edit

attribute: departmentGroup
syntax: dirString
description: Department Group
guiLabel: Department group
guiControl: edit

attribute: subDepartment
syntax: dirString
description: SubDepartment
guiLabel: SubDepartment
guiControl: edit

attribute: facilityLocation
syntax: dirString
description: Location of nearest facility
guiLabel: Location
guiFlags: values="Location1|Location2|Location3"
guiControl: comboBox

#####

3. DISPLAY PARAMETERS

#####

#####

3.1 Property Page Name

displayPage: Company Page

#####

3.2 Display Attributes

property: middleInitial
property: department
property: departmentGroup


```
property:      subDepartment
property:      facilityLocation
```

```
#####
# 4. END OF FILE
#####
```

Step 2. Build LDAP Connector Plug-ins

Perform the following procedure to build the extended LDAP connector plug-ins.

To build Extended LDAP Connector plug-ins

1. Open a DOS screen.
2. Change the DOS prompt to the following directory:
PSSDKHOME/Samples/LDAP-SDK
3. Edit the Makefile file, and update the definition of LDAP_EXTENSIONS to include the extension definition files needed to extend the object. The order in which the files are listed corresponds to the order of the property pages in the object's property sheet.

Note: Verify that all your extension definition files are in %PSSDKHOME%/Samples/LDAP-SDK.

4. Run NMAKE.

Note: Verify that your development environment settings are correct, ensure that DEVROOT is correctly set, and run the VCVARS32.BAT file before you run NMAKE.

The following files are generated, which you can find under the *PSSDKHOME\bin* and *PSSDKHOME\data* directories:

- LDAGuiPlugin.DLL
- LDANamespacePlugin.DLL
- LDAParse.PTT

Step 3. Add Extended LDAP Connector Objects to the Provisioning Directory

Perform the following procedure to add the extended LDAP connector objects to the Provisioning Directory.

To add the extended LDAP connector objects to the Provisioning Directory

1. Double-click the Services icon in the Control Panel. The Services window appears.
2. Select Identity Manager Provisioning service and click Stop.
3. Click Close.
4. Close and exit the current Provisioning Manager session.
5. Copy the following files from *PSSDKHOME\bin* to *PS_HOME\bin*:
 - LDAGuiPlugin.DLL
 - LDANamespacePlugin.DLL
6. Copy the LDAParse.ptt file from *PSSDKHOME\data* to *PS_HOME\data*.

Note: You should save a copy of the original LDAParse.ptt for recovery purposes.

Step 4. Update the Schema to Include Extended Attributes

Perform the following procedure to update the schema to include the extended attributes.

To update the schema to include extended attributes

1. Open a DOS screen and change the directory to *PS_HOME\data*.
2. Run the following command:

```
..\bin\schemagen -genoid -n lda
```
3. Edit *PS_HOME\Data\etrust_admin.conf*, making sure that the *etrust_lda.schema* file is included.
4. Edit *PS_HOME\Data\etrust_admin.schema* to remove all LDA definitions.
5. Move the file *PS_HOME\Data\etrust_lda.dxc* to the *DXHOME\config\schema* folder.
6. Edit *DXHOME\config\schema\etrustadmin.dxc* , making sure that it includes the following line:

```
source "etrust_lda.dxc";
```
7. Edit *DXHOME\config\schema\etrust_admin.dxc* to remove all LDA definitions.
8. Run the following command so that eTrust Directory can read the configuration:

```
dxserver init etrustadmin
```
9. Start the Identity Manager Provisioning service.

Step 5. Restart the Services

You must restart the services that were previously stopped.

To restart the services

1. Double-click the Services icon in the Control Panel. The Services window appears.
2. Select Identity Manager Provisioning Service and click Start.
3. Click Close.
4. Close and exit from your current Provisioning Manager session.

How to Write Auxiliary Object Class Definition Files

Auxiliary object class definition files are used when extending the Generic LDAP connector's Account object. You can define the following:

- The name of the auxiliary object class
- Attributes that are managed on the property page
- Information on sequence and how each of the attributes is displayed

The source for the auxiliary object class definition file consists of keywords, properties, and comments. When creating your own definition file, you must perform the following steps:

1. Define the auxiliary object class.
2. Define the attributes within this object class.
3. Define the order of the attributes displayed on the property sheet.

The following is an example:

```
#####
# companyPerson
#
# LDAP definition:
#
# objectclass companyPerson
#         oid companyPerson-oid
#         superior inetOrgPerson
#         allows
#             middleInitial,
#             department,
#             departmentGroup,
#             subDepartment,
#             facilityLocation
#####

#####
# 1. EXTENSION FILE HEADER
#####

# 1.1 EAO2 Object Type
objectType:         account

# 1.2 Auxiliary Object Class Name
AuxiliaryObjectClass: companyPerson
```

```
#####
```

```
# 2. ATTRIBUTES
```

```
#####
```

```
attribute: middleInitial
syntax:      dirString
description: Middle initial
guiLabel:    Middle initial
guiFlags:    minLen=0, maxLen=1
guiControl: edit
```

```
attribute: department
syntax:      dirString
description: Department
guiLabel:    Department
guiControl: edit
```

```
attribute: departmentGroup
syntax:      dirString
description: Department Group
guiLabel:    Department group
guiControl: edit
```

```
attribute: subDepartment
syntax:      dirString
description: SubDepartment
guiLabel:    SubDepartment
guiControl: edit
```

```
attribute: facilityLocation
syntax:      dirString
description: Location of nearest facility
guiLabel:    Location
guiFlags:    values="Location1|Location2|Location3"
guiControl: comboBox
```

```
#####
```

```
# 3. DISPLAY PARAMETERS
```

```
#####
```

```
#####
```

```
# 3.1 Property Page Name
```

```
displayPage:    Company Page
```

```
#####
```

```
# 3.2 Display Attributes
```

```
property:       middleInitial
property:       department
```

```
property:      departmentGroup
property:      subDepartment
property:      facilityLocation
```

```
#####
# 4. END OF FILE
#####
```

Definition File Names

The standard naming convention for auxiliary object class definition files is `XXXX.txt`, where `XXXX` is the name of the auxiliary object class.

The files should be installed in the `%PSSDKHOME%/Samples/LDAP-SDK` folder.

Extend Definition File Parser

You can generate the stubs manually instead of using the provided Makefile.

To manually generate the stubs

1. Open a DOS prompt.
2. Change the DOS prompt to the following directory:

PSSDKHOME/Samples/LDAP-SDK

3. Run the createHeaders.exe utility as follows:

```
bin/createHeaders.exe extension_input_file1 extension_input_file2 ...
```

To display a list of all the createHeaders command line options, enter the following:

```
Bin/createHeaders -h
```

The following command line options appear:

Usage: ./createHeaders.pl [switches] [--] extension1 [extension2, ...]

-d

Enables the printing of debugging messages

-h

Displays the usage information

-n endpoint type

Specifies a three-letter abbreviation for Endpoint Type name. It is recommended not to use this option.

-u string

Specifies the string to use when substituting the underscore ('_') characters in an LDAP attribute when generating the corresponding Identity Manager LDAP attribute name. The default is to strip them out.

-v

Enables verbose messages

extension1, 2. . .

Specifies the names of the auxiliary object class extension files. The files should be listed in the order that you want the property pages to appear.

Extension File Header

The first statement in the definition file must be an OBJECTTYPE statement. This statement specifies the object type that is being extended.

Note: Only account type objects are supported. In the future, groups and containers may be extended using the same mechanism.

The OBJECTTYPE statement has the following format:

```
OBJECTTYPE: object_type
AUXILIARYOBJECTCLASS: ldap_objectclass_name

[ SUSPENSIONATTRIBUTE: ldap_attribute_name ]
[ SUSPENSIONACCOUNTACTIVEVALUE: activation_value ]
[ SUSPENSIONACCOUNTDISABLEDVALUE: disabled_value ]
```

OBJECTTYPE: object_type

Required keyword. Specifies the object being extended. Currently, only objects of type 'account' are supported.

AUXILIARYOBJECTCLASS: ldap_objectclass_name

Required keyword. The LDAP name of auxiliary object class as defined in the schema of the LDAP Server.

SUSPENSIONATTRIBUTE: ldap_attribute_name

Optional keyword. The LDAP name of the attribute which conveys the suspension facility on the LDAP endpoint.

SUSPENSIONACCOUNTACTIVEVALUE: activation_value

Required keyword if SUSPENSIONATTRIBUTE is defined. The value to set the suspension attribute to indicate that the account is active.

SUSPENSIONACCOUNTDISABLEDVALUE: disabled_value

Required keyword if SUSPENSIONATTRIBUTE is defined. The value to set the suspension attribute to indicate that the account is disabled.

Attribute Definitions

This section of the definition files defines the LDAP attributes associated with the auxiliary object class. As a minimum, only the attributes intended to be managed by Identity Manager must be included.

Each attribute definition has the following format:

```
ATTRIBUTE: ldap_attribute_name
SYNTAX: string_type
[LENGTH: integer]
[FLAGS: additional_properties_list]
[DESCRIPTION: report_value]
[GUI LABEL: gui_attribute_label]
[GUI FLAGS: display_properties_list]
GUICONTROL: display_attribute
[GLOBALUSER: internal_LDAP_attribute_name]
```

ATTRIBUTE: *ldap_attribute_name*

Required keyword. The LDAP name of the attribute.

SYNTAX: *string_type*

Required keyword. Specifies the type of string data for this LDAP attribute. The valid string types are as follows:

- **dirString**-String of type directory string.
- **octString**-String of type octet string.
- **ia5String**-String of type IA5 string.

LENGTH: *integer*

Optional keyword. Indicates the maximum length of a string in characters or binary value in octets for this LDAP attribute. This should usually be set to the upper-bound value of an attribute's LDAP definitions (as defined in its schema).

FLAGS: *additional_properties_list*

Optional keyword that sets additional properties and characteristics on the LDAP attribute. The value of *additional_properties_list* is a comma-delimited list of one or more of the following:

- **mandatory**-Indicates that the attribute must be in the LDAP definition of the auxiliary object class.
- **singleValue**-Indicates that the attribute can only have a single value.
- **virtual**-Indicates that this attribute does not exist in LDAP. Reserved for future use.

DESCRIPTION: *character_string*

Optional keyword used to report error messages. The Provisioning Server uses this value when reporting errors while validating attribute values.

GUILABEL: *gui_attribute_label*

Optional keyword used to label the field in the Provisioning Manager's object property page. The property page automatically adds the trailing colon (":") when displaying the attribute.

GUIFLAGS: *display_properties_list*

Optional keyword. Sets the display properties of this attribute when being displayed in the Provisioning Manager's object property page. The value of *display_properties_list* consists of a comma-separated list of properties. The following properties are available:

- **none**-indicates that there are no flags set.
- **readOnly**-indicates that the attribute can not be edited in the Provisioning Manager.
- **mandatory**-indicates that this attribute must have a value. The Provisioning Manager prefixes the '*' character to the attributes Provisioning Manager label.

Note: Use with caution. The Provisioning Server will not allow operations to go through unless all mandatory attributes are filled. The auxiliary object class must also be present in the entry when add or modify operations are performed on the object which is particularly crucial when creating or updating LDAP account templates.

- **multiValued**-Indicates whether the attribute can be assigned more than one value.
- **minLen=integer**-Minimum length of a string.
- **maxLen=integer**-Maximum length of a string. While this property is optional, you should specify it so that you can query Manager and set the maximum length for an edit box. This prevents administrators from entering too many characters in a field.

Note: The min and max lengths apply to both account and policy objects. This may restrict the values of the account template object, especially when specifying long rulestrings.

- **values=value_list**-Defines valid property values or alias values. Multiple properties can be specified for a single attribute. When the alias value is specified, it is converted to the real value before storing. The format of *value_list* is a '|' -separated sequence of *choice*, which has the following EBNF:

choice ::= valid_string or valid_string '=' alias_string

Note: The use of this facility strictly limits the values to which the corresponding attribute may be set. If the account template object requires a rulestring, it may be necessary to add it to the values list.

- **numeric=low_integer|high_integer**-Specifies the minimum and maximum values for an LDAP attribute storing integers.
- **default=default_value**-Provides a default value for an attribute if none is specified during an ADD operation. This feature is used only when creating new objects, and not when adding the attribute to an existing object.

Note: Use with caution. The use of the default parameter forces the Provisioning Server to set the subject attribute to this value. If the auxiliary object class is not present in the object a failure occurs. This is particularly crucial when creating LDAP account templates.

GUICONTROL: *display_attribute*

Required keyword. Specifies the Provisioning Manager control used to display the content of the attribute in the object's property page. A valid *display_attribute* is one of the following:

- **edit**-Standard single-line text edit field.
- **numEdit**-Standard single-line text edit field restricted to numeric characters.
- **textBox**-Multi-line text edit field. Useful for comments and longer text values.
- **listBox**-List control to display and manipulate multi-valued attributes.
- **comboBox**-Pull-down selection control. The values displayed in the pull-down control are taken from the *values* property of the GUIFLAGS keyword.
- **button**-Control for boolean values: '1' when set, '0' when unset.
- **spinButton**-Control for integer values. The *minLen* and *maxLen* properties of the GUIFLAGS keyword should be set.

GLOBALUSER: *internal_LDAP_attribute_name*

Optional keyword. Applies only to extensions to objects of type ACCOUNT. This property identifies the relationship between an extended account and a global user. If, during the correlation process, a new global user must be created for the account, the global user attribute that is identified here is initialized to have the same attribute value as the account.

Display Parameters

This section of the definition file defines the parameters used to display the LDAP attributes that have been defined.

This section has the following format:

DISPLAYPAGE: *property_page_label*
PROPERTY: *ldap_attribute_name*

DISPLAYPAGE: *property_page_label*

Specifies the string used to label the Provisioning Manager's object property page.

PROPERTY: *ldap_attribute_name*

Specifies the name of the LDAP attribute as defined in the attribute definition section. This keyword is repeated for each attribute to be displayed. The order in which the keywords appear dictates the order of the attributes on the property page.

Note: Both DISPLAYPAGE and PROPERTY are required keywords.

NDS Connector

The NDS connector provides a single point for all user administration by letting you do the following:

- Register NDS trees, explore them for objects to manage, and correlate their accounts with global users
- Administer accounts and groups on Novell Directory Services (NDS) on all available versions of Netware servers, up to 6.x.
- Create and manage NDS accounts using NDS-specific account templates
- Change account passwords and account activations in one place
- Synchronize global users with their roles or synchronize global users' accounts with their account templates.
- Assign an NDS account template to each of your NDS endpoints
- Use the default Account Template to create accounts with the minimum level of security needed to access an NDS endpoint
- Manage NDS groups from the NDS tree
- Assign NDS groups to accounts
- Generate and print reports about NDS accounts and groups

This connector is managed using the Connector and C++ Server installation process.

Note: For more information and requirements, see *Connector and C++ Connector Server Installation* (see page 23).

Additional NDS Install Requirements

The following is needed to use the NDS connector:

Novell Client for Windows, v 4.7 or later must be installed on the machine running the C++ Connector Server that manages the NDS connector.

The Novell Client latest version can be downloaded from the download.novell.com site.

NDS Support for FIPS and IPv6

For release r12 of Identity Manager, the NDS Connector does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

NDS Trees

In Identity Manager, an NDS tree is an NDS endpoint.

Acquire an NDS Tree

You must acquire the NDS Tree before you can administer it with Identity Manager. In order to acquire an NDS Tree, perform the following steps from the Endpoint Type task view:

1. Register the NDS tree as a tree in Identity Manager, using the NDS Tree property.

Note: Verify that the administrative account is authorized to manage the NDS Tree. If this account does not have the proper privileges, you cannot manage the NDS Tree in Identity Manager.

2. Explore the objects that exist in the tree.

After registering the Tree in Identity Manager, you can explore its contents using the Explore and Correlate Endpoint dialog. The Exploration process identifies the hierarchy of the NDS Tree and finds all NDS accounts and groups in this hierarchy. You can correlate the accounts with global users at this time, or later.

3. Correlate the explored accounts with global users.

When correlating accounts, you can specify which fields are matched with global user fields. By default, when you correlate accounts, Identity Manager creates or links the accounts in an endpoint with global users, as follows:

- a. Identity Manager attempts to match the NDS login name with each existing global user name. If a match is found, Identity Manager associates the NDS account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the NDS full name with each existing global user's full name. If a match is found, Identity Manager associates the NDS account with the global user. If a match is not found, Identity Manager performs the next step.
- c. If the Create Global Users as Needed button is checked, Identity Manager creates a new global user and then associates the NDS account with the global user. If the Create Global Users as Needed button is unchecked, Identity Manager performs the next step.
- d. Identity Manager associates the NDS account with the *default user* object.

4. Re-exploring NDS Trees.

If you want that changes made directly to the tree hierarchy, such as adding or removing an organization using native tools, be taken into account during a re-exploration, you must perform one of the following steps, *prior to re-exploring*:

- Restart the C++ Connector Server service from the Service Manager.
- Open a command prompt and enter the following commands:

```
net stop im_ccs
```

```
net start im_ccs
```

NDS Default Account Templates

The NDS Default Account Template, provided with the NDS connector, can be used as a model to create new account templates.

NDS Accounts

An NDS account is also known as Login. When viewed in a new window, the NDS Account property sheet is labeled NDS Login.

NDS Groups

Identity Manager lets you create and maintain NDS groups from the Endpoint Type task view, using the NDS Group property sheet to manage your groups.

NDS Containers

NDS has several different container objects that can be used to organize NDS accounts and groups. Because some of these containers are optional, each NDS environment can be different. Whatever containers your environment uses, you can use the Endpoint Content dialog to view basic information about them. The following list describes all the possible NDS containers and the property sheets that are available in Identity Manager:

- **Country**-Optional container that designates the country where the network site is located. Use the Country property sheet to view information about the country.
- **Locality**-Optional object that designates a portion of the network. Use the Locality property sheet to view information about the locality.
- **Organization**-Mandatory object that organizes objects according to accounts and groups at a certain location. Use the Organization property sheet to view the organizations within an NDS endpoint.
- **Organizational Unit**-Optional object that organizes the account and groups objects into units. Use the Organizational Unit property sheet to view the organizational units within an organization.

Each of these objects is displayed in the list view as shown in the following table:

Object	Appears In List View As
Country	C:country_name
Locality	L:locality_name S:state_name
Organization	O:organization_name
Organizational Unit	OU:organizational_unit_name

For more information, see the following:

- *Identity Manager Provisioning Guide*
- The Working with Endpoints NDS topic in the *Procedures* help

Chapter 9: Non-Managed System Connector

This section contains the following topics:

[Universal Provisioning Connector](#) (see page 243)

The following connector is considered to be a non-managed system connector:
Universal Provisioning

Note: The UPO connector operates in one of two modes; managed and non-managed.

The information in this section describes how to install and manage this connector.

Universal Provisioning Connector

Universal Provisioning Connector (UPO) provides a mechanism for Identity Manager to invoke user-specified external programs when user-provisioning requests are received.

UPO Installation Requirements

Before beginning the installation, install and configure Provisioning Server and any connectors required to manage your accounts.

If you are going to work in non-managed mode and use the SendMail exit, you need the details of an SMTP server that can be used to submit emails.

More information:

[SendMail Program Exit](#) (see page 255)

Install UPO

The UPO Connector is installed using the Connector and C++ Connector Server installation process. For more information on this installation process, click [here](#) (see page 23).

UPO and UPO Monitoring Server Post Installation Step

After UPO has been installed, the automatically installed global user eTUpAd, is set to Suspended. eTUpAd must be activated before UPO or the UPO Monitor Service is used and at the same time, the password must be changed to something you will remember.

How to Configure UPO

After installing your Universal Provisioning client, perform the following steps to configure the configure:

1. Create a new endpoint.
2. Create one or more exit objects.
3. Add the appropriate exit references to the endpoint.
4. Explore the endpoint.
5. Create a suitable account template and provisioning role.
6. Assign global users to the provisioning role and account template.

Create a UPO Endpoint

In the Provisioning Manager, the Endpoint Configuration property sheet is used to configure an endpoint object.

To create a UPO endpoint

1. Select an object type of UPO Endpoint in the Endpoint Types task frame of Provisioning Manager, and click New.

The UPO Endpoint dialog appears. You must select or enter the following data:

Endpoint Name

Specifies a name for identifying the endpoint object.

Comments

Provides comments that you might have about this endpoint object.

Operating Mode

Includes the following options:

- Non-Managed
- Managed

Authentication Group Box

Provides the details of the Identity Manager global user and the Provisioning Server that this endpoint will use to manage the accounts. The connector utilizes this user to bind to and manage the content of the repository through the selected Provisioning Server.

Service Level Agreement Group Box

The exceeded delay provides the number of seconds an operation can be in the pending state before the UPO Monitor invokes the SLA-exceeded Exit.

The warning delay sets the number of seconds that an operation can be in a pending state before the UPO Monitor invokes the SLA-exceeded warning Exit to forewarn of an impending SLA breach. The Warning delay should be smaller than the SLA exceeded delay, otherwise the warning will never be called.

Setting these fields is optional, and are only relevant only to non-managed mode operations. If you want to monitor the endpoint for compliance with a service level agreement, the Exceeded delay should be set to a non-zero value.

2. (Optional) Select the default account template for this endpoint object from the drop-down list in the Default Account Template field.
3. Click OK to have the endpoint object created. It appears in the list box on the left.

UPO Support for FIPS and IPv6

For release r12 of Identity Manager, the UPO Connector does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

UPO Operational Modes

Two operational modes are provided by UPO:

- **Non-Managed Mode (Asynchronous Mode)**--This mode is for, but not limited to, those Endpoint Types with which Identity Manager does not have an existing direct management interface.

The non-managed mode uses asynchronous exits and lets UPO send requests alerting the system administrator of a non-managed system. The system administrator is then expected to carry out the request on the target endpoint type, and, after it is completed, manually update the Identity Manager account request status to Completed.

In the non-managed mode, UPO includes a SendMail program exit that sends an email message to one or more recipients, who should be system administrators of the non-managed system. UPO can also invoke user-created exits.

- **Managed Mode (Synchronous Mode)**--The UPO's managed system mode calls exits in a synchronous fashion and is designed to provide a direct management interface to remote endpoint types by interpreting the result of the external program invocation as the success or failure of the requested user-provisioning operation.

This mode provides a simpler alternative for developing connectors for new endpoint types that may not need the whole functionality provided by the SDK or ODBC-SDK.

Note: Identity Manager SDK includes a sample program exit that implements a connector to the SDK endpoint type data.

UPO Running in Managed Mode Account Creation Issue

If you created a UPO account in a UPO endpoint running in "Managed" mode, but the account was removed after an exploration, this might be due to the UPO Program Exits Reference not being correctly set up. You will need to check to make sure the UPO Program Exits Reference is set up correctly. This also applies to other functions related to UPO account operations.

Data Passed to Program Exits

In both managed and non-managed operation modes, the data passed to the exit depends on the value(s) set in the user data field of the UPO account or account template property sheet. Global user properties are mapped to the data field by using the appropriate rule strings. The user data field can also contain user-specific data, as deemed necessary. UPO operation-specific information is also passed to the invoked exit.

As a minimum, the content of the exit payload conveys the following information about the account owner:

- The account name
- An optional password for new accounts
- Any optional user data

Exits

Program exits let you execute custom code during certain Identity Manager actions. Program exits extend the framework of Identity Manager, permitting additional functionality that can change or augment the standard functionality.

Exit Types

UPO provides support for the following types of exits:

- Identity Manager Common Exits by means of the Program Exit References tab in the Endpoint Configuration property sheet.
- UPO Exits by means of the UPO Program Exit Reference tab.

Common Program Exits

Identity Manager common exits are for PRE- and POST- operations, while UPO exits apply to the actual operations.

For more information, see the *Programming Guide for Provisioning*.

UPO Program Exits

In the non-managed mode, thirteen entry points are available (including three for SLA Monitor Service) and seven are available in the managed mode. This provides complete coverage where custom code can be referenced.

For example, in non-managed mode, you may want to send an email message every time a UPO account is created. You could write a program exit containing code that submits an email message, and have that program exit run whenever a UPO account is created.

Service Level Agreement (SLA)

The Service Level Agreement (SLA) specifies the time, in seconds, that an account can have an operation in pending state.

More Information

[Non-Managed Mode Operations](#) (see page 259)

Program Exit Objects

Program exits are referred to in terms of the operation that Identity Manager is used to perform.

Program exits have a single common interface for their calling structure. This interface consists of a single argument and a single return value. The single argument is an XML buffer representation, encoded in Unicode Transformation Format 8 (UTF-8), of the object being acted upon combined with any custom information from the definition of the exit. The return value is status information on the result of the program exit execution, as well as any documented custom information that is required for the particular exit.

For further instructions on writing program exits, see Writing Common Program Exits and Writing UPO Program Exits in the *Identity Manager C++ Developer Guide*.

More Information

[Creating a SendMail Program Exit](#) (see page 258)

How to Add the Exit Reference to the UPO Endpoint

Use the Property Exit References page to define exits that can be performed from the UPO endpoint. References to the exit types can be created, and more than one exit of a given type can be defined.

The following groups of exits are available:

- Exits common to the non-managed and managed modes
- Exits unique to the non-managed mode
- Exits unique to the managed mode

Non-Managed and Managed Modes Exits

The following exits that can be referenced to the endpoint are available:

INVOCATION_ERROR

Called by UPO when a previous exit returned an unsuccessful result.

ADD_ACCOUNT

Called by UPO when a new account is invoked.

DELETE_ACCOUNT

Called by the DELETE request. The account name is the only account attribute available to this exit.

MODIFY_ACCOUNT

Called by UPO when an account modification is invoked. The account information passed to the MODIFY account request is also passed to this program. In the non-managed mode, the password attribute is omitted, as it is handled by UPO_CHANGE_ACCOUNT_PASSWORD exit type.

RENAME_ACCOUNT

Called by the MODIFYRDN request. The current account name and the new account name are the only account attributes available to this exit.

Exits Unique to the Non-Managed Mode

The following exits are available to non-managed systems only:

CHANGE_ACCOUNT_PASSWORD

This is a special case of MODIFY. This exit is called when the UPO receives a password change attribute for the account. If the password attribute is the only change, no other modify code is triggered. If other attributes change, this code is triggered in addition. The account name and the password attributes contain the only information available to the program exit.

ENABLE_ACCOUNT

Called when the "enable" attribute for the account changes. If the "enable" attribute is the only change, no other modify code is triggered. If other attributes change, this is triggered in addition. The account name is the only account attribute available to this exit.

DISABLE_ACCOUNT

This value is a special case of MODIFY. This exit is called when the "disable" attribute for the account changes. If the "disable" attribute is the only change, no other modify code is triggered. If other attributes change, this is triggered in addition. The account name is the only account attribute available to this exit.

REQUEST_PENDING

Called by UPO when a MODIFY, ADD, or DELETE request has been successfully sent out to the administrator of a non-managed system. The request status of the account is set to "pending."

REQUEST_COMPLETED

Called by UPO when a MODIFY request successfully changes the request status from "pending" to "completed."

SLA_EXCEEDED

Called by the UPO Monitoring service when pending requests that exceed the Service Level Agreement time that was set for the corresponding directory are found.

SLA_WARNING

Called by the UPO Monitoring service when pending requests that exceed the SLA warning time that was set for the corresponding directory are found.

SLA_TIMEOFFSET

Called by the UPO Monitoring service before searching for outstanding pending accounts. The called program returns an offset, in seconds, that is subtracted from the current time, when calculating the time period. This program is required to return a zero (0) if there is no offset (during working hours) or an offset until the next work day. This lets SLA_EXCEEDED and SLA_WARNING exits skip non-work hours, weekends, and holidays.

Exits Unique to the Managed Mode

The following exits are available to managed systems only.

READ_ACCOUNT

Called when a SEARCH for a UPO account is requested. The account name is the only account attribute available to this exit. The exit returns the properties of the account in the user table.

LIST_ACCOUNTS

Called when a SEARCH for enumerating accounts is requested. A list of accounts is returned.

For more information on exits unique to the managed mode, see the *Programming Guide for Provisioning*.

How to Explore the UPO Endpoint

Exploring is the process of finding the objects in an endpoint and storing instances of them in the Provisioning Directory, the primary data repository for Identity Manager data.

You must explore a newly created Universal Provisioning endpoint once before you can administer it with Identity Manager. Exploring is needed to create UPO-specific objects within the Provisioning Directory. In managed mode, the appropriate exit will be invoked to populate the Provisioning Directory with the objects found in the remote system.

To explore a Universal Provisioning system, perform the following steps from the Endpoint Type task view:

1. For managed-mode UPO endpoints, make sure that the UPO program exits have been configured.
2. Register the system as an endpoint in Identity Manager by means of the Universal Provisioning Endpoint property sheet.
3. Explore the endpoint. After registering the server in Identity Manager, you must explore it to initialize the various nodes needed for correct operation. To do this, use the Explore and Correlate Endpoint dialog. The Exploration process creates a number of entries. It will not find any users, however, because they are not directly accessible from Identity Manager.

For complete instructions on exploring an endpoint, see the online *Procedures*.

Exploration Results in Managed Mode

In managed mode, UPO exploration returns accounts or creates new accounts, as for any other connector.

Exploration Results in Non-Managed Mode

UPO, operating in the non-managed mode, is different from other connectors in that, although you still need to explore, it does not return nor create new accounts, because they are not directly accessible. When exploring, UPO simply creates the placeholders needed for correct operation. Exploring needs to be done only once, when the endpoint has been created.

Assigning Account Templates to Provisioning Roles

Identity Manager lets you manage every user in your enterprise individually or by groups. Identity Manager uses the term provisioning *role* to represent a job function. Provisioning Roles also permit administrators to group users with corresponding access privileges. Therefore, you must have a provisioning role for each existing job function in your enterprise.

An *account template* is a template that defines the access privileges that a user has on an endpoint.

Provisioning roles define the job functions in your organization using one or more account templates. Provisioning roles and account templates let you manage endpoint of users without having to keep track of the access privileges for individual users.

Because UPO is a generic type of connector, the information that is made available to the Exit is dictated by the content of the user data field.

For complete instructions on creating account templates and provisioning roles, see the online *Procedures*.

Managing Exits

The following sections detail UPO exits.

UPO Exits

UPO program exits are implemented as separate objects within the Provisioning Directory, and are only referenced by a UPO Endpoint object. When a UPO Endpoint object has a reference to a particular Exit object, the exit represented by the Exit object will be invoked as appropriate.

A UPO endpoint object is able to reference multiple program exits, including multiple exits of the same type. For example, an endpoint can reference two ADD_ACCOUNT exits.

Note: The UPO Monitor uses the references defined in the endpoint object.

Order of Program Exits in the Process Flow

Program exits are serialized in the UPO process flow, according to their priority value, which is set in the UPO Program Exit Reference property page. The lower this number is, the higher the corresponding priority. If a Program Exit invocation fails, exits with lower priorities are not executed.

Define UPO Exits Using Manager

For complete instructions on writing program exits, see the *Programming Guide for Provisioning*.

UPO includes a number of ready-made DLL exits, including one called UPOSendMailExit.dll, which is described in detail in SendMail Program Exit.

The first step in setting up UPO exit functionality is to define a UPO exit.

To define a UPO exit

1. List the UPO endpoints.
2. Select the required endpoint and right-click it.
3. Select Custom, and UPO Program Exits to display the UPO Program Exits property sheet.
4. Click Add, and use the tabs to enter information to define the exit. The defined exit can then be used by any UPO directory.
 - On the Program Exit tab, enter the name and description of the exit to be invoked. If the Disabled box is checked, the program exit is not invoked, even if it is referenced by an endpoint object.
 - On the Common Parameters tab, specify whether the Provisioning Server is to use the Simple Object Access Protocol (SOAP) or a Dynamic Link Library (DLL) file to invoke the exit. In the Method field, you must provide the name of the exported function within the DLL file or the name of the function defined by the SOAP service.

- On the Authentication tab, provide information for authentication data that is passed to the program exit when it is invoked. This data is not used by Identity Manager.
 - Selecting None passes no authenticated data to the exit.
 - Selecting Current User passes the authenticated data of the global user that is logged on at the time the exit is invoked.

For more information, see Using Task Frames in *Identity Manager Provisioning Guide*.

- Selecting Proxy User enables the Authentication Details group box and lets you substitute another global user whose authentication data will be passed to the exit. Enter the appropriate Domain, Name, and password information.

Note: When you select Proxy User, the information that you provide must belong to a valid global user defined in Identity Manager.

- Selecting Other enables the Authentication Details group box and lets you choose any desired person whose name or global user name may not be a valid global user defined by Identity Manager.
- On the SendMail tab, if using the UPO SendMail program exit, provide information for connecting and sending RFC2822-compliant email messages to an ESMTP mail server. When defining a Custom Exit, an XML document can be entered in the custom data field. This data is passed to the program exit when it is invoked, but is not used by Identity Manager.

SendMail Program Exit

UPO includes a DLL exit called UPOSendMailExit.dll. This exit provides the SendMail method to send mail to one or more recipients. The body of the message consists of the exit XML document. This exit is designed to work with UPO operating in the non-managed mode, and can be invoked for any non-managed exit type, as relevant to the target system.

DLL Methods

The exit DLL provides the following methods:

- SendMail_Raw
- SendMail

The SendMail_Raw method simply dumps the incoming XML document into the body of the message. The SendMail method provides a more user-friendly version of the XML document.

UPO Program Exit Property Sheet

The SendMail property page of the UPO Program Exit property sheet is used to configure the messaging parameters for the SendMail exit.

When using the SendMail Program exit, ensure that Use SendMail Fields is checked. This causes discrete entry fields to display on the property page, rather than only a single field containing raw XML.

When Use SendMail Fields is checked, the following input fields are displayed:

Mail Server Host

Specifies the hostname or IP address of the SMTP mail server used for message submission.

SMTP Port

Specifies the TCP port for the SMTP or ESMTP message submission protocol. **Default:** 25.

SSL/TLS Enabled

Specifies whether SSL or TLS is needed when submitting a message. This requires an ESMTP server which supports EHLO and the RFC 2487's STARTTLS extension.

Mail Server Requires Authentication

Indicates that an account and password credentials are required to connect to the SMTP mail server. This facility requires that the SMTP mail server support the "AUTH" keyword as specified in RFC 2554. The only supported authentication method is the RFC 2595 - PLAIN SASL mechanism.

Account Name

Specifies the name of the account used for the authentication process.

Password

Specifies the password of the account used for the authentication process.

Originator (FROM)

Specifies the email address used by the originator (sender) of the message. This appears in the FROM field of all emails sent by this instance of the program exit.

Primary Recipient(s)

(Optional) Indicates the email address(es) of the recipient(s) that appear in the TO field of the message. Multiple recipients are separated by either commas or semicolons. Any value entered here is overridden by the one specified in the Exit Reference in the Directory object.

Secondary Recipient(s)

(Optional) Indicates the email address(es) of the recipient(s) that appear in the CC field of the message. Multiple recipients are separated by either commas or semicolons. Any value entered here is overridden by the one specified in the Exit Reference in the endpoint object.

Subject

(Optional) Specifies the subject line of the message. Any value entered here is overridden by the one specified in the Exit Reference in the endpoint object.

Body

(Optional) Specifies the default content of the message body. Any value entered here is overridden by the one specified in the Exit Reference in the endpoint object.

UPO Endpoint Property Sheet

When editing a UPO Program Exit Reference from the UPO Program Exits Reference property page, you can set some of the SendMail properties to override the ones specified in the Exit Object itself.

When using the SendMail Program exit, ensure that Use SendMail Fields is checked. This causes the display of the following fields:

To

Indicates the email address(es) of the recipient(s) that appear in the TO field of the message. Multiple recipients are separated with either commas or semicolons. This parameter must be defined by either the Program Exit object or the Exit Reference.

CC

(Optional) Indicates the email address(es) of the recipient(s) that appear in the CC field of the message. Multiple recipients are separated with either commas or semicolons.

Subject

(Optional) Specifies the subject line of the message.

Body

(Optional) Specifies the default content of the message body.

Creating a SendMail Program Exit

Use the property sheet described in Defining UPO Exits with Manager.

Create an object with the following values:

Program Exit

Specify the name and description of the exit and enable or disable it.

Common Parameters

Select "DLL" as the type of communication protocol, select the file "UPOSendMailExit.dll" as the location (the file is installed in *PS_HOME\bin*) and enter either "SendMail" or "SendMail_Raw" as the method.

Authentication

Select "None".

SendMail

Check "Use SendMail Fields", and then fill in the fields with the appropriate values for your environment.

Add the SendMail Exit to a Universal Provisioning Endpoint

The following example shows the steps to add the SendMail Exit to an endpoint.

To add the SendMail Exit to a Universal Provisioning Endpoint

1. Click the Users task button, and enter etaupoad in the Search value text box.
2. Click Search, right-click etaupoad in the list view, and select Properties from the pop-up menu.
3. Change the password in the Password tab to password and click OK.
4. Click the Endpoint Types task button, and select Universal Provisioning from the Object type drop-down list.
5. Click New, and then complete the Endpoint tab with the details for the global user who will manage the accounts and the server to which Identity Manager is connected. Click OK.
6. Right-click your new endpoint in the list view, and select Custom/UPO Program Exits from the pop-up menu. Click the Search button in the UPO Program Exits dialog to display a list of Exit objects.
7. Click Add to create a new Exit.
8. Enter SendMail in the Name text box of the Program Exits tab. Select the Common Parameters tab, and click DLL for type.

9. Click the Select button and double-click UPOSendMailExit.dll in the Open dialog list. The DLL name populates the Location text box. Enter SendMail as the Method in the Common Parameters tab and then select the SendMail tab.

Note: No action needs to be taken in the Authentication tab.

On the SendMail tab:

- a. Make sure the Use SendMail Fields box is checked.
Enter the name of your mail server host and the SMTP port number.
 - b. Make sure Mail Server Requires Authentication is *not* checked.
 - c. Enter the appropriate email addresses for the Originator and the Primary and Secondary recipients.
 - d. Enter UPO SENDMAIL EXIT for the Subject and UPO SAMPLE BODY for the Body.
 - e. Click OK.
10. Right-click your new endpoint, and from the pop-up menu, select Properties.
 11. Select the UPO Program Exits Reference tab, and click Add.
 12. On the Edit UPO Program Exit Reference tab:
 - a. Select the new SendMail exit from the Name drop-down list.
 - b. Enter 1 for the Priority.
 - c. Select ADD_ACCOUNT from the Type drop-down list.
 - d. Check the Use SendMail Fields box.
 - e. Complete the Custom Data using the recipient names you entered when creating the SendMail Exit.
 - f. Add a message to the Body text box and click OK.
 13. Repeat step 6 through step 12 to add other Exits.

More Information

[Create a UPO Endpoint](#) (see page 244)

Non-Managed Mode Operations

The following sections detail describe the non-managed mode operations.

Marking a Pending Operation as Completed

In non-managed mode, UPO marks an account as Pending after invoking the operation exit (if any). While an account is in the Pending state, any new user-provisioning operation immediately fails, ensuring that there is only one outstanding operation against an account. To clear this state, the outstanding operation must be marked as Completed.

You can mark a Pending operation as Completed by using either the Provisioning Manager, etautil utility, or the LDAP protocol directly. A short note can be added when completing the operation.

Mark a Pending Operation as Completed with Provisioning Manager

You can use the following procedure to mark a pending operation as completed.

To mark a Pending operation as Completed with Provisioning Manager

1. Right-click on the account that is in Pending state in list view, and select Properties.
2. On the Pending tab, select Completed from the Status drop-down list in the Update section. Click Apply.

Mark a Pending Operation as Completed with etautil Utility

The following is an example of how to use the etautil utility to mark a Pending operation as Completed:

```
etautil [-d domain] [-u user [-p password]] update
```

```
'eTUPOAccountContainerName=Accounts,eTUPODirectoryName=<DIRECTORY_NAME>,eTNamespaceName=Universal Provisioning' eTUPOAccount eTUPOAccountName=<ACCOUNT_NAME> to  
eTUPOOperationStatus=' C' eTUPOOperationStatusNote=' Completed the request. Assigned to default group as  
requested. '
```

Mark a Pending Operation as Completed with the LDAP Protocol Directly

You can use an LDAP client to send a Modify operation for the UPO account entry in the repository. This operation sets the eTUPOOperationStatus attribute to C. To have an LDAP client send a Modify operation, use the global user configured in the endpoint property sheet as the user for the Bind request.

The bind parameters are as follows:

```
DN: eTGlobalUserName=etaupoad,eTGlobalUserContainerName=Global Users,  
eTNamespaceName=CommonObjects,  
dc=DOMAIN_NAME,dc=eta
```

where password is the etaupoad password.

The modify operation detail using the LDIF format is as follows:

```
dn: eTUPOAccountName=ACCOUNT_NAME,eTUPOAccountContainerName=Accounts,  
eTUPODirectoryName=DIRECTORY_NAME,eTNamespaceName=Universal Provisioning,  
dc=DOMAIN_NAME,dc=eta
```

```
changetype: modify  
replace: eTUPOOperationStatus  
eTUPOOperationStatus: C
```

```
replace: eTUPOOperationStatusNote  
eTUPOOperationStatusNote: Completed the request. Assigned to default group as requested.
```

Removing an Account from the UPO Endpoint

When deleting an account, the LDAP delete operation is passed on to UPO as a Delete operation request, not as an actual request to remove the account from the endpoint.

To remove an account from the endpoint using the Provisioning Manager, perform the following:

1. Select the account to be removed.
2. Right-click the account, select Custom menu..., Remove Account From Endpoint...
3. Refresh the list view to reflect the removal of the account.

Monitor Service

UPO, when operating in the non-managed mode, includes a Windows service to monitor endpoints for Service Level Agreements (SLA). Each endpoint can be configured with an SLA time value to specify the time, in seconds, that an account can have an operation in the Pending state. They can also be configured to provide a warning of accounts approaching the SLA time period.

The Monitor service is installed in *PS_HOME\bin\UPOMonitorService.exe*.

Note: The Service Level Agreement (SLA) specifies the time, in seconds, that an account can have an operation in pending state.

The command line options are the following:

- -install
- -remove

More Information

[Service Installation](#) (see page 262)

[Service Removal](#) (see page 262)

Service Installation

The -install command line option is described in the following table:

Command Line Connector	Description
-install	Install the monitor service into the service control manager database.

For example:

```
> UPOMonitorService -install
UPO Monitor: successfully installed as service 'CAUPOMonitor'
```

Service Removal

The -remove command line option is described in the following table:

Command Line Connector	Description
-remove	Removes the monitor service from the service control manager database

For example:

```
> UPOMonitorService -remove
```

```
> UPO Monitor: Successfully removed 'CAUPOMonitor' as a service
```

User Credentials

Monitor needs access to the Provisioning Server to search for operations that exceed the SLA. The following command line options permit you to set the global user name and password to be used. They must be entered together, as specified in the example below. This command updates values in the registry.

Command Line Option	Description
-user <i>globalUser</i>	Global User name
-password <i>password</i>	Password of the global user

For example:

```
> UPOMonitorService -user etaupoad -password somepassword
```

The global user that Monitor binds to the Provisioning Server must have read access privileges to the following:

- UPO Endpoint objects
- UPO Account objects
- UPO Program Exit objects

The default UPO administrator, ETAUPOAD, has the appropriate privileges.

Polling Frequency

This value determines how often the monitor scans each directory for SLA checks. This command updates a value in the registry.

Command Line Option	Description
-delay <i>time</i>	Time period, in seconds, for polling directories. The value must be equal to or greater than 60 seconds.

For example:

```
> UPOMonitorService -delay 600
```

Adding an Exit Reference for Monitor

You must use the Endpoint property sheet's UPO Program Exits Reference property tab to add an exit reference to be used by the Monitor service. The following exits are available for Monitor:

UPO_SLA_EXCEEDED

This program is called by the UPO Monitoring service when pending requests are found that exceed the Service Level Agreement delay that was set for the corresponding endpoint.

SLA_WARNING

This program is called by the UPO Monitoring service when pending requests are found that exceed the SLA warning delay that was set for the corresponding endpoint.

SLA_TIMEOFFSET

This program is called by the UPO Monitoring service before searching for outstanding pending accounts. The called program returns an offset, in seconds, that is subtracted from the current time when calculating the time period. This program is required to return a zero (0) if there is no offset (during working hours) or an offset until the next work day. This lets SLA_EXCEEDED and SLA_WARNING exits skip non-work hours, weekends, and holidays.

Chapter 10: Groupware Connectors

This section contains the following topics:

[Microsoft Exchange 200x Connector](#) (see page 265)

[Lotus Notes/Domino Connector](#) (see page 280)

The following connectors are considered to be groupware connectors:

- Exchange 200x
- Lotus Notes/Domino

The information in this chapter describes how to install and manage these connectors.

Microsoft Exchange 200x Connector

The Microsoft Exchange 200x Connector lets you administer mailboxes on Active Directory Services (ADS) servers.

The Microsoft Exchange 200x Connector is designed to run with the Active Directory Services Connector. It provides a single point for all user administration by letting you do the following:

- Create and manage Microsoft Exchange 200x mailboxes for any existing ADS account
- Create and manage new Microsoft Exchange 200x mailboxes
- Create and manage contacts with email addresses and create and delete email addresses of the contacts
- Create and manage Microsoft Exchange 200x distribution lists and groups
- Create and delete email addresses of an existing distribution group
- Generate and print reports about Microsoft Exchange 200x mailboxes
- Explore an ADS/Microsoft Exchange 200x computer, distribution groups, and Microsoft Exchange 200x users

This connector is managed using the Connector and agent installation process. For more information and requirements, click [here](#) (see page 21).

This connector can also be managed using the [Connector and C++ Server](#) (see page 23) installation process as well.

How to Manage Mailboxes

Understanding how to manage your mailboxes is helpful in understanding how the Microsoft Exchange 200x Connector is integrated into Identity Manager. The Microsoft Exchange 200x Connector is designed to run in conjunction with the ADS Connector. With the technology from both connectors, you can manage ADS users who have mailboxes.

Because the Microsoft Exchange 200x Connector runs with the ADS Connector, you will manage ADS objects, not Microsoft Exchange objects. For example, if you want to do the following:

- Acquire a Microsoft Exchange 200x server, you must acquire its ADS server
- Use an account template that creates Microsoft Exchange 200x mailboxes, you must create an ADS account template and assign Microsoft Exchange attributes to the account template
- Create mailboxes for global users, you must create the mailboxes using the global users' ADS accounts
- Create distribution lists, you must assign an email address to each of the ADS groups
- Perform a synchronization on mailboxes, you must synchronize the ADS accounts

To manage Microsoft Exchange 200x mailboxes, install the ADS and Microsoft Exchange 200x connectors on your Provisioning Server.

For more information about installing and configuring the ADS Connector, see the *Active Directory Services* section of *this guide*.

Exchange 200x Log Files

Log files generated for "Move Mailbox" and "Manage Mailbox Rights" can be found in the following directory:

`PS_HOME\Logs\ADS`

Configuring the Exchange 200x Remote Agent Component

The remote agent component is required to manage mailbox permissions and to perform the operation of moving mailboxes between mailbox stores. The remote agent uses APIs that have to be launched on the Microsoft Exchange computer.

The Exchange Remote Agent should be installed and the CAM and CAFT Service should be configured on Exchange 200x servers that you plan to use as a gateway server to manage the Exchange site.

In the Provisioning Server, for each managed ADS/Exchange 200x endpoint, you should define which Exchange gateway server should be used. This can be done from the Exchange General tab of the managed endpoint property page.

For Windows 2003 Machines:

When installing the Exchange Remote Agent, you need to perform post installation steps to grant the remote agent enough rights to perform the required operations. The CA Message Queuing Server service needs to be updated to be started with an account granted enough rights on both the Exchange Server and the ADS Domain (for move mailbox operations and management of mailbox rights).

If the Remote Agent is installed on a Windows 2003 box, the accounts natively created by default may miss rights needed for the CA Message Queuing Server service to have enough rights to run commands. Therefore, the account starting this service needs to have the following rights (found under Local Security Policy\Local Policies\User Rights Assignment):

- Act as part of the operating system
- Log on as a batch job
- Log on as a service

Step 1. Configure the CAM and CAFT Service

You must configure CAFTHOST to recognize the C++ Connector Server and the Identity Manager clients.

1. Issue the following command on the computer where the remote agent is installed:

```
$ CAFTHOST -a Windows_node_name
```

where *Windows_node_name* is the name of the C++ Connector Server host.

Note: If the C++ connector Server is networked using DHCP or you do not use DNS for name resolution, the network name will not be recognized. Under these conditions, use the TCP/IP address for the Windows node name or add a Windows node entry in the local hosts file on your Microsoft Exchange server.

2. Verify this command by issuing:

```
$ CAFTHOST -I
```

The previously mentioned steps can also be preformed by using the Host to Caft Definition Provisioning Manager that can be selected from the following location:

Start/Program Files/CA/Identity Manager/Host to Caft Definition

For more information about viewing, starting, or stopping the CAM and CAFT Service, see Managing the CAM and CAFT Service in this section.

Step 2. Update the CAM and CAFT Service Logon Account

By default, the CAM and CAFT Service is started by the system account when you install the Remote Agent. Identity Manager needs this service to be started by an account that has exchange administrative rights in the domain; therefore, you must change the account that starts this service.

For Windows 2000

To change the account on a Windows 2000 computer, do the following:

1. Click Start on the taskbar, and then choose Programs, Administrative Tools, Services.
2. Double-click the CA Message Queuing Server service.
3. Click the Log On tab.
4. Select This Account and enter the name of an account with administrative rights to the domain and its password.
5. OK.
6. Restart the CA Message Queuing Server service.

For Windows NT

To change the account on a Windows NT computer, do the following:

1. Click Start on the taskbar, and then choose Settings, Control Panel.
2. Double-click the Services icon.
3. Select CA Message Queuing Server, and then click the Startup button.
4. In the Log On As group box, select This Account and enter an account with administrative rights to the domain.
5. Click OK.
6. Restart the CA Message Queuing Server service.

Activating CAM and CAFT Encryption

To install the encryption key, follow these steps:

1. Enter the following command at the command prompt to generate your key file:

```
#PATH=`cat/etc/catngcampath`/bin:$PATH  
  
#export PATH  
#caftkey -g keyfile password
```

where:

keyfile is the name you assign to the key file.

password is the password you assign to the key file.

Note: The caftkey command and attributes are the same for Win32 platforms.

2. Install your Public Key on both CAFT Agent and CAFT Admin computers using the previously-generated key file by entering the following command at the command prompt:

```
#PATH=`cat/etc/catngcampath`/bin:$PATH  
#export PATH  
#caftkey -policy_setting keyfile password
```

- *keyfile* and *password* must be the same values you specified in Step 1. -*policy_setting* is -i, -m, or blank.
- The *policy_setting* governs the communication between this computer (the local computer) and other computers that have the CAM and CAFT Service installed, but may or may not have the CAM and CAFT encryption certificates installed.

Policy -1 (caftkey -i keyfile password)

The -i option specifies Policy -1. This policy lets computers running previous versions of the CAM and CAFT Service execute commands on this computer and lets this computer execute commands on those computers. Policy -1 encrypts messages if the other computer has these certificates installed. This policy does not encrypt messages if the other computer does not have these certificates installed.

Policy 1 (caftkey -m keyfile password)

The -m option specifies Policy 1. This policy prohibits other computers from executing commands on this computer if they are running previous versions of the CAM and CAFT Service without the encryption certificates. This policy also prohibits this computer from executing commands on those computers.

If both computers have the CAM and CAFT encryption certificates installed, but have different Public Key Files installed when Policy 1 is set, the command requests between the two computers always fail.

The Blank Option

The blank option specifies Policy 0. This policy is set if no Public Key File is installed, the CAM and CAFT encryption certificates were not installed properly, or if you do not specify a policy setting when you enter the caftkey command. Policy 0 specifies no encryption.

Note: The CAM and CAFT Service must already be installed on the computer in your network. For example, to install the encryption key on Linux computers, run the following commands:

```
#tar xvf LINUX_V1.07_20020319_Build230.tar
#cd ./cam/scripts
#./install
```

3. Recycle the CAM Service on each computer where you install the new Key as follows:

```
prompt> camclose      //stop Cam/Caft service and processes
prompt> cam start      //start CAM service and process
```

Check the Policy setting:

To see what mode the computer is operating in, look in the following file:

```
%CAL_MSQ%\ftlogs\dg000
```

Managing the CAM and CAFT Service for Microsoft Exchange 200x

Note: The CAM and CAFT Service allows encryption using certificates.

The CAM and CAFT Service is a daemon process. You can control this process using the Services dialog on your Control Panel. To view the Services dialog, click the Services icon. The CAM and CAFT Service is called CA Message Queuing Server.

View the CAM/CAFT Process

To view this process, perform the following steps:

1. Open the Windows Task Manager.
2. Click the Processes tab on the Windows Task Manager.

The CAM and CAFT daemon processes appear. The following is a sample of these processes:

Image name	PID	CPU	CPU Time	Mem Usage
Caft.exe	1364	00	0:00:16	1 600 K
Cam.exe	516	00	0:00:08	704 K

Start the CAM/CAFT Service

Although the CAM and CAFT Service starts automatically, there may be times when you have to manually start it.

To start the CAM and CAFT Service, do the following:

1. Double-click the Services icon on the Control Panel.

The Services dialog appears.

2. Select CA Message Queuing Server from the Service window and click Start.
3. Click Close.

Note: After you stop the CAM and CAFT Service, you must restart it so Identity Manager can communicate with the Microsoft Exchange Remote Agent.

Stop the CAM/CAFT Service

To stop the CAM and CAFT Service, do the following.

1. Double-click the Services icon on the Control Panel.

The Services dialog appears.

2. Select CA Message Queuing Server from the Service window and click Stop.
3. Click Close.

Note: After stopping the CAM and CAFT Service, you must restart it so that Identity Manager can communicate with the Microsoft Exchange Remote Agent.

E2K Support for FIPS and IPv6

For release r12 of Identity Manager, the Exchange 200x Connector does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Managing Exchange Users

Identity Manager manages mailboxes on ADS endpoints. Information about these mailboxes is stored in the ADS endpoint associated with the Microsoft Exchange server. For this reason, you must acquire the ADS endpoint, when you want to manage your Microsoft Exchange users.

When you acquire an ADS endpoint, the Endpoint Content dialog displays the following containers:

- **Builtin** contains all security groups that are built into ADS, such as Administrators and Backup Operators
- **Computers** contains computers that belong to the ADS directory
- **Microsoft Exchange System Objects** contains Microsoft Exchange system mailboxes
- **Users** contains all ADS accounts and groups, including those with mailboxes

Other organizational containers may appear in this dialog. These containers reflect the structure that exists within the ADS directory. For example, an ADS directory may contain a Human Resource container for all Human Resource users and groups.

Authentication Process

Exchange management for the ADS Endpoint is now enabled or disabled when you supply a new Userid/Password during authentication. If the new Userid/Password is incorrect, management is disabled, and conversely for the other.

ADS Account Templates

The ADSSContactPolicy, provided with the Active Directory Services Connector, provides a user with the minimum-security level needed to access an ADS endpoint. You can use it as a model to create new account templates with Exchange specific options.

When working with ADS Contact account templates, the Exchange General tab defines the Microsoft Exchange attributes that you can set, for example:

- Any delivery restrictions, such as the messages that are accepted by the mailbox and their maximum size
- Any delivery options, such as any forwarding addresses or permissions that the user has when sending messages on other user's behalf
- All storage limits, such as the size limit and the length of time that a user can keep deleted items

This tab contains all the attributes that are necessary for you to create a Microsoft Exchange 200x mailbox.

Specifying Datastore Names in an ADS Policy

Important: The following is an alternate way to specify the Datastore Name. You can also specify the datastore name using the Home Server and Mailbox Datastore fields on the Provisioning Manager.

Note: When you create an ADS policy that uses a rule string for the name of a Microsoft Exchange server, do not specify the datastore location. Datastores are dependent on the server name.

To create a default location for the datastore, follow these steps:

1. Open the default.e2k file located in the following directory:

\PS_HOME\data\ads

Note: This file should be manually created if needed and should contain two columns. The first column contains the Relative Distinguished Name (RDN) of the Microsoft Exchange server. The second column contains the RDN of the datastore that is used as the default.

2. Update this file by entering the RDN of the datastore or a complete DN in the second column. If you enter the RDN, you must specify a unique value. In addition, the second column must be delimited with a double quote.
3. After saving and closing the file, you must restart the C++ Connector Server.

Microsoft Exchange Distribution Lists and ADS Policies Using Strong Synchronization

Microsoft Exchange uses Active Directory Server for its directory. Microsoft Exchange distribution lists are implemented as Active Directory groups. Because the groups are defined in ADS, the members of these groups are subject to ADS policies. If an ADS policy does not list the Microsoft Exchange distribution list, which is now an Active Directory group, in the Groups (Member Of) tab of the policy and the policy uses strong synchronization, the accounts that are synchronized by this policy will lose their membership in the Microsoft Exchange Distribution list. To prevent this loss of membership, the ADS group representing the Microsoft Exchange distribution list needs to be added to the policy.

For some installations, adding the distribution lists to the ADS policies is not desired. From the Exchange General tab of the Active Directory directory property sheet, you can choose the Active Directory container used to store the Microsoft Exchange Distribution lists. During policy synchronization, the synchronization mechanism will not remove the account from a group that exists in the specified container or any container owned by the specified container. This lets you use current ADS strong synchronization policies with having accounts lose membership in ADS groups used for Microsoft Exchange Distribution lists.

Install Without a Microsoft Certificate Authority

If you are not able to install a Microsoft certificate authority, you must acquire the certificates for all of your domain controllers and any you add in the future.

To acquire the certificates

- Purchase the certificates using the fully qualified machine names for the domain controllers from trusted certificates vendors and handle the renewal activity based on the terms that are agreed upon.
- Create and maintain your own certificates using tools that are available to you.

Note: Two certificates are usually received for each domain controller, a certificate authority certificate and a certificate for the machine.

Once you have received the certificates, you can use the mmc.exe utility to install them. You must also install the certificate authority certificate on all of your Provisioning Servers and the machines where the C++ Connector Server is installed.

Exchange Accounts

Accounts give users access to the resources on an endpoint. Identity Manager lets you manage Microsoft Exchange mailboxes from the Endpoint Type task view.

- Use the Active Directory Services Account property sheet when managing mailboxes
- Use the New User property sheet when creating a new ADS account that will have a mailbox
- Use the Custom menu to create or delete mailboxes that are associated with ADS accounts
- Use the Active Directory Services Account property sheet when deleting a mailbox of an Active Directory Services account without deleting the account

There are the following three Account-specific tabs for Exchange on the ADS Account Property Sheet:

Email Addresses Tab

Contains email addresses for the corresponding mailbox.

Exchange General Tab

Contains Exchange attributes such as Server, Alias name, or the limits for a mailbox.

Exchange Advanced Tab

Contains all of the advanced Exchange properties such as custom attributes, protocol settings, ILS settings, and mailbox rights.

Setting Live Communication Server 2003 Attributes for ADS

You can now set Live Communication Server 2003 attributes for ADS accounts and ADS account policies. There is no Provisioning Manager support for setting or viewing these attributes. However, they can be set through the eTA Batch Utility (etautil) or through an LDAP browser (for example, JXplorer).

The following attributes need to be set to enable live communications:

- eTADSmsRTCSIP-IsMaster. For example, TRUE.
- eTADSmsRTCSIP-PrimaryHomeServer. For example:
`CN=RTC Services,CN=Microsoft,CN=<YourDC>,OU=Domain Controllers,DC=<YourDOMAIN>,DC=com`
- eTADSmsRTCSIP-PrimaryUserAddress. For example:
`sip:<account>@DOMAIN.com.`
- eTADSmsRTCSIP-UserEnabled. For example, TRUE.

The attributes of an account that has Live Communications enabled can be viewed as an example.

Note: It is possible to use rule strings for the attributes in the ADS account policy. For example, eTADSmsRTCSIP-PrimaryUserAddress can be set as `sip:%AC%@DOMAIN.com.`

Custom Menu on Accounts and Contacts

The following Exchange operations are available from the Custom Menu:

- Create Mailbox (accounts)
- Enable e-mail addresses (contacts)
- Move Mailbox (accounts)
- Delete Mailbox (accounts)
- Disable e-mail addresses (contacts)
- Remove Exchange Attributes

Custom Menu on Group

The following Exchange operations are available from the Custom Menu:

- Enable e-mail address
- Disable e-mail address

Creating E-mail Addresses for Groups

E-mail addresses for groups can be created:

- During the creation of a group using the Content menu
- On an existing group using the Custom Menu

Distribution Lists

An ADS security group that has an active email address is called a *distribution list*. Identity Manager lets you create and maintain distribution lists using the Endpoint Type task view. Use the Active Directory Services Group property sheet when managing these lists.

E2K Etautil Conventions

Use the following ADS conventions in your etautil commands:

- The endpoint type name (eTNamespaceName) is ActiveDirectory
- The endpoint type prefix is ADS. Therefore, the Active Directory Services class names are:
 - eTADSDirectory for an endpoint
 - eTADSPolicyContainerName for an account template container
 - eTADSPolicy for an account template
- The Description line for all Microsoft Exchange attributes contain the phrase (Exchange2000 only)

Lotus Notes/Domino Connector

The Lotus Notes/Domino Connector lets you administer accounts and groups on Lotus Notes/Domino servers and provides a single point for all user administration by letting you do the following:

- Register multiple endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage Lotus Notes/Domino accounts using Lotus Notes/Domino-specific account templates
- Create and manage Lotus Notes/Domino groups and organizational units
- Activate accounts in one place
- Synchronize global users with their roles or synchronize global users' accounts with their account templates
- Assign Lotus Notes/Domino endpoints to your Lotus Notes/Domino endpoints
- Use the default Endpoint Type account template to create accounts with the minimum level of security needed to access the Lotus Notes/Domino endpoints
- Recertify, rename, and move Lotus Notes/Domino accounts in the hierarchy
- Generate and print reports about Lotus Notes Domino accounts and groups

The Lotus Notes/Domino Connector uses the inherent object model and administrative processes underlying the Lotus Notes/Domino product. The next sections introduce the native Lotus Notes/Domino object model, the security application databases, and the administrative processes used by the Lotus Notes/Domino Connector to perform user management.

The following versions of the endpoint software are supported:

- Version 6.5 (6.5.3 or later)
- Version 7
- Version 8

LND Installation

The Lotus Notes/Domino Connector is installed using the Java Connector Server installation process. Click [here](#) (see page 26) to view installation requirements and examples for this process.

LND Java Implementation Considerations

The Java version of the LND Connector (installed with the Java Connector Server) provides the same functionality as the previous (eTrust Admin r8.1) C++ version of the LND Connector with the added benefit of DJX support, but there are a few things to consider when switching from the C++ version.

- To add the Java LND connector to an existing system:
 1. Run the Provisioning Server install to reconfigure and add the LND connector.
 2. Run the JCS installer and select Register with the Provisioning Server.
Doing this routes requests from the Provisioning Server to the JCS for the LND endpoint type. See "[Install the LND Connector with JCS](#) (see page 286)" for more information.
- For this release it is no longer required to have Lotus Notes Client software installed, or a Notes.ini file available on the computer where the Provisioning Manager is running. The same is true for the computer where the Java CS hosting the LND connector is run.
- Connector operation, as well as migration requires the Lotus Notes Domino remote Corba interface jar (NCSO.jar) to be copied to
`<jcs-home>/extlib/`
from the server installation directory
`<lotus-home>\Data\domino\java`
and JCS restarted.
- If existing Regarc and Regcert databases are used, these encryption Keys must be imported from the Admin ID used previously, to the Server ID, or all documents in the database created using the old encryption keys will show an error.
- ID file paths should be specified local to the endpoint or use a UNC style path.
- Organization names and organizational unit names no longer appear prefixed by "O:" and "OU:".
- There are new choices for mail systems that correspond to the available choices in Domino R6 and R7
 - Lotus Notes
 - POP
 - IMAP
 - Domino Web Access
 - Other Internet
 - Other

- None

- The mapping information for the custom attributes is now contained in

`<jcs-home>/conf/override/lnd/lnd_custom_metatdata.xml`

Two sample files with the prefix of "SAMPLE" are provided including one covering DJX mappings. the override connector.xml file of the Java LND Connector. For existing LND endpoints, the credential information currently stored in the registry is no longer required.

- The following settings must be made to the Domino server's notes.ini file:

- *\$Reg_TemDir* variable representing the temp directory on the Domino server, must be added to the notes.ini file. If you add or change the variable, the Domino server must be restarted.

Note: The directory specified must already exist. The LND Connector will not create the directory. An example of this setting:

\$Reg_TemDir=c:\lotus\domino\data\temp

Or

\$Reg_TemDir=\\user01w2k3\c\$\lotus\domino\data\temp

The value is necessary for the temporary placement of ID files during ID password changes. For ID password changes to be successful, the ID must be located on the server at the time of the change. The value is also necessary for the temporary placement of ID files during account and certifier creation if another location is not specified for the ID.

- Two new fields are required when acquiring an LND Endpoint:

- DIIOP Port

The LND connector uses the remote access support provided in the Lotus Domino Toolkit for Java/CORBA 2.1 to communicate with the endpoint.

- Use SSL/TLS

- The connector is now case insensitive. Objects existence is checked ignoring case. If duplicate objects already exist during an explore operation, only one is managed (silently) by the Provisioning Manager.
- If connection to the endpoint is temporarily lost, the default retry settings of six retries 10 seconds apart are enforced. As for all JCS connectors, these settings can be adjusted using the *exceptionRetryMap* settings in the connector.xml file located in:

`<jcs-home>/conf/override/lnd/.`

- Search results are streamed so the connector passes search results on to the Provisioning Server immediately rather than buffering all results before passing any on. This keeps the memory usage of the Connector Server to a minimum, regardless of the number of objects in an LND endpoint.

- Custom delete of an account in a secondary directory is not supported by the Administration Process (adminp) JAVA API used by the connector so the connector explicitly deletes only the person document itself. You must manually clean up any references to the person's name in group members/ACL's and so forth, where they exist.

Note: Account operations are not supported on accounts that are in a secondary directory and require the intervention or spawning of requests from the Domino administration process 'adminp'.

LND Migration

The LND migration is driven by a text menu-driven interface. When run for the first time, information to be used to connect to a Provisioning Server is requested (defaults are presented in square [] brackets) and password characters are displayed as stars (*) to prevent snooping. Once a successful connection to the Provisioning Server is achieved, all of the connection details (except password) are saved and can be recalled using a simple number displayed as a label the next time the migration interface is run.

When connecting to a Provisioning Server, the migration script queries for all of the existing LND endpoints that are registered within the Provisioning Server. You are then given an opportunity to review the settings for each endpoint (port number, use SSL, and so forth). Once a successful connection to the LND endpoint is achieved, the details entered for it are saved in the Provisioning Server and the data migration itself takes place.

From the bin folder of the Connector Server, find LNDMigrate.bat for Windows and LNDMigrate (without .sh) for UNIX environments to start the migration.

```
<jcs-home>\bin
```

Log files are generated in

```
<jcs-home>/resources/lnd/LNDMigrate.log
```

Following are the migration updates that take place:

- eTLNDAccount attributes
 - eTLNDInternetOnlyUser and eTLNDNoIDFileCreation attribute values are copied from the Provisioning Manager to account's matching LND Person document.
 - Values for the eTLNDLocalAdmin and eTLNDManager attributes are now Provisioning Server LDAP DN's and must be translated from the previous Notes internal name syntax.
 - A new field for the UserUniqueOU name is calculated and stored in the Person document for accounts that have a "Unique OU" value set. This improves performance when re-exploring as it saves the connector from having to look up every account's immediate parent to decide whether it is a real OU or a unique OU.
- eTLNDAccountTemplate attributes
 - Values for the eTAccountContainer, eTLNDGroupMembership, eTLNDLocalAdmin, and eTLNDManager attributes are now Provisioning Server LDAP DN's and must be translated from the previous Notes internal name syntax.

Note: The values of eTLNDLocalAdmin and eTLNDManager contain special syntax to differentiate "Unique OU". Refer to section [How Managed Objects are Referred to in the Java LND Connector](#) (see page 296) for details. If the account template does not have an LND endpoint assigned, the values of eTLNDLocalAdmin and eTLNDManager will not be converted to "Unique OU" syntax for unique OUs, instead all unique OUs will be treated as real OUs.

Important! After the migration tool has run successfully, you must re-run the exploration and correlation for those endpoints that have just been migrated in order to update the account DN syntax stored in the repository and fix all the broken links between the accounts and global users due to the changed account DN syntax.

Migrate Passwords

All passwords are decrypted and encrypted by the connector using FIPS support. Passwords are migrated automatically as they are changed and new passwords can be distinguished as they are prefixed with "{AES}".

SSL Prerequisites

Before using SSL to secure connections to an LND endpoint, the following must be done:

1. You must configure the endpoint itself to allow inbound SSL connections. This process is documented for Domino r6 and above here:

http://www-128.ibm.com/developerworks/lotus/library/ls-Java_access_2/index.html

2. Copy the TrustedCerts.class file (created in Step 1 in

install_path\Lotus\Domino\data\domino\Java\

to

<cs-home>/extlib/

Note: A port cannot be provided when acquiring the endpoint using SSL.

Install the LND Connector with JCS

Follow this procedure to install the LND connector with the Java Connector Server.

To install the LND Connector

1. Run the setup.exe program of the Provisioning Server installation package and select LND connector.
2. Run the setup.exe program of the JCS installation package to install the Java Connector Server with the LND option.
3. Copy the NCSO.jar file from the following locations (if the connector is going to communicate with endpoints of differing versions, use the file from the lowest numbered version):

Version 6.5 and 7: *lotus-home*\NCSO.jar

Version 8: *lotus-home*\NCSO.jar

where

lotus-home

Specifies the Lotus Notes Domino installation directory. Defaults to: C:\Program Files\Lotus\Domino\data\domino\java (Versions 6.5 and 7) or C:\Program Files\IBM\Lotus\Domino\data\java (Version 8)

to *jcs-home*\extlib\

where

jcs-home

Specifies C:\Program Files\CA\Identity Manager\Connector Server\ of the JCS installed machine. If you use SSL to communicate with the Domino Server, the "TrustedCerts.class" file (generated by Domino when SSL is set up on the Domino server) must be in the classpath on the JCS system. For more information on prerequisites for SSL, click [here](#) (see page 285).

4. Restart the CA Identity Manager - Connector Server (Java) service.

Configure the Lotus Notes/Domino Connector

Before installing the Lotus Notes/Domino Connector, do the following:

1. Verify your access privileges to the Lotus Notes/Domino databases.
2. Enable the Administration Process (Adminp).
3. Encrypt your local databases.
4. Create and add Identity Manager encryption keys to the administrator ID.

Step 1. Verify Your Access Privileges

To ensure that Identity Manager can manage users on a Lotus Notes/Domino server, verify that the server ID used by the Lotus Notes/Domino Administrator has the same access level, privileges, and roles as the Lotus Notes/Domino domain administrator in the following databases:

- names.nsf
- admin4.nsf
- certlog.nsf

Important! We recommend that you log into the Lotus Notes/Domino Administrator application using the ID file for the same administrative user configured to be used by the connector to access the endpoint. By doing this, you ensure that this user has the necessary access level, privileges, and roles that are required to complete user management actions, including access to all certifiers in the domain.

Step 2. Enable the Administration Process (Adminp)

This step ensures that you can use all of the Administration Process (Adminp) features. By default, Adminp runs when a Lotus Notes/Domino server is started; however, it is not automatically enabled for the domain. Therefore, to set up Adminp in the domain, you must do the following:

1. Designate a server in the domain to be an administration server for the Domino endpoint (Public Address Book).
2. Ensure the administration server for the domino endpoint is running the most recent version of Lotus Notes/Domino.

Note: After assigning an administration server to the endpoint, you must always use the server (rather than local) copy of the Public Address Book for Adminp tasks.

Step 3. Add Encryption Keys to the Server ID

For Lotus Notes Client 6.5 or higher

To add the encryption keys to the server ID, perform the following steps on the Domino Server for Domino 6 or higher:

1. From the File menu, select Security, User Security.
2. Enter the ID password
3. Expand the Notes Data section and select Documents
4. Click New Secret Key
5. Enter RegXArchive as the name of the key.
6. Select Use international encryption if the key is to be used outside North America (U.S. and Canada) on versions of Notes prior to 5.0.4.
7. (Optional) Enter a comment that explains the key's use.
8. Repeat steps 4 and 5 to create the RegXCertifier key.
9. Click OK
The User Dialog appears.
10. Click Done.

Note: To import existing encryption keys into the server ID, complete Steps 1, 2, and 3 and then click the Other Actions button and choose Import Secret Key. After you have found the respective files containing the keys to import, click open and specify the password if no password has been associated with the key.

Step 4. Encrypt Local Databases

If REGARC.NSF or REGCERT.NSF is copied to a local workstation, you should encrypt them. To do this, complete the following steps in the Lotus Notes Client workspace:

1. Open the database using the Lotus Notes Client workstation.
2. Select Database/Properties.
The Database Properties dialog appears.
3. Select Encryption settings.
The Encryption dialog appears.
4. Select the "Locally encrypt this database using option" check box.
5. Select the Strong, Medium, or Simple Encryption option depending on the type of encryption you want.
6. Click OK to close the Encryption dialog.
7. Click OK to close the Database Properties dialog.

By encrypting them, unauthorized users cannot view the contents of fields, such as passwords and certifier IDs.

Set Up the Domino Server

The LND Connector uses the Domino Java API to access the Domino server using CORBA (Common Object Request Broker Architecture). The server's initial object is obtained as an IOR (Interoperable Object Reference) using HTTP, and further objects are obtained using an IIOP (Internet Inter-ORB Protocol) connection. The Notes client is not required on the client system, but it must contain NCSO.jar in the classpath.

To set up the Domino server for this remote access, the server must be accessible through TCP/IP. You must be able to ping the server using its Internet host name (for example, dominoserver.yourcompanyname.com). Additionally, the HTTP and DIIOP tasks must be enabled on the Domino server, by either adding them to the ServerTasks variable in the server's notes.ini file or, manually loading them at the server console.

The server document should be modified to allow and restrict access as desired. The following are some suggested settings:

On the Security tab, in the Server Access section:

- Access server – All users can access this server
- Not access server – blank
- Create new databases – blank (= everyone)
- Create replica databases – LocalDomainAdmins, LocalDomainServers, and the Domino Administrator account used by the LND Connector if that account is not a member of LocalDomainAdmins

On the Security tab, in the Programmability Restrictions section:

- Run unrestricted methods and operations – the Domino server name, the Domino Administrator account used by the LND Connector
- Run restricted LotusScript Java agents – the Domino Administrator account used by the LND Connector

On the Security tab, in the Internet Access section:

- Internet authentication – Few name variations with higher security

On the Ports tab, under Internet Ports, for DIIOP:

- Authentication options
- Name & password - Yes
- Anonymous - Yes

On the Internet Protocols tab, under HTTP, in the R5 Basics section

- Allow HTTP clients to browse databases – Yes

You must also copy the database templates, REGARCHV.NTF and REGCERTS.NTF from

<jcs-home>\resources\lnd

to the data folder of the Domino Server endpoint.

<lotus-home>\Data

While logged in as the Domino Administrator account used by the LND Connector,

1. Open each of the database templates Regarchv.ntf and Regcerts.ntf using Domino Designer.
2. In the database view window on the right-hand side, expand 'Shared Code' and select 'Agents'.

A list of agents located in each template is displayed.

3. Select each agent and click the Sign button.

This signs each of the agents by the administrative user that the connector is deployed within your environment.

4. After signing is complete, close the designer.

Note: This step is necessary before you acquire the endpoint for the first time. If you already have the databases regarc.nsf and regcert.nsf created, in addition to doing this step you will need to perform a 'Refresh design' in the Domino Administrator (switch to file view and select each regarc.nsf and regcert.nsf in turn and perform refresh by navigating to the file menu and selecting 'Database followed by Refresh Design').

More Information

[Install the LND Connector with JCS](#) (see page 286)

Administration Process (Adminp)

Identity Manager uses the Lotus Domino Administration Process (Adminp) to automate tasks that involve accounts, such as recertifying, renaming, upgrading to hierarchical naming conventions, and deleting. It also uses the ADMIN4.NSF database to store requests and responses.

The Adminp process works with Identity Manager so that when requests are made, Adminp transforms the account ID on the Lotus Notes/Domino server from a normal state to a pending state. The following states are maintained with the account status information:

- Change Expiration Pending
- Rename Pending
- Move in Hierarchy Pending
- Delete Pending

Both the Rename and Move in Hierarchy requests require accounts to accept the modification when the account connects to the server for the first time after the request is made. If the account accepts the change, Adminp performs a series of actions that are detailed and logged in the ADMIN4.NSF database and the REGARC.NSF database. These actions include modifications to the following databases:

- Address Book
- Access Control List
- Person Documents
- Free Time Database
- Calendar Entries and Profiles
- Reader and Author fields

When making a Rename request, the account must accept the change within the period of time that is specified in the Name_Change_Expiration_Days NOTES.INI setting. The account is prompted upon every subsequent server connection until the change request becomes obsolete.

Even though no defaults for this setting exist, change requests are deleted after 21 days if there is no specified setting. If the request becomes obsolete, Adminp deletes it from the ADMIN4.NSF database. You are then responsible for backing out the change made in the REGARC.NSF database.

As part of any of the three Custom requests—specifically Change Common Name, Move In Hierarchy, and Upgrade to Hierarchical, Identity Manager creates a secondary account ID and stores it in the REGARC.NSF database. The secondary account ID is for recovery purposes only and follows this format:

- If this is the first rename for the account ID, the secondary account ID will contain a new password with no encryption keys.
- If the account ID has been renamed before, the secondary account ID inherits the password of the previous account ID, but does not inherit its encryption keys.

To replace the secondary account ID with a copy of the real renamed account ID that was accepted by the user, you must establish a procedure to retrieve a copy of the real account ID from the user and store it in the REGARC.NSF database. If such a procedure is not established, any encryption keys associated with the real account ID will be lost if this ID is lost or becomes unusable. To make the account ID accessible, explore the Lotus Notes/Domino server again and remove the old account IDs. You will then be able to manage the renamed account IDs as though they were original account IDs.

If a user must retrieve mail or access documents that were encrypted with a previous version of the account ID you can retrieve a previous generation of the account ID from the REGARC.NSF database as long as you archived the proper version of that account ID.

Etautil Script Considerations

For this release of the Java LND connector, there are several things to be aware of regarding etautil scripts:

- The connector now uses Provisioning Server DNs to refer to all managed objects (except the administrative user value during an acquire). Any existing scripts will have to be changed to comply.
- AddCert objects only need to appear in LDAP ADD requests achieve their intended goal as effective function calls. No error will be returned if an LDAP DELETE request targeting an AddCert object is received, but such a request is not necessary (the AddCert object is transient and immediately deleted once the ADD request is processed).

Note: An AddCert request targeting an object can be used to update the object where it already exists. It behaves like an LDAP MODIFY request.

- The **etautil_addarchive.bat** script installed into **C:\Program Files\CA\Identity Manager\Provisioning Server\etc\Ind** has been updated to match the new connector behaviour.

LND Support for FIPS and IPv6

For release r12 of Identity Manager, the LND Connector does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire a Lotus Notes/Domino Server

Acquiring an LND endpoint is the first task you must perform before you can manage Lotus Notes/Domino accounts.

From the Endpoint Type task view

1. Register the server as an LND endpoint using the Provisioning Manager.

Use the LND Endpoint property sheet to register a Domino Server. During the registration process, the Provisioning Server identifies the Domino Server you want to administer and gathers information about it.

Note: Regarc and Regcert databases are created when a new endpoint is acquired. For the LND Connector to work, you must copy the database templates, REGARCHV.NTF and REGCERTS.NTF, from the <jcs-home>\resources\lnd folder to the data folder of the domino server endpoint, (<lotus-home>\Data), prior to acquiring the endpoint for the first time.

2. Explore the objects that exist on the endpoint.

After registering the server with the Provisioning Manager you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all LND accounts, organizations, organizational units, and groups that exist on the server. You can correlate the accounts with global users at this time or later

3. Correlate the explored accounts with global users.

When you correlate accounts, the Provisioning Server creates or links the accounts on an endpoint with global users. By correlating accounts, you can specify what fields are matched with global user fields. The Provisioning Server provides a default correlation account template for Lotus Notes/Domino endpoints. This account template performs the following actions in this order:

1. The Provisioning Server attempts to match the Lotus Notes/Domino Account short name with each existing global user's unique name. If a match is found, the Provisioning Server associates the Lotus Notes/Domino account with the global user. If a match is not found, the Provisioning Server performs the next step.
2. The Provisioning Server attempts to match the Lotus Notes/Domino Account name with each existing global user's full name. If a match is found, The Provisioning Server associates the Lotus Notes/Domino account with the global user. If a match is not found, the Provisioning Server performs the next step.
3. If Create Global User is checked, the Provisioning Server creates a new global user; otherwise, it associates the Lotus Notes/Domino account with the [default user] object.

Once the endpoint is acquired, your Lotus Notes/Domino server appears as an endpoint object and your accounts are organized according to their respective certifier container.

The position of user objects within the endpoint is specified by its context. The complete context or path from an object to the country of the endpoint tree identifies and forms the object's hierarchical name, for example, eTLNDAccountName=xx,eTLNDOrganizationUnitName=yy,eTLNDOrganizationName,eTLNDCountryName=zz. All hierarchical names must be unique in the endpoint.

Before accounts can be created under the explored organizations or organizational units, each of the account containers must have their certifier details registered in the regcert.nsf database.

To register the certifier details

1. Search for the appropriate certifier (eTLNDOrganization or eTLNDOrganizationalUnit) in Provisioning Manager, and select it.
2. Right click the certifier, and from the popup menu, select custom.
3. Select 'Certifier Detail' and provide the certifier ID location on the Domino server and the certifier ID password and type.

Managed Objects

Identity Manager organizes the following objects into a hierarchical endpoint tree:

- **Country object** depicts the country that is selected as the organizational root. This object is generally implicit in the Lotus Notes/Domino representation of the organizational hierarchy. Countries appear directly under the root container and their use is optional. Only Organization objects can be their direct children.
- **Organization objects** represent the Lotus Notes/Domino organization level certifiers that are registered with the Domino Administration Server and stored in the Domino Address Book. These can contain organizational unit objects or account objects. They can only appear under a Country object or root level.
- **Organizational Unit objects** represent the Lotus Notes/Domino organizational unit level certifiers that are registered with the Domino Administration Server and stored in the Domino Address Book. These can contain other organizational unit objects or account objects. (Maximum four OU objects).
- **Group objects** represent the groups on the Lotus Notes/Domino server. Group objects are leaf objects, but all appear directly under the single eTLNDGroupContainer container.

Note: LND groups cannot be added to other LND groups that are not in the same Domino directory. A group in the primary Domino directory cannot be added to a group that is in the secondary Domino directory, and vice versa.

- **Account objects** represent the accounts on the Lotus Notes/Domino server. Account objects are leaf objects and can appear under any Organization or Organizational Unit.

For more information about the managed objects or the endpoint schema, see the appendix "Endpoint Schema and Structure."

How Managed Objects are Referred to in the Java LND Connector

The LND Connector uses Provisioning Server DNs to refer to all managed objects (except the DN of the administrative account used to connect to the endpoint). This includes syntax used to distinguish LND "Unique OUs" from real Organizational Units. For example, previously a group may have named an account (with a Unique OU) that was a member of the group as, "CN=user,OU=uou,O=Acme". An equivalent reference using the new connector is "eTLNDAccountName=user/uou,eTLNDOrganizationName=Acme".

System Databases

The Provisioning Server uses five system databases to manage users. The first three databases originate from the Lotus Notes/Domino product. The last two databases are created when an LND endpoint is acquired by the Provisioning Manager and their templates have been copied to the Domino\data folder from `<jcs-home>\resources\lnd` folder.

Database	Description
ADMIN4.NSF	The Administration Process (Adminp) uses this database to post and respond to requests. You can approve requests that move users to different organization hierarchies, delete objects, delete mail files, and monitor Administration Process errors.
CERTLOG.NSF	Lists the names of all registered and certified users in a domain. This database is required if you want to use the Administration Process to simplify user management.
NAMES.NSF	Provides a domain-wide directory of the server, including its users, certifiers, foreign domains, and groups. This database includes documents that manage server-to-server communication and server programs.
REGARC.NSF	Stores archive documents for all managed accounts. Each archive document includes the login name, password, certificate expiration date, and a copy of the user ID file. Note: Agents in this template must be signed by the Admin account used by the Provisioning Server to connect to the Domino Server.
REGCERT.NSF	Stores certifier documents for all organization and organizational unit certifiers that certify accounts. Each certifier document includes the certifier name, type, password, and ID file. Note: Agents in this template must be signed by the Admin account used by the Provisioning Server to connect to the Domino Server.

Note: For details on the access privileges that you need to perform user management in your Lotus Notes/Domino domain, see the section, [Configure the Lotus Notes/Domino Connector](#).

Each time a request is sent or received, the Provisioning Server opens these databases and makes changes to the information stored in them. See [Step 1: Verify Your Access Privileges](#) (see page 287) for an important warning about correct access privileges. For more information about connecting to the system databases, see the Identity Manager Provisioning Help's LND Connector-Specific section.

Locations for Storing User IDs

When creating users, check the "User ID File Path" check box on the UserID tab, if you want the ID file to be stored on the LND server machine. You must enter the absolute path (on the LND server machine) appended with the ID file into the text field (for example, C:\Program Files\Lotus\Domino\data\user.id). However, if you want the ID File to be stored on a machine other than the LND server machine, you must enter the full UNC path that includes the proper drive's name, (for example, \\server\c\$\share\user.id, the c\$ cannot be omitted).

As the paths being selected are used on the LND endpoint, the "Browse..." and "Change ID..." buttons which allowed paths on the Provisioning Manager to be selected, have been removed from this release.

Locations for Storing Certifier IDs

When creating organizational units, fill in the "Specify a Location for the Certifier ID (Path and Filename)" field on the Organization Certifiers tab, if you want the ID file to be stored on the LND server machine. You must enter the absolute path (on the LND server machine) appended with the ID file into the text field (for example, C:\Program Files\Lotus\Domino\data\certifier.id). However, if you want the ID File to be stored on a machine other than the LND server machine, you must enter the full UNC path including a proper drive name, (for example, \\server\c\$\share\certifier.id, the c\$ cannot be omitted).

As the paths being selected are used on the LND endpoint, the "Browse..." and "Change ID..." buttons which allowed paths on the Provisioning manager machine to be selected, have been removed from this release.

DJX Support

DJX extensions are now supported by LND Connector and are managed through the Custom Attributes tab.

See [Custom Attribute Support](#) (see page 299), for more information on this feature.

Custom Attribute Support

Several enhancements have been made in this release for custom attribute support. They include:

- Custom attributes eTLNDCustomAttribute01-50 and eTLNDCustomCapabilityAttribute01-50 (policySync="yes") can be mapped using the XML file <jcs-home/conf/override/lnd/lnd_custom_metatdata.xml. This file adheres to the XML schema that can be found here:

<jcs-home>/conf/xsd/datamodel.xsd

The custom metatdata file, if present, is merged with the static lnd_metadata.xml file included in the connector to provide the complete runtime metadata to be used by the connector.

- Two sample custom metadata files are included for reference. To active either of these mappings, make a copy of the sample file (not changing the directory) and rename this copy to lnd_custom_metatdata.xml.

SAMPLE.lnd_custom_metadata.xml includes two simple mappings to provide easy verification of the custom mapping functionality.

- DJX_SAMPLE.lnd_custom_metadata.xml includes mappings for each of the attributes used for the DJX extension to LND. This file provides a useful starting point (or complete solution) for users interested in DJX. This file also includes an example of an LND "Dialog" field being handled as a Boolean value.

- Only power users should modify the custom metatdata file and should take precautions like saving a backup copy of any existing file before updating. Tests to verify mapping changes should be conducted immediately after modifications are made, as any syntax errors introduced will render any LND connector hosted by the modified JCS inoperable until a valid custom file is reinstated (or the offending custom mapping file deleted). Error messages in the Provisioning Manager clearly distinguish failures in the parsing of this file so remedial action can be taken immediately.

- If customized mappings need to be active on multiple JCS installations, the same lnd_custom_metadata.xml file needs to be copied to

<jcs-home>/conf/override/lnd/

on each of them.

- All metadata settings that can be used in custom mapping files are documented in the MetaDataUtil.java class of the Java CS available as either JavaDoc with the Java CS SDK or as part of the "Programming Guide for Java Connector Server r12. You can find this guide in the r12 Bookshelf.
- Attribute values entered on the Custom Attributes tab are subject to validation by the connector. For example, integer fields emit a validation failure when non-digit characters are present.

- The values provided for any custom attribute configured to be date or dateTimes on the Custom Attributes tab, must be entered in the UTC time zone, not local time, unless the computers on which the client is running (for example, Provisioning Manager / etautil script) and the LND endpoint are configured to use the same time zone..

Alternative Languages Support for both Organization and Organizational Unit Certifiers

You can now Add, Delete, Query, and Modify the alternative language list from the Provisioning Manager. When adding an alternate language, both the normal (long) name and short name for the language is accepted. For example, both Japanese and ja are valid names. You must separate multiple languages by commas. The normal (long) name of the language is now displayed from the Provisioning Manager. The short name is stored in the repository the same as the previous C++ Connector.

The following are the alternative languages available for this Connector:

- Arabic [ar]
- Byelorussian [be]
- Bulgarian [bg]
- Catalan [ca]
- Czech [cs]
- Danish [da]
- German [de]
- Greek [el]
- English [en]
- Spanish [es]
- Estonian [et]
- Finnish [fi]
- French [fr]
- Gujarati [gu]
- Hebrew [he]
- Hindi [hi]
- Croatian [hr]
- Hungarian [hu]
- Indonesian [id]
- Icelandic [is]
- Italian [it]
- Japanese [ja]
- Korean [ko]
- Lithuanian [lt]
- Latvian [lv]

- Macedonian [mk]
- Marathi [mr]
- Malay [ms]
- Dutch [nl]
- Norwegian [no]
- Polish [pl]
- Portuguese [pt]
- Romanian [ro]
- Russian [ru]
- Slovak [sk]
- Slovenian [sl]
- Albanian [sq]
- Serbian [sr]
- Swedish [sv]
- Tamil [ta]
- Telugu [te]
- Thai [th]
- Turkish [tr]
- Ukrainian [uk]
- Vietnamese [vi]
- Konkani [x-KOK]
- Chinese (Simplified) [zh-CN]
- Chinese (Traditional) [zh-TW]

Organizations and Organizational Units Handling Extended

Organizations and Organizational Units can now be deleted. However, it is not possible to delete these objects while they still contain child objects.

Note: Organizations cannot be created using the Provisioning Manager, although explored and listed organizations can be deleted.

Correlate on Shortname

The default correlation attribute used by the Lotus Notes/Domino Connector is the Shortname attribute. When creating new global users, they will, by default, have the LND account's shortname as the global user name. Accounts are then correlated to existing global users if the global user name matches the LND account's shortname value.

To use another attribute as the correlation attribute, for example, the full name, follow these steps:

1. From the System task in Provisioning Manager, select Domain Configuration.

Several folders appear in the right-hand frame.

Select Explore and Correlate, Correlation attribute from the right-hand frame

The Domain Configuration tab appears.

2. From the Domain Configuration tab, modify the Correlation attribute appropriately.

For example, to change the correlation attribute to the full name, you would set the Correlation Attribute to "GlobalUserName=Lotus Domino Server:Name".

Attribute Mapping

In order to improve performance, only a bare minimum of attributes is retrieved from the Domino server during exploration. By default, most Domino attributes are not mapped to the Global Users. Should you need to populate Global User information from the Domino database, this information can be retrieved by defining more attribute mappings than the default. Follow these steps to set up attribute mapping:

1. Select Use custom settings from the Attribute Mapping tab.
2. Click Set Default and define at least one additional attribute mapping.

The LND Connector is now forced to retrieve all data from the Domino server.

Note: Exploration times are likely to increase due to extra information retrieval from the Domino endpoint.

LND Account Templates

The Lotus Notes/Domino Default Policy, provided with the Lotus Notes/Domino Connector, gives a user the minimum security level needed to access an endpoint. You can use this account template as a model to create new account templates.

Identity Manager lets you manage provisioning roles and account templates from the Roles task view. You can use the Role property sheet to manage your provisioning roles and the Lotus Notes/Domino Account Template property sheet to manage your account templates. For example, with Lotus Notes/Domino you give a person access to the Lotus Notes/Domino server by registering the person using the Lotus Notes Domino Client. When registering a user, the connector creates a Person document in the Public Address Book (PAB), a user ID file, and a server-based mail file that defines the types of mail the user can receive. (The PAB is also called the Domino Endpoint.)

Similarly, an Internet user is defined as someone who is required to provide a password when accessing a Lotus Notes/Domino server or someone who uses client authentication with Secure Sockets Layer (SSL). In addition, this user uses either no mail or Internet mail, in which case a user ID and mail file are not necessary. An Internet user can be added by the connector creating a Person document in the Public Address Book (PAB). The document contains information about the user's name and Internet password.

In Identity Manager, you register both of these users by dragging them onto a provisioning role that has a Lotus Notes/Domino account template defined and a Lotus Notes/Domino endpoint associated with the account template.

Archive Database Data Collection

Before password synchronization can take place, all current Notes account ID files with their passwords need to be obtained. The repository for these account IDs and passwords is the existing Archive database. Keeping this repository current allows for ID and password recovery. If you lose your account ID, the Administrator can retrieve the current account ID and password from the Archive database and send them to you.

To obtain the current account IDs and passwords, the archive database on the Domino server needs to be designated as "Mail-in" database and the *Send ID to Archive* DB hidden agent needs to be copied to all user mailfiles by the Administrator. The agent can be copied in one of the following ways:

- Using the Domino Designer client, copy the hidden agent from the Archive DB to each mailfile individually.
- Using the Domino Designer client, copy the hidden agent from the Archive DB to the mail template and let the Designer task automatically update the mailfiles. (recommended) By default, the Designer task runs daily at 1:00 a.m.

This agent gets the user's Notes account ID specified by the "KeyFilename" entry in their notes.ini file on the Domino Client, prompts the user to enter his or her password and then mails these items to the Archive database. The Archive DB must be configured as a Mail-in Database in the Domino endpoint using the Mail-in name "Archive Database".

Once the agent is present in the user mailfiles, a mail message is sent notifying them that their account ID and password need to be sent to the Archive database. This message contains a button that activates the *Send ID to Archive DB* hidden agent which retrieves the ID file and mails both ID and password to the Archive database.

You must sign the agents with a signature that is valid in your organization in order for the new agents to run successfully. To do this, edit and save each agent in the Domino Designer client.

If a database is designed to receive mail, you must create a Mail-In Database document in the Domino Directory. This document must exist in the Domino Directory of every server that stores a replica of the database. The database cannot receive mail until you create this document.

To create a Mail-In Database Document

1. Make sure you have at least Author access with the Create Documents privilege selected.
2. From the People & Groups tab of the Domino Administrator, choose the Mail-In Databases Resources view, and click Add Mail-In database.
3. On the Basics tab, complete these fields:

Mail-in name: "Archive Database"

Domain: <Your domain name>

Server: <Your server>

File name: regarc.nsf

4. Save the document.

Another hidden agent called *Update ID File* has been added to the Archive database. This agent gets the current Archive documents for the user whose ID has been received and replaces the ID and password values on the document with those received in the mailed-in document. If a previous Archive document exists for that user, a new document containing the new ID and password is linked to the Archived document.

The RegXArchive encryption key must also be available in the current User.ID of the Administrator as well as the Server.ID of the Registration server to let the mail-triggered background agent in the Archive database run successfully. Alternatively, the agent can be run manually in the foreground by the Administrator if the encryption key cannot be added to the Server.ID.

You must have at least Designer access with Create LotusScript/Java agent to the user mailfiles in order to copy the hidden agent.

Add the following parameter to the NOTES.INI file on the Registration server:

```
Mgr_DisableMailLookup = 1
```

This parameter lets the mail-triggered agent in the Archive database to run even if the server is not the mail server for the Administrator.

A third, optional agent, *Remove ID Agent from User Mailfiles* can be added to the Archive database. This agent can be run manually by the administrator to remove the hidden agent from user mailfiles after the ID repository has been created.

Password Synchronization

The administrative user of Identity Manager can change the password associated with an account's ID file in one of the following ways:

- Directly modifying the account
- Propagate a Global User password change to associated accounts.

Once the password is changed, an email is sent to the account, optionally, including the new server ID file.

To customize the subject and body of the email that is sent, set the following parameters in the NOTES.INI file on the Domino server:

"\$Password_Change_Subject=" specifies the message body to be used: If not specified, the parameter defaults to a generic subject.

"\$Password_Change_Message=" specifies the message body to be used. If not specified, the parameter defaults to a generic body.

"\$Password_Change_AttachID=" specifies whether the new ID file is attached to the message. If not specified, the default is "Yes". Any value other than a case-insensitive match to "Yes" is interpreted as "No."

"\$Password_Send_To=" specifies who receives the message.

Suspend/Resume

A combobox called Status located on the Profile tab of the Account and Account Template Property sheets provides a form of suspension using Deny Access groups.

When Status is set to Active, the account is not in a Deny Access group on the Domino server. When Status is set to Suspended, the account has been added to a Deny Access group on the Domino server.

Note: This functionality is currently limited in the number of accounts that can be concurrently suspended. For Domino 6.x, the limit is 64 KB. For Domino 7.x, the limit is 32 KB.

LND Accounts

To manage LND Accounts, some manual steps are required.

Each Organization or Organizational Unit must have an entry in RegCert.nsf to permit Identity Manager access.

To create this entry, do the following:

1. Explore the Lotus/Domino endpoint.
2. Expand Organizations or Organizational Units in the List Tab.
3. Select an item and right-click it to select Custom, then Certifier Details.
4. Fill in all mandatory fields (Name, Storage location, and Password of the Certifier ID).

Account Custom Operation (Rename, Recertify, Move In Hierarchy)

For Account Custom Operation (Rename, Recertify, and Move In Hierarchy), you must add an entry in RegArc.nsf for explored accounts. This is only for Accounts created with native tool and explored with the Provisioning Server.

To create this entry, do the following:

1. Explore the Lotus/Domino endpoint.
2. Expand Accounts in a List Tab.
3. On the History tab of each account for which you want to add an archive entry, click the Add/Update Archive button.
4. Fill in all mandatory fields (Location and Password of the Certifier Id).
5. Click OK.

Modify Home Server

The Provisioning Manager allows the Home Server field to be modified on LND accounts. Changing the Home Server only changes the value on the LND account. The mail file is not moved to the new server. You must still ensure that a mail file exists on that server.

Management of Alternate Names on LND Accounts

The LND connector supports the management of alternate names on your LND accounts. The account ID must be certified by a certifier ID that has at least one alternate name configured for it in order to add alternate name information. To include the management of alternate languages on certifier files, the administrator must perform the following steps prior to using this new functionality:

1. Use the Domino Administrator (see Domino Administrator Help for more information) to configure the certifier ID with alternate names.
2. Update the existing Certifier documents for each certifier in the Certifiers database by using the Domino Administrator client to delete the existing certifier ID file from the Certifier document and then attach the updated certifier ID. You must also supply a password for the password field.

Note: This step is necessary any time the alternate name information is changed in a certifier ID file.

3. Update each Organization or Organizational Unit certifier that contains alternate information within the Provisioning database. The multi-valued attribute eTLNDOrgCertAltLanguageList for Organization and Organizational Unit objects must contain all the languages supported by the certifier.

You can Add, Delete, Query, and Modify the language list from the Provisioning Manager by using the LND Organization and Organization Unit management dialogs. The language codes are automatically expanded to language names when added. However, you can still use etauil to add or update the list. See [Sample etauil Commands](#) (see page 310) for an example.

Only those valid languages added to the Organization or Organizational Unit objects in the Provisioning database are displayed as choices when creating accounts using that Org or OU. For a list of languages and associated codes, see [Alternative Languages Support for both Organization and Organizational Unit Certifiers](#). (see page 301)

LND Etauil Conventions

Use the following Lotus Notes/Domino conventions in your etauil commands:

- The endpoint type name (eTNamespaceName) is Lotus Domino Server
- The endpoint type prefix is LND. For example, some of the Lotus Notes/Domino object class names are as follows:
 - eTLNDDirectory for an endpoint
 - eTLNDPolicyContainer for an account template container
 - eTLNDPolicy for an account template

Sample etautil Commands

Update Language List

The following is an example etautil command to update the Language List for an Organization or Organizational Unit using etautil:

```
etautil -d DOEJO03W2K3 -u etaadmin -p password update  
'eTLNDDirectoryName=doeja03w2k3,eTNamespaceName=Lotus Domino Server,dc=DOEJA03W2K3,dc=eta'  
eTLNDOrganization eTLNDOrganizationName=' eTLNDOrganizationName=cai' to  
+eTLNDOrgCertAltLanguageList='ko' +eTLNDOrgCertAltLanguageList='fr'
```

Modify Home Server

The following is an example etautil command to modify the Home Server on an LND account using the Provisioning Manager and etautil:

```
etautil -d DOEJA01XP -u <eta administrator> -p <eta password>  
'eTLNDOrganizationName=cai,eTLNDDirectoryName=doeja01w2k3,eTNamespaceName=Lotus Domino  
Server' eTLNDAccount eTLNDAccountName='mail location03' eTLNDHomeServer='CN=doeja01w2k3/O=cai'
```

You must ensure that the Mail Server attribute in the user's person document is updated correctly.

Note: The mail file must be manually created on the new server.

Modify Mail Quota and Warning Threshold

The following is an example etautil command to modify the Mail Quota and Warning Threshold values of an LND account using the Provisioning Manager and etautil:

```
etautil -d DOEJA01XP -u <eta administrator> -p <password> update  
'eTLNDOrganizationName=cai,eTLNDDirectoryName=doeja01w2k3,eTNamespaceName=Lotus Domino  
Server' eTLNDAccount eTLNDAccountName='file test02' eTLNDMailFileQuota=500  
eTLNDMailFileThreshold=400
```

Delete Accounts (With and Without Mail Files)

The following is an example etautil command to delete LND accounts using etautil:

```
etautil -d DOEJA01XP -u <eta administrator> -p <password> update  
'eTLNDOrganizationName=cai,eTLNDDirectoryName=doeja01w2k3,eTNamespaceName=Lotus Domino  
Server' eTLNDAccount eTLNDAccountName='mail file02' eTLNDAdminp=1 eTLNDAccountOperation=2  
eTLNDAccountState=2 eTLNDDelFlag=0
```

The possible values for eTLNDDelFlag, which controls deletion of the mail file are as follows:

- 0 = Don't delete mail file(s)
- 1 = Delete primary mail file
- 2 = Delete primary mail file and all replicas

Rename Accounts

The following is an example etautil command to rename LND accounts using etautil:

```
etautil -d DOEJA01XP -u <eta administrator> -p <password> update  
  
eTLNDOrganizationName=cai,eTLNDDirectoryName=doeja01w2k3,eTNamespaceName=Lotus Domino Server'  
eTLNDAccount eTLNDAccountName='rename me107' eTLNDAdminp=1 eTLNDAccountOperation=3  
eTLNDAccountState=3 eTLNDNewLastName=me107 eTLNDNewInitials="" eTLNDNewFirstName=renaming
```

Move Accounts in Hierarchy

The following is an example etautil command to move LND accounts using etautil:

```
etautil -d DOEJA01XP -u <eta administrator> -p <password> update  
'eTLNDOrganizationalUnitName=ou1,eTLNDOrganizationName=cai,eTLNDDirectoryName=doeja01w2k3,eTNa  
mespaceName=Lotus Domino Server' eTLNDAccount eTLNDAccountName='move me202' eTLNDAdminp=1  
eTLNDAccountOperation=5 eTLNDAccountState=5 eTLNDNewOrganization='eTLNDOrganizationName=cai'
```

Recertify Accounts (by Exact Date and by Number of Months)

The following are example etautil commands to recertify users by using both Provisioning Manger and etautil.

By exact date:

```
etautil -d DOEJA01XP -u <eta administrator> -p <password> update  
'eTLNDOrganizationName=cai,eTLNDDirectoryName=doeja01w2k3,eTNamespaceName=Lotus Domino  
Server' eTLNDAccount eTLNDAccountName='exp date01' eTLNDAdminp=1 eTLNDAccountOperation=1  
eTLNDAccountState=1 eTLNDExpiration='10/13/2010 12:00:00 PM'
```

By number of months:

```
etaultil -d DOEJA01XP -u <eta administrator> -p <password> update  
'eTLNDOrganizationName=O:cai,eTLNDDirectoryName=doeja01w2k3,eTNamespaceName=Lotus Domino  
Server' eTLNDAccount eTLNDAccountName='exp date01' eTLNDAAdminp=1 eTLNDAccountOperation=1  
eTLNDAccountState=1 eTLNDExpireMonths=24
```

To confirm that the ID was recertified correctly, perform a “Refresh Status” on the account in Provisioning Manager to update the Archive document. Also view the account in Provisioning Manager to ensure the expiration date displays properly.

Chapter 11: Host Access Management Connectors

This section contains the following topics:

[CA Access Control Connector](#) (see page 314)

[CA SSO Connector for Advanced Policy Server](#) (see page 337)

[Kerberos Connector](#) (see page 345)

The following connectors are considered to be host access management connectors:

- CA Access Control
- CA Single Sign-On for Advanced Policy Server
- Kerberos

The information in this chapter describes how to install and manage these connectors.

CA Access Control Connector

The CA Access Control Connector lets you administer accounts and groups on CA Access Control servers.

The CA Access Control Connector provides a single point for all user administration by letting you do the following:

- Register endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage CA Access Control accounts using account templates specific to CA Access Control
- Change account passwords and account activations in one place
- Synchronize global users with their roles or synchronize global users' accounts with their account templates
- Assign a CA Access Control account template to each of your CA Access Control endpoints
- Use the default Endpoint Type account template to create accounts with the minimum level of security needed to access a CA Access Control endpoint
- Create and manage CA Access Control groups
- Generate and print reports about CA Access Control accounts and groups
- Create and manage objects of the supported CA Access Control resource classes.

This connector is managed using the Connector and C++ Server installation process.

Note: For more information and requirements, see *Connector and C++ Connector Server Installation* (see page 23).

Recommended Patch Levels

If you are using the Solaris, HP-UX, Linux, or AIX version of CA Access Control UNIX r5.3, you must apply the mandatory patch for the corresponding version of CA Access Control. Consult CA Access Control Customer Support to obtain the latest revisions of these mandatory patches.

If you are using the CA Access Control Connector for UNIX, you must install the latest revision of CA Access Control UNIX r8.0 on the UNIX system where the C++ Connector Server is to be run.

ACC Connector Multi-Threading Support

The current release of the CA Access Control interface library does not support multiple threads. Hence, the CA Access Control connector cannot be multi-thread enabled.

Runtime Environment Settings

The following are the runtime environment settings for the CA Access Control Connector for Windows and the CA Access Control Connector for UNIX.

Setting the Encryption Key for the CA Access Control Connector

If the CA Access Control Connector has to use an encryption key other than the default one to manage your CA Access Control systems, issue the following commands at the prompt on the Provisioning Server to enter a new encryption key:

```
cd PS_HOME\Provisioning Server\etc\acc  
CHANGE_EAC_KEY
```

Important! Restart the Windows service Identity Manager C++ Connector Server after the new encryption key is set.

Resetting the Encryption Key for the CA Access Control Connector Back to the Default Key

If the CA Access Control Connector has to use the default encryption key to manage your CA Access Control systems, do not change the encryption key. If you need to change your new encryption key back to the default encryption key, issue the following commands at the prompt on the Provisioning Server:

```
cd PS_HOME\Provisioning Server\etc\acc  
RESET_EAC_KEY
```

Important! Restart the Windows service Identity Manager C++ Connector Server after the new encryption key is set.

Changing the Encryption Method for the CA Access Control Connector

If the CA Access Control Connector has to use an encryption method other than the default one to manage your CA Access Control systems, edit the following Windows registry entry on the Provisioning Server and set the value to the path name of the DLL for the new encryption method:

```
HKEY_LOCAL_MACHINE\SOFTWARE\ComputerAssociates\Identity Manager\Provisioning  
Server\NSOptions\ACC\Trust Access Control SDK\Encryption Package
```

For example, you can change the value of the **Encryption Package** from C:\Program Files\CA\Identity Manager\Provisioning Server\etc\acc\defenc.dll for the default encryption to C:\Program Files\CA\Identity Manager\Provisioning Server\etc\acc\tripleDESenc.dll for triple-DES encryption.

Note: The directory *PS_HOME*\Provisioning Server\etc\acc contains the encryption DLLs for the default encryption, DES encryption, triple-DES encryption, and AES encryption.

Important! Restart the Windows service C++ Connector Server after the encryption method is changed.

CA Access Control Connector for UNIX

CA Access Control UNIX r8.0 must be running on the same UNIX system where the CA Access Control Connector is installed. Otherwise, the CA Access Control Connector will not work. For Solaris, if CA Access Control UNIX r8.0 is installed after the CA Access Control Connector is installed, you must add the library pathname for CA Access Control UNIX r8.0 to the environment variable LD_LIBRARY_PATH. For example, if CA Access Control UNIX r8.0 is installed in directory /opt/CA/eTrustAccessControl, add the pathname /opt/CA/eTrustAccessControl/lib to LD_LIBRARY_PATH. You do not have to add the path name if CA Access Control UNIX r8.0 is already installed before the CA Access Control Connector is installed.

The CA Access Control Connector shares the same runtime environment with CA Access Control UNIX r8.0. Therefore, they will have the same encryption settings. In order to change the encryption method or key of the CA Access Control Connector, you have to change the corresponding one of CA Access Control UNIX r8.0. Refer to the documentation for CA Access Control UNIX r8.0 for information about changing the encryption settings. You need to restart the C++ Connector Server process after the encryption settings are changed.

Configuring CA Access Control UNIX on the C++ Connector Server System

Note: This section is only applicable to the CA Access Control Connector for UNIX.

The UNIX user who invokes the C++ Connector Server process for Identity Manager must be properly defined to CA Access Control UNIX. By default, UNIX user *etaslappd* is to run the C++ Connector Server process.

Start CA Access Control command *selang* on the UNIX system where the C++ Connector Server is installed and issue the following commands in *selang*:

```
eu etaslappd admin
```

```
auth terminal superagent_workstation_name uid(etaslappd) acc(a)
```

where

superagent_workstation_name

Is the machine name of the UNIX system where the superagent is installed.

Configuring a CA Access Control UNIX or Windows Server

To configure your CA Access Control UNIX or Windows server for Identity Manager, follow these steps:

1. Start the *selang* command interpreter.
2. Create the system administrator's account on the CA Access Control server if it does not already exist.
3. Authorize Identity Manager to connect to the CA Access Control server.
4. Enable the administrator's account to connect from the Provisioning Server.
5. Install Filtering Rules for the Policy Model Database (PMDB).

Note: You can also use the Identity Manager for Access Control utility (SeAM) to perform these authorizations.

Starting the Slang Command Interpreter

To begin the configuration process, start the *selang* command interpreter on the CA Access Control server system as follows:

- a. Change directory to the home directory of CA Access Control from a UNIX or Window prompt.
- b. Change directory to *bin* and then enter the command *selang*

Creating the System Administrator's Account

Create an administrator's account on the CA Access Control server using the user ID and password that you use when logging on to the Provisioning Server. To do this, issue the following commands in selang:

```
nu administrator_name password (administrator_password) admin auditor
```

administrator_name

Is the user ID that you use to log on to the Provisioning Server.

administrator_password

Is the administrator's password for the user ID.

Important! It is strongly recommended that you do not use a user ID named "Administrator" to define a CA Access Control directory for Windows 2000, because doing so may cause login failures when you try to access the directory.

Ensure that you add the admin and auditor keywords to the command. This gives you administrative privileges.

Next, you must create the administrator's account and password in the native operating system (UNIX). To do this, issue the following commands in selang:

```
env(native)
```

```
eu administrator_name password(administrator_password)
```

```
env(seos)
```

administrator_name

Is the user ID that you use to log on to the Provisioning Server.

administrator_password

Is the administrator's password for the user ID.

Authorizing Access to the CA Access Control Server

To give the Provisioning Server access to the CA Access Control server, issue the following command in selang:

```
nr TERMINAL workstation_name owner(terminal_owner) defacc(R)
```

workstation_name

Is the machine name of the Provisioning Server.

terminal_owner

Is the owner of the terminal.

For example, if the *workstation_name* is cacc.la.com and the *terminal owner* is nobody, enter the following:

```
nr TERMINAL cacc.la.com owner(nobody) defacc(R)
```

Enabling the Administrator's Account

Issue the following command in selang so that the administrator's account can access the CA Access Control server:

```
auth TERMINAL workstation_name acc(a) uid(administrator_name)
```

workstation_name

Is the machine name of the Provisioning Server or Identity Manager clients.

administrator_name

Is the administrator's account that was created in Creating the System Administrator's Account.

For example:

```
auth TERMINAL cacc.la.com acc(a) uid(accadmin)
```

Installing Filtering Rules for the Policy Model Database (PMDB)

The following step should be performed after you enable the administrator's account.

The following PMDB filtering rules should be specified for each PMDB on the CA Access Control server if you want to administer the PMDB. These rules prevent internal updates to the pre-defined account `__ACCAgt` (use two underscores with this account name) from being propagated to the subscribers of the PMDB.

```
#-----  
#ACCESS ENV. CLASS OBJECTS PROPERTIES ACTION  
#-----  
MODIFY eTrust USER __ACCAgt * NOPASS  
CREATE eTrust USER __ACCAgt * NOPASS  
DELETE eTrust USER __ACCAgt * NOPASS
```

For example, if the PMDB is for CA Access Control for UNIX, add these rules to the filter file specified in the *pmd* section of the *pmd.ini* file for the PMDB. For CA Access Control for Windows, the filter file is specified in the registry for the PMDB. For either platform, create the filter file if it does not exist.

The *Utilities Guide* for CA Access Control for UNIX and the *Administrator Guide* for CA Access Control for Windows provide the instructions for setting up filtering rules for PMDB propagation.

ACC Support for FIPS and IPv6

For release r12 of Identity Manager, the CA Access Control Connector does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire the CA Access Control Server

You must acquire the CA Access Control server before you can administer it with Identity Manager. When acquiring a CA Access Control server, perform the following steps from the Endpoint Type task view:

1. Register the server as an endpoint in Identity Manager.

Use the CA Access Control property sheet to register a CA Access Control server. During the registration process, Identity Manager identifies the CA Access Control server you want to administer and gathers information about it.

Note: Ping the node name from the Provisioning Server. If the ping is successful, then you know that Identity Manager will find the CA Access Control node.

2. Explore the objects that exist on the directory.

After registering the server in Identity Manager, you can explore its contents, using the Explore and Correlate Endpoint dialog. The Exploration process finds all CA Access Control accounts and groups. You can correlate the accounts with global users at this time, or you can correlate them later.

3. Correlate the explored accounts with global users.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users. By correlating accounts, you can specify what fields are matched with global user fields. Identity Manager provides a default correlation account template for CA Access Control endpoints. This account template performs the following actions in this order:

- a. Identity Manager attempts to match the account name with each existing global user's unique name. If a match is found, Identity Manager associates the CA Access Control account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the full name with each existing global user's full name. If a match is found, Identity Manager associates the CA Access Control account with the global user. If a match is not found, Identity Manager performs the next step.
- c. If the Create Global Users as Needed button is selected, Identity Manager creates a new global user and then associates the CA Access Control account with the global user. If the Create Global Users as Needed button is cleared, Identity Manager performs the next step.
- d. Identity Manager associates the CA Access Control account with the [default user] object.

ACC Account Templates

The CA Access Control Default Policy, provided with the CA Access Control Connector, gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

Important! When you associate a new endpoint to an account template, the new endpoint must contain all applications, application groups, and groups that have already been selected for the associated account template. For details, see [Associating Account Templates with Endpoints](#).

File Class Administration

A property sheet called the CA Access Control File has been added so that you can protect files on a Windows or UNIX operating system.

General Tab on CA Access Control File Property Sheet

From this tab you can add file records to the Access Control database to be protected on a CA Access Control Windows or UNIX system.

Default Access Tab on CA Access Control File Property Sheet

From this tab you can grant access to an account or group who is not defined to CA Access Control or do not appear in the access control list of the resource. However, an account or group must exist in the CA Access Control database.

Authorization Tab on CA Access Control File Property Sheet

From this tab you can set the access control lists or ACLs of a file. You can also set the desired file permissions in the Permissions window on this tab.

Audit Tab on CA Access Control File, Program, Terminal, Loginappl, Surrogate, and Regkey Property Sheets

From this tab you can specify various properties for auditing access to a resource.

Day/Time Restrictions Tab on CA Access Control File, Program, Terminal, Loginappl, Surrogate, and Regkey Property Sheets

From this tab you can specify the days and hours when access to a resource is allowed.

ACC Statistics Tab on CA Access Control File, Program, Terminal, Loginappl, Surrogate, and Regkey Property Sheets

From this tab, you can view the usage statistics for a resource that has been created in the CA Access Control database.

TERMINAL Class Administration

A property sheet called CA Access Control Terminal has been added so that you can create, modify, delete, or duplicate terminal records in the CA Access Control database.

General Tab on CA Access Control Terminal Property Sheet

From this tab you can add terminal records to the Access Control database to be protected on a CA Access Control Windows or UNIX system.

Default Access Tab on CA Access Control Terminal Property Sheet

From this tab you can grant access to an account or group who is not defined to CA Access Control or do not appear in the access control list of the resource. However, an account or group must exist in the CA Access Control database.

Authorization Tab on CA Access Control Terminal Property Sheet

From this tab you can set the access control lists or ACLs of a terminal.

PROGRAM Class Administration

A property sheet called CA Access Control Program has been added so that you can protect programs on a Windows or UNIX operating system.

General Tab on CA Access Control Program Property Sheet

From this tab you can add program records to the CA Access Control database to be protected on a CA Access Control Windows or UNIX system.

Trusted Program Tab on CA Access Control Program Property Sheet

From this tab you can specify the unique security properties of a program, such as the program information to monitor.

Default Access Tab on CA Access Control Program Property Sheet

From this tab you can grant access to an account or group who is not defined to CA Access Control or do not appear in the access control list of the resource. However, an account or group must exist in the CA Access Control database.

Authorization Tab on CA Access Control Program Property Sheet

From this tab you can set the access control lists or ACLs of a program.

Note: You must be running at minimum, CA Access Control r5.3 UNIX or CA Access Control Windows r8.0 in order for the Provisioning Manager to enable the program Condition field.

LOGINAPPL Class Administration

A property sheet called CA Access Control Loginappl has been added so that you can create, modify, delete, or duplicate loginappl records in the CA Access Control database. Only the UNIX version of CA Access Control supports this resource.

General Tab on CA Access Control Loginappl Property Sheet

From this tab you can add loginappl records to the CA Access Control database to be protected on a UNIX system.

Default Access Tab on CA Access Control Loginappl Property Sheet

From this tab you can grant access to an account or group who is not defined to CA Access Control or do not appear in the access control list of the resource. However, an account or group must exist in the CA Access Control database.

Authorization Tab on CA Access Control Loginappl Property Sheet

From this tab you can set the access control lists or ACLs of a loginappl.

Surrogate Class Administration

A property sheet called CA Access Control Surrogate has been added so that you can create, modify, delete, or duplicate surrogate records in the CA Access Control database.

Note: Surrogate Class is not supported by the Windows NT version of CA Access Control.

General Tab on CA Access Control Surrogate Property Sheet

From this tab you can add surrogate records to the CA Access Control database to be protected on an Access Control Windows or UNIX system. The name of a surrogate must be in the form of *USER.name* or *GROUP.name*.

Default Access Tab on CA Access Control Surrogate Property Sheet

From this tab you can grant access to an account or group who is not defined to CA Access Control or do not appear in the access control list of the resource. However, an account or group must exist in the CA Access Control database.

Authorization Tab on CA Access Control Surrogate Property Sheet

From this tab you can set the access control lists or ACLs of a surrogate.

REGKEY Class Administration

A property sheet called CA Access Control REGKEY has been added so that you can create, modify, delete, or duplicate regkey records in the CA Access Control database. Only the Windows version of CA Access Control supports this resource class.

General Tab on CA Access Control Regkey Property Sheet

From this tab you can add regkey records to the CA Access Control database to be protected on a CA Access Control Window system.

Default Access Tab on CA Access Control Regkey Property Sheet

From this tab you can grant access to an account or group who is not defined to CA Access Control or do not appear in the access control list of the resource. However, an account or group must exist in the CA Access Control database.

Authorization Tab on CA Access Control Regkey Property Sheet

From this tab you can set the access control lists or ACLs of a regkey.

ACC Etautil Conventions

Use the following CA Access Control conventions in your etautil commands:

The endpoint type name (eTNamespaceName) is Access Control.

- The endpoint type prefix is ACC. The CA Access Control class names are:
 - eTACCDirectory for an endpoint class
 - eTACCPolicyContainerName for an account template container
 - eTACCPolicy for an account template object

Associating ACC Account Templates with Endpoints

When associating account templates with endpoints, you must follow the intersection *requirement* for account templates: For any new endpoint that you want to associate with an account template, all groups, categories, and security labels that have previously been selected for the account template *must* exist in the new endpoint. If any of the previously selected groups, categories, or security labels for the account template do not exist in the new endpoint, the attempt to associate the new endpoint to the account template fails, and an error message is displayed.

Sample Scenario Using Groups

This sample scenario uses groups to illustrate the intersection requirement. At the beginning of this scenario, the following are true:

- Groups 1 through 4 exist in the Endpoint 1.
- Group 1 and Group 4 exist in Endpoint 2.
- Account Template 1 is not associated to any endpoints.

The following table illustrates this setup:

Endpoint 1	Endpoint 2	Account Template 1
Group 1	Group 1	(none)
Group 2		
Group 3		
	Group 4	

Suppose you perform the following steps:

1. Associate Endpoint 1 with Account Template 1, creating the first endpoint association for Account Template 1.

Endpoint 1	Endpoint 2	Account Template 1
Group 1	Group 1	+ Endpoint 1 (succeeds)
Group 2		
Group 3		
	Group 4	

2. Select Group 1 and Group 2 for Account Template 1. The selection succeeds because these groups exist in Endpoint 1.

Endpoint 1	Endpoint 2	Account Template 1
Group 1	Group 1	Endpoint 1
Group 2		+ Group 1 (succeeds)
Group 3		+ Group 2 (succeeds)
	Group 4	

- Attempt to associate Endpoint 2 with Account Template 1. The attempt fails because one of the account template's selected groups, Group 2, does not exist in Endpoint 2.

Endpoint 1	Endpoint 2	Account Template 1
Group 1	Group 1	Endpoint1
Group 2		Group 1
Group 3		Group 2
	Group 4	+ Endpoint 2 (fails)

When the attempt fails, the Manager displays an error message similar to the following one:

Resources not found

Endpoint 2 is not necessarily required to contain the same groups as Endpoint 1. However, Endpoint 2 **must** contain all groups from Endpoint 1 that have already been selected for Account Template 1. In other words, for any new endpoint that you want to associate with an account template, all previously selected groups for the account template must exist in the new endpoint.

Sample Scenario Using Additional Groups

A new endpoint that you want to associate to an account template is permitted to contain *additional* groups; that is, groups that have not been selected for the account template. After you associate the new endpoint to the account template, you may optionally select any of these additional groups for the account template. If you do, these groups are **added** to the list of required groups that must exist in all new endpoints that you attempt to associate with the account template in the future.

The following scenario illustrates this principle. At the beginning of this scenario, the following are true:

- Groups 1 through 4 exist in Endpoint 1
- Groups 1 through 3 exist in Endpoint 2
- Account Template 1 has no associated endpoints

The following table illustrates this setup:

Endpoint 1	Endpoint 2	Account Template 1
Group 1	Group 1	(none)
Group 2	Group 2	

Group 3	Group 3
Group 4	

Suppose you perform the following steps:

1. Associate Endpoint 1 with Account Template 1, creating the first endpoint association for Account Template 1.

Endpoint 1	Endpoint 2	Account Template 1
Group 1	Group 1	+ Endpoint 1 (succeeds)
Group 2	Group 2	
Group 3	Group 3	
Group 4		

2. Select Group 1 and Group 2 for Account Template 1. The selection succeeds.

Endpoint 1	Endpoint 2	Account Template 1
Group 1	Group 1	Endpoint 1
Group 2	Group 2	+ Group 1 (succeeds)
Group 3	Group 3	+ Group 2 (succeeds)
Group 4		

Consequently, any endpoints that you attempt to associate with Account Template 1 in the future must contain Group 1 and Group 2; otherwise, the attempt will fail.

3. Associate Endpoint 2 with Account Template 1, creating the second endpoint association for Account Template 1. The association succeeds, because all groups already selected for Account Template 1 (Group 1 and Group 2) exist in Endpoint 2.

Endpoint 1	Endpoint 2	Account Template 1
Group 1	Group 1	Endpoint 1
Group 2	Group 2	Group 1
Group 3	Group 3	Group 2
Group 4		+ Endpoint 2 (succeeds)

4. Select Group 3 for Account Template 1. The selection succeeds.

Endpoint 1	Endpoint 2	Account Template 1
Group 1	Group 1	Endpoint 1
Group 2	Group 2	Group 1
Group 3	Group 3	Group 2
Group 4		Endpoint 2
		+ Group 3 (succeeds)

Consequently, any endpoints that you attempt to associate with Account Template 1 in the future must contain groups Group 1 through Group 3; otherwise, the association will fail.

Availability Requirements for Groups

On the Manager, the groups that you can select for an account template are displayed in the account template's list of available groups. Available groups must meet *both* of the following criteria:

- Are not already selected for the account template
- Exist in all endpoints that are associated with the account template

In the Sample Scenario Illustrating Additional Groups, Group 4 is an available group for Account Template 1 when Endpoint 1 is the only endpoint associated with the account template. However, when Endpoint 2 is associated with Account Template 1, Group 4 is no longer an available group for Account Template 1, because Group 4 does not exist in Endpoint 2.

Availability Requirements for Categories and Security Labels

The intersection principle described previously for groups also applies to categories and security labels. That is, when you associate a new endpoint to an account template, the new endpoint is not necessarily required to contain all the same categories and security labels as the previous endpoint or endpoints that have already been associated with the account template. However, the new endpoint *must* contain all categories and security labels that have already been selected for the associated account template.

Similarly, a new endpoint that you want to associate to an account template is permitted to contain additional categories or security labels that have not already been selected for the account template. After you associate the new endpoint to the account template, you may optionally select any such categories or security labels for the account template. If you do so, these categories or security labels are *added* to the list of required categories and security labels that must exist in all new endpoints that are associated with the account template in the future.

Finally, the availability requirements discussed previously for groups also apply to categories and security labels. That is, an available category or security label remains available to an account template until an endpoint that does not contain the category or security label is associated with the account template.

Removing ACC Endpoints from Account Templates

If an associated endpoint is removed from an account template, the list of selected items does not change. This is true even if the last endpoint is removed from an account template, but the list of available items is recomputed. For example, the connector attempts to compute the new list of available groups for the remaining endpoints that are still associated to the account template.

Password Synchronization

The Identity Manager password synchronization agent supports the interception of Windows password changes.

Reconfiguring the Password Synchronization Component

These are the steps to reconfigure an Identity Manager password synchronization component:

1. Install the Windows NT Connector after installing the Identity Manager Windows Connector (Windows NT or Active Directory).
2. Install Windows NT Remote Agent for a Windows NT system. Identity Manager Active Directory Services Connector directly manages Active Directory using LDAP.
3. Acquire the Windows directory to create an internal representation of the Windows system in Identity Manager.
Note: Do not explore and correlate the Windows accounts, because they are managed as CA Access Control accounts. Explore and correlate these as CA Access Control accounts.
4. Install the Identity Manager Password Synchronization agent. During the installation process, the Identity Manager Password Synchronization Configuration wizard guides you through the process to set the component as a Windows password interceptor.
5. Install the CA Access Control Connector to manage the CA Access Control accounts.
6. Install CA Access Control on the Windows system if it is not already installed.
7. Acquire the CA Access Control Directory to manage Identity Manager.
8. Explore and correlate CA Access Control accounts to global users in Identity Manager.
9. Revise the Identity Manager Password Synchronization Configuration File to reflect the changes from Windows to the eTrust Access Control Connector.

Architecture

The out-of-box configuration does not support intercepting CA Access Control password changes. However, because CA Access Control also manages Windows password changes, you can reconfigure the password synchronization component to propagate CA Access Control password changes.

When a Windows password changes, the password synchronization component intercepts the change and forwards it to Provisioning Server, which then propagates the change to other accounts belonging to the same global user.

You can reconfigure the CA Admin password synchronization component for synchronizing CA Access Control passwords, using the same Windows password interception:

Thus, users make changes to their passwords using CA Access Control tools. The password changes affect the CA Access Control environment and the native Windows environment. When you make password changes to the Windows environment, the Identity Manager Provisioning password synchronization component intercepts the password changes.

The reconfiguration of the Provisioning password synchronization component sends the password changes to the Provisioning Server, indicating that the password changes are from CA Access Control, instead of a native Windows system.

Provisioning Server discovers the global user associated with the CA Access Control accounts that originate the password changes, and then propagates password changes to other accounts belonging to the same global users.

Comparing PMDB to Local Database

Identity Manager manages CA Access Control identities in an identity store that can be either a CA Access Control PMDB or a local database. Since Identity Manager and CA Access Control both manage users and passwords, it is an architecture decision as to which users are managed by Identity Manager and which by CA Access Control. A general guideline is that Identity Manager manages a PMDB and the PMDB handles the propagation of all its subscribers.

Changing Passwords Using Windows Tools

Besides password changes from CA Access Control tools, users can also change their passwords using Windows native utilities. The Identity Manager password synchronization component intercepts the password change and propagates it to other Identity Manager managed accounts associated with the same global users. However, the CA Access Control managed accounts require a separate mechanism to synchronize passwords initiated from the native Windows environment. CA Access Control also provides a password intercept mechanism for this purpose. We recommend the following guidelines:

- Disable the password quality control of the Identity Manager password synchronization agent.
- Let the CA Access password synchronization component manage the password quality control.

Mapping Configuration from Windows

The following two configuration files are an example of a conversion from Windows to CA Access Control. The information that you should modify is in *italics*.

```
;
; This configuration file is used by the Identity Manager Windows Password
; Synchronization Facility.
;
[Server]
host=<Provisioning Server host>
port=20389
use_tls=yes
admin_suffix=dc=<domain suffix>
admin=etaadmin
password=k4tpGDJ8Djg=

;; Identity Manager domain information
;;
;; If the search fails, and the container dn is specified, the account dn is
;; constructed as "<acct_attribute_name>=<native acct name>,<container dn>".
;; The container DN should contain "dc=eta".
;;
[EtaDomain]
domain=<domain name>
etrust_suffix="dc=eta"
domain_suffix=dc=<domain suffix>
Namespace=Windows NT
directory=chete03

directory_dn=eTN16Directory/Name=chete03,eTNamespaceName=Windows NT,dc=129-731-CHOPIN,dc=eta
container_dn=eTN16Account/ContainerName=Accounts,eTN16Directory/Name=chete03,eTNamespaceName=
Windows NT,dc=129-731-CHOPIN,dc=eta
acct_attribute_name=eTN16Account/Name
acct_object_class=eTN16Account
```

```
;
; This configuration file is used by the Identity Manager Password Synchronization
; Facility for CA Access Control
;

[Server]
host=<Provisioning Server host>
port=20389
use_tls=yes
admin_suffix=dc=<domain suffix>
admin=etaadmin
password=k4tpGDJ8Djg=

;; Identity Manager domain information
;;
;; In order to find the account DN, a search operation will be performed, using
;; the directory dn as the search base, and objectClass and account name as the
;; search filter.
;;
;; If the search fails, and the container dn is specified, the account dn is
;; constructed as "<acct_attribute_name>=<native acct name>,<container dn>".
;; The container DN should contain "dc=eta".
;;
;; Currently, domain, etrust_suffix, Endpoint Type, and directory keys are not used,
;; because all DNs are hardcoded. The future enhancement is to provide "domain",
;; "Endpoint Type" and, "directory name". Identity Manager will find out the DNs based on
;; the supplied information.

[EtaDomain]
domain=<domain name>
etrust_suffix="dc=eta"
domain_suffix=dc=<domain suffix>
Namespace=Windows NT
directory=pmdb
;; Directory name of the CA Access Control system
```

```
directory_dn=eTACCDirectoryName=pmdb,eTNamespaceName=Access Control,dc=129-731-CHOPIN,dc=eta
container_dn=eTACCAccountContainerName=Accounts,eTACCDirectoryName=pmdb,eTNamespaceName=Access Control,dc=129-731-CHOPIN,dc=eta
acct_attribute_name=eTACCAccountName
acct_object_class=eTACCAccount
```

```
:: Password Profile Configuration
:: profile_enabled = [yes|y|no|n] --> Unknown values default to "no"
:: profile_dn = "<the DN of the password profile>"
[PasswordProfile]
profile_enabled = no
profile_dn = eTPasswordProfileName=Password Profile,eTPasswordProfileContainerName=Password Profile,eTNamespaceName=CommonObjects,dc=129-731-CHOPIN,dc=eta
```


CA SSO Connector for Advanced Policy Server

The CA SSO Connector for Advanced Policy Server (PLS) is a Endpoint Type connector for Identity Manager that lets you administer CA Single Sign-On, version 7.0 or higher. The CA SSO Connector for Advanced Policy Server provides a single point for all user administration by letting you do the following:

- Manage Endpoint, Account, Group, Terminal, Authentication Host, Application, Application Group and Account Template object classes.
- Create, modify, or delete an account or group in a user data store.
- Add accounts to a group, or remove them.
- Authorize an account or group to access selected applications and application groups.
- Administer passwords for the SSO and LDAP authentication methods.
- Administer login information for applications.
- Administer various pre-defined account and group properties, such as expiration date, suspension date, and resumption date.
- Administer date and time restrictions for Account, Account Template, and Terminal objects.
- Specify user attribute values for accounts in a user data store.
- Create, modify, or delete Terminal or Authentication Host objects in SSO endpoints
- Authorize users and groups to access Terminal or Authentication Host objects

Note: Terminal and Authentication Host classes are only available to be managed in the PLS Connector when the SSO servers are v8.0 and higher.

This connector is managed using the Connector and C++ Server installation process.

Note: For more information and requirements, see *Connector and C++ Connector Server Installation* (see page 23).

Configuring the CA Single Sign-On Server

Follow the steps below to configure your CA Single Sign-On server for Identity Manager.

1. Start the selang command interpreter.
2. Create the system administrator's account on the CA Single Sign-On server if it does not already exist.
3. Enable the administrator's account to connect from the Provisioning Server.

Step 1. Start the Selang Command Interpreter

To begin the configuration process, start the selang command interpreter.

Step 2. Create the System Administrator's Account

Create the CA Single Sign-On administrator's account on the CA Single Sign-On server. Ensure that you add the admin and auditor keywords to the command. This gives the administrator administrative privileges. In selang, enter the following command:

```
nu administrator_name password(administrator_password) admin auditor
```

administrator_name

Is the user ID that the administrator uses to log on to the eTrust Single Sign-On Server.

administrator_password

Is the administrator's password for the user ID.

Note: We recommend that you do **not** use a user ID named "Administrator" to define a CA Single Sign-On endpoint for Windows 2000.

Enter the following command to add *administrator_name* to the predefined group *_ps-adms*.

```
join administrator_name group(_ps-adms)
```

Enter the following commands to ensure the administrator's account is created in the native operating system with the same password.

```
env(native)
```

```
eu administrator_name password(administrator_password)
```

```
env(seos)
```

Step 3. Enable the Administrator's Account

Enter the following command to enable the CA Access Control and CA Single Sign-On authentication methods for the administrator.

```
eu administrator_name auth_type(method5, method20)
```

Enter the following command to set the CA SSO password for the administrator's account to the same password you specified in Step 1.

```
el administrator_name appl(__SSO__) currpwd(administrator_password)
```

Give the administrator access to the CA Single Sign-On server by issuing the following command.

```
auth terminal server_name uid(administrator_name) acc(access_type)
```

server_name

Is the machine name of the eTrust Single Sign-On Server.

administrator_name

Is the administrator's account.

access_type

Is the access that the administrator needs. Read and write access is necessary. The keywords for access_type are READ, WRITE.

Using Failover

When using the PLS Connector to connect to a policy server farm, you can set up a failover system that automatically switches from a failed server to a running server to let you keep working without interruption. For large sites that use a policy server farm, failover can provide reliable and rapid service.

When discovering the SSO endpoint, the policy server that is to be the primary policy server must be provided. After the discovery, the Fail-Over property page in the Endpoint Property Sheet shows the policy server that was specified. You can then add more policy servers to the list. Once the policy servers have been added, they can be edited or even removed as needed.

The PLS Connector always tries to connect the first policy server in the list, so the order of the policy servers in the list is significant. If the connection fails to the first policy server then the PLS Connector tries connecting to the second policy server and so on. Once a connection is successfully made, PLS continues to work with the server. Every 60 seconds, PLS checks whether failed servers are available again.

Note: When changing the policy server list in the Fail-Over tab, the primary server, (for example, the first entry in the list) must be responsive for the changes to be accepted and applied.

Enable Application Password Propagation

Currently, in an SSO endpoint, every SSO user record contains a login application and every login application record contains a username and password. This username and password does not have to be the same as the SSO username. For example:

```
SSOuser1 Username=Doe Password=Doe
```

```
TelnetApp1 Username=Doe1 Password=Doe1 (Unix Host Srv1)
```

```
TelnetApp2 Username=Doe2 Password=Doe2 (Unix Host Srv2)
```

SSO has password synchronization. If you (or SSO) change the password from TelnetApp1, SSO also changes the password for TelnetApp2.

If you put Identity Manager into this equation, Admin is able to do password synchronization and has an SSO Connector and a UNIX Connector. You now have the following scenario:

```
Global User=Doe
```

```
SSO User=Doe
```

```
Inside SSO TelnetApp1 username=Doe, TelnetApp2 username=Doe
```

```
Unix User on Srv1=Doe
```

```
Unix User on Srv2=Doe
```

If you change the password for the global user Doe and propagate the password to all of the global user accounts, the password will change on the following Endpoint Types: SSO, Unix Srv1 and Unix Srv2. However, the password in the loginapplications (TelnetApp1, TelnetApp2, and so forth) for the SSO user will not be changed and those using SSO cannot use SSO to log into their applications anymore because the password stored in their loginappl record is out of sync.

To solve this problem, a master application, for example, eTrustIAM, can be defined and TelnetApp1 and TelnetApp2 can be set to use eTrustIAM as the master application. The PLS Connector can then update the password of the master application eTrustIAM when it receives the password propagation request caused by the Identity Manager global user password change. As a result, the Policy Server updates the passwords for TelnetApp1 and TelnetApp2. Because the UNIX Connector updates the passwords for the user in both Unix Srv1 and Unix Srv2, and the PLS Connector updates the SSO password if the user uses the SSO authentication method, the passwords in all levels are in sync.

If you are using an older Policy Server version that does not have the eTrustIAM master application defined automatically after installation, do the following to use this feature:

- Using Policy Manager, create a master application "eTrustIAM" in the Policy Server and set _SSO_ as the master application.
- Like the _SSO_ application, the eTrustIAM application should be available for every user, so set the default access rights to EXECUTE. And, since the eTrustIAM application should not be shown in the SSO client, the access rights must also be set to HIDDEN.
- Set eTrustIAM as the master application for all applications where you want password propagation.

If you want to integrate admin applications (Provisioning Manager, IA Manager, Self Service, SPML Manager) with SSO, do the following to start these Admin applications through the SSO client:

1. Using Policy Manager, create SSO applications for each Admin application (Provisioning Manager, IA Manager, and so forth).
2. Set eTrustIAM as the master application for these SSO applications.
3. Create TCL scripts for each Admin application, (These are used to start the applications through SSO.), and put these TCL scripts in the following directory:

eTrust Policy Server\Scripts

PLS Support for FIPS and IPv6

For release r12 of Identity Manager, the PLS Connector does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire a CA Single Sign-On Server

You must acquire the CA Single Sign-On server before you can administer it with Identity Manager. When acquiring an CA Single Sign-On server, perform the following steps from the Endpoint Type task view:

1. Register the server as an endpoint in Identity Manager.

Use the PLS Endpoint property sheet to register an CA Single Sign-On server. During the registration process, Identity Manager identifies the CA Single Sign-On server you want to administer and gathers information about it.

Note: Ping the node name from the Provisioning Server. If the ping is successful, then you know that Identity Manager will find the PLS node.

2. Explore the objects that exist in the endpoint.

After registering the server in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all accounts and groups in the SSO server.. You can correlate the accounts with global users at this time or you can correlate them later.

3. Correlate the explored accounts with global users.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the account name with each existing global user name. If a match is found, Identity Manager associates the PLS account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the full name with each existing global user's full name. If a match is found, Identity Manager associates the PLS account with the global user. If a match is not found, Identity Manager performs the next step.
- c. If the Create Global Users as Needed button is checked, Identity Manager creates a new global user and then associates the PLS account with the global user. If the Create Global Users as Needed button is unchecked, Identity Manager performs the next step.
- d. Identity Manager associates the PLS account with the [default user] object.

Roles and Policies

In addition to defining privileges for users, you can also set login information for applications associated with account templates. Once this information is set, users have access to the applications if they provide the correct login information.

The PLS Default Policy, provided with the CA SSO Connector for Advanced Policy Server, gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

PLS Control Applications

Identity Manager allows you to view certain basic properties of an application on the PLS Application property sheet. You are not allowed to add, delete, or modify an application.

PLS Control Application Groups

Identity Manager allows you to view certain basic properties of an application group on the PLS Application Group property sheet. You are not allowed to add, delete, or modify an application group.

PLS Terminal

Identity Manager allows you to assign accounts and groups to access the current terminal objects. Use the PLS Terminal property sheet to set the profile, day/time restrictions, and account and group access to terminals.

PLS Authorization Hosts

Identity Manager allows you to assign accounts and groups to access the current authorization host objects. Use the PLS Authhost property sheet to set the profile, and account and group access to authorization hosts.

Kerberos Connector

You can use the Kerberos Connector to administer Kerberos principals and Kerberos password policies on Solaris servers. The Kerberos Connector provides a single point for all user administration by letting you do the following:

- Register endpoints, explore them for objects to manage, and correlate their accounts with global users.
- Create and manage Kerberos principals using Kerberos-specific account templates.
- Change principal passwords and principals activations in one place.
- Synchronize global users with their provisioning roles or synchronize global users' accounts with their account templates.
- Assign a Kerberos account template to each of your Kerberos endpoints.
- Create accounts with the minimum level of security needed to access a Kerberos endpoint using the default endpoint type account template.
- Create, edit and delete password policies.

This connector is installed using the Connector and Java Connector Server installation process. For more information and requirements, click [here](#) (see page 26).

Kerberos Connector Limitations

When you use the Kerberos Connector, you should be aware of the following limitations:

- The connector is based on the Solaris implementation of Kerberos version 5.
- The Windows Java CS does not support Kerberos, therefore you can only use the Solaris Java CS to manage Kerberos.
- The connector can be installed with both the Windows and Solaris Server version of the Provisioning Server and manage the connector using the Java CS.
- The connector does not currently support keytab management of kadmin.
- The connector will generate an error if you use any characters other than non-control ASCII characters in principal names, password policy names, and passwords, as Kerberos accepts only non-control ASCII characters.

Unsupported kadmin Options

The Kerberos Connector is integrated with the kadmin interface to let you provision KRB principal and password policies; however you should be aware of the following:

- The connector does not support kadmin.local. Thus, options that are available only through kadmin.local are not supported.
- The keytab management (ktadd, ktremove) and administration privileges (ACL) aspects of kadmin are currently not supported.
- The -c option of kadmin is not supported since kadmin requests new service tickets from the KDC.
- The -kvno and -keepold password related options are not currently supported.

Naming Limitations

Because the Kerberos Connector relies on kadmin to communicate with the Kerberos server, kadmin limitations are limitations of the connector.

Principal names, passwords, and password policy names can include any printable ASCII character. However, the following kadmin limitations are applicable, as described in the following sections:

Principal Naming Limitations

- The double quote character (") is used by kadmin only as a quoting character. kadmin does not accept this character as part of a principal name. As a result, the connector will reject principal names containing this character.
- The @ character delimits principal names from realm names, and as such cannot be part of a principal or realm name.

The connector and kadmin will accept an account name in the form name@realm, but if realm is not the same as the realm specified by the endpoint, kadmin will treat this as a cross-realm principal. As a result, even though an entry for this principal will be included in the Kerberos database, unless you configure cross-realm authentication properly, this principal may not be able to authenticate to any KDC. If an account name with more than one @ character is used, kadmin will display a *Malformed name* error.

- The backslash character (\) is not properly supported. There are cases where, in a sequence of one or more backslash characters, one character may be dropped depending on the character immediately succeeding the backslash. The connector will not prevent the creation of principal names with backslash characters, but we recommend that you use the backslash character with caution.
- The hash (#) character can be used to start a principal name in kadmin. However, due to DN syntax limitations, the hash at the beginning of a principal name will be escaped with a backslash character (\). Within The Provisioning Manager, this escape character will always be present, but in the Kerberos system, the principal name will not have the escape character.

Password Policy Naming Limitations

- kadmin uses the double quote character (") only as a quoting character. kadmin does not accept this character as part of a password policy name. Thus this connector will reject password policy names containing this character.
- kadmin will accept a password policy name that starts with a hash (#). However, due to DN syntax limitations, the hash at the beginning of a name will be escaped with a backslash character. Within the Provisioning Manager and the Kerberos system, this escape character will always be present.

Password Limitations

The double quote character (") is used by kadmin only as a quoting character. kadmin does not accept this character as part of a password.

Kerberos Installation and Deployment

This section provides information about installing and deploying the Kerberos Connector, including firewall configuration and keytab and cross-realm paths setup.

Installation Prerequisites

The Kerberos server (KDC) must be Sun's Kerberos V5 implementation, and installed on Solaris 10. You must install the following packages.

- SUNWkdcr (Kerberos V5 KDC - root)
- SUNWkdcd (Kerberos V5 Master KDC - user)
- SUNWkrbr (Kerberos version 5 support - Root)
- SUNWkrbu (Kerberos version 5 support - Usr)

The Java CS host must have the SUNWkdcd (Kerberos V5 Master KDC - user) packages installed, and you must configure them as a Kerberos client (that is, you must configure `krb5.conf`).

Kerberos Connector Installation Options

You can only deploy the KRB connector, and thus Java CS, on a Solaris host that satisfies either of the following requirements:

- The host is a member of the realm managed by the target KDC.
- The host is a member of a realm that can reach the target KDC through cross-realm configuration.

The connector communicates with the KDC directly through the `kadmin` interface.

Install and Deploy the Connector

The installation package contains the components required to install the Kerberos Connector.

Note: If you have any standalone Provisioning Manager installations that require access to Kerberos, re-install the Provisioning Manager to add the Kerberos Connector.

Note: For information on installing the Java CS, see the *Java CS Implementation Guide*.

To install and deploy the connector

1. (Windows and Solaris) Run the Provisioning Server install, and add the Kerberos Connector when prompted.

The server and directory components are updated with the schema for the Kerberos Connector.

2. Install the Identity Manager – Connector Server, and register it to the domain during the installation.

The connector is deployed and tells the server where to send Kerberos requests. When complete, you can start to acquire Kerberos endpoints.

Note: For more information on setting up hosts, keytabs, and configuration files on a computer that hosts the Java CS where it is not the same computer as the KRB endpoint, see [How to Set Up the Java CS Host to be a Member of the Target Realm](#) (see page 352).

Firewall Configuration

There are three main Kerberos components:

- Kerberos client applications (for example, kinit, telnet, pop)
- Server applications (for example, telnetd, popper)
- Kerberos KDC

Different types of traffic go between each pair of components your firewall is between. Depending on the pair of components your firewall is between, you will need to allow different types of traffic through your firewall.

Note: The notation `xxxx/udp` or `xxxx/tcp` used in the following table refers to an ephemeral port number (that is, >1024). This refers to a return port that the system assigns. The only assumption you can make about the port number is that it will be greater than 1024.

You may need to configure your firewall to allow traffic between a client program and the KDC on the following ports and protocols:

Client Application	To KDC	Return Traffic
Ticket requests (for example, kinit)	88/udp	xxxx/udp
Kerberos 5-to-4 ticket conversion	4444/udp	xxxx/udp
Changing password (kpasswd under Unix)	749/tcp	xxxx/tcp
Changing password (under Windows, old interface)	464/tcp	xxxx/tcp
Changing password (under Windows, new interface)	464/udp	xxxx/udp
Running kadmin (also requires initial ticket, 88/udp)	749/tcp	xxxx/tcp

You may need to configure your firewall to allow traffic between an application server and the KDC on the following ports/protocols:

Application Server	To KDC	Return Traffic
Initial ticket request (for example, kinit)	88/udp	xxxx/udp
Kerberos 5-to-4 ticket conversion	4444/udp	xxxx/udp

You may need to configure your firewall to allow traffic between a client program and an application server on the following ports/protocols:

Application Program Server	To Server	To ClientTraffic
rlogin/rlogind (w/o encryption)	543/tcp	xxxx/tcp
rlogin/rlogind (w/encryption)	2105/tcp	xxxx/tcp
rsh/rshd	544/tcp	xxxx/tcp
pop/popper	1109/tcp	xxxx/tcp
telnet/telnetd	Same as non-kerberos telnet/telnetd	
ftp/ftpd	Same as non-kerberos ftp/ftpd	

Keytab and Cross-realm Paths Setup

Depending upon the Administrative principal's authentication options, and whether the host where the Java CS is deployed is in the realm specified for the endpoint, you may need to set up keytabs and cross-realm paths on the Java CS host.

Note: For more information, see the *Solaris 10 System Administration Guide: Security Services*.

Authentication Methods

You can set up authentication using several different methods:

- [The Java CS host principal](#) (see page 354)
- [The Java CS host principal and a custom keytab](#) (see page 355)
- [A principal other than the Java CS host principal and the default keytab](#) (see page 356)
- [A principal other than Java CS host principal and a custom keytab](#) (see page 358)
- [Principal and password authentication](#) (see page 359)

How to Set Up the Java CS Host to be a Member of the Target Realm

The following section shows an example you how you can set up the host for use with the Java CS where the host will be a member of the target realm.

1. Copy the file `/etc/krb5/krb5.conf` from the key distribution center to the Java CS host. Ensure that:
 - The `default_realm` entry in the `libdefaults` section points to the target realm.
 - The KDC entry in the appropriate realm relation in the `realms` section points to the target KDC.
 - The `domain_realm` section has the correct mapping of the Java CS host to the target realm.
2. Modify the logging and `appdefaults` sections in the `/etc/krb5/krb5.conf` file as required.
3. On the KDC, create a host principal for the Java CS host and give it a random key. For example, use the following command in `kadmin` to create a new host principal:

```
add_principal -randkey host/jcs_host.ca.com
```

4. Set up authentication to use one of the following:
 - [The Java CS host principal](#) (see page 354)
 - [The Java CS host principal and a custom keytab](#) (see page 355)
 - [A principal other than the Java CS host principal and the default keytab](#) (see page 356)
 - [A principal other than Java CS host principal and a custom keytab](#) (see page 358)
 - [Principal and password authentication](#) (see page 359)

Note: For information on using the host for other Kerberos-related purposes, such as hosting other Kerberos applications or services, see the relevant sections on `kadmin`, `ktutil` and `krb5.conf` in the *Solaris 10 System Administration Guide: Security Services*.

How you set Up Keytab Authentication Using the Host Principal

To set up keytab authentication using the host principal, do one of the following:

- If the default keytab file exists, [add the entries into a temporary keytab](#). (see page 354)
- If the default keytab file does not exist, [create a new keytab file](#). (see page 355)

Set Up Keytab Authentication Using the Java CS Host Principal if Keytab File Does Not Exist

To set up keytab authentication using the host principal if the default keytab file does not exist, you need to create a new keytab file.

To specify keytab authentication using the Java CS host principal if keytab file does not exist

1. Enter the following command in kadmin:

```
kadmin: ktadd -k temp_keytab jcs-host-principal
```

Kerberos adds the entries into a temporary keytab.

Note: This creates a new randomized password for the host principal, thus any entries for the host principal in any existing keytab file are no longer valid.

2. In the KDC, modify the kadm5.acl file using a text editor.

The connector adds the necessary privileges to the host principal.

Note: Use * to specify all privileges.

3. In the Provisioning Manager, on the Endpoint Property sheet, click the Properties tab.

The Properties tab is displayed.

4. Select the Keytab option.
5. Leave the Keytab and Principal fields blank.
6. Click Apply.

The Kerberos Connector uses the Java CS host principal for keytab authentication.

Set Up Keytab Authentication Using the Java CS Host Principal if Keytab File Exists

To set up keytab authentication using the host principal if the keytab file exists, you need to add keytab entries for the Java CS host principal to the default `/etc/krb5/krb5.keytab` file.

To specify keytab authentication using the Java CS host principal if keytab file exists

1. Enter the following commands in ktutil:

```
ktutil: read_kt temp_keytab
```

```
ktutil: read_kt /etc/krb5/krb5.keytab
```

Kerberos reads both keytabs.

2. Enter the following command in ktutil:

```
ktutil: write_kt /etc/krb5/krb5.keytab
```

Note: Make sure that the entries for the host principal are the same, and are the latest key version number.

Kerberos writes the entries to the default keytab file and the temporary keytab file is merged into the default keytab.

3. In the KDC, modify the `kadm5.acl` file using a text editor.

The connector adds the necessary privileges to the host principal.

Note: Use `*` to specify all privileges.

4. In the Provisioning Manager, on the Endpoint Property sheet, click the Properties tab.

The Properties tab is displayed.

5. Select the Keytab option.
6. Leave the Keytab and Principal fields blank.
7. Click Apply.

The Kerberos Connector uses the Java CS host principal for keytab authentication.

Set Up Keytab Authentication Using a Custom Keytab and the Java CS Host Principal

To set up keytab authentication using a custom keytab file rather than the default keytab file and the Java CS host principal, you can add keytab entries for the Java CS host principal to your custom keytab file.

To set up keytab authentication using a custom keytab and the Java CS host principal

1. If the keytab file you want to use does not exist, use the following command to add entries to your custom keytab file.

```
kadmin: ktadd -k keytab jcs-host-principal
```

Note: This creates a new randomized password for the host principal, therefore any entries for the host principal in any existing keytab file are no longer valid.

2. If the keytab file exists, do the following:
 - a. Enter the following command in kadmin to add entries into a temporary keytab:

```
kadmin: ktadd -k temp_keytab jcs-host-principal
```

Note: This creates a new randomized password for the host principal, thus any entries for the host principal in any existing keytab file are no longer valid.

- b. Enter the following command in ktutil to read both keytabs:

```
ktutil: read_kt temp_keytab
```

- c. Enter the following command in ktutil to write it to the keytab file you want to use:

```
ktutil: write_kt keytab
```

The temporary keytab file is merged into the keytab file you want to use.

Note: Make sure that the entries for the host principal are the same, and are the latest key version number.

3. In the KDC, modify `kadm5.acl` using a text editor to add necessary privileges to the host principal.

Note: Use `*` to specify all privileges.

4. In the Provisioning Manager, on the Endpoint Property sheet, click the Properties tab.
5. Specify the keytab file you want to use, but leave the Principal field blank.
6. Click Apply.

The Kerberos Connector uses the keytab you specified for authentication.

Set Up Keytab Authentication Using the Default Keytab and a Principal Other than the Java CS Host Principal

To specify keytab authentication using the default keytab and a principal other than the Java CS host principal, you can add keytab entries for the principal to the keytab file.

To specify keytab authentication using the default keytab and a principal other than the Java CS host principal

1. If the principal has a random password and the default keytab file does not exist, enter the following command in kadmin to add entries to the file:

```
kadmin: ktadd principal
```

Note: This creates a new randomized password for the target principal, therefore any entries for the target principal in any existing keytab file are no longer valid.

2. If the principal has a random password and the keytab file exists, do the following:

- a. Enter the following command in kadmin to add entries into a temporary keytab:

```
kadmin: ktadd -k temp_keytab principal
```

Note: This creates a new randomized password for the target principal, thus any entries for the target principal in any existing keytab file are no longer valid.

- b. Enter the following commands in ktutil to read both keytabs:

```
ktutil: read_kt temp_keytab
```

```
ktutil: read_kt /etc/krb5/krb5.keytab
```

- c. Enter the following command in ktutil to write the entries to the target keytab file you want to use.

```
ktutil: write_kt /etc/krb5/krb5.keytab
```

The temporary keytab file is merged into the target keytab file you want to use.

Note: Make sure that the entries for the target principal are the same, and are the latest key version number.

3. If the principal has a specific password, do the following:

- a. Enter the following command in ktutil:

```
ktutil: read_kt /etc/krb5/krb5.keytab
```

- b. Enter the following command in ktutil:

```
ktutil: addent -password -p principal -k kvno -e encype
```

- c. Repeat Step b for all enttypes.

ktutil adds the entries to the default keytab file.

Note: Ensure you add all keys for the principal, and that all resulting entries for the principal are the same and latest key version number.

4. Enter the following command in ktutil to verify that the list contains all required keys:

```
ktutil: list
```

5. Enter the following command in ktutil to write the entries to the keytab file:

```
ktutil: write_kt /etc/krb5/krb5.keytab
```

6. In the KDC, modify kadm5.acl using a text editor to add necessary privileges to the target principal.

Note: Use * to specify all privileges.

7. In the Provisioning Manager, on the Endpoint Property sheet, click the Properties tab.
8. Specify the principal you want to use, but leave the Keytab field blank.
9. Click Apply.

The Kerberos Connector uses the keytab you specified for authentication.

Set Up Keytab Authentication Using a Custom Keytab and a Principal Other than the Java CS Host Principal

To specify keytab authentication using a keytab file other than the default keytab and a principal other than the Java CS host principal, you can add entries for the desired principal to the desired keytab file.

To set up keytab authentication using a custom keytab and a principal other than the Java CS host principal

1. If the principal has a random password and the keytab file you want to use does not exist, use the following command to add entries:

```
kadmin: ktadd -k keytab principal
```

Note: This creates a new randomized password for the target principal, therefore any entries for the target principal in any existing keytab file are no longer valid.

2. If the principal has a random password and the keytab file exists, do the following:
 - a. Enter the following command in ktutil to add entries into a temporary keytab:

```
kadmin: ktadd -k temp_keytab principal
```

Note: This creates a new randomized password for the desired principal, thus any entries for the desired principal in any existing keytab file are no longer valid.

- b. Enter the following commands in ktutil to read both keytabs:

```
ktutil: read_kt keytab
```

```
ktutil: read_kt temp_keytab
```

- c. Enter the following command in ktutil to write the entries to the keytab file you want to use.

```
ktutil: write_kt keytab
```

The temporary keytab file is merged into the target keytab file you want to use.

- Note:** Make sure that the entries for the desired principal are the same, and are the latest key version number.

3. If the principal has a specific password, do the following:

- a. Enter the following command in ktutil:

```
ktutil: read_kt /etc/krb5/krb5.keytab
```

- b. Enter the following command in ktutil:

```
ktutil: addent -password -p principal -k kvno -e enctype
```

- c. Repeat Step b for all enttypes.

ktutil adds the entries to the keytab file you want to use.

Note: Ensure you add all keys for the principal, and that all resulting entries for the principal are the same and latest key version number.

4. Enter the following command in ktutil to verify that the list contains all required keys:

```
ktutil: list
```

5. Enter the following command in ktutil to write the entries to the keytab file:

```
ktutil: write_kt /etc/krb5/krb5.keytab
```

6. In the KDC, modify kadm5.acl using a text editor to add necessary privileges to the target principal.

Note: Use * to specify all privileges.

7. In the Provisioning Manager, on the Endpoint Property sheet, click the Properties tab.
8. Specify the principal and keytab you want to use.
9. Click Apply.

The Kerberos Connector uses the keytab you specified for authentication.

Set Up Principal and Password Authentication

You can specify authentication using principal and password authentication.

To set up principal and password authentication

1. In the KDC, modify kadm5.acl to add necessary privileges to the target principal.

Note: Use * to specify all privileges.

2. In the Provisioning Manager, on the Endpoint Property sheet, click the Properties tab.
3. Specify the principal and keytab you want to use.
4. Click Apply.

Kerberos uses the principal and password for authentication.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Tools for Managing Data in Endpoints

You can manage the accounts on your directories using any of the client interfaces. Each of these interfaces offers unique functionality:

- **Provisioning Manager**—Lets you perform all administrative tasks. It is the most commonly used interface that all administrators can access.
- **Batch Utility**—Lets you perform repetitive and time-consuming tasks offline through a command line interface.

KRB Etautil Conventions

Use the following Kerberos conventions in your etautil commands:

- The endpoint type name (eTNamespaceName) is KRB
- The endpoint type prefix is KRB. Therefore, the Kerberos class names are:
 - - eTKRBDirectory for an endpoint
 - - eTKRBAccountContainer for an account container
 - - eTKRBAccount for an account
 - - eTKRBPASSWORDPolicyContainer for a password policy container
 - - eTKRBPASSWORDPolicy for a password policy
 - - eTKRBPOLICYContainer for an account template container
 - - eTKRBPOLICY for an account template

Program Exits (Common or Native)

Program Exits let you write software that executes during certain actions that the Provisioning Manager carries out. Program exits extend the framework of the Provisioning Manager and allow for additional functionality that can change or augment the standard Provisioning Manager behaviors. Of the two types of exits, the Kerberos Connector supports *Native Exits*.

Native exits are program exits executed from within the managed endpoint types. Program exits let you reference custom code from within the Provisioning Manager process flow.

Information about the Kerberos exit program is entered on the Kerberos Program Exit property sheet.

Note: For more detailed information about how to write program exits, see the *Programming Guide for Provisioning* for Common Exits.

How You Acquire and Manage Kerberos Endpoints

You must acquire the Kerberos endpoint before you can administer it with the Provisioning Manager. When acquiring a Kerberos endpoint, perform the following steps from the Endpoint task view:

1. Acquire the machine as an endpoint in the Provisioning Manager.

Note: There are two ways to authenticate to the endpoint: use a keytab or use a principal and a password. To specify the authentication method, complete the fields on the [Properties Tab on the KRB Endpoint Property Sheet](#) (see page 382).

2. Explore the objects that exist in the endpoint.

After registering the machine in the Provisioning Manager, you can explore its contents. The exploration process finds all Kerberos objects. You can correlate the principals with global users at this time, or you can wait to correlate them.

3. Correlate the explored principals to global users. You can choose to:

- Use existing global users. Do this when there are already global users in the Provisioning Manager and you want to connect the existing global users to the Kerberos principals.
- Create global users as needed. Do this when there are no global users and you want to populate the Provisioning Manager from the Kerberos principals.

When you correlate principals, the Provisioning Manager creates or links the principals on an endpoint with global users, as follows:

- The Provisioning Manager attempts to match the Kerberos principal name with each existing global user name. If a match is found, the Provisioning Manager associates the Kerberos principal with the global user. If a match is not found, the Provisioning Manager performs the next step.
- The Provisioning Manager attempts to match the Kerberos principal name with each existing global user's full name. If a match is found, the Provisioning Manager associates the Kerberos principal with the global user. If a match is not found, The Provisioning Manager performs the next step.
- The Provisioning Manager associates the Kerberos principal with the [default user] object or a new global user is created depending on your choice.

More Information:

[Acquire a New Endpoint](#) (see page 363)

[Explore and Correlate Principals](#) (see page 366)

[View All Principals](#) (see page 366)

Acquire a New Endpoint

You must acquire and register a Kerberos endpoint before you can administer it with the Provisioning Manager.

To acquire a new endpoint

1. In the Provisioning Manager, click the Endpoints button.
2. In the Object Type list, select KRB Policy, then click New.
The KRB Account Template dialog appears.
The KRB Directory dialog appears.
3. Complete the fields on the KRB Directory tab, then click OK.
The parameters you need to acquire and register a KRB Directory are specified.
4. Complete the fields on the Properties tab, and then click OK.
The Kerberos Server, Realm and the credentials used for the connection are specified.
5. Complete the fields on the Endpoint Settings tab.
The various settings that apply to controlling endpoints, such as password propagation and synchronization are specified.
6. Complete the fields on the Program Exits Reference tab.
Program exits are viewed added edited or removed as specified.
7. Complete the fields on the Custom Settings tab.
The supported encryption types and salt pairs for this endpoint are specified.
8. Complete the fields on the Attribute Mapping tab.
The default attribute mapping defined in the schema file for the endpoint type are specified.
9. Complete the fields on the Logging tab.
The logging settings for the new endpoint are specified.
10. Click Apply.

More information:

[KRB Directory Tab](#) (see page 380)

[Properties Tab](#) (see page 382)

[Custom Settings Tab](#) (see page 384)

[KRB Endpoint Property Sheet](#) (see page 379)

Modify an Endpoint

You can modify the parameters of an already registered KRB endpoint, such as the default account template used and the authentication mode used.

Note: If you modify any of the connection-related fields (for example, kerberos server, port, realm, security credentials), the connector will, as when acquiring a new endpoint, run a kadmin command to validate the changed values.

To modify an endpoint

1. In the Provisioning Manager, acquire the endpoint you want to modify.
2. Explore and correlate the endpoint you want to modify.
3. In the EndpointName column on the leftmost side of the Provisioning Manager, double-click the endpoint you want to modify.

The KRB Endpoint Property Sheet dialog appears.

4. Modify the fields on the KRB Endpoint tab as required, and then click OK.

Note: You can only change comments and the default account template. You cannot change the name of the endpoint.

5. Modify the fields on the Properties tab as required, and then click OK.

You have specified the Kerberos Server, Realm and the credentials used for the connection.

6. Complete the fields on the Endpoint Settings tab.

The various settings that apply to controlling endpoints, such as password propagation and synchronization are specified.

7. Complete the fields on the Program Exits Reference tab.

Program exits are viewed added edited or removed as specified..

8. Complete the fields on the Custom Settings tab.

The supported encryption types and salt pairs for this endpoint are specified.

9. Complete the fields on the Attribute Mapping tab.

The default attribute mapping defined in the schema file for the endpoint type are specified.

10. Complete the fields on the Logging tab.

The logging settings for the new endpoint are specified.

11. Click Apply.

More information:

[KRB Directory Tab](#) (see page 380)

[Properties Tab](#) (see page 382)

[Custom Settings Tab](#) (see page 384)

Change Administrator Passwords

If the admin principal password has been changed or reset or due to expire, you can update the Provisioning Directory with the new password.

Note: You cannot update the password for an endpoint that uses keytab.

To change administrator passwords

1. In the Provisioning Manager, acquire the endpoint you want to view principals for.
2. Explore and correlate the endpoint you want to view principals for.
3. In the EndpointName column on the leftmost side of the Provisioning Manager, double-click on the endpoint you want to change the administrator password for.

The KRB Endpoint Property Sheet appears.

4. Click the Properties tab.

The Properties tab appears.

5. Complete the Password field on the Properties tab.

The password for the principal is specified.

6. Click Apply.

The updated password is applied.

Explore and Correlate Principals

You can correlate the explored principals with global users.

To explore and correlate principals

1. In the Provisioning Manager, right-click on an acquired KRB endpoint and then click Explore/Correlate.

The Explore and Correlate Endpoint dialog appears.

2. Complete the fields on the Explore and Correlate Endpoint dialog.
3. Click Start.

The Explore & Correlate Principals process starts.

When the process is finished, the Provisioning Manager displays the number of objects created in the endpoint in the Results section of the Explore and Correlate Endpoint dialog.

View All Principals

After you acquire an endpoint you can view all principals, or you can specify search criteria to view specific principals.

To view all principals

1. In the Provisioning Manager, acquire the endpoint you want to view principals for.
2. Explore and correlate the endpoint you want to view principals for.
3. In the EndpointName column on the leftmost side of the Provisioning Manager, right-click on an endpoint and click Content.

The Endpoint Content dialog appears.

4. In the Container tree, select KRB Accounts.
5. Complete the fields on the Endpoint Content dialog as required, then click Search.

The Provisioning Manager displays the KRB accounts in the endpoint in the KRBAccount column.

6. In the main the Provisioning Manager window, double-click the KRB principal you want to view the properties for.

The KRB Account Property sheet appears and displays the KRB account settings.

More Information:

[KRB Account Property Sheet](#) (see page 385)

Add a Principal

After you acquire an endpoint, you can add principals as required.

To add a principal

1. In the Provisioning Manager, acquire the endpoint you want to add principals to.
2. Explore and correlate the endpoint you want to add principals to.
3. In the EndpointName column on the leftmost side of the Provisioning Manager, right-click on an endpoint and then click Content.
4. The Endpoint Content dialog appears.
5. In the Container tree, select KRB Accounts.
6. Click New.
The KRB Account Property sheet appears and displays the KRB account settings.
7. Complete the fields on the Profiles tab.
The userid and provisioning status information are specified.
8. Complete the fields on the Account Properties tab.
The account properties of the principal and account template are specified.
9. (Optional) Click enc:salt.
The Encryption Type and Salt Pairs dialog appears.
Complete the fields on the Encryption Type and Salt Pairs dialog, and then click OK.
The encryption types and salt pairs are specified.
10. Click OK.
Kerberos adds the principal you specified.

More Information:

[Acquire a New Endpoint](#) (see page 363)
[Naming Limitations](#) (see page 346)
[Profiles Tab](#) (see page 385)
[Account Properties Tab](#) (see page 386)
[Encryption Type and Salt Pairs Dialog](#) (see page 391)

Modify a Principal

You can modify the properties of a principal such as user options and the associated the KRB account templates.

To modify a principal

1. In the Provisioning Manager, acquire the endpoint that contains the principal you want to modify.
2. Explore and correlate the endpoint that contains the principal you want to modify.
3. In the EndpointName column on the leftmost side of the Provisioning Manager, right-click on an endpoint and then click Content.
The Endpoint Content dialog appears.
4. In the Container tree, select KRB Accounts.
5. Complete the fields on the Endpoint Content dialog as required, then click Search.
The Provisioning Manager displays the KRB accounts in the endpoint in the KRBAccount column.
6. In the main Provisioning Manager window, double-click the KRB principal you want to modify the properties for.
The KRB Account dialog appears and displays the KRB account settings.
7. Complete the fields on the Profiles tab.
The userid and provisioning status information are specified.
8. Complete the fields on the Account Properties tab.
The account properties of the principal and account template are specified.
9. (Optional) Click enc:salt.
Note: The enc:salt button is only available if you select the Choose Random Password check box or if you modify the Password field.
The Encryption Type and Salt Pairs dialog appears.
Complete the fields on the Encryption Type and Salt Pairs dialog, and then click OK.
The encryption types and salt pairs are specified.
10. Complete the fields on the Account Templates tab.
The account properties of the principal are specified.
11. Click OK.
Kerberos modifies the principal you specified.

More Information:

[Acquire a New Endpoint](#) (see page 363)

[Explore and Correlate Principals](#) (see page 366)

[Profiles Tab](#) (see page 385)

[Account Properties Tab](#) (see page 386)

[Encryption Type and Salt Pairs Dialog](#) (see page 391)

Delete a Principal

Once you have explored an endpoint, KRB principals can be deleted as required.

To delete a principal

1. In the Provisioning Manager, acquire the endpoint that contains the principal you want to delete.
2. Explore and correlate the endpoint that contains the principal want to delete.
3. In the KRB Account column, right-click the principal you want to delete, then click Delete.
4. When prompted, confirm that you want to delete the principal.

The Provisioning Manager removes the Kerberos principal from the Kerberos database.

Kerberos Default Account Template

The Kerberos default account template, provided with the Kerberos Connector, gives a user the minimum security level needed to log in using Kerberos authentication. You can use it as a model to create new account templates. The account template contains the following values:

Account Template	Value
-expiry dates	Never
-ticket lives	Connector specified defaults
-flags	Default Kerberos flags
-password policy	None

View KRB Account Templates

You can view all KRB account templates, or you can specify search criteria to view specific KRB account templates.

To view password policies, you can specify search criteria to view all or specific KRB password policies.

To view KRB account templates

1. Click the Roles button.
2. In the Object Type list, select KRB Policy, then click Search.

The Provisioning Manager displays the KRB account templates in the AccountTemplateName column.

Add a KRB Account Template

After you acquire an endpoint, you can add KRB account templates as required.

To add a KRB account template

1. Click the Roles button.
2. In the Object Type list, select KRB Policy, then click New.
The KRB Account Template dialog appears.
3. Click the Account Template tab.
The Account Template tab appears.
4. Complete the fields on the Account Template tab.
The Provisioning Manager specifies general information about the Account Template.
5. Complete the fields on the Profiles tab.
The Provisioning Manager specifies the attribute values of the principal.
6. Complete the fields on the Account Properties tab.
The Provisioning Manager specifies the attribute values of the principal.
7. (Optional) Click enc:salt.
The Encryption Type and Salt Pairs dialog appears.
8. Complete the fields on the Encryption Type and Salt Pairs dialog, and then click OK.
The Provisioning Manager specifies the encryption types & salt pairs.
9. Complete the fields on the Program Exits Reference dialog.
The Provisioning Manager specifies the Program Exits.
10. Complete the fields on the Workflow tab.
The Provisioning Manager assigns approvers to an account template.
11. Complete the fields on the KRB Endpoints tab.
The Provisioning Manager populates the password policies on the Account Properties tab.
12. Complete the fields on the Roles tab.
The Provisioning Manager associates the provisioning roles with KRB account templates.
13. Click OK.
The Provisioning Manager creates a new KRB account template.

More Information:

[KRB Account Property Sheet](#) (see page 385)

[Profiles Tab](#) (see page 385)

[KRB Endpoints Tab](#) (see page 390)

Modify a KRB Account Template

You can modify the properties of a KRB account template such as attributes and the encryption types and salt pairs.

To modify a KRB account template

1. Click the Roles button.
2. In the Object Type list, select KRB Policy, then click Search.
The KRB Account Template dialog appears.
3. Click Search.
The Provisioning Manager displays the KRB account templates in the AccountTemplateName column.
4. In the AccountTemplateName column, right click the account template you want to modify.
5. To modify general information about an account template, complete the fields on the Account Template tab.
6. To modify the attribute values of a principal, complete the fields on the Profiles tab.
The Account Template Property dialog displays.
7. To modify the attributes of a principal, complete the fields on the Account Properties tab.
8. To modify the encryption types & salt pairs, do the following:
 - a. Click enc:salt.
The Encryption Type and Salt Pairs dialog appears.
Note: The enc:salt button is only available if you select the Choose Random Password check box or if you modify the Password field.
 - b. Complete the fields on the Encryption Type and Salt Pairs dialog, and then click OK.
Note: The enc:salt and random passwords are account template-only attributes, and so changes will not be propagated to associated accounts.
9. To modify program exits, complete the fields on the Program Exits Reference dialog.
10. To assign approvers to a account template, complete the fields on the Workflow tab.

11. To associate provisioning roles with KRB account templates, complete the fields on the Roles tab.
12. Click OK.
13. When prompted, confirm that you want to apply the changes to the associated accounts.

More information:

[Profiles Tab](#) (see page 385)

[KRB Endpoints Tab](#) (see page 390)

Delete a KRB Account Template

You can delete KRB account templates as required.

Note: You may not be able to delete the account template if there are accounts that have this template assigned.

To delete a KRB account template

1. Click the Roles button.
2. In the Object type list, select KRB Policy.
3. Click Search.

The Provisioning Manager displays the KRB account templates in the AccountTemplateName column.

4. In the Provisioning Manager, right click the KRB account template you want to delete and then click Delete.
5. When prompted, confirm that you want to delete the KRB account template.

Synchronize Accounts with Account Templates

To synchronize an account with a KRB account template, right-click on the KRB account template and select Synchronize Accounts with Account Template.

View a KRB Password Policy

After you acquire an endpoint you can view all KRB password policies, or you can specify search criteria to view specific KRB password policies.

To view a KRB password policy

1. In the Provisioning Manager, acquire the endpoint you want to view password policies for.
2. Explore and correlate the endpoint you want to view password policies for.
3. In the EndpointName column, right-click the endpoint you want to view password policies for, then click Content.

The Endpoint Content dialog appears.

4. In the Container tree, select KRB Password Policies.
5. Complete the fields on the Endpoint Content dialog as required, then click Search.
6. In the main the Provisioning Manager window, double-click the KRB password policy you want to view the properties for.

The Password Policy Properties dialog appears and displays the parameters of the password policy.

More information:

[Acquire a New Endpoint](#) (see page 363)

[Explore and Correlate Principals](#) (see page 366)

[Password Account Template Properties Tab](#) (see page 389)

Add a KRB Password Policy

After you acquire an endpoint, you can add KRB password policy as required.

To add a KRB password policy

1. In the Provisioning Manager, acquire the endpoint you want to view password policies for.
2. Explore and correlate the endpoint you want to view password policies for.
3. In the EndpointName column, right-click the endpoint you want to add a password policy for, then click Content.

The Endpoint Content dialog appears.

4. In the Container tree, select KRB Password Policies.
5. Click New.

The Password Policy Properties dialog appears.

6. Complete the fields on the Password Policy Properties dialog, and then click OK.

The connector creates the password policy in the endpoint, and is now available to be assigned to principals.

More information:

[Acquire a New Endpoint](#) (see page 363)

[Explore and Correlate Principals](#) (see page 366)

[Password Account Template Properties Tab](#) (see page 389)

Modify a KRB Password Policy

After you acquire an endpoint, you can modify KRB password policies as required.

When you modify a Kerberos password policy, the accounts that refer to that policy are not affected until the account's password is changed. When the password is changed, the new password must conform to the properties of the modified policy.

Note: If the password policy is assigned to principals, you can only modify the account template in a way that is below the current rule enforcement. For example, if the minimum password length is specified as eight characters, you cannot change it to ten characters.

To modify a KRB password policy

1. In the Provisioning Manager, acquire the endpoint you want to view password policies for.
2. Explore and correlate the endpoint you want to view password policies for.
3. In the EndpointName column, right-click the endpoint you want to view password policies for, then click Content.

The Endpoint Content dialog appears.

4. In the Container tree, select KRB Password Policies.
5. Complete the fields on the Endpoint Content dialog as required, then click Search.

The Provisioning Manager displays the password policies in the main Provisioning Manager window.

6. In the main Provisioning Manager window, double-click the KRB password policy you want to modify the properties for.

The Password Policy Properties dialog appears.

7. Modify the fields on the Password Properties dialog as required, and then click OK.

The Provisioning Manager modifies the password policy.

More information:

[Password Account Template Properties Tab](#) (see page 389)

Delete a KRB Password Policy

After you acquire an endpoint, you can delete KRB password policies as required.

Note: You cannot remove a password policy if a principal is assigned to it.

To delete a KRB account template

1. In the Provisioning Manager, acquire the endpoint you want to view password policies for.
2. Explore and correlate the endpoint you want to view password policies for.
3. In the EndpointName column, right-click the endpoint you want to view password policies for, then click Content.

The Endpoint Content dialog appears.

4. In the Container tree, select KRB Password Policies.
5. Complete the fields on the Endpoint Content dialog as required, then click Search.

The Provisioning Manager displays the password policies in the main Provisioning Manager window.

6. In the Provisioning Manager, right click the KRB password policy you want to delete, then click Delete.
7. When prompted, confirm that you want to delete the password policy.

The connector removes the password policy from the endpoint.

More information:

[Acquire a New Endpoint](#) (see page 363)

[Explore and Correlate Principals](#) (see page 366)

KRB Endpoint Property Sheet

Use this property sheet to register or view the properties of a Kerberos endpoint. The following property pages apply to Kerberos endpoints:

KRB Endpoint

Specifies information about the endpoint.

Endpoint Settings

Specifies endpoint attributes.

Properties

Specifies the Kerberos Server and Realm and the credentials used for connection.

Custom Settings

Specifies the supported encryption type and salt pairs of the endpoint and how the disablement of the principal is implemented.

Program Exits Reference

Specifies the priority, name, and type of program exit to be used.

Attribute Mapping

Specifies endpoint mapping configuration information.

Logging

Specifies logging information.

Statistics

Provides read-only information reporting on activity for this object.

The fields in this property sheet are listed below:

OK/Apply

Accepts or saves the information in the property sheet or dialog.

Cancel/Reset

Exits or resets the values in the property sheet or dialog.

KRB Directory Tab

Use this tab to register or view the properties of a Kerberos endpoint.

The fields in this tab are listed below:

Directory Name

Specifies the name of the Kerberos directory.

Size/Type: 1 to 100 characters

This is a required field.

Comments

User-supplied description field.

Size/Type: 1 to 128 characters

Default Domain

Specifies the name of the domain where the default account template exists.

This drop-down list box displays your personal domain list only. This list is a local list and is unique for each account on the client workstation. This list is intended for the domains that you use frequently. Selecting a domain from this list does not require a request to a network server and is therefore very fast.

To display and access every domain in your entire network, click the Domains button. When you do so, you can add one or more domains to your personal domain list.

Default Account Template

Specifies the default account template of the endpoint.

The default account template is used to create new accounts on the endpoint. You must define a default account template to activate the drag-and-drop feature on an organization or organizational unit. If the organization or organizational unit is not associated with an account template, the default account template at the tree level is applied.

The drop-down list box contains all the account templates defined for KRB endpoints.

Domain

Displays every domain in the entire network and makes them available for you to access. Clicking this button displays the Full Domain List Selector dialog.

If desired, you can add one or more of these domains to your personal domain list, which is stored in the Default Domain drop-down list box.

Clicking the Domain button issues a request to a network server, and may take longer than using the Default Domain drop-down list box.

More Information:

[KRB Endpoint Property Sheet](#) (see page 379)

Properties Tab

Use this tab to enter the Kerberos Server and Realm and the credentials used for the connection.

The fields in this tab are listed below:

Kerberos Server

Specifies the host name or IP address of the server where the Kerberos Server is deployed.

Size Type: 1 to 100 alphanumeric characters.

This is a required field.

Note: This host may be used by multiple KRB endpoints managing multiple Kerberos realms.

Port

Specifies the port number the server listens on. If this field is left blank, the default port is used. If not left blank, the number is passed to the kadmin command.

Kerberos Realm

Specifies the Kerberos realm to be managed.

Size/Type: 1 to 100 alphanumeric characters.

This is a required field.

Note: The same naming rules that apply to principals also apply to Kerberos realms.

Security controls

Use these controls to specify the authentication method. When the Keytab option is selected, the Keytab and Principal fields are enabled. When the Principal option is selected, the Principal and Password fields are enabled.

Keytab

Specifies the fully qualified keytab filename. If this field is left blank, the connector uses the default keytab. On Solaris the default keytab file is:

/etc/krb5/krb5.keytab

Principal

Specifies the principal to authenticate with. If you use the keytab option, the principal's encrypted password must be in the keytab file.

If you use the keytab option and you leave this field blank, the host principal of the host where the Java CS is deployed, host/localhostname@REALM, will be used to authenticate using its key from the keytab. When the principal is specified, its key from the keytab will be used to authenticate.

Note: If you use the keytab option and add a principal to a keytab file using the ktadd command of kadmin, its password will be randomized. So if the principal is being used for another KRB directory by specifying the password, that directory will not be operable since the password is not correct.

This is a required field when using the password option.

Size/Type: 1 to 128 alphanumeric characters.

Note: The naming rules should follow principal naming rules.

Password

Specifies the password for the principal.

Explore Timeout

Specifies the time that the Provisioning server waits for a response from the remote server before generating an error for the Explore and Correlate operation.

Default Value: 3600 seconds or 1 hour

More Information:

[KRB Endpoint Property Sheet](#) (see page 379)

Custom Settings Tab

Use this tab to specify the supported encryption type and salt pairs for the endpoint.

The fields in this tab are listed below:

Supported Encryption Type and Salt Pairs

Specifies the supported encryption type and salt pairs to be added for this endpoint.

Add Button

Adds the pairs into the Encryption Type and Salt Pairs list box.

Supported Encryption Type and Salt Pairs List Box

When acquiring a directory for the first time, this list is pre-populated with the following six items:

- aes128-cts-hmac-sha1-96:normal
- arcfour-hmac-md5:normal
- arcfour-hmac-md5-exp:normal
- des-cbc-crc:normal
- des-cbc-md5:normal
- des3-cbc-sha1:normal

You can also manually add the type and pairs here and they will be made available at the Principal and Account Template property sheets. Once the pairs have been added, you can select a pair to be removed or edited.

Disable Principal: Set "ALLOW TIX" to false Check Box

When checked, specifies that the `DISALLOW_ALL_TIX` flag is set when the principal is disabled. This results in the suspended principal's ALLOW TIX check box on the Account Properties Tab to be unchecked.

Disable Principal: Expire the Principal Check Box

When checked, specifies that the expiration date is set to a date in the past so that the principal is disabled.

More Information:

[KRB Endpoint Property Sheet](#) (see page 379)

KRB Account Property Sheet

Use this property sheet to create, view, or modify the current properties of a KRB account. The following property pages apply to KRB accounts:

Profiles

Specifies user information such as userid and status information of the account.

Account Properties

Specifies account properties including the password, password account template, and advanced Kerberos options.

Account Templates

Maintains account template inclusions.

Statistics

Provides read-only information reporting on activity for this object.

The fields in this property sheet are listed below:

OK/Apply

Accepts or save the information in the property sheet or dialog.

Cancel/Reset

Exits or reset the values in the property sheet or dialog.

Profiles Tab

Use this tab to view and modify the userid and provisioning status information.

The fields in this tab are listed below:

UserId

Specifies the name of the Kerberos principal.

Rule String: %AC%

This is a required field.

Provisioning information: Status

Specifies whether or not the user profile is suspended:

- 0 - active
- 1 - suspended

More Information:

[KRB Account Property Sheet](#) (see page 385)

[KRB Account Template Property Sheet](#) (see page 388)

Account Properties Tab

Use this tab to view and modify the account properties of a principal and account template.

The fields in this tab are listed below:

Password

Specifies the password of the Kerberos principal.

Size/Type: 0 to 64 alpha-numeric characters

Rule String: %p%

Choose Random Password Check Box

When checked, specifies that the password of the principal is randomly generated. The generated password conforms to the selected password account template.

enc:salt Button

Displays the Encryption Type and Salt Pairs dialog where you can view or edit the encryption types and key salt pairs.

Note: If there is no password change, the dialog is read-only. If the password is changed or Choose Random Password is selected, the dialog is enabled and can be edited.

Maximum Ticket Life

Specifies the maximum ticket life of the Kerberos principal in days, hours minutes and seconds.

Maximum Renewable Life of a Ticket

Specifies the maximum renewable life of a ticket for the Kerberos principal in days, hours, minutes and seconds.

Password Account Template

Specifies the password account template that is assigned to the principal.

Size/Type: up to 128 alpha-numeric characters

Expiration Date Time: User

Specifies the expiration date of the principal.

Expiration Date Time: Password

Specifies the expiration date of the principal's password.

ALLOW POST DATED Check Box

When checked, specifies that the principal is can receive post-dated tickets.

ALLOW PROXIABLE

When checked, specifies that the principal can receive proxiable tickets.

REQUIRES HWAUTH

When checked, specifies that the principal must pre-authenticate using a hardware device before being allowed to kinit.

ALLOW TIX

When checked, specifies that the principal can be issued tickets.

ALLOW FORWARDABLE

When checked, specifies that the principal can receive forwardable tickets.

ALLOW DUP SKEY

When checked, specifies that the principal can receive a session key for another user.

ALLOW SVR

When checked, specifies service tickets can be issued to this principal.

NEED PWD CHANGE

When checked, specifies that a flag is set to force a password change.

ALLOW RENEWABLE

When checked, specifies that the principal is *not* prohibited from obtaining renewable tickets.

REQUIRES PREAUTH

When checked, specifies that the principal must pre-authenticate before being allowed to kinit.

ALLOW TGS REQ

When checked, specifies that a Ticket-Granting Service request for a service ticket is permitted for this principal.

PWD CHANGE SERVICE

When checked, specifies that a flag is set to mark this principal as a password change service.

More Information:

[KRB Account Property Sheet](#) (see page 385)

[KRB Account Template Property Sheet](#) (see page 388)

KRB Account Template Property Sheet

Use this property sheet to create, view, or modify the current properties of a KRB account template. The following property pages apply to KRB account templates:

Account Template

Specifies general information about an account template.

Profiles

Specifies the userid of the principal.

Account Properties

Specifies the account properties of the account template including password, password account template, and advanced Kerberos options.

Program Exits Reference

Specifies the priority, name, and type of program exit to be used.

Workflow

Specifies the name of the technical approver for this account template.

KRB Endpoints

Maintains endpoint group inclusions.

Roles

Maintains provisioning role inclusions.

Statistics

Provides read-only information reporting on activity for this object.

The fields in this property sheet are listed below:

OK/Apply

Accepts or saves the information in the property sheet or dialog.

Cancel/Reset

Exits or resets the values in the property sheet or dialog.

Password Account Template Properties Tab

Use this tab to add a new or modify an existing password account template.

The fields in this dialog are listed below:

Account Template Name

Specifies the name of a Kerberos password account template.

Size/Type: 1 to 128 alphanumeric characters.

This is a required field.

Description

Specifies the description of the password account template.

This description only exists in the Provisioning repository.

Size/Type: 1 to 128 alphanumeric characters.

Maximum Password Life

Specifies the maximum password life in days, hours, minutes, and seconds.

Minimum Password Life

Specifies the minimum password life in days, hours, minutes, and seconds.

Minimum Password Length

Specifies the minimum password length in number of characters.

Minimum Number Password Character Class

Specifies the minimum number of password characters. You can choose one of the following valid values:

- 1 - only letters
- 2 - both letters and numbers
- 3 - letters, numbers, and punctuation

Number of Old Keys Kept

Specifies the number of old keys to be kept to disallow reuse.

Reference Count

Specifies the number of principals that reference this password account template.

This is a read-only field.

KRB Endpoints Tab

Use this tab to associate one or more KRB endpoints in order to pre-populate password policies on the Account Property Tab.

The fields on this page are listed below:

Available

Specifies the objects that are available for inclusion.

Included

Specifies the objects that have been added as inclusions.

Add

After selecting an object in the Available list, click the Add (>) button to add it to the Included list.

Add All

After selecting an object in the Available list, click the Add All (>>) button to add all the objects to the Included list.

Remove

After selecting an object in the Included list, click the Remove (<) button to remove it to the Available list.

Remove All

After selecting an object in the Included list, click the Remove All (<<) button to remove all the objects to the Available list.

Domain

Specifies the domain for the search.

Domain

Displays the Full Domain List Selector dialog to add domains to the Domain field.

Attribute

Specifies a simple attribute that is used to search.

Advanced

Displays the Advanced Search Attributes dialog. Use this dialog to set more advanced search criteria.

Tip: This is useful if you want to narrow down the list of objects in the class. Click the Search button to start the search.

Value

Specify a value in the Value field to restrict the search criteria, and click the Search button. By default, the wildcard character (*) is specified, which causes the search to return all entries.

Note: If you perform an advanced search for an attribute, this field is disabled.

Search

Starts the search.

More Information:

[KRB Account Template Property Sheet](#) (see page 388)

Encryption Type and Salt Pairs Dialog

Use this dialog to view and edit the encryption type and salt pairs.

The fields in this dialog are listed below:

Available List Box

Specifies the encryption types and salt pairs that are available.

Add Button

Adds the selected available pair into the Included List box.

Note: If you try to add two pairs of the same type, only one will be added.

Remove Button

Removes the selected pair from the Included List Box.

Included List Box

Specifies the encryption type and salt pairs that have been included. Once you have clicked OK, the pairs are kept in memory and committed to the KDC together with the password change of the principal.

If there are invalid entries, an error message is displayed. To correct the error, you must go back to the Customization Tab of the endpoint property sheet.

Note: The encryption type and salt pairs listed here are sorted per Java's implementation of the natural ordering of strings. When two similar encsalts are passed to Kerberos, only the first one is used. For example, if both des-cbc-crc:normal and des-cbc-md5:normal are listed, only des-cbc-crc:normal is used because "c" has a lower value than "m".

Chapter 12: Midrange Connectors

This section contains the following topics:

[OS/400 Connector](#) (see page 393)

[OpenVMS Connector](#) (see page 418)

[HP NonStop SAFEGUARD Connector](#) (see page 431)

The following connectors are considered to be mid-range connectors:

- OS/400 Connector
- OpenVMS Connector
- HP NonStop SAFEGUARD (NSK)

The information in this chapter describes how to install and manage these connectors.

OS/400 Connector

The OS/400 Connector lets you administer accounts and groups on OS/400 machines and provides a single point for all user administration by letting you do the following:

- Register endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage OS/400 accounts using OS/400-specific account templates
- Change account passwords and account activations in one place
- Synchronize global users with their provisioning roles or synchronize global users' accounts with their account templates
- Assign an OS/400 account template to each of your OS/400 endpoints
- Use the default endpoint type account template to create accounts with the minimum level of security needed to access an OS/400 endpoint
- Create and manage OS/400 groups
- Generate and print reports about OS/400 accounts and groups

OS/400 Installation

The OS/400 Connector be managed with the Java Server Connector installation process. For more information on this installation process, click [here](#) (see page 26).

OS/400 Migration Steps

To migrate from the C++ OS/400 connector to the Java OS/400 connector, you must do the following:

- Install the OS/400 Java connector using the JCS installation
- Using Connector Xpress, switch the OS400 Endpoint Type Connector Server from the C++ Connector Server to JCS

Once this has been done, all types of operations can be executed against the existing OS400 endpoints seamlessly.

How to Configure your Machines

You must configure your OS/400 system to use the OS/400 connector. To do this, install and configure programs on your OS/400 system.

Install and Configure Programs on OS/400

The JTOPEN toolkit used by the OS/400 connector requires the following programs to be installed and configured on your OS/400 system:

- TC1 Licensed Program (TCP/IP Connectivity Utilities for OS/400)
- Host Server Option of OS/400

These programs are necessary so the OS/400 connector can connect to your OS/400 system and access its data and services.

How to Secure Your Information (Optional)

You can send information through secured or unsecured channels.

For security purposes, we recommend that you secure the communications between all your machines. To do this, you must configure the following:

- Provisioning Server
- Java Connector Server
- OS/400 system

Connect Using SSL

Communication between the Provisioning Server/Java Connector Server and the OS/400 machine is secured by SSL. Using SSL is optional in both links and can be switched on when acquiring the OS/400 machine. Certificates are used to authenticate the server and encrypt communications and the username and password are used to authenticate the client request on the OS/400 machine.

To use SSL, the Java Connector Server machine must have the endpoint certificate installed in the java certificate store in the JRE in which the Java Connector Server machine is running. Click [here](#) (see page 401) for information on how to configure the Java Server Connector.

Configure Your OS/400 System

Secure the channel between the JCS and your OS/400 system by performing these steps:

1. Prepare the system
2. Select the certificate location
3. Import the certificate authority
4. Request a server certificate from the CA
5. Request a server certificate for your system
6. Import the server certificate
7. Assign the Server Certificate to your OS/400 applications

Prepare the System

To prepare your OS/400 system, perform the following procedure:

On your OS/400 system

1. Verify that one of the following client encryption licensed programs is installed:

5722-CE2

IBM iSeries Client Encryption (56-bit) Version 5, Release 1. This program is used in countries other than the United States or Canada.

5722-CE3

IBM iSeries Client Encryption (128-bit) Version 5, Release 1. This program is used in the United States and Canada only.

5769-CE2

IBM iSeries Client Encryption (56-bit) Version 4, Release 5. This program is used in countries other than the United States or Canada.

5769-CE3

IBM iSeries Client Encryption (128-bit) Version 4, Release 5. This program is used in the United States and Canada only.

Note: These programs are an installation option on your OS/400 system.

2. Verify that one of the following server encryption licensed programs is installed:

5722-AC2

IBM iSeries Server Encryption (56-bit) Version 5, Release 1. This program is used in countries other than the United States or Canada.

5722-AC3

IBM iSeries Server Encryption (128-bit) Version 5, Release 1. This program is used in the United States and Canada only.

5769-AC2

IBM iSeries Server Encryption (56-bit) Version 4, Release 5. This program is used in countries other than the United States or Canada.

5769-AC3

IBM iSeries Server Encryption (128-bit) Version 4, Release 5. This program is used in the United States and Canada only.

Note: These programs are an installation option on your OS/400 system.

3. Verify that the following licensed programs are installed:

5722-SS1

Product Option 34 - Digital Certificate Manager

5722-DG1

IBM HTTP Server

4. Create a file share from your OS/400 system to your Provisioning Server/Java Connector Server.

Select the Certificate Location

Select the location where you will import the certificate on your OS/400 system.

To select the location

1. Start the HTTP Administration Server using the Operations Navigator or run the following command at your OS/400 command prompt:
`STRTCPSVR SERVER(*HTTP) HTTPSVR(*ADMIN)`
2. Connect to the HTTP Administration Server by pointing your browser at the following location and logging on with your system credentials:
`http://server:2001`

server

Specifies the name of the system running OS/400.

Note: Your logon ID must have the All Object Access and System Configuration permissions.

3. Select the Digital Certificate Manager link.
The Digital Certificate Manager window appears. The left frame contains navigational buttons and the right frame contains command buttons.
Note: The steps that reference the Digital Certificate Manager are based on Version 5, Release 1. If you are using another version, these steps may vary slightly.
4. Click the Select a Certificate Store button in the left frame.
5. Select the *SYSTEM store radio button and then click Continue.
6. Enter the password for the *SYSTEM certificate store and then click Continue.

Import the Certificate Authority

Once you have selected the certificate location, import the certificate from your Certificate Authority (CA).

From the left frame

1. Expand the Manage Certificates link.
2. Select the Import Certificate link.
The Import Certificate window appears.
3. Select the Certificate Authority (CA) radio button and then click Continue.
4. Enter the directory location that contains the certificate for the Integrated File System (IFS) on your OS/400 system and then click OK.

For example, enter: `\home\etadmin\certificate_file_name`.

5. Enter a unique name in the Label field for the certificate, for example etaCACert, and then click Continue.
6. Click the OK button.

The Digital Certificate Manager reads the certificate file and imports it into the system.

Request a Server Certificate from the CA

After importing the Certificate Authority, you must now request a server certificate.

From the CA

1. Select the Create Certificate option.
The Create Certificate window appears.
2. Select the Server or client certificate radio button and then click Continue.
3. Select the Internet Certificate Authority radio button, for example VeriSign, and then click Continue.
4. Enter at least the following information and then click Continue:

Key size

1024 bits

Certificate Label

The name of your certificate

Common Name

The name of your server

Organization Name

The name of your organization

State or province

The name of your state or province

Country

The name of your country

5. Copy the generated lines (including the BEGIN and END lines) into a file and then save that file on your OS/400 system.

Request a Server Certificate for Your System

To request a server certificate for your OS/400 system, follow this procedure:

From a Certificate Authority (CA)

1. Install and configure Microsoft Certificate Services on your Windows 2000 server.
2. Point your browser to `http://computer-name/certsrv`.
where *computer-name* is the name of the computer for which you are generating the certificate. The Microsoft Certificate Services Wizard appears.
3. Select Request a certificate, and click Next.
4. Select Advanced request, and click Next.
5. Select Submit a certificate request using a base64 encoded PKCS #10 file or a renewal request using a base64 encoded PKCS #7 file, and click Next.
6. Open the certreq.txt file with Notepad and cut its contents.
7. Paste the contents of certreq.txt in the Saved Request box, and click Submit.
8. Select Base 64 Encoded, and click the Download CA Certificate.
9. Save the certificate to your hard drive.

Note: Remember the location where you save the certificate.

Import the Server Certificate

Once you generate a server certificate, you can import it into the system.

From the CA

1. Expand the Manage Certificates link in the left frame.
2. Select the Import Certificate link.
The Import Certificate window appears.
3. Select the Server or client radio button and then click Continue.
4. Enter the directory path that contains the certificate for the IFS on your OS/400 system and click Continue.
For example, enter: `\home\etadmin\usildaaj.cer`.
5. Click OK.

Assign the Server Certificate to Your OS/400 Applications

After importing the certificate, you must assign the server certificate to the following applications:

- OS/400 TCP Central Server
- OS/400 TCP Remote Command Server
- OS/400 TCP Signon Server

From the CA

1. Expand Manage Applications in the left frame.
2. Select Update certificate assignment.
3. Select Server and then click Continue.
The Update Certificate Assignment window appears.
4. Perform the following steps for each of the applications:
 - a. Select the radio button for the application and then click the Update Certificate Assignment button.
 - b. Select the server certificate and then click the Assign New Certificate button.

5. Stop the applications by issuing the following command with each argument:

```
ENDHOSTSVR *CENTRAL  
ENDHOSTSVR *RMTCMD  
ENDHOSTSVR *SIGNON
```

6. Start the applications by issuing the following command with each argument:

```
STRHOSTSVR *CENTRAL RQDPCL(*TCP)  
STRHOSTSVR *RMTCMD RQDPCL(*TCP)  
STRHOSTSVR *SIGNON RQDPCL(*TCP)
```


Configure the Java Connector Server

If you are using a certificate from one of the following CAs, you do not need to perform this step:

- IBM World Registry
- Integrion Financial Network
- RSA Data Security, Inc.
- Thawte Consulting
- VeriSign, Inc.

If, you are using a certificate from a different CA, you must configure the Java Connector Server. If you use the same certificate for each OS/400 system, you will perform these steps only once.

To configure the Java Connector Server

1. Stop the JCS service..
2. Copy the CA certificate from your certificate authority to the following folder:

`C:\Program Files\CA\Identity Manager\Connector Server\_uninst\_jvm\lib\security`
3. Open a DOS screen and change the DOS prompt to the following directory:

`cd C:\Program Files\CA\Identity Manager\Connector Server\_uninst\_jvm\lib\security`
4. Issue the following command to import the CA certificate into the CA certificate store for Java:

`..\..\bin\keytool -import -alias "eTrust Admin CA Certificate" -file
certificate_name.cer -keystore cacerts`
 - a. Enter **changeit** at the "Enter a keystore password" prompt.
 - b. Enter **yes** at the "Trust this certificate" prompt.
5. Restart the JCS service.

How the Password Synchronization Agent Works

The Password Synchronization agent lets password changes, made on the OS/400 endpoint system, be propagated to your other accounts managed by Identity Manager. The Password Synchronization agent works as follows:

1. Install and execute the agent on the OS/400 endpoint system
As part of the installation, the program is registered with the OS/400 system so that when users change their passwords, the agent sends the password changes on to the Provisioning Server.
2. The Provisioning Server propagates the password change to the associated accounts.
Password changes initiated from the Change Password command (CHGPWD) or Change Password (QSYCHGPW) API are received by the agent.
3. The agent logs operation success or failure to a log file located and PWDSYNCH/LOG.

The Password Synchronization agent is supported on the following platforms:

- OS400 V5R2
- OS400 V5R3
- OS400 V5R4

To install the Password Synchronization Agent

1. Locate the Identity Manager Provisioning Components download or other media.
2. Start the Product Explorer.
3. Select Install Products, Infrastructure
4. Select Password Synchronization Agent for OS/400
5. Follow the onscreen instructions to complete the installation.

Note: The installation instructions in the Endpoint Agent Software link are included in the following sections.

Install the Password Synchronization Agent

You must have *ADDOBJ privileges and the following are necessary for the agent to receive password change notifications:

- System value QPWDVLDPGM must be set to *REGFAC
- Program must be registered with the command WRKREGINF
EXITPNT(QIBM_QSY_VLD_PASSWRD)

The agent is initiated only when a password change is made. To change the password, issue the CHGPWD command.

Note: The Global User must be flagged for password synchronization. See the section "Password Synchronization" in the *Provisioning Guide* for more information.

On the iSeries

1. Log on as a user with *ALLOBJ and *SECADM privileges (for example, QSECOFR).

2. Create a user called PWDSYNCH:

```
CRTUSRPRF USRPRF(PWDSYNCH) PWDEXP(*YES)
```

Note: As a security measure, the user is created with the password expired.

3. Create a savefile to store the installation package in a library of your choice (for example, MYLIB):

```
CRTSAVF MYLIB/PWDSYNCH
```

4. On the Windows machine with the savefile, use FTP to transfer the savefile to the iSeries:

```
ftp <hostname>  
binary  
cd MYLIB  
put PWDSYNCH.FILE
```

5. On the iSeries, extract the program from the savefile:

```
RSTLIB SAVLIB(PWDSYNCH) DEV(*SAVF) SAVF(MYLIB/PWDSYNCH)
```

This command extracts and installs the synch agent into the PWDSYNCH library.

6. Verify the installation:

```
DSPLIB PWDSYNCH
```

The following objects should be displayed:

Object	Type	Attribute
PWDSYNCH	*PGM	CLE
CONFIG	*FILE	PF
LOG	*FILE	PF

7. Set up the iSeries to use PWDSYNCH as the password validation exit program:

```
CHGSYSVAL SYSVAL(QPWDVLDPGM) VALUE(*REGFAC)
ADDEXITPGM EXITPNT(QIBM_QSY_VLD_PASSWRD) FORMAT(VLDP0100) PGMNBR(1)
PGM(PWDSYNCH/PWDSYNCH) TEXT('eTrust Admin Password Synch Agent')
```

8. On the iSeries, specify the connection parameters for your Provisioning Server:

```
EDTF FILE(PWDSYNCH/CONFIG)
```

SSL Configuration

SSL is used to encrypt communication between the synch agent and the Provisioning server. This is important for the synch agent because SSL sends passwords across the network. SSL is recommended to always be used.

The synch agent must trust the Provisioning Server's certificate in order to connect with SSL. Therefore, the certificate must be installed on the iSeries machine and configured so that the certificate is trusted by the synch agent. These tasks are performed by the Digital Certificate Manager, an optional component of OS/400. Please follow the OS/400 documentation regarding installation and setup of the Digital Certificate Manager.

Install the Provisioning Server Certificate

The following operating system components must be installed on your iSeries machine to use SSL:

- Cryptographic access provider licensed program (5722-AC3)
- Digital Certificate Manager (Option 34 of OS/400)
- IBM HTTP Server for iSeries (5722-DG1)

On the iSeries

1. Upload the Provisioning server certificate from the Provisioning server machine to the iSeries. The certificate can be found at:

```
C:\Program Files\CA\Identity Manager\Provisioning Server\Data\Tls\et2_cacert.pem
```

2. Log onto the DCM.

Using a web browser, go to `http://<hostname>:2001`. When prompted, log on as QSECOFR and click the Digital Certificate Manager link

3. Work with the *SYSTEM certificate store

Click the 'Select a Certificate Store' button and select the *SYSTEM certificate store. If this store does not exist, create a new store called *SYSTEM and then enter the certificate store password.

4. Import the certificate as a CA Certificate using the DSM.

Click Manage Certificates, Import Certificate and select the Certificate Authority (CA) option and then enter the file name of the Provisioning server certificate. (This is where you uploaded the certificate in step 1). Enter a label for the certificate: Provisioning Server Importing the certificate is complete.

5. Configure the 'Directory Services client' application to trust the Admin server certificate by opening Manage Applications, Define CA trust list and choosing Directory Services Client.

The Provisioning Server certificate should be listed here if imported correctly from step 4.

Click 'Trusted' for the Provisioning Server certificate and click OK at the bottom of the list.

6. Give PUBLIC read permission to the SSL files and grant read access to the *SYSTEM certificate store:

(/QIBM/userdata/ICSS/Cert/Server/default.kdb)

Grant read and execute permission to the parent folder

(/QIBM/userdata/ICSS/Cert/Server)

Note: Adopting authority of user PWDSYNCH does not work in the / file system, so access must be granted for all users.

Un-Install the Password Synchronization Agent

If you need to un-install the Password Synchronization Agent, follow this procedure.

From the password validation exit point

1. Remove PWDSYNCH:

```
RMVEXITPGM EXITPNT(QIBM_QSY_VLD_PASSWRD) FORMAT(VLDP0100) PGMNBR(1)
```

2. Delete the synch agent library:

```
DLTLIB PWDSYNCH
```

3. Delete PWDSYNCH user

```
DLTUSRPRF PWDSYNCH
```

4. Remove the Provisioning server certificate by following the SSL instructions to log onto the DCM and work with the *SYSTEM certificate store:

Click Manage Certificates, Delete Certificate and select 'Certificate Authority (CA)'

Select 'Provisioning Server' certificate and click Delete.

OS/400 Password Agent Parameter Must be Set Correctly

The "pwd_case_action" parameter must have the value set correctly for it to work. Correct values include:

- pwd_case_action = pwd_case_unchanged
- pwd_case_action = pwd_to_uppercase
- pwd_case_action = pwd_to_lowercase

If pwd_case_action = [invalid value] the password will be forced to uppercase.

Note: You need to pay attention to the QPWDLVL (Password Level) system value when configuring the Password Synch Agent:

- When QPWDLVL is set to 0 (default value) on the AS400 system, passwords with a length of 1 to 10 uppercase characters are supported.
- When QPWDLVL is set to 2 or 3, passwords from 1 to 128 characters with mixed case are allowed.

As the default behavior (when "pwd_case_action" is set to "pwd_case_unchanged" or not specifying a value for "pwd_case_action", or specifying an invalid value for "pwd_case_action", or not specifying "pwd_case_action" at all), the PSA propagates the unchanged password (all upper case when QPWDLVL equals 0, and mixed case when QPWDLVL equals 2 or 3) to the Provisioning Server. However, regardless of the value of QPWDLVL, you can force the PSA to propagate passwords with upper case or lower case by setting "pwd_case_action" to "pwd_to_uppercase" or "pwd_to_lowercase" respectively.

OS/400 Support for FIPS and IPv6

For release r12 of Identity Manager, the OS/400 Connector does not support FIPS or IPv6.

The OS/400 Password Synchronization Agent also does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire an OS/400 Machine

You must acquire the OS/400 machine before you can administer it with Identity Manager.

From the OS/400 Endpoint Property Sheet

1. Register the machine as an endpoint in Identity Manager.

Provide the OS/400 server machine name, the user ID and password when acquiring an OS/400 system.

Note: Before acquiring the endpoint, make sure that it is registered to use the Java connector. To do this:

1. In Connector Xpress, right-click the OS400 endpoint
2. Select Set Managing CS
3. Select Java Connector

During the registration process, Identity Manager identifies the OS/400 machine you want to administer and gathers information about it.

2. Explore the objects that exist on the endpoint.

After registering the machine in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all OS/400 objects. You can correlate the accounts with global users at this time, or you can wait to correlate them.

3. Correlate the explored accounts to global users by choosing either of the following Connectors:

- Use existing global users
- Choose this option when there are already global users in Identity Manager and you want to connect the existing global users to the OS/400 accounts
- Create global users as needed

Choose this option when there are no global users and you want to populate Identity Manager from the OS/400 accounts.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the user profile name with each existing global user name. If a match is found, Identity Manager associates the OS/400 account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the user profile name with each existing global user's full name. If a match is found, Identity Manager associates the OS/400 account with the global user. If a match is not found, Identity Manager performs the following step.
- c. Identity Manager associates the OS/400 account with the [default user] object or a new global user is created depending on your choice.

Note: More information on enabling Secure Socket Layer (SSL) communications between the Provisioning Server and the OS/400 system exists in the Provisioning Manager Help.

Streaming Search Results

During the explore operation, the connector returns accounts to the Provisioning server as soon as possible instead of waiting until all accounts have been reviewed. This reduces memory usage resulting in a more efficient explore process.

OS/400 Provisioning Roles and Account Templates

The OS/400 Default Policy, provided with the OS/400 Connector, gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

Policy Default Values

The new account templates are created with default values for most attributes. The new account templates are valid as soon as they are created and the attributes can be customized as necessary.

OS/400 Cascading Delete

In previous versions, if an OS400 account owned objects, the account could not be deleted from Identity Manager. In this version, a flag called "cascadingDelete" in the OS400 connector.xml in the Java Connector Server can be used to change this behavior. When the flag is set to true, the account and all objects owned by the account will be deleted. The default value is set to true.

If you want override the default value, you must:

1. Copy the connector.xml to

```
<jcs-home>/conf/override/as400/
```

2. Restart the JCS.

Note: See the *Identity Manager JCS Implementation Guide* for more information on "override" connector.xml files.

Connector.XML File

The following is an example of the connector.xml file to be copied to <jcs-home>/conf/override/as400/:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE beans (View Source for full doctype...)>
<beans default-lazy-init="false" default-autowire="no" default-dependency-check="none">
  <bean class="com.ca.jcs.ImplBundle" parent="as400" lazy-init="default" autowire="default" dependency-
check="default">
    <!--
      Note that all settings here will override values in the conf/connector.xml
      within the jcs-connector-*.jar, but the JCS will still run fine if this file
      is deleted entirely as it will fallback to the version within the jcs-connector-*.jar.
    -->
    <property name="defaultConnectorConfig">
      <bean class="com.ca.jcs.as400.AS400ConnectorConfig" lazy-init="default" autowire="default" dependency-
check="default">
        <!-- When cascadingDelete is set to true, all objects owned by the user profile will be deleted as well while the
user --
        >
        <!-- is deleted. -->
        <!-- If cascadingDelete is set to false, the default behavior is used that is the user profile will NOT be deleted if --
        >
        <!-- it owns objects. -->
        <property name="cascadingDelete">
          <value>false</value>
        </property>
        <property name="poolConfig">
          <bean class="com.ca.jcs.cfg.GenericObjectPoolConfigBeanWrapper" lazy-init="default" autowire="default"
dependency-      check="default">
            <!--
              refer to Jakarta Commons Pool documentation

              http://jakarta.apache.org/commons/pool/apidocs/org/apache/commons/pool/impl/GenericObjectPool.html
              for complete listing of configurable settings.
            -->
            <!--
              The maximum number of active connections that can be allocated from this
              pool at the same time, negative for no limit.
            -->
            <property name="maxActive">
              <value>10</value>
            </property>
            <!--
              The minimum number of connections that can remain idle in the pool, without extra ones
              being created, or zero to create none. Best to create it on demand instead.
            -->
            <!-- <property name="minIdle"><value>1</value></property> -->
            <!--
```

The maximum number of connections that can remain idle in the pool, without extra ones being released, or negative for no limit.

-->

```
<property name="maxIdle">
```

```
<value>4</value>
```

```
</property>
```

<!-- The maximum number of milliseconds to wait for connection to be returned to the pool. -->

```
<!-- <property name="maxWait">-1</property> -->
```

<!--

The minimum amount of time an object may sit idle in the pool before it is eligible for eviction.

10 * 60 * 1000 = 10 minutes

-->

```
<property name="minEvictableIdleTimeMillis">
```

```
<value>600000</value>
```

```
</property>
```

<!--

The number of milliseconds to sleep between runs of the idle object evictor thread.

1 * 60 * 1000 = 2 minutes

-->

```
<property name="timeBetweenEvictionRunsMillis">
```

```
<value>60000</value>
```

```
</property>
```

```
</bean>
```

```
</property>
```

```
<property name="validators">
```

```
<bean class="com.ca.jcs.cfg.MetaPluginConfigSuite" lazy-init="default" autowire="default" dependency-check="default">
```

```
<property name="typeToPluginMap">
```

```
<map>
```

```
<entry key="FLEXI_STR:suspendState">
```

```
<bean class="com.ca.jcs.validator.attr.MapAttributeValidator" lazy-init="default" autowire="default" dependency-check="default">
```

```
<constructor-arg>
```

```
<value>>false</value>
```

```
</constructor-arg>
```

```
<constructor-arg>
```

```
<ref bean="suspendStateMap" />
```

```
</constructor-arg>
```

```
</bean>
```

```
</entry>
```

```
</map>
```

```
</property>
```

```
</bean>
```

```
</property>
```

```
<property name="converters">
```

```
<bean class="com.ca.jcs.cfg.MetaPluginConfigSuite" lazy-init="default" autowire="default" dependency-check="default">
```

```
<property name="typeToPluginMap">
```

```
<map>
<entry key="FLEXI_STR:suspendState">
<bean class="com.ca.jcs.converter.attr.MapAttributeConverter" lazy-init="default" autowire="default"
dependency-check="default">
<constructor-arg>
<value>false</value>
</constructor-arg>
<constructor-arg>
<ref bean="suspendStateMap" />
</constructor-arg>
</bean>
</entry>
</map>
</property>
<!-- enable the functionality that convert the multi-valued attributes to String arrays -->
<property name="propertyPluginConfigs">
<list>
<bean class="com.ca.jcs.cfg.MetaPluginConfig" lazy-init="default" autowire="default" dependency-
check="default">
<property name="pluginClass">
<value>com.ca.jcs.converter.meta.StringArrayPropertyConverter</value>
</property>
<property name="pluginConfig">
<bean class="com.ca.jcs.converter.meta.ConverterConfig" lazy-init="default" autowire="default"
dependency-check="default">
<property name="activeByDefault">
<value>true</value>
</property>
<property name="order">
<value>100</value>
</property>
</bean>
</property>
<property name="metadataPropNames">
<list>
<value>isMultiValued</value>
</list>
</property>
</bean>
<bean class="com.ca.jcs.cfg.MetaPluginConfig" lazy-init="default" autowire="default" dependency-
check="default">
<property name="pluginClass">
<value>com.ca.jcs.converter.meta.BoolPropertyConverter</value>
</property>
<property name="pluginConfig">
<bean class="com.ca.jcs.converter.meta.BoolPropertyConverter$BoolConfig" lazy-init="default"
autowire="default" dependency-check="default">
<property name="activeByDefault">
<value>true</value>
```

```

    </property>
    <property name="numeric">
      <value>false</value>
    </property>
    <property name="order">
      <value>100</value>
    </property>
  </bean>
</property>
</bean>
<bean class="com.ca.jcs.cfg.MetaPluginConfig" lazy-init="default" autowire="default" dependency-
check="default">
  <property name="pluginClass">
    <value>com.ca.jcs.converter.meta.JavaObjectPropertyConverter</value>
  </property>
  <property name="pluginConfig">
    <bean class="com.ca.jcs.converter.meta.JavaObjectPropertyConverter$Config" lazy-init="default"
autowire="default" dependency-check="default">
      <property name="activeByDefault">
        <value>true</value>
      </property>
      <property name="order">
        <value>200</value>
      </property>
    </bean>
  </property>
</bean>
<!--
  Specifying no metadata property names
  means checked against all properties,
  where plug-in constructor decides on
  relevance
-->
</bean>
</list>
</property>
</bean>
</property>
<property name="validateToConnector">
  <value>true</value>
</property>
<property name="validateFromConnector">
  <value>false</value>
</property>
<property name="caseSensitive">
  <value>true</value>
</property>
<!--
  <property name="typeToPluginMap">
    <map>

```

```
        <entry key="TIME">
            <bean class="com.ca.jcs.converter.attr.XMLSQLTimeAttributeConverter"/>
        </entry>
    </map>
</property>

->
<property name="containerBaseObjectClass">
    <value>eTAS4Directory</value>
</property>
<property name="virtualContainers">
    <list>
        <bean class="com.ca.jcs.VirtualContainerConfig" lazy-init="default" autowire="default" dependency-
check="default">
            <property name="name">
                <value>Accounts</value>
            </property>
            <property name="objectClass">
                <value>eTAS4AccountContainer</value>
            </property>
            <property name="namingAttribute">
                <value>eTAS4AccountContainerName</value>
            </property>
            <property name="containedObjectClasses">
                <list>
                    <value>eTAS4Account</value>
                </list>
            </property>
        </bean>
        <bean class="com.ca.jcs.VirtualContainerConfig" lazy-init="default" autowire="default" dependency-
check="default">
            <property name="name">
                <value>Groups</value>
            </property>
            <property name="objectClass">
                <value>eTAS4GroupContainer</value>
            </property>
            <property name="namingAttribute">
                <value>eTAS4GroupContainerName</value>
            </property>
            <property name="containedObjectClasses">
                <list>
                    <value>eTAS4Group</value>
                </list>
            </property>
        </bean>
    </list>
</property>
</bean>
```

```
</property>
</bean>
</beans>
```

OS/400 Security Requirements

The OS/400 Connector issues remote commands to the endpoint system to manage accounts. The managing user profile must have permission to issue remote commands for creating, reading, modifying, and deleting accounts. Areas of security to consider include, special authorities of the managing account (*SECADM is mandatory), exit programs implementing security, and authorization to user profiles.

OS/400 Groups

Identity Manager lets you create and maintain OS/400 groups using the Endpoint type task view. Use the OS/400 Group property sheet when managing your groups.

When a new group is defined, you should perform another exploration on the endpoint so Identity Manager has an updated group list.

Deleting Account Members from Groups

Account members cannot be deleted from a group if that group is designated as the primary group. You must remove the group from the account member . For example, ProvisioningGroup has two account members, Prov1 and Prov2, and ProvisioningGroup is the primary group of Prov1. Prov2 has a primary group FinancialGroup and a supplement group called ProvisioningGroup. If you try to delete Prov1 and Prov2 from ProvisioningGroup, only Prov2 is removed successfully. Prov1 remains as an account member of ProvisioningGroup.

OS/400 Directory Entry Names

When an account or group is created, a directory entry is created to store personal information about the user. Previously, the directory entry name was assumed to be the same as the user profile name. The attributes can now be set independently. If the Directory Entry Name is not specified, then a directory entry is not created for that user and many attributes cannot be set. Directory entry names must be unique across accounts and groups.

Changing Connection Settings

The connection settings associated with each endpoint cannot be changed using the Endpoint property sheet. To change incorrect connection settings, follow these steps:

1. Right-click the endpoint name.
The context sensitive menu appears.
2. Select Custom..., Change Admin Password.
The Change Password Dialog appears.
3. Fill in the dialog and select OK.
The dialog closes.

After the connection settings are changed, they are verified by attempting a connection to the OS/400 machine. The new settings are only saved if the connection is successful.

Conventions

Use the following OS/400 conventions in your etutil commands:

- The endpoint type name (eTNamespaceName) is OS400
- The endpoint type prefix is AS4. Therefore, the OS/400 class names are:
 - eTAS4Directory for an endpoint
 - eTAS4PolicyContainerName for an account template container
 - eTAS4Policy for an account template

OS/400 Native Program Exits

The Java OS/400 Connector supports Native Program Exits in the same way as the eTrust Admin 8.1 SP2 OS/400 Connector did with the following limitations:

- Only one parameter of the Command Call format can be specified in Program Exits.
- The Program Exits can target only account objects, not groups.

Note: The JCS provides a Scripting Style Processor interface for connectors. You can write code in the JavaScript scripting language to add extra logic to, or change the behavior of the OS/400 connector's operations. This approach is much more powerful than the C++ OS/400 Connector's Native Program Exits approach because you can access the full operation's details and write whatever you want to achieve for both account and group objects.

An example of this approach follows:

To change the description for each new account to the value of 'To demo scripting program exit concept works', use the `conf/as4script_opbindings.xml` file within the OS/400 Connector's archive file: `<jcs-home>/lib/jcs-connector-as400.jar`. Uncomment the `"staticMethodScriptStyleMetaDataFile"` in `<jcs-home>/conf/override/as400/connector.xml` and restart the JCS to turn on this behaviour.

See the *JCS Programming Guide* for more information on scripting style programming.

OpenVMS Connector

The OpenVMS Connector lets you administer accounts on OpenVMS systems and provides a single point for all user administration by letting you do the following:

- Register endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage OpenVMS accounts using OpenVMS-specific account templates
- Change account passwords and account activations in one place
- Synchronize global users with their provisioning roles or synchronize global users' accounts with their account templates
- Assign an OpenVMS account template to each of your OpenVMS endpoints
- Use the default endpoint type account template to create accounts with the minimum level of security needed to access an OpenVMS endpoint
- Generate and print reports about OpenVMS accounts

OpenVMS Installation and Configuration

OpenVMS is managed using the Connector Only installation process. Click [here](#) (see page 24) to view the installation requirements for this process.

The following sections describe what is needed to configure the OpenVMS Connector.

OpenVMS Disk Space and Environment Requirements

During the installation, your system needs 85,000 blocks of temporary disk space besides the amounts needed for installed files:

Files	Blocks of Disk Space
Alpha and Itanium systems	1,550 blocks
VAX systems	1,450 blocks
Log files and temporary files on the ETA\$ROOT disk	10,000 blocks

Your OpenVMS environment must be running OpenVMS, versions 7.2 or 7.3 on VAX, versions 7.2, 7.3, or 8.2 on Alpha, or version 8.2 on Itanium.

How to Install the OpenVMS Agent

The OpenVMS Connector requires that the OpenVMS Agent be installed to work.

To install the OpenVMS Agent, follow this procedure:

1. Locate the Identity Manager Provisioning Components download or other media.
2. Start the Product Explorer.
3. Select Install Products, Endpoint Agent Software
4. Choose Agent for OpenVMS
5. Follow the onscreen instructions to complete the installation.

You can install the OpenVMS Agent in one of the following ways:

- From a Windows machine
- From an OpenVMS system

Install the OpenVMS Agent From a Windows Machine

To install the OpenVMS Agent from a Windows machine, follow this procedure.

From an FTP utility

1. Transfer the following installation files from the \OpenVMS folder on the CD to your OpenVMS system.

- For Alpha systems, transfer the following files:

ETA081.ZIP
SETUP_AXP.EXE
UNZIP_ALPHA.EXE

- For Itanium systems, transfer the following files:

ETA081.ZIP
SETUP_IA64.EXE
UNZIP_IA64.EXE

- For VAX systems, transfer the following files:

ETA081.ZIP
SETUP_VAX.EXE
UNZIP_VAX.EXE

Note: Ensure that you select the binary transfer option in the ftp utility.

2. From a TELNET utility, log on to your OpenVMS system as the system account using the TELNET Windows tool.
3. Set the default directory to the directory where you copied the files.
4. Issue one of the following commands to begin the installation program.

- For Alpha systems, enter:

\$ RUN SETUP_AXP

- For Itanium systems, enter:

\$ RUN SETUP_IA64

- For VAX systems, enter:

\$ RUN SETUP_VAX

Install the OpenVMS Agent From an OpenVMS System

To install the OpenVMS agent from an OpenVMS system, follow this procedure.

Using the DCL MOUNT command

1. Mount the CD:

```
$ MOUNT/OVER=ID dev
```

Dev

Specifies the CD-ROM drive where the CD is mounted.

Example: \$ MOUNT/OVER=ID DKA400

2. Ensure that the CD is accessible:

```
$ SHOW DEVICE dev
```

Dev

Specifies the CD-ROM drive where the CD is mounted.

Example: \$ SHOW DEVICE DKA400

If the CD is properly mounted, the volume label appears as ETRADMWIN.

3. Point to the CD:

```
$ SET DEF dev:[OPENVMS]
```

Dev

Specifies the CD-ROM drive where the CD is mounted.

Example: \$ SET DEF DKA400[OPENVMS]

4. Begin the installation program.

- For Alpha systems, enter:

```
$ RUN SETUP_AXP
```

- For Itanium systems, enter:

```
$ RUN SETUP_IA64
```

- For VAX systems, enter:

```
$ RUN SETUP_VAX
```

5. Dismount the CD before removing it from the CD-ROM:

```
$ DISMOUNT dev
```

Dev

Specifies the CD-ROM drive where the CD is mounted.

Example: \$ DISMOUNT DKA400

How to Configure and Start the OpenVMS Agent on your OpenVMS System

Once the OpenVMS agent has been installed on your OpenVMS system, you must:

- Configure the agent
- Start the agent

Configure the Agent

To configure the agent on your OpenVMS system, issue the following command to define the symbols and logical names for Identity Manager:

```
S@SYS$STARTUP:ETA$STARTUP
```

Automatically Start the Agent

To automatically start the agent when you start your OpenVMS system, add the following line to the SYS\$STARTUP:SYSTARTUP_VMS.COM file:

```
$@SYS$STARTUP:ETA$STARTUP.COM
```

Define a CAFTHOST Symbolic Name

To define a CAFTHOST symbolic name you must add the following command to your LOGIN.COM file:

```
CAFTHOST :==$CAPOLY$BIN:CAFTHOST.EXE
```

and then logon to your OpenVMS system again.

Configure the CAM and CAFT Service--VMS

CAFTHOST must be configured to recognize the Provisioning Server and the Identity Manager clients. To configure CAFTHOST, follow these steps:

1. Issue the following command:

```
$ CAFTHOST -a Windows_node_name
```

Windows_node_name

Represents the name of the Provisioning Server.

Note: If the Provisioning Server is networked using DHCP or you do not use DNS for name resolution, the network name is not recognized. Under these conditions, use the TCP/IP address for the Windows node name or add a Windows node entry in the local hosts file on your OpenVMS system.

2. Issue the following command:

```
$ CAFTHOST -I
```

How to Manage the CAM and CAFT Service

To manage the CAM and CAFT Service on your OpenVMS system, use the view, start, and stop commands.

View the CAM and CAFT Service

The CAM and CAFT Service is a daemon process. To view this process, enter the following command at the DCL prompt:

```
$ SHOW SYSTEM /PROCESS=CAM$CA*
```

The CAM\$CAM and CAM\$CAFT processes appear. Sample processes follow.

```
$ SHOW SYSTEM/PROC=CAM$CA*
```

```
OpenVMS V7.1-2 on node MICKEY 28-JUN-2000 13:41:40.28 Uptime 103 18:28:45
```

Pid	Process Name	State	Pri	I/O	CPU	Page fts	Pages
212005E2	CAM\$CAM	LEF	4	1841622	0 00:16:55.71	488	83
2120182F	CAM\$CAFT	LEF	3	2131410	0 00:30:28.75	702	142

Start the CAM and CAFT Service

To start the CAM and CAFT Service, enter the following command at the DCL prompt:

```
@SYS$STARTUP:CAM$STARTUP
```

Stop the CAM and CAFT Service

To stop the CAM and CAFT Service, enter the following command at the DCL prompt:

```
@SYS$STARTUP:CAM$SHUTDOWN
```

OpenVMS Sample Installation Log File

The following code represents a sample installation log file that you can follow as you install the OpenVMS Agent.

```
$ r SETUP_AXP
UnZipSFX 5.20 of 30 April 1996, by Info-ZIP (Zip-Bugs@wku.edu).
  inflating: ca$setup.com
  inflating: ca$setup.ini
Platform type : ALPHA
OS version   : V8.2
Setup location: OVMSQA$DKA100:[KVYDM01.ETA_NEW_KIT.KIT]
Product Kit   : OVMSQA$DKA100:[KVYDM01.ETA_NEW_KIT.KIT]ETA081.ZIP;1
* Enter the device and directory to unzip the save-sets, 85000 blocks of temp
space needed [sys$common:[CA$SETUP]]:
%CREATE-I-EXISTS, SYS$COMMON:[CA$SETUP] already exists
UNZIP location: OVMSQA$DKA100:[KVYDM01.ETA_NEW_KIT.KIT]UNZIP_ALPHA.EXE;1
Unzipping installation media ... Please be patient ...
```

OpenVMS Software Product Installation Procedure V8.2

It is 23-MAY-2005 at 14:27.

Enter a question mark (?) at any time for help.

%VMSINSTAL-W-ACTIVE, The following processes are still active:

TCPIP\$FTP_1

TCPIP\$XDM_1

KVYDM01

* Do you want to continue anyway [NO]? YES

* Are you satisfied with the backup of your system disk [YES]?

The following products will be processed:

ETA V8.1

Beginning installation of ETA V8.1 at 14:27

%VMSINSTAL-I-RESTORE, Restoring product save set A ...

%VMSINSTAL-I-RESTORE, Restoring product save set Y ...

(C) 2000-2005, Computer Associates International, Inc. All Rights Reserved.

* Do you want this software to start right after the installation [YES]?

* Do you want to run the IVP after the installation [YES]?

* Do you want to purge files replaced by this installation [YES]?

This product needs a TCP/IP network stack installed and started
on this node. You can use TCPWARE, MULTINET or Compaq TCP/IP Services.

The TCP/IP stack TCPIP is up and running...

Copyright 1995-1997, Compaq Computer Corporation.
All rights reserved. Reproduction or translation
without express permission is prohibited.

```
%VMSINSTAL-I-RESTORE, Restoring product save set Z ...
Providing ETA agent files

Providing Startup, Shutdown, Installation Verification
and Deinstallation procedures ...
%VMSINSTAL-W-NOFILE, New file ETA$SHUTDOWN.COM does not exist.
%VMSINSTAL-E-BADSPEC, File spec SHUTDOWN.COM cannot be parsed

Installing Unicenter Message Service (CAM/CAFT)
Shutdown CAM Daemon...
caftabort: server aborted.
Shutdown CAFT Daemon...
camabort: server aborted.
%CREATE-I-EXISTS, SYS$COMMON:[CAI$REGISTRY.] already exists
%NONAME-F-NOMSG, Message number 00000004
%VMSINSTAL-I-MOVEFILES, Files will now be moved to their target directories...
Copyright (c) 2005, Computer Associates International, Inc. All rights reserved

Starting the CA Messaging Component (CAM)...
Starting CAM Server...
%RUN-S-PROC_ID, identification of created process is 0000027F
Starting CAFT Server...
%RUN-S-PROC_ID, identification of created process is 00000280

(C) 2000-2005, Computer Associates International, Inc. All Rights Reserved.

eTrust Admin Remote Agent version 8.1 IVP starting

Testing the agent...

Trace level is now 0
Trace indentation gap is now 3
%NONAME-W-NOMSG, Message number 00000000

%ETA-S-OK eTrust Admin Remote Agent version 8.1 IVP completed

Installation of ETA V8.1 completed at 14:29
Adding history entry in VMI$ROOT:[SYSUPD]VMSINSTAL.HISTORY
Creating installation data file: VMI$ROOT:[SYSUPD]ETA081.VMI_DATA

VMSINSTAL procedure done at 14:29

%DELETE-I-FILDEL, SYS$COMMON:[000000]CA$SETUP.DIR;1 deleted (18 blocks)
%DELETE-I-FILDEL, USR$SYSROOT:[KVYDM01.ETA_NEW_KIT.KIT]CA$SETUP.INI;1 deleted (18 blocks)
$
```

OpenVMS Support for FIPS and IPv6

For release r12 of Identity Manager, the OpenVMS Connector does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Endpoint Overview

OpenVMS users are called *accounts* in Identity Manager. You manage your accounts in secured databases or security systems called endpoints. The term *OpenVMS endpoint* traditionally refers to a collection of files. It has a physical location similar to a file folder on a Windows system. You can move from endpoint to endpoint and view or edit the files in the endpoint.

Besides the traditional type of endpoint, OpenVMS also uses another type of endpoint called a *database* endpoint. OpenVMS systems store accounts and access permissions in database endpoints using the AUTHORIZE utility. This is the database endpoint that Identity Manager manages. Other database endpoints, such as OpenVMS Mail and Oracle OpenVMS, include account and access information specific to these applications only. These applications are not supported by Identity Manager.

Acquire an OpenVMS System

You must acquire the OpenVMS system before you can administer it with Identity Manager. When acquiring an OpenVMS system, perform the following steps from the Endpoint type task view:

1. Register the server as an endpoint in Identity Manager.

Use the OpenVMS Endpoint property sheet to register an OpenVMS system. During the registration process, Identity Manager identifies the OpenVMS system you want to administer and gathers information about it.

Note: Ping the node name from the Provisioning Server. If the ping is successful and your CAM and CAFT Services are configured correctly, then Identity Manager finds the OpenVMS node.

2. Explore the objects that exist in the endpoint.

After registering the server in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all OpenVMS accounts and groups. You can correlate the accounts with global users at this time or you can correlate them later.

3. Correlate the explored accounts to global users by choosing either of the following options:

- Use existing global users
- Choose this option when there are already global users in Identity Manager and you want to connect the existing global users to the OpenVMS accounts
- Create global users as needed

Choose this option when there are no global users and you want to populate Identity Manager from the OpenVMS accounts.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the user profile name with each existing global user name. If a match is found, Identity Manager associates the OpenVMS account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the user profile name with each existing global user's full name. If a match is found, Identity Manager associates the OpenVMS account with the global user. If a match is not found, Identity Manager performs the following step.
- c. Identity Manager associates the OpenVMS account with the [default user] object or a new global user is created depending on your choice.

Provisioning Roles and Account Templates

The OpenVMS Default Account Template, provided with the OpenVMS Connector, gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

If you use the OpenVMS Default Account Template without modifying it, be aware that:

- Identity Manager creates a unique UIC for each account using group 201 and assigns the next available member number in the group to the OpenVMS system.
- Identity Manager creates the default logon directory in the SYS\$SYSDISK device with an endpoint path the same as the unique name.

If these defaults do not meet your needs, you can always change the UIC group number or logon directly and save these changes to the account template.

Create Account Templates

The Default Account Template, provided with each connector, gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

To create an account template

1. Click the Provisioning Roles task button, select the connector's Account Template in the Object Type drop-down list box and click New.

The Account Template Property Sheet for the specified connector appears.

2. Complete the Account Template Property Sheet by:
 - a. Selecting an endpoint to populate the drop-down and group selection lists.
 - b. Selecting group memberships and other account settings.
 - c. Clicking OK.

A new account template is created for your connector.

VMS Program Exits

Program Exits (Common or Native) let you write software that executes during certain actions that Identity Manager carries out. Program exits extend the framework of Identity Manager and permit additional functionality that can change or augment the standard Identity Manager behaviors.

The OpenVMS Connector supports the use of Program Exits, which are incorporated as 'Common Exits'. Program Exits provide you with the capability to perform certain actions before or after Identity Manager creates, modifies, or deletes an account. These exits can be referenced either on the Endpoint property sheet to execute custom code in a single endpoint, or on the Account Template property page to execute custom code in multiple endpoints. Actions can include native SI (Security Integrator) commands that modify account privileges or access to resources on the OpenVMS system.

For more detailed information about how to write program exits, see the *Programming Guide for Provisioning*. If you want further information on using SI commands, see the SI SDK.

HP NonStop SAFEGUARD Connector

The HP NonStop SAFEGUARD Connector lets you administer all your accounts and groups on HP NonStop Himalaya machines.

Note: SAFEGUARD is an optional HP security product and must be purchased through HP.

The HP NonStop SAFEGUARD Connector provides a single point for all user administration by letting you do the following:

- Register endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage SAFEGUARD accounts using HP NonStop SAFEGUARD-specific account templates
- Change account passwords and control account activations in one place
- Synchronize global users with their roles or synchronize global users' accounts with their account templates
- Assign a SAFEGUARD policy to each of your SAFEGUARD endpoints
- Use the default Endpoint Type policy to create accounts with the minimum level of security needed to access a SAFEGUARD directory
- Create and manage SAFEGUARD groups
- Generate and print reports about SAFEGUARD accounts and groups.

This connector is managed using the Connector and agent installation process. For more information and requirements, click [here](#) (see page 21).

This connector can also be managed using the [Connector and C++ Server](#) (see page 23) installation process as well.

Installing the HP NonStop SAFEGUARD Agent

To install the Connector, perform the following steps:

1. Transfer the file to the HP NonStop SAFEGUARD system.
2. Configure you system.

Step 1. Transfer the Files to the HP NonStop SAFEGUARD System

On each HP NonStop SAFEGUARD Server, you should install the HP NonStop SAFEGUARD remote agent by performing the following procedure.

From the Identity Manager Installation Wizard

1. Start the Identity Manager HP NonStop SAFEGUARD agent.
2. Enter the following information:
 - The TCP/IP host name of the HP NonStop SAFEGUARD Server
 - The password for SUPER.SUPER
 - The name of the disk volume where the agent will be installed

Note: The installation program uses this information to transfer the required files to the HP NonStop SAFEGUARD system.

Step 2. Configure Your System

Perform the following procedures to configure your HP NonStop SAFEGUARD system.

As **SUPER.SUPER**

1. Log on to your HP NonStop SAFEGUARD system
2. Issue the following commands at the TACL prompt:

- VOLUME *instvol*.CARELSCR

instvol

The name of your installation volume.

- RUN SFGSETUP

SFGSETUP prompts you for information required to configure TCP/IP for communication with the administrative server. This information includes:

- The name of the TCP/IP services file
- The name of the host file
- The TCP/IP host name of the HP NonStop SAFEGUARD system
- The TCP/IP host name of the administrative server

Note: The installation asks for the name of the host file only if a Domain Name Server (DNS) is not used. If the installation detects more than one TCPIP process, it prompts for the name of the TCPIP process.

For example, SFGSETUP prompts with the following:

- Enter the fully qualified name of the SERVICES file: \$SYSTEM.ZTCPIP.SERVICES
- Do you want to run INSTALL now ([Y]es/No)? Y
- Do you want to install all functions ([Y]es/No)? Y
- Do you want the Compaq NSK SAFEGUARD option to use a DNS ([Y]es/No)? N
- Enter the fully qualified name of the hosts file: \$SYSTEM.ZTCPIP.HOSTS
- Enter the local TCP/IP node name for \CLXA: hpNonStop_server
- Enter the remote TCP/IP node name of the eTRUST server responsible for managing this node: etrust_server

Run the Agent Processes

To manage your HP NonStop SAFEGUARD endpoints, you must run the agent processes. Perform the following procedures to start the agent processes:

From the TACL prompt on your HP NonStop SAFEGUARD system

1. Log on as SUPER.SUPER.
2. Issue the following commands:

```
instvol.CARELSCR.CONNECT
```

instvol

The disk volume where the HP NonStop SAFEGUARD agent is installed.

```
CSTART ALL
```

3. To stop the agent processes, type the following command:

```
CASHUTDOWN ALL
```

Common Communication Interface (CCI) Installation for HP NonStop Safeguard

The CCI component must be installed to enable the HP NonStop Safeguard connector to communicate with endpoint targets. You can install the CCI component by installing either Unicenter or CCISA (CCI Stand Alone). Follow these steps to install and configure CCISA:

1. Locate the Identity Manager Provisioning Components download or other media.
2. Start the Product Explorer.
3. Select Install Products, Infrastructure
4. Choose CCISA
5. Follow the onscreen instructions to complete the installation.
6. In the ~\Program Files\CA\Identity Manager\CCI\Caiuser folder, create a CCIRMTD.RC file with the following lines:

```
local = <Host Name> <Host Name> 32768 STARTUP
remote = <Host Name> <Host Name> 32768 STARTUP PORT=<Port Number> RETRY=1
```

The first *<Host Name>* value in the local or remote line can be the IP address.

Note: You must set a port number. If the port has not been changed on the target, the port number is 1721.

7. To improve CCI performance, add the following key to the Windows Registry:

```
HKEY_LOCAL_MACHINE\Software\ComputerAssociates\CA
Unicenter\CurrentVersion\Components\COM\ENV\OPT\RMTHOSTS
```

8. Under the RMTHOSTS key, create a string value named Default, and set the HP NonStop Safeguard host name in this string value.

Note: If you do not add this key to the Registry, the CCI response time will be very long.

9. Stop and then restart the CCI service.
10. Open a command prompt and use the following CCI commands to start CCI and test the connection:
 - To start CCI, enter `ccicntrl start`.
 - To display the CCI status, enter `ccicntrl status`.
 - To test the CCI connection, enter `ccii`.
11. To stop CCI after testing the connection, enter `ccicntrl stop`.

NSK Support for FIPS and IPv6

For release r12 of Identity Manager, the NSK Connector does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire an HP NonStop SAFEGUARD Machine

You must acquire the HP NonStop SAFEGUARD machine before you can administer it with Identity Manager. When acquiring an HP NonStop SAFEGUARD machine, perform the following steps.

From the Endpoint Type task view

1. Register the machine as an endpoint in Identity Manager.

Use the HP NonStop SAFEGUARD Endpoint Property Sheet to register an HP NonStop SAFEGUARD machine. During the registration process, Identity Manager identifies the HP NonStop SAFEGUARD machine you want to administer and gathers information about it.

2. Explore the objects that exist on the endpoint.

After registering the machine in Identity Manager, you can Explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all HP NonStop SAFEGUARD objects. You can Correlate the accounts with global users at this time, or you can wait.

3. Correlate the Explored accounts with global users.

When you Correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the account name with each existing global user name. If a match is found, Identity Manager associates the HP NonStop SAFEGUARD account with the global user. If a match is not found, Identity Manager performs the following step.
- b. Identity Manager attempts to match the account name with each existing global user's full name. If a match is found, Identity Manager associates the HP NonStop SAFEGUARD account with the global user. If a match is not found, Identity Manager performs the following step.
- c. If the Create Global Users as Needed button is checked, Identity Manager creates a new global user and then associates the HP NonStop SAFEGUARD account with the global user. If the Create Global Users as Needed button is unchecked, Identity Manager performs the next step.
- d. Identity Manager associates the HP NonStop SAFEGUARD account with the default user object.

NSK Accounts

The HP NonStop SAFEGUARD Connector manages GUARDIAN users and SAFEGUARD alias accounts. Use the HP NonStop SAFEGUARD User Property Sheet when managing your accounts.

Accounts appear in the Provisioning Manager with the account name and user ID fields showing the GUARDIAN user name. If the account name and the user ID are identical, the account is a GUARDIAN user. Otherwise, the account is a SAFEGUARD alias. The user ID must be a valid GUARDIAN user name.

Note: SAFEGUARD alias names are case-sensitive. GUARDIAN user names are not case-sensitive.

NSK Groups

You can create and maintain SAFEGUARD groups using the Endpoint Type task view. Use the SAFEGUARD Group Property Sheet when managing your groups.

You can define your GUARDIAN users and SAFEGUARD aliases to any group that exists on your HP NonStop SAFEGUARD system.

A GUARDIAN user is defined by a group name and a user name. A group name has a group number, and a user name has a user number. When defining a GUARDIAN user, you must make sure that the group number and group name match. The first user name defined in a GUARDIAN group determines the name of that group.

For example, if you create a user named ETRUST.USER, and the user ID is 20,100, by default the GUARDIAN user name ETRUST.USER belongs to the group ETRUST and has the number 20, and the user has 100 as user number. All user names for group 20 must then be in the form of ETRUST.*name*. GUARDIAN group numbers can range from 0 - 255.

SAFEGUARD allows groups to be defined with numbers up to 32767. SAFEGUARD and Identity Manager allow accounts to be associated with multiple groups and defines a primary group for any given account. When a new group is defined, you should perform another Exploration on the endpoint so Identity Manager has an updated group list.

Chapter 13: Middleware Connectors

This section contains the following topics:

[Embedded Entitlements Manager Connector](#) (see page 439)

The following connector is considered to be a middleware connector:
Embedded Entitlements Manager (EEM)

The information in this chapter describes how to install and manage this connector.

Embedded Entitlements Manager Connector

The Embedded Entitlement Manager (EEM) Connector lets you create management interfaces for EEM servers and provides a single point for all user administration by letting you do the following:

- Administer two types of EEM applications: the built-in "Global" application and the user-defined application
- Add accounts and account containers to any level of the tree
- Delete accounts and account containers from any level of the tree
- Modify and search accounts from any level of the tree
- Add groups to the accounts
- Modify user attributes for accounts in user-defined applications
- Create, search, modify, and delete access policies
- Create, search, modify, and delete calendars
- Add, search, and delete groups
- Search resource classes

The EEM Connector supports multiple, simultaneous connections to different EEM servers and multiple applications on the same server.

Note: The EEM Connector refers to the EIAM Connector for this release.

EEM Installation

This connector is managed using the Connector and C++ Server installation process.

Note: For more information and requirements, see *Connector and C++ Connector Server Installation* (see page 23).

EEM Support for FIPS and IPv6

For release r12 of Identity Manager, the Embedded Entitlements Manager Connector does not support FIPs or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire an EEM Server Machine

You must acquire the EEM server machine before you can administer it with Identity Manager.

From the Endpoint Type task view

1. Register the machine as an endpoint in Identity Manager.

Use the EEM Endpoint property sheet to register an EEM server machine. During the registration process, Identity Manager identifies the EEM server machine you want to administer and gathers information about it.

Then choose the application name and specify the name and password for the management account that has been configured on the EEM Backend Server.

2. Explore the accounts that exist on the endpoint.

After registering the machine in Identity Manager, you can explore its accounts. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all EEM accounts. You can correlate the accounts with global users at this time, or you can wait to correlate them.

3. Correlate the explored accounts with global users.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the username with each existing global user name. If a match is found, Identity Manager associates the EEM account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the account name with each existing global user's full name. If a match is found, Identity Manager associates the EEM account with the global user. If a match is not found, Identity Manager performs the next step.
- c. If the Create Global Users as Needed button is checked, Identity Manager creates a new global user and associates the EEM account with the global user. If the Create Global Users as Needed button is unchecked, then Identity Manager performs the next step.
- d. Identity Manager associates the EEM account with the [default user] object.

EEM Account Templates

The EEM Connector account template can be associated with two types of EEM applications, (build-in and user-defined). If the associated directory is the Global application, the "Application Properties tab in the account template is disabled, whereas, the Global Properties tab is disabled for user-defined application directories.

Since the Connector is a hierarchical Endpoint Type, the account container must be specified for the account template in order to create accounts in the right place.

Global and Application Endpoints are Managed Separately

No accounts can be added to the build-in Global application endpoint when adding accounts to user-defined application endpoints and conversely the opposite applies. You can now view which endpoint the accounts are being associated to on the Account Template Property Sheet.

EEM Accounts

When creating application users using the EEM Web User Interface (not recommended), all users will be created under the root folder regardless of what folder is specified, although the Web User Interface appears to display the user with the correct path. This indicates that such application users created using the EEM Web User Interface will all appear under the "EEM Accounts" container after exploration in the Provisioning Manager.

If you use Identity Manager or Safex to create users, all users will be created in the correct folders or they will be displayed correctly after an exploration.

EEM Access Policies

You can create and maintain EEM access policies using the Endpoint Type task view. Use the EEM Access Policies property sheet when defining and managing your access policies.

Any changes made in this object using native tools are kept in sync with the views in the Provisioning Server without requiring a re-exploration.

EEM Calendars

Calendars define dates and times that users can access system functions. Use the EEM Calendars property sheet to set user access times.

Any changes made in this object using native tools are kept in sync with the views in the Provisioning Server without requiring a re-exploration.

EEM Groups

You can create and maintain EEM groups using the Endpoint Type task view. Use the EEM Group property sheet when managing your groups.

Any changes made in this object using native tools are kept in sync with the views in the Provisioning Server without requiring a re-exploration.

Available Groups in Account Template Property Sheet

You can now view and choose from all the available groups, where to add group members to an account template.

EEM Resource Classes

You can view the resource classes for the EEM server.

Chapter 14: Business Application Connectors

This section contains the following topics:

[SAP Connector](#) (see page 446)

[Siebel Connector](#) (see page 461)

The following connectors are considered to be business application connectors:

- SAP
- Siebel CRM

The information in this chapter describes how to install and manage these connectors.

SAP Connector

The SAP Connector provides a single point for all user administration by letting you perform any of the following actions:

- Retrieve existing users from the SAP repository
- Display, create, modify, or delete a user
- Retrieve the existing authorization profiles from the SAP repository
- Display authorization profiles
- Assign or unassign an authorization profile to a user
- Retrieve the existing SAP roles from the SAP repository
- Display SAP roles
- Assign or unassign a SAP role to a user
- Register endpoints, explore them for objects to manage, and correlate their accounts with global users
- Create and manage SAP accounts using SAP-specific account templates
- Change account passwords and account activations in one place
- Assign a SAP account template to each of your SAP endpoints
- Use the default endpoint type account template to create accounts with the minimum level of security needed to access a SAP endpoint
- Generate and print reports about SAP accounts, SAP profiles, and SAP roles
- Manage SAP CUA environments

SAP Installation

The SAP Connector must be managed by the Java Connector Server installation process.

Click [here](#) (see page 26) for more information on this process.

SAP Prerequisites

The following are required for this connector.

- The SAP GUI must be installed on the machine where the Java Connector Server is installed if you want to use SAP Logon ID when creating SAP endpoints. You must define the SAP Logon ID with SAP Logon, which is installed together with the SAP GUI.
- The SAP JCO Library must be installed according to the instructions on the machine where the Java Connector Server is installed.

How to Install the SAP Connector with the Java Connector Server

Follow this procedure to install the SAP connector with the Java Connector Server.

To install the SAP Connector

1. Run the setup.exe program of the Provisioning Server installation package and select SAP option.
2. Run the setup.exe program of the JCS installation package to install the Java Connector Server with the SAP option.
3. Install the SAP GUI on the machine where JCS is installed using the SAP GUI installation instructions, if you want to use SAP Logon ID when creating SAP endpoints.
4. Install the SAP JCO library on the machine where JCS is installed. A SAP Service Marketplace Profile/Account is required for this step.

From <http://service.sap.com/connectors>

1. Select 'SAP Java Connector' to display the JCo overview.
2. Select 'Tools and Services' from the left-hand menu.
3. Enter the project description and agree to the license terms.
4. Download the appropriate version of SAP JCo Release 2.1.8.

Click [here](#) (see page 448) for instructions on installing the SAP JCO library on a SUN Solaris machine or [here](#) (see page 449) for instructions on installing the SAP JCO library on a Windows machine.

SAP JCO Library Installation on SUN Solaris

Important! You must install JCS before installing the SAP JCO library. The SAP JCO library is only to be installed on the JCS machine. After the installation of the SAP JCO library, you must restart JCS.

The following are the distribution packages that are available for the various JRE versions and hardware processors:

- sapjco-sun-2.1.8.tgz for a 32-bit JRE running on a 32- or 64-bit SUN SPARC processor
- sapjco-sun_64-2.1.8.tgz for a 64-bit JRE running on a 64-bit SUN SPARC processor
- sapjco-sunx86_64-2.1.8.tgz for a 64-bit JRE running on a 64-bit AMD or INTEL x86 processor

To install the SAP JCO Library on SUN Solaris

1. Copy the appropriate distribution package into an arbitrary directory {sapjco-install-path}
2. Change to the installation directory:
`cd {sapjco-install-path} [return]`
3. Extract the archive:
`gunzip sapjco-sun*2.1.8.tgz [return]`
`tar xvf sapjco-sun*2.1.8.tar [return]`
4. Copy the file sapjco.jar to the extlib directory of the JCS installation.
5. Copy the files librfccm.so and libsapjcorfc.so to the bin directory of the JCS installation.
6. Restart the JCS service by running the command `./JCS stop` and then `./JCS start` from within the bin directory of the JCS installation

SAP JCO Library Installation on Windows NT, Windows 2000, and Windows 2003

Important! You must install JCS before installing the SAP JCO library. The SAP JCO library is only to be installed on the JCS machine. Restart the JCS after the installation of the SAP JCO library.

The following are the distribution packages for the various JRE versions and hardware processors available:

- sapjco-ntintel-2.1.8.zip for a 32-bit JRE running on a 32-bit INTEL x86 or a 64-bit INTEL Itanium processor
- sapjco-ntia64-2.1.8.zip for a 64-bit JRE running on a 64-bit INTEL Itanium processor
- sapjco-ntamd64-2.1.8.tgz for a 64-bit JRE running on a 64-bit AMD or INTEL x86 processor

To install JCo for Windows

1. Unzip the appropriate distribution package into an arbitrary directory {sapjco-install-path}
2. Copy sapjcorfc.dll and librfc32.dll to the Windows system32 directory and replace any file that might already be there.
3. Copy the sapjco.jar file to the extlib directory of the JCS installation.
4. Restart the CA Identity Manager - Connector Server service.

SAP Migration Steps

The Java version of the SAP Connector (installed with the Java Connector Server) provides all of the functionality of the previous (eTrust Admin r8.1) C++ version of the SAP Connector with the added benefit of full CUA management, but there are a few things to consider when switching from the C++ version.

- When a SAP CUA master endpoint has already been acquired and explored, the endpoint will be managed as a CUA engine after the switch from the C++ connector to the Java connector. Since all existing SAP roles and account templates are still valid after the migration, existing admin account templates targeting the master directory are still usable. Existing managed objects are valid as well. *You must re-explore the endpoint to include the managed objects that exist on child systems.*
- When SAP CUA member endpoints have already been acquired and explored, they should be removed. Account templates pointing to these member endpoints should be pointing to the CUA master endpoint.

Note: Management of local account attributes (for example, default printer) according to the SCUA parameter is still possible by keeping the CUA member endpoint and managing these attributes through this endpoint.

- To add the SAP connector to an existing system:
 1. Run the Provisioning Server install to reconfigure and add the SAP connector.
 2. Run the JCS installer and select Register with the Provisioning Server.Doing this will route requests from the Provisioning Server to the JCS for the SAP endpoint type.
- **Important!** Before migrating from C++ to JCS, the following must be filled out and selected on the Endpoint Tab of the SAP Endpoint property sheet:
 - The check box for 'Use LogonID' must not be selected.
 - The application server name and number must be entered.

SAP Support for FIPS and IPv6

For release r12 of Identity Manager, the SAP Connector does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire a SAP Endpoint

SAP must be installed on the machine that you want to administer through the Identity Manager SAP connector.

From the Endpoint type task view

1. Register the SAP system as a SAP/R3 endpoint.

Use the SAP Endpoint property sheet to register a SAP system. During the registration process, the Provisioning Manager identifies the SAP system you want to administer and gathers information about it.

Note: You must provide a SAP user/password with administrator rights when registering a SAP system.

More information:

[Create a New User and SAP Role with Minimum Rights to Administer SAP](#) (see page 454)

2. Explore the objects that exist on the endpoint.

After registering the system in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all SAP Accounts, SAP Roles (except Generated Roles), and SAP Profiles (except Generated Profiles). You can correlate the accounts with global users at this time or later.

3. Correlate the explored accounts with global users.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the eTSAPAccountName with each existing global user name. If a match is found, Identity Manager associates the SAP account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the eTSAPLastName with each existing global user's full name. If a match is found, Identity Manager associates the SAP account with the global user. If a match is not found, Identity Manager performs the next step.
- c. Identity Manager associates the SAP account with the *default user* object.

Changed Passwords are Expired

The SAP Endpoint Property Sheet contains a parameter called Changed passwords are expired. When this field is checked, the user is prompted to change their password when they next log on. If the password has been propagated from a global user password change, the user is also prompted to change their password. This is the SAP recommended behavior for password changes and has been checked by default on all newly created endpoints. Click [here](#) (see page 459) for further details on account password management in a CUA environment, or [here](#) (see page 456) for account password management with stand-alone SAP systems

SAP Provisioning Roles and Account Templates

The SAPDefaultPolicy, provided with the SAP Connector, gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

Modify SAP Account Template Properties

To modify or view a SAP account template's properties, perform the following steps:

1. Click the Roles task button.
2. Select SAP Endpoint Account Template in the Object Type drop-down list box.
3. Search for the account template that you want.
4. Right-click on the account template in the list view and choose Properties from the pop-up menu.
5. Modify or view the properties in the SAP Account Template property sheet.
6. Click Apply.

Create a New User and SAP Role with Minimum Rights for Administration

To set the minimum authorization that a user should have to administrate a SAP system from Identity Manager, you must create a new SAP role.

Note: If you are administering a CUA environment, see the notes on CUA below.

To create a new user with a SAP role with minimum rights to administer SAP

1. Create a new communications user with no authorizations.
2. Create a new authorization role by using transaction *PFCG*.
3. On the descriptions tab, enter a meaningful description.
4. On the menu tab, copy the "Tools>Administration>**User Maintenance**" menu by selecting '*copy menus>from the SAP menu*'.
5. Select the '*Change Authorization Data*' button on the Authorizations tab:
 - Do not assign the role an organizational level
 - Manually add the authorization **S_RFC**. If you are using SAP Kernel 6.40 or later, you must also manually add the authorization **S_TABU_DIS**
 - Assign the full authorization for all trees by setting the authorization fields to '*'. All authorizations must be active (green light) before proceeding.
 - If necessary, drill down and manually set the 'Human Resources>Personnel Planning>Personnel Planning>**Plan Version**' to full authorization, '*'.
 - Generate the profile.
6. On the user tab, add the user ID of the previously created communications user and then perform a 'user comparison' to immediately assign the authorizations to the account.

Notes for SAP CUA

- You should perform the above steps on the CUA master system only.
- The communications user must be added to the CUA master system (Maintain User Properties>System Tab) **before** completing a user comparison during role creation.

Grace Interval on Logon Tab of SAP Account Template Property Sheet

When an account is created, you can choose a date from the Valid From and Valid To fields that indicate when the logon credentials are valid or you can specify the number of days from the Valid from date until the credentials become valid, by selecting the Grace Interval field and entering the number of days.

Valid From and Valid To are two SAP account template attributes that can be propagated to the associated accounts. Grace Interval is only valid at the time of account creation. For example, modification applies only for the SAP account template and not the accounts associated to it.

SAP Accounts

Use the SAP Account property sheet when managing your accounts.

Note: The extended attributes include the following:

- Name at birth
- Middle name initial
- Second academic title
- Name prefix
- Second prefix
- Name supplement
- Nickname
- Name format
- Internal mail
- Code
- Second family name
- Second given name

Note: Each attribute that may be specified through a rule string in an account template could possibly be truncated. A warning message will display the name of all the attributes that were truncated.

SAP Profiles

SAP Profiles grant access to the user for SAP transaction menus and other SAP objects.

Note: A SAP Profile differs from the Admin Profile, which is attached to the Global User and grants access to Identity Manager objects.

User License Contractual Type

The License Data tab enables you to manage the account license contractual type, selecting a value for the Contractual User Type.

Explore Non-Dialog Accounts Without Correlation

Non-dialog accounts are accounts that are used to run the batch process, remotely connecting from a foreign application. To prevent non-dialog accounts from being correlated to a global user, perform the following steps:

1. In the SAP Endpoint tab, check the "Only manage dialog accounts" checkbox prior to exploration and correlation.
2. Uncheck the "Only manage dialog accounts" checkbox, and explore the endpoint accounts without correlate.

Account Password Management

When connecting to a stand-alone SAP system, if not using a pre-expired password, the following occurs:

- **On Account creation:**

The password is pre-expired. You must change the password upon first logon.

- **On Account Modify**

The password is changed.

Note: With SAP Kernel 6.40, it is not possible to change the password on a locked account unless the endpoint is set to use pre-expired passwords. The account must be unlocked before the password change can be applied.

When using a pre-expired password, the following occurs:

- **On Account Creation and Modify**

The password is pre-expired. You must change the password upon first logon and first logon after the change.

SAP Conventions

Use the following SAP conventions in your etutil commands:

- The endpoint type name (eTNamespaceName) is SAP R3.
- The endpoint type prefix is SAP. Therefore, the SAP class names are the following:
 - eTSAPAccountContainer for a SAP user container
 - eTSAPAccount for a SAP user name
 - eTSAPProfileContainer for a SAP profile container
 - eTSAPProfile for a SAP profile name
 - eTSAPRoleContainer for SAP role container
 - eTSAPRole for a SAP role name
 - eTSAPPolicyContainer for a SAP account template container
 - eTSAPPolicy for a SAP account template

Managing SAP Central User Administration (CUA) Environments

The SAP connector lets you manage CUA environments. The following sections apply to CUA and how they are managed by this connector.

SAP (CUA) Management

CUA is a tool that can be used to manage SAP account on multiple SAP systems centrally on a single Master System. The Java-based SAP connector processes CUA master systems as a CUA engine.

Note: While using the new Java-based connector, it is not possible to manage a CUA Master system as a standalone.

The SAP Connector (Java) can manage all SAP systems that are part of a CUA. A new read-only field on the SAP Endpoint property page "CUA Status" displays the status of a SAP endpoint against CUA. When the SAP system is a CUA master, the field shows "CUA master system managed as a CUA engine".

Note: CUA management is only effective when the field distribution parameters using transaction SCUM are set to 'GLOBAL'.

As a CUA Engine Processing Mode

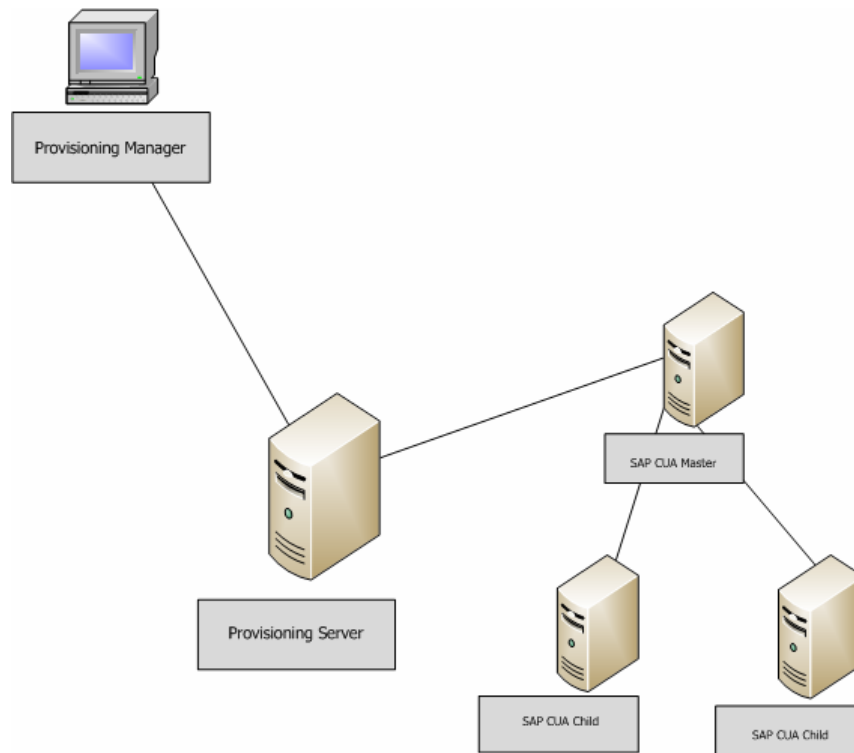
For example, a directory called CUAMAST (CUA master) can grant an account the following roles:

- role1
- CUACHI1/role3
- CUACHI2/role4

The following then occurs:

- An account is created on the CUA master directory by Identity Manager.
- The SAP CUA mechanism grants this account role1 privilege on the directory.
- The SAP CUA mechanism propagates the account to both CUACHI1 and CUACHI2 directories.
- The SAP CUA mechanism grants the account on CUACHI1 the role role3.
- The SAP CUA mechanism grants the account on CUACHI2 the role role4.

The following diagram shows the Java-based connector's As a CUA engine processing mode:



Account Password Management in CUA Environment

The following sections show how account password management is handled in a CUA environment.

Connecting to a CUA Master

When connecting to a CUA master system, if not using a pre-expired password, the following occurs:

- **On Account Creation for both CUA Master and CUA Child**

The password is pre-expired. You must change the password upon first logon.

- **On Account Modify**

CUA Master - The password is changed.

CUA Child - The password change is not distributed to child systems. Password management must be done locally.

Note: With SAP Kernel 6.40, an attempt to change the password of an account that does not reside on the Master system will return PASSWORD NOT ALLOWED.

When connecting to a CUA master system using a pre-expired password, the following occurs:

- **On Account Creation for both CUA Master and CUA Child**

The password is pre-expired. You must change the password upon first logon.

- **On Account Modify**

CUA Master - The password is pre-expired. You must change the password upon first logon after the change.

CUA Child - The password change is not distributed to child systems. Password management must be done locally.

Connecting to a CUA Child

We recommend using the connector to manage locally managed attributes of the account. To be able to change passwords when connecting to a child system, the distribution model of the initial password should be set to "proposal" using the SAP transaction SCUM.

Password Management in a CUA Environment

As password changes applied to the CUA Master System are not distributed to other CUA members, you will need to acquire separate SAP endpoints to the Child Systems to be able to manage account passwords on Child Systems. After creating a new account on the CUA Master System, you must re-explore and correlate the users container on the endpoint set up to manage such child systems. The passwords can then be managed by a modification to the Global User associated with these accounts, or directory to the accounts in these managed endpoints. This is a limitation imposed by SAP.

Remove an Account from the CUA Master System

Removing an account from the CUA Master System with the Java connector removes the account on the Master System as well as all the Child systems.

CUA Distribution Settings

For further details on the distribution parameters for fields within transaction SCUM, refer to the SAP Central User Administration documentation available at <http://service.sap.com>. It is important to be aware of the distribution settings within your CUA environment as some settings may end with unexpected results. In particular:

- When the distribution model for an attribute has been set to "Global", these attributes must be managed by the Provisioning Manager using the endpoint connecting to the CUA Master system.
- When the distribution model for an attribute has been set to "Local", the attribute can only be managed from the endpoint(s) connecting directly to each individual member system, regardless of its status within the CUA.
- When changing an attribute that conflicts with the distribution model, the modification attempted by the Connector Server may be ignored. In some cases, an error is returned. You should be aware of the distribution settings and manage accordingly.
- In these cases, the Provisioning Manager may not give a visual indication that the attribute change is permitted under the current distribution settings.
- Passwords cannot be managed as "Global", regardless of the distribution settings. Any changes applied to the password on a CUA Master system are not distributed to the child systems by design. Click [here](#) (see page 459) for information on password management using the Provisioning Manager.
- With the exception of the password management, we recommend that where possible, the distribution settings be set to "Global".

Siebel Connector

The Siebel Connector lets you manage user accounts on Siebel machines and provides a single point for all user administration by letting you do the following:

- Retrieve the existing user accounts (users,employees) positions, and responsibilities, organizations, internal divisions and views from a Siebel server. Entities from lists of values (LOV) are also retrieved; this happens only when LOV values are used in custom user account mapping.
- Display properties of each of these components
- Create and modify user accounts (users,employees).
- Associate user accounts with responsibilities
- Associate user accounts with positions
- Associate user accounts with organizations
- Remove associations between user accounts and responsibilities
- Remove associations between user accounts and organizations
- Remove associations between user accounts and positions
- Delete a user account. Since deletion of a user in Siebel is not recommended, Identity Manager lets you choose from the following behaviors:
 - Do nothing. Deletion is ignored. A user account disappears from Identity Manager but remains intact in Siebel.
 - Simulate suspension. Identity Manager changes the Siebel user's password and removes all user responsibilities (if possible, with used custom mapping and Siebel configuration).
 - Delete. Identity Manager deletes a user in Siebel. This is not recommended. A user ID (login name) of a deleted account can no longer be used in Siebel.
- Simulate user account suspension/resumption by removing/restoring user's responsibilities.
- Create, modify, and delete positions.
- Create, modify, and delete responsibilities.
- Associate responsibilities with views
- Associate responsibilities with organizations
- Remove associations between responsibilities and views
- Remove associations between responsibilities and organizations
- Create, modify, and delete internal divisions.
- Map up to 20 fields of a Siebel user to Identity Manager attributes

- Map up to 10 fields of exposed Siebel objects other than users, to Identity Manager attributes

Note: The Siebel Connector supports Siebel 7.5, 7.7, and higher.

Siebel Installation

This connector is managed using the Connector and C++ Server installation process.

Note: For more information and requirements, see *Connector and C++ Connector Server Installation* (see page 23).

The following section contains additional requirements needed for the connector.

Siebel Requirements

The following are required for the Siebel connector:

- The Siebel mobile web client or the Siebel dedicated web client has to be manually installed on a machine, before or after Identity Manager installation, where the C++ Connector Server is running. The Siebel web client version must be the same as a version of a managed Siebel server. See the Siebel documentation for more information.
- The Siebel Application Object Manager has to be running on a Siebel Server. See the Siebel documentation about Siebel Object Manager installation and configuration.

Siebel Support for FIPS and IPv6

For release r12 of Identity Manager, the Siebel Connector does not support FIPs or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire a Siebel Server

You must acquire the Siebel server before you can administer it with Identity Manager. When acquiring a Siebel server, use this procedure.

From the Endpoint type task view

1. Register the server as an endpoint in Identity Manager.

Use the Siebel Server Endpoint property sheet to register a Siebel Server. Necessary mapping information should be provided to associate Siebel fields and Identity Manager attributes. This includes information about Siebel business objects, Siebel business components, and necessary Siebel fields for user accounts and required multi-value groups (positions, responsibilities, organizations, and views).

During the registration process, Identity Manager identifies the Siebel Server you want to administer and gathers information about it.

2. Explore the objects that exist on the endpoint.

After registering the machine in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all Siebel Server objects. You can correlate the user accounts with global users at this time, or you can wait to correlate them.

3. Correlate the explored user accounts with global users.

When you correlate user accounts, Identity Manager creates or links the user accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the username with each existing global user name. If a match is found, Identity Manager associates the Siebel Server user with the global user. If a match is not found, Identity Manager performs the next step.
- b. If the Create Global Users as Needed button is checked, Identity Manager creates a new global user and associates the Siebel user account with the global user. If the Create Global Users as Needed button is unchecked, then Identity Manager performs the next step.
- c. Identity Manager associates the Siebel user account with the [default user] object.

Siebel Endpoint Property Sheet

The Siebel Endpoint Property Sheet consists of ten property pages with seven being specific to the Siebel Connector. The following property pages are Siebel specific:

Siebel Server

Use the Siebel Server Endpoint property page to register a Siebel Server.

Note: Siebel employee or user records that are used as a proxy account in the SBL endpoint object, just have a blank "New Responsibility" field.

Mapping Table: User

Use the Mapping Table: User property page to configure custom mappings for user single-value fields. To edit the custom attributes in the custom attributes list box, click the Edit Button to bring up the Attribute Mapping Dialog. You can then add a new entry in a mapping table or edit an existing entry.

Note: When more than one item corresponding to the same Identity Manager attribute appears in a mapping table, it is not considered an error. The last item found in the mapping table will be taken. This can happen when a client other than Provisioning Manager is used.

Note: Only one user account type is supported per endpoint.

Mapping Table: MVG

Use the Mapping Table: MVG property page to view and set mapping information for fields in user accounts and responsibilities associated with multi-value groups. This property page contains three similar groups of controls for the user's positions, responsibilities, and organizations, and two groups of controls for responsibility's views and organizations.

Mapping Table: Position

Use the Mapping Table: Position property page to configure custom mappings for Position's single-value fields.

Mapping Table: Organization

Use the Mapping Table: Organization property page to configure custom mappings for Organization's single-value fields.

Mapping Table: Responsibility

Use the Mapping Table: Responsibility property page to configure custom mappings for Responsibility's single-value fields.

Mapping Table: Division

Use the Mapping Table: Division property page to configure custom mappings for Internal Division's single-value fields.

Mapping Table: View

Use the Mapping Table: View property page to configure custom mappings for View's single-value fields

Mapping Table: LOV

Use the Mapping Table: LOV property page to configure mappings for list of values' single-value fields.

Siebel User Property Sheet

The Siebel User Property Sheet consists of six property pages with four being specific to the Siebel Connector. The following property pages are Siebel specific:

Profile

Use the Profile property page when managing your users and configuring custom mappings for user single-value fields. To edit the custom attributes in the custom attributes list box, click the Edit Button to bring up the Edit custom attribute Dialog. Depending on the type of control specified in the mapping table for the attribute being edited, you will see either combo box to select a value of the user attribute or an edit box to type in the value you want for the user attribute.

Positions

Use this property page to manage employee positions. You can view a list of all available and occupied positions. You can select a position in the All Positions list box and move it to the Occupied Positions list box to un-assign a position. You can also move a select position from the Occupied Positions list box into the All Positions list box to un-assign an employee position. To assign an employee's primary position, you can copy a selected position from the Occupied Positions list to the Primary Positions text box by clicking the v (Down) button.

Note: Only the user account of "employee" type can be associated with positions. Employee must hold at least one position.

Responsibilities

Use this property page to manage user account's responsibilities. You can view a list of all available and assigned responsibilities. You can select a responsibility in the All Responsibilities list box and move it to the Assigned Responsibilities list box to un-assign a responsibility. You can also move a selected responsibility from the Assigned Responsibilities list box into the All Responsibilities list box to un-assign an employee responsibility. To assign an employee's primary responsibility, you can copy a selected responsibility from the Assigned Responsibilities list to the Primary Responsibility text box by clicking the v (Down) button.

Member of (Organizations)

Use this property page to manage employee's organizations. You can view a list of all available and assigned organizations. You can select an organization in the All Organizations list box and move it to the Member of list box to assign an organization. You can also move a selected organization from the Member of list box into the All Organizations list box to un-assign an employee organization. To assign an employee's primary organization, you can copy a selected organization from the Member of list to the Primary Organization text box by clicking the v (Down) button.

Note: Associating a user account with an organization is not required as this is done implicitly by Siebel. Primary organization cannot be removed from a user.

Siebel Responsibility Property Sheet

This Siebel Responsibility Property Sheet consists of four property pages with three being specific to Siebel.

Responsibility

Use this property page to view the Siebel responsibility name and the custom attributes associated with the Siebel responsibility.

Member of (Organizations)

Use this property page to manage a responsibility's organizations. You can view a list of all available and assigned organizations. You can select an organization in the All Organizations list box and move it to the Member of list box to assign an organization. You can also move a selected organization from the Member of list box into the All Organizations list box to un-assign a responsibility's organization. To assign a responsibility's primary organization, you can copy a selected organization from the Member of list to the Primary Organization text box by clicking the v (Down) button.

Views

Use this property page to manage responsibility's views. You can view a list of all available and assigned views. You can select a view in the All Views list box and move it to the Assigned list box to assign an view. You can also move a selected view from the Assigned list box into the All Views list box to un-assign a responsibility's view.

There's no primary view attribute for a responsibility object.

Siebel Position Property Sheet

The Siebel Position Property Sheet consists of four property pages with three being specific to Siebel.

Position

Use this property page to view the Siebel position name and the custom attributes associated with the Siebel position.

Parent Position

Use this property page to manage a position's parent position. You can view a list of all available positions. You can select a position in the All Positions list box and move it to the Parent Position list box to assign a parent position. You can also move a position from the Parent Position list box into the All Positions list box to un-assign a position's parent position.

Division

Use this property page to manage a position's division. You can view a list of all available divisions. You can select a division in the All Divisions list box and move it to the Associated Division list box to assign a division

Siebel Organization Property Sheet

The Siebel Organization Property Sheet consists of two property pages with one, the Organization property page, being specific to Siebel.

Organization

Use this property page to view the Siebel organization name and the custom attributes associated with the Siebel organization.

Siebel View Property Sheet

The Siebel View Property Sheet consists of two property pages with one, the View property page, being specific to Siebel.

View

Use this property page to view the Siebel view name and the custom attributes associated with the Siebel view.

Siebel Internal Division Property Sheet

The Siebel Internal Division Property Sheet consists of three property pages with two being specific to Siebel.

Internal Division

Use this property page to view the Siebel division name and the custom attributes associated with the Siebel division.

Parent Division

Use this property page to manage a division's parent division. You can view a list of all available divisions. You can select a division in the All Divisions list box and move it to the Parent Division list box to assign a parent division. You can also move a division from the Parent Division list box into the All Divisions list box to un-assign a division's parent division.

Siebel LOV Property Sheet

The Siebel LOV Property Sheet consists of two property pages with one, the Properties property page, being specific to Siebel.

Properties

Use this property page to view the LOV code, type of value, and the display value.

Chapter 15: Remote Access Connector

This section contains the following topics:

[The RSA ACE \(SecurID\) Connector](#) (see page 471)

The following connector is considered to be a remote access connector: RSA ACE/Server

The information in this chapter describes how to install and manage this connector.

The RSA ACE (SecurID) Connector

The RSA ACE (SecurID) Connector lets you administer the users, groups of users, and tokens of RSA ACE/Server machines and provides a single point for all user administration by letting you do the following:

- Retrieve the existing users from the RSA ACE/Server database
- Display, create, modify, or delete a user
- Assign or un-assign a token to a user
- Create remote users
- Add or remove users on an Agent Host
- Add or remove a user to a group
- Retrieve existing groups from the ACE/Server repository
- Create and delete groups
- Enable or disable a group on an Agent Host
- Retrieve a token's details
- Active operations on a token

RSA Installation

This connector is managed using the Connector and agent installation process. For more information and requirements, click [here](#) (see page 21).

This connector can also be managed using the [Connector and C++ Server](#) (see page 23) installation process as well.

The following sections detail the post installation and configuring requirements for this connector.

RSA Post Installation Requirements

The following must be done after the connector installation:

- The user named SYSTEM must be added to the Primary RSA ACE/Server and registered as an Administrator.
- CAM CAFT service must be configured on the Primary RSA/ACE Server. For more information, see the following section.
- The RSA Authentication Manager 5.x and higher Administration Toolkit must be installed on the Primary RSA ACE/Server. For token management, the 6.1 Administration Toolkit is required.

RSA Limitations

For this release, the following limitations should be considered when using the RSA Connector:

- If the PIN change option is selected for an eTPassword change event propagation, only numeric values for the password change event will be accepted regardless of the PIN options settings specified in the System Parameters of the RSA ACE/Server Administration Tool. This limitation is due to handling of the PIN change by RSA Administration Toolkit function Sd_SetPin(). This restriction is also imposed by the type of the devices (like RSA SecurID PINPAD Token) that are not allowed the use of alphanumeric PINs.
- Management for multiple tokens is not supported. The Agent component processes modify requests for token objects one at a time.
- The assignment of the tokens to the accounts created for global users cannot be done using the RSA Account Template. A token cannot be associated with more than one user at the same time. To do this, you must create the accounts first and then assign tokens using the RSA Connector GUI or RSA native administration tools.

How to Configure the CAM and CAFT Service

Install the RSA Remote Agent and configure the CAM and CAFT Service on any RSA ACE/Server machine that you want to administer.

To configure the CAM and CAFT Service, perform the following procedure.

From the RSA ACE/Server machine

1. Log on as the domain or local administrator
2. Issue the following command from a command window:

```
caftthost -a RSA_node_name
```

RSA_node_name

Specifies the name of the Connector Server.

Note: If the Connector Server is networked using DHCP or you do not use DNS for name resolution, the network name will not be recognized. Under these conditions, use the TCP/IP address for the RSA ACE node name or add an RSA ACE node entry in the local hosts file on your RSA ACE/Server machine.

3. Verify this command by issuing the following command:

```
caftthost -l
```

RSA Support for FIPS and IPv6

For release r12 of Identity Manager, the RSA Connector does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

Acquire an RSA ACE/Server Machine

You must acquire the RSA ACE/Server machine before you can administer it with Identity Manager. When acquiring an RSA ACE/Server machine, perform the following steps.

From the Endpoint type task view

1. Register the machine as an endpoint in Identity Manager.

Use the RSA ACE (SecurID) Endpoint property sheet to register an RSA ACE/Server machine. During the registration process, Identity Manager identifies the RSA ACE/Server machine you want to administer and gathers information about it.

2. Explore the objects that exist in the endpoint.

After registering the machine in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all RSA ACE (SecurID) objects. You can correlate the accounts with global users at this time or you can correlate them later.

3. Correlate the explored accounts to global users

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the username with each existing global user name. If a match is found, Identity Manager associates the RSA ACE (SecurID) account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the account name with each existing global user's full name. If a match is found, Identity Manager associates the RSA ACE (SecurID) account with the global user. If a match is not found, Identity Manager performs the following step.
- c. If the Create Global Users as Needed button is checked, Identity Manager creates a new global user and associates the RSA ACE (SecurID) account with the global user. If the Create Global Users as Needed button is unchecked, then Identity Manager performs the next step.
- d. Identity Manager associates the RSA ACE (SecurID) account with the [default user] object.

RSA Endpoint Property Sheet

The RSA Endpoint Property sheet lets you register or view the properties of an RSA ACE server. From the RSA Endpoint Tab you can specify the endpoint name, host name, account template information, and the password change propagate mode.

Password Change Propagate Mode

The password change propagate mode on the Endpoint Tab of the Endpoint Property Sheet, specifies the way that the Password and PIN changes are handled during a change event. The following scenarios are possible:

1. If neither the Password Change nor PIN Change check boxes are checked, the password change will not occur.
2. If the Password Change check box is selected, but the PIN Change check box is not selected, only the user password will be changed to the value provided in the eTPassword attribute. No modifications will be applied to the assigned tokens.
3. If the Password Change check box is not selected, but the PIN Change check box is selected, only the value of the PIN for the assigned tokens will be changed to the value provided in the eTPassword attribute.
4. If both the Password Change check box and PIN Change check box are selected, both the user password and assigned tokens PINs will be changed to the value provided in the eTPassword attribute.

Note: For 4, if the user does not have any tokens assigned to them, the request to modify the eTPassword attribute is treated as a request to assign the password to the user using the value provided in the eTPassword attribute.

Note: For 3 and 4, if the user has more than one token assigned, the PIN reset applies to ALL of the tokens that are in possession of the user. The PIN associated with each assigned token is changed to the value provided in the eTPassword attribute.

RSA Groups

Identity Manager lets you create and maintain RSA ACE (SecurID) groups using the Endpoint type task view. Use the RSA Group property sheet when managing your groups.

Token Management

The Token management features of the RSA Connector let users view and manage RSA Tokens and simulate suspending RSA accounts through a global user or Token properties.

RSA Tokens Property Sheet

The RSA Tokens property sheet lets you view a token's details and initiate operations on the token. The following property pages apply to RSA tokens:

- Token Profile
- Token Operations

Tokens Profile Tab

The Tokens Profile tab is a read-only page that displays the following details of a token:

- Serial Number
- Token Type
- Assigned...to..
- Enabled
- Lost
- New PIN Mode
- Expired
- Replacement Status

Token Operations Tab

The Token Operations tab is used to initiate operations on a token.

Note: Operations are initiated on a single token at a time.

Using this tab, you can initiate the following operations:

Enable Token

Enables the token.

Disable Token

Disables the token

Set New PIN Mode

When the Set New PIN operation is selected, you can check the Clear PIN check box to clear the token immediately. A new PIN is assigned the next time you log in with your token code only.

If the Clear PIN check box is not checked, the PIN is cleared the next time you log in with your current PIN and token code.

Set Emergency Mode Off

Sets the emergency mode to off.

Set Emergency Mode On

When the set emergency mode operation is selected, you can specify the temporary password to be used, how long the emergency access mode is to last, and that the token is not automatically declared lost during the emergency access mode.

Set Replacement Mode

When the replacement mode operation is selected, you can search for a replacement token from all the available tokens or narrow the search by specifying specific attributes to search.

Enable a Token

Perform this procedure to run the enable token operation.

From the Token Operations Tab

1. Select Enable Token from the Operation field drop-down list.
2. Click Apply/OK to enable the token.

The enable token operation is performed.

Disable a Token

Perform this procedure to run the disable token operation.

From the Token Operations Tab

1. Select Disable Token from the Operation field drop-down list.
2. Click Apply/OK to disable the token.

The disable token operation is performed.

Set Emergency Mode Off

Perform this procedure to run the set emergency mode off operation.

From the Token Operations Tab

1. Select Set Emergency Mode Off from the Operation field drop-down list.
2. Click Apply/OK to turn emergency mode off.

The set Emergency Mode Off operation is performed.

Set New PIN Mode

Perform the following procedure to run the set new PIN mode operation.

From the Token Operation Tab

1. Select Set New PIN Mode from the Operations field drop-down list.

The Set New PIN Mode controls are activated.

2. Check the Clear PIN check box if the PIN for the token is to be cleared immediately.

A new PIN must be assigned the next time you log in with your token code. Your current PIN will not work.

If the Clear PIN check box is not checked, the PIN is cleared when you log in again with you current PIN and token code.

3. Click Apply/OK to put the token into new PIN mode.

The Set New PIN Mode operation is performed.

Set Emergency Mode On

Perform the following procedure to run the set emergency mode on operation.

From the Token Operation Tab

1. Select Emergency Mode On from the Operations field drop-down list.
The Set Emergency Mode On controls are activated.
2. Enter the temporary password in the Temporary Password field to be used during the emergency operation.
3. Enter the length in the Life time field, in hours, that the emergency mode will be in effect.
4. Check the Auto not lost check box if the token should not be declared lost during emergency mode. If the check box is not selected, the token will be declared lost during the emergency operation.

Note: Auto not lost is only available for RSA 6.1 or higher.

5. Click Apply/OK to turn Emergency Mode on.

The Set Emergency Mode On operation is performed.

Set Replacement Mode

Perform the following procedure to run the set replacement mode operation.

From the Token Operation Tab

1. Select Set Replacement Mode from the Operations field drop-down list.
The Set Replacement Mode Operations controls are activated.
2. Check the Keep Current Pin check box if the replacement token should be given the same PIN as the token being replaced. Leave this check box unchecked if the replacement token should start in new PIN mode.
3. Search for the available tokens by clicking the Search button to list all of the available tokens or specify the seed size, token type, and serial number to narrow the token search.

The available tokens appear in the Available Tokens List Box.

4. Select a token from the Available Tokens list box and click the Add (>) button to add the token to the replacement serial number field.
5. Click Apply/OK to perform the selected replacement.

The Set Replacement Mode operation is performed.

Suspending and Resuming RSA Accounts

The RSA Connector can simulate account suspension by removing all tokens from an account. This approach restricts a user's ability to access the system. Resumption of an account is implemented by re-assigning tokens to an account. Two following two attributes are included in the connectors account class:

- eTSuspensionState
- eTPreSuspensionState

If these two attributes are defined, the following requests on an RSA account are affected:

- Account Search
- Account Modify
- Account Suspend
- Account Resume

Account Search

If the eTSuspensionState attribute is explicitly mentioned in a search request the agent plug-in retrieves a list of tokens assigned to an account, generates a corresponding XML document and returns it as an eTSuspensionState value.

Account Modify

If the eTSuspensionState attribute is included in a modifications list, an account is considered to be already suspended. If a list of modifications in a request contains any updates of the eTRSATokenNumber multi-value attribute, that request is rejected with an LDAP_OPERATIONS_ERROR code and proper message being sent.

Account Suspend

If the eTSuspended attribute is set to "1", the agent plug-in removes all eTRSATokenNumber values from an account. The account is then suspended.

Account Resume

To resume an account, a modify operation with eTSuspended set to 0 must be run. If the eTSuspensionState attribute is present in the modifications list, the attribute must be used to restore the eTRSATokenNumber values of an account.

Chapter 16: Public Key Infrastructure Connector

This section contains the following topics:

[Entrust PKI Connector](#) (see page 481)

The following connector is considered to be a public key infrastructure connector: Entrust PKI

The information in this chapter describes how to install and manage this connector.

Entrust PKI Connector

The Entrust PKI Connector lets you administer the Entrust Authority Security Management System as a managed endpoint and provides a single point for all user administration by letting you do the following:

- Acquire and explore Entrust Systems
- Correlate Entrust users with global user identities
- Perform create, modify, and delete operations on Entrust accounts.

Note: This connector does not replace the administration facility provided by native Entrust Security Manager Administration tools, but leverages those to manage the Entrust System.

Entrust PKI Installation

This connector is managed using the Connector and C++ Server installation process.

Note: For more information and requirements, see *Connector and C++ Connector Server Installation* (see page 23).

The following sections detail the pre-and post-installation requirements of this connector.

Entrust PKI Connector Installation Prerequisites

The following are the Entrust PKI prerequisites:

- Entrust Authority Security Manager v7.0 or v7.1 running on Windows 2000/2003 Server or Solaris 8
- The Entrust Authority Administration Toolkit for C Release 6.0 Service Pack 2 must be installed on the machine where the C++ Connector Server is installed. The environment variable PATH must be modified to include the location where the Administration Toolkit libraries (etadmintk.dll and enterr.dll) can be found.

Entrust PKI Connector Limitations

For this release, the following limitations should be considered when using the Entrust PKI Connector:

- Configuration of the Entrust System (setting certificate types, Entrust user roles and templates, and so forth) should be done using the Entrust native tools. The Identity Manager Entrust PKI account template or Add New Account operation can retrieve the range of the valid values for these attributes from the Entrust endpoint.
- New accounts are created with the Use default key update policy set. For some accounts, the individual key update policy may be required to be set using the Security Manager Administration tool.
- Due to restrictions in the Entrust Administration toolkit, password synchronization/propagation cannot be enabled for the PKI Accounts.
- The functionality that is provided in the Entrust PKI Connector is dependent on the functionality that is supported or restricted by the Entrust Administration toolkit.
- For this release, the assumption is made that for each host, a user can have only one profile defined. The reason for this limitation is based on the assumption that only one Entrust Authority Security System is associated with a given host and the user uses only one profile to access the System.
- Due to limitations in the Entrust Administration toolkit, the PKI connector does not provide the functionality to operate on search bases. The Container DN control does not let you search the LDAP repository that is used internally by the Entrust Authority Security Manager system so you must type in the value of the Container DN.
- Entrust Authority implements the Change DN operation in such a way that once the operation is executed, the operation enters a pending state and remains in that state until either the operation is canceled or the user logs in. During the pending state, two entries exist for the same user. If an exploration from the Provisioning Server is performed during this time, it will fail to explore the accounts container.

- Due to restrictions imposed by the Entrust Administration toolkit and implementation of the Provisioning Server, only the user's common name will be used for correlation purposes (to correlate existing Entrust accounts with global users).
- An attempt to create a PKI account by synchronizing a suspended global user with a PKI account template will fail to create the account. This operation only works for active global users.
- PKI accounts cannot be activated or deactivated by performing a direct operation on the account object. These operations can only be performed by propagating global user status to the account.
- Only global users with Identity Manager administrative privileges, as well as PKI administrative privileges, are allowed to modify properties of managed PKI endpoint objects.
- Hardware security devices are not supported.

Entrust PKI Post Installation Requirements

The following configurations must be made after the PKI connector has been installed:

- The Provisioning administrator must have an existing account with administrative privileges in the Entrust Authority domain.
 - The administrator's profile (.epf file) must be made accessible from the PKI Agent plug-in
 - The administrator's password must be the same as their Entrust PKI password protecting their profile.
- The entrust.ini file that is used to establish a connection with the Entrust Authority Manager System must be accessible from the PKI Agent plug-in to acquire an Entrust PKI Endpoint. You must modify the FIPS Mode section of this file by replacing FipsMode=1 with FipsMode=0.
- To create Entrust profiles, the PKI Agent plug-in must have write access to the location(s) where the Entrust profiles are to be created.

Create a PKI Environment

Once the PKI connector has been installed and configured, you can create your PKI environment.

To create your PKI environment

1. Create an administrative user and profile.

The Identity Manager administrator that is going to use the PKI connector must have an existing account with administrative privileges in the Entrust Authority domain and the corresponding profile. This user should have the Security Officer role assigned.

2. For this administrator, create a Global User in Identity Manager with the Admin profile that lets the Global User perform administrative actions (for example, Domain Administrator), and assign the password that matches exactly the password that was used to create the profile for this administrative user in the Entrust Security System.

3. (Optional) Define a primary email address for this administrator to enable the email notification feature.

Use the Entrust PKI Domain Access tab on the Global User Property Sheet to define the host name of the managed PKI system and the PKI profile file (the corresponding PKI account that is used to manage the PKI system).

4. Log into Identity Manager using the credentials of the global user that was just created.

Your PKI environment is now established.

PKI Support for FIPS and IPv6

For release r12 of Identity Manager, the Entrust PKI Connector does not support FIPS or IPv6.

Connector-Specific Features

This section details your connector's specific management features, such as how to acquire and explore your endpoint. Also included are account, provisioning roles, account template, and group information specifically for your connector.

Note: For a general overview of the Provisioning Manager and its main features, see [Managing the Connectors](#) (see page 29). For more detailed information on the Provisioning Manager, see the *Provisioning Guide*.

How PKI Endpoints are Authenticated

There are two different ways for the PKI Connector to authenticate with Entrust PKI endpoints:

- Proxy profile approach
- Individual Entrust profile approach

Proxy Profile Approach

A proxy profile, defined on an endpoint type level can be used to authenticate PKI endpoints. Any Provisioning administrator is able to manage such PKI endpoint types. An Administrator's Provisioning User will not be required to have Entrust profile related information in its eTOptionDataPKI attribute.

All Provisioning administrators will have the same level of access to an Entrust system. This eliminates any possible strange exploration results, where an exploration is initiated by a person with low level PKI access. If you find this approach insecure, you can use individual profiles and discussed in the next section.

Individual Entrust Profile Approach

You must acquire the Entrust Authority Security Manager System before you can administer it with Identity Manager. This lets you join Entrust users' domain with Admin Global Users' domain into one domain to enable centralized administration.

To manage the Entrust target system, the Identity Manager administrator must have an account on the Entrust System with administrative privileges and have a profile (.epf file) and password to protect the profile. This information must be assigned as properties of the Global User who represents the Administrator that is going to manage the Entrust system using the PKI Connector.

The Entrust PKI Domain Access tab of the Global Users Property Sheet lets you define Global User's credentials to access Entrust Systems by specifying the hostname where the system is located and the profile that was assigned to the Global User's account on this system. The password that is assigned to the Global User must match exactly the password that is used to protect the Entrust profile.

Acquire an Entrust Authority Security Manager System

To acquire an Entrust Authority Security Manager System, you must be an administrative user defined in the Entrust Authority System.

From the Endpoint Type task view

1. Register the Entrust Authority Security Manager System as a managed endpoint in Identity Manager.

Use the PKI Endpoint property sheet to register an Entrust Authority Security Manager System.

Important! The host name string specified in the host name control, must match exactly the host name string specified for this Entrust Authority system in the Global User settings on the Entrust PKI Domain Access property page.

During the registration process, Identity Manager identifies the Entrust Authority Security Manager System you want to administer and gathers information about it.

2. Explore the objects that exist on the endpoint.

After registering the Entrust Authority Security Manager System in Identity Manager, you can explore its contents. Use the Explore and Correlate Endpoint dialog. The Exploration process finds all PKI users/accounts. You can correlate the accounts with global users at this time or you can correlate them later.

3. Correlate the explored accounts with global users.

When you correlate accounts, Identity Manager creates or links the accounts on an endpoint with global users, as follows:

- a. Identity Manager attempts to match the logon name with each existing global user name. If a match is found, Identity Manager associates the PKI account with the global user. If a match is not found, Identity Manager performs the next step.
- b. Identity Manager attempts to match the display name with each existing global user's full name. If a match is found, Identity Manager associates the PKI account with the global user. If a match is not found, Identity Manager performs the next step.
- c. If the Create Global Users as Needed button is selected, Identity Manager creates a new global user and then associates the PKI account with the global user. If the Create Global Users as Needed button is cleared, Identity Manager performs the next step.
- d. Identity Manager associates the PKI account with the [default user] object.

Endpoint Connections

The PKI Endpoint object does not maintain any connection with the target system. The possibility to connect to the target system or the target system availability is checked during the creation of the PKI Endpoint object. Access to the target system and login to Entrust is performed at the beginning of the processing of every request to manage objects that belong to the PKI Endpoint.

Also, the PKI Endpoint object does not contain Administrator's credentials required to login to Entrust. The login credentials of the Entrust user that exists on the Security Manager System must be used to create an association with that Security Manager System.

PKI Provisioning Roles and Account Templates

The PKI Default Account Template, provided with the PKI connector, gives a user the minimum security level needed to access an endpoint. You can use it as a model to create new account templates.

Note: You can create PKI account templates that are associated with multiple endpoints. These account templates can only be used to grant privileges to existing accounts.

PKI Account Templates

Each PKI account template must have at least one endpoint inclusion. We recommend that account templates not be associated with more than one endpoint. This recommendation is due to certain certificate types not being available on all endpoints.

PKI Accounts

The PKI Account represents the Global User in the Security Manager System instance and binds the Global User to the existing Entrust user (or creates a new user).

The PKI Account Property Sheet consists of six property pages with the following four pages specific to the PKI connector:

- Account Properties
- Operations
- Certificates
- Member Of

Create PKI Accounts

Once an endpoint has been explored, PKI accounts can be created as needed.

To create PKI accounts

1. Click the Endpoint Type task button, select the PKI Endpoint in the Object Type drop-down list box, and click Search to search for the endpoint that you want to create an account on.

A list of explored endpoints appears.

2. Right-click on the endpoint in the list view and choose Content from the pop-up menu.

The Endpoint Content dialog appears.

3. Select Accounts in the Container Tree box and then click New.

4. Complete the PKI Account Property Sheet by:

- a. Entering the required information on the Account Properties tab. This information includes naming properties, certificate information, and profile options for the account

- b. Clicking OK.

A new PKI Account is created.

Account Properties

The Account Properties page is used for both Accounts and Account Templates. This property page lets you specify the naming properties, certificate information, and profile options for the accounts and account templates.

For existing Accounts or Account Templates, this page displays values of attributes that are assigned to the existing objects. This page is also used to initiate the following operations:

- Change associated Entrust role
- Change Certificate Type

Note: Certificate updates take effect only at the time when the next automated key update is scheduled. You can force the update using the Update Key Pairs option in the Security Manager Administration tool.

- Create profile for existing users

Upon completion of this operation, a profile is created and placed into the specified location. The user status is changed from Added to Active.

Change a User Role

You can modify a PKI account to run the change associated Entrust user role operation.

To change the associated role

1. Select the Account Properties tab on the PKI Account Property Sheet.

The Account Properties tab appears.

2. From the User Role drop-down menu, choose the user role you want associated to the account and Click OK.

The operation to change the associated user role is initiated.

Change the Certificate Type

You can modify a PKI account to run the change certificate type operation.

To change the certificate type

1. Select the Account Properties tab on the PKI Account Property Sheet.

The Account Properties tab appears.

2. From the Certificate type drop-down menu, choose the certificate type you want and Click OK.

Note: Certificate updates take effect only at the time when the next automated key update is scheduled. You can force the update using the Update Key Pairs option in the Security Manager Administration tool.

Create a Profile for an Existing User

You can modify a PKI account to create a profile for an existing user.

To create the profile

1. Select the Account Properties tab on the PKI Account Property Sheet.
The Account Properties tab appears.
2. Check the Create profile check box to force the creation of an Entrust user's profile and then fill in the following fields:

- a. Select the type of profile to create from the drop-down menu.
- b. Enter and confirm the password for the account.
- c. Enter the full path of the Entrust profile to be created.

Note: The name of the profile file including the full path must be provided. Use the Browse Location button to browse for the profile name.

3. Click OK.

Upon completion of this operation, a profile is created and placed into the specified location. The user status is changed from Added to Active.

Operations

The Operations Property page lets you perform the following operations:

Revoke All Certificates

Revokes all certificates that are associated with the account. You must specify an appropriate reason in the Revocation area of the GUI.

If you are using the credentials of an Entrust Administrator (not a Security Officer role), you do not have permission to issue a new CRL (Certificate Revocation List) and should not select the Issue CRL option.

Note: This operation is irreversible for all reasons except the "ON HOLD" reason unless it is part of a Global User suspend/un-suspend operation. If a certificate is put ON HOLD using this operation's option, the Entrust Administration tool is required to take it out of hold status

Rollover User

Rolls over a user's encryption key. No extra information is required to be specified.

Recover User

Performed by an administrator upon receiving a corresponding request from the user (for example, an end user's password needs to be re-established).

Recover User Operation

The administrator can run the compound recover operation to re-establish the user's password when requested.

To recover the user when Recover profile is also checked

1. Select Recover User and click Apply.

The user is put into Key Recovery state.

2. Make sure the name of the .epf file here is the same name as the revoked file.
3. Deliver the password to the end user out of bound.

If the Recover profile option is disabled or not checked, new activation codes are generated after the user is put into Key Recovery state. The administrator can then deliver the activation codes to the end user out of bound.

Certificates

The Certificates Property page lets you view the list of certificates available to the PKI Account (user). You can either select a certificate from the list and view it in detail using the View Certificate Contents button, or you can revoke the selected certificate by providing the required reason.

Member Of

The Member Of property page is used for both Accounts and Account Templates. This page lets you view the PKI Groups that are available for both the account and account template. You can also assign a user to the selected group or remove the user from the group.

This page contains the All groups control that when selected, implies implicit membership. (For further information on group handling, see the Entrust documentation.) Assigned groups that have been selected from this list are explicitly assigned. You cannot have both implicit and explicit memberships.

The Entrust user (account) must be a member of at least one group. This is required by the Entrust Security Manager System.

Chapter 17: Feed Connector

This section contains the following topics:

[PeopleSoft Feed](#) (see page 493)

The following connector is considered to be a feed connector: PeopleSoft Feed

The information in this chapter describes how to install and manage this connector.

PeopleSoft Feed

The PeopleSoft Feed is a command line utility that uses Identity Manager's Bulk Loader task to bulk load user objects from authoritative data sources. For this release, only PeopleSoft can be loaded.

With the PeopleSoft Feed, you can do the following:

- Create a user when the user does not exist
- Modify the user if the user already exists
- Delete a user is no longer needed

Install the PeopleSoft Feed

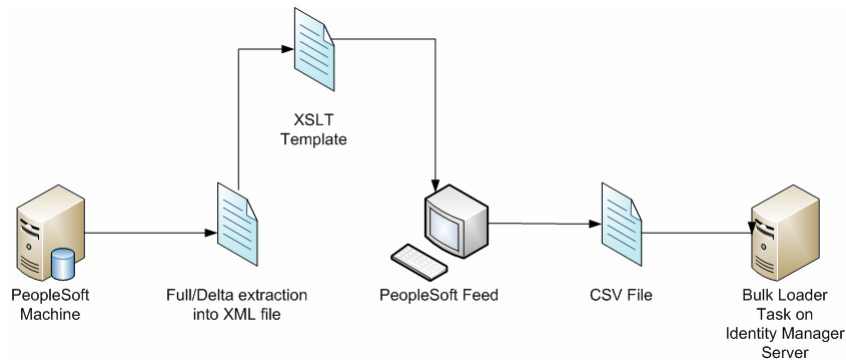
To install the PeopleSoft Feed utility, run the setup.exe program of the Connector Factory installation package and select PeopleSoft Feed option.

During installation you can choose to enter the Server URL, as well as the user name and password.

How the PeopleSoft Feed Works

The relevant user and account information is extracted from the authoritative data source, (For this release the data source is PeopleSoft.), in the form of an XML or CSV file. If the information is stored in an XML, the file is transformed to CSV format by the PeopleSoft Feed using XSL transformation. The resultant CSV file is then used by the PeopleSoft Feed to invoke the Bulk Loader task on Identity Manager. Based on the mapped Identity Manager admin tasks, users will be created, modified, or deleted.

The following is an example of the PeopleSoft Feed cycle:



From the PeopleSoft machine

1. Extract either a full or delta dump of user records in the form of an XML file.
2. Create an XSLT template to convert the XML file into the standard CSV form that can be used by the Identity Manager Bulk Loader task.
3. Feed the XML file into Identity Manager PeopleSoft Feed, together with the XSLT template.

PeopleSoft Feed will internally transform the XML file into CSV format required by the Identity Manager Bulk Loader task. You can either just write this CSV file to a disk file or use the CSV file to invoke the Bulk Loader task.

Note: The resultant CSV file can be loaded in smaller chunks;

4. PeopleSoft Feed uses the CSV file to invoke the Bulk Loader task using the TEWS interface.

If the CSV file has been broken up into smaller chunks, PeopleSoft Feed invokes the Bulk Loader task for each of the chunks. Subsequent chunks will be sent to the Bulk Loader task once the SOAP response from the previous request is received and the response indicates that the previous request was submitted successfully.

5. The SOAP response is logged to a file/written to the standard output.

Command Line Options

The following options are used to run the PeopleSoft Feed:

Command	Description
-b, --batchSize <number>	This option specifies the maximum number of user data records to be sent to the server in each request. This option can be used to break up the large number of records into smaller chunks in order to avoid overloading the server.
-c, --configFile <file>	This option specifies a properties file that contains the configuration options for invoking the Bulk Loader task. The default is "imbulkloadclient.properties".
-e, --endpointInfoFile	This option is used together with the (-s, --storeEndpointInfo) option. This option specifies a properties file that contains the key/value pairs for "user", "password", and "serverUrl".
-f, --format (CSV XML)	This option specifies the format of the input file (-i, --inputFile) that contains the data records to be sent to the server. The default is XML. When the input file format is XML, you must use the -t, --transformFile <file> option to specify the XSLT template for carrying out the transformation. If the input file format is CSV, the file will be submitted to the Bulk Loader task directly without transformation.
-h, --help	Displays the help and exit.
-i, --inputFile <file>	The input file that contains the user data records to be sent to the server. It can be in XML or CVS format.
-o, --outputFile <file>	If this is option is specified, the result will also be written to this file when the input file is transformed.
-p, --password <pass>	Specifies the password used for server authentication.
-s, --serverUrl <url>	Specifies the URL of the TEWS interface.

-S, --storeEndpointInfo	Stores the specified server URL and the Admin user name and password in the configuration file (-c, --configFile). The password is obfuscated before it is stored. The information that is going to be stored can be provided through (-e, --endpointInfoFile).
-u, --user <username>	Specifies the user name for server authentication.
-v, --verbose	Specifies to output as much of the message as available.
-V, --version	Displays the version information of the program.
-T, --transformOnly	Specifies to only carry out the XSLT transformation of the input XML file into CSV format without submitting to the server. If a valid file name is also specified by -o, --outputFile <file>, the CSV result will be written to that file.
-t, --transformFile <file>	Specifies the file that contains the XSLT template for XSLT transformation of the input file, if the file format is in XML.

Supported Use Cases

The following use cases are supported for the PeopleSoft Feed:

- Full Dump
- Delta Dump
- Scheduling of the full or delta load

Delta Dump

A delta dump is made of all user changes since the last time a delta/full dump was made. This dump presents the current state of each record modified since the previous delta/full dump, or identifies the record as deleted if the user/account no longer exists.

With PeopleSoft HRMS, there is a pre-defined message (PERSON_BASIC_SYNC) that publishes every change made to the user data records. You can set up PeopleSoft Integration Broker to publish those changes to a local XML file. This XML file can then be loaded by PeopleSoft Feed. A sample message files comes with the installation and is located under the "samples" directory. This file contains a sample message for PERSON_BASIC_SYNC.

Note: PERSON_BASIC_SYNC publishes every change you made to the user data record to its own file, so there could be many files to load.

You can create your own message definition that publishes all the changes you made since last full/delta dump into one single file. For additional assistance with how to create the custom message, refer to Oracle Support.

Full Dump

A full dump is a complete dump of all users. The full dump data extraction must present the current state of each record at the time of the extraction.

With PeopleSoft HRMS, you can use the full table synchronization message (PERSON_BASIC_FULLSYNC) to do a full table publish which can publish all user data records to a local XML file. That XML file can then be used to feed into PeopleSoft Feed. You might need to customize the PERSON_BASIC_FULLSYNC message to suit your specific needs. Also, you will need to map all the records in the message to a view that only extracts currently effective data. A sample message file (peoplesoft2.xml) comes with the installation and is located under the "samples" directory. This file contains sample messages for PERSON_BASIC_FULLSYNC.

Refer to the *PeopleSoft Integration Broker PeopleBook* for detailed information on how to set up PeopleSoft Integration Broker and a full table data publish.

Scheduling a Load

Scheduling the load is done using native OS capabilities and not as part of PeopleSoft Feed.

Using the XSLT Template

If the data extracted from the PeopleSoft machine is in an XML file, an XSLT template file called `PeopleSoft.xslt` (which imports another XSLT file called `import.xslt`) has been created to carry out the transformation of the extracted XML file into the CSV format that can be used by the Bulk Loader task of Identity Manager. For information on how the CSV file should be formatted, see "Feeder File Format" in the *Identity Manager Administration Guide*.

This template file is created to work with PeopleSoft Rowset-based message format and is customizable. For instructions on how to customize this file, check the comments in the template.

Note: The two XSLT files are located under the "samples" directory.

Using PeopleSoft Feed

When using PeopleSoft Feed, we recommend that you use PeopleSoft Feed with the out-of-the-box configuration of the Identity Manager Bulk Loader task. Changes to the tag of the Bulk Load task can cause the client of the web service to be unusable.

You must also enable the execution of Web Services on the Identity Manager Server.

PeopleSoft Feed Localization

PeopleSoft Feed uses the default locale of the Java Virtual Machine when starting up, and the default locale corresponds to system locale of the host platform. All user messages are externalized to the Java ResourceBundles to allow localization. The default resource file (Java Properties file) that contains the English resource Imbulkloadclient_msg.properties file is built into the imbulkloadclient.jar file and is used by default.

To use a resource file that contains a different language resource, create the resource file by translating the default resource file and putting the new resource file under

```
$INSTALLATION_DIR\conf\com\caliam\imbulkloadclient
```

The file name of the new resource file should have the language and country code appended. For example, for Canadian French, the file name should be

```
Imbulkloadclient_msg_fr_CA.properties
```

where

fr

Specifies the lowercase two-letter ISO-639 language code

CA

Specifies the uppercase two-letter ISO-3166 country code

Note: A Java resource file is a Java properties file. The encoding of a properties file is ISO-8859-1, also known as Latin-1. All non-Latin-1 characters must be entered by using Unicode escape characters. For example, \uHHHH, where HHHH is a hexadecimal index of the character in the Unicode character set. You can use the JDK tool native2ascii.exe to convert files which contain other character encodings into files containing Latin-1 and/or Unicode-encoded characters (using Unicode escape characters).

Authenticating to the Identity Manager Server

PeopleSoft Feed uses user name / password to authenticate to the Identity Manager Server.

When the Identity Manager Server is protected by SiteMinder™, SiteMinder™ basic authentication is supported by setting "isProtectedBySiteMinder = true" in \BulkLoadClient\conf\imbulkloadclient.properties.

SSL Support

If you want to use SSL to protect data submitted to Identity Manager, the Identity Manager Server must be configured to accept HTTPS requests. The following procedure is a sample configuration when using JBoss, on a Windows platform, as the application server for Identity Manager Server. The sample uses the Java utility "keytool.exe" to create and manage key stores for the server. You can refer to Identity Manager documentation for information on how to set up SSL when Identity Manager is running on another application server.

From the "\$JBOSS_HOME/server/default/conf directory

1. Create a keystore by issuing the following command to create a self-signed certificate for the Server. When prompted for the first name and last name, use the domain name of the host machine.

```
keytool -genkey -alias server -keyalg RSA -keystore server.keystore -validity 3650
```

2. From the Tomcat SAR subdirectory of your server configuration's deployment directory \$JBOSS_HOME/server/default/deploy/jbossweb-tomcat55.sar, edit the server.xml file and uncomment the <Connector> element for SSL that listens on port 8443. Make sure the values to "keystoreFile" and "keystorePass" matches what you have created.

```
<!-- SSL/TLS Connector configuration using the admin devl guide keystore -->
<Connector port="8443" address="{jboss.bind.address}"
    maxThreads="100" strategy="ms" maxHttpHeaderSize="8192"
    emptySessionPath="true"
    scheme="https" secure="true" clientAuth="false"
    keystoreFile="{jboss.server.home.dir}/conf/server.keystore"
    keystorePass="password" sslProtocol = "TLS" />
```

3. Restart the server.
4. Test out the SSL connection by starting a browser on your local machine and go to <https://localhost:8443/jmx-console/index.jsp>. Your browser should send an error about an un-trusted certificate signer. To avoid this, either import the server certificate into you browser or obtain a certificate from a well known certificate authority.
5. Specify the server URL using HTTPS URL. For example: <https://nisda03-imr12:8443/idm/TEWS6/feedtst>
6. From the machine where you are running the PeopleSoft Feed, use the Java keytool utility to create a keystore and import the server certificate as trusted certificate. Enter your keystore password when prompted. Answer yes when asked whether to trust the certificate being imported.

```
keytool -import -alias imserver -file <your_server_cert_file> -keystore
"%HOMEDRIVE%%HOMEPATH%\imbulkloadclientkeystore"
```

7. Edit the imbulkloadclient.bat or the imbulkloadclient.sh file to set TRUSTSTORE_PASSWORD to the value you entered in the previous step.

Configuring the PeopleSoft Feed

The following needs to be configured through a Property file when using the PeopleSoft Feed:

- Identity Manager parser class to be used for the Bulk Loader task. At the moment, only "com.ca.identitymanager.feeder.parser.CSVParser" is supported.
- The unique identifier attribute name (column name) in the CSV file.
- The action attribute name (column name) in the CSV file.
- The primary object for the Bulk Loader task. (This is always "USER")
- The action to admin task mapping (in the form of "create.Create User;modify.Modify User;delete.Delete User;")
- Whether or not the web service is protected by SiteMinder.

Anything on the command line can be specified in the configuration as well. Just add a *key/value pair* to the configuration file with the key being the command line option's long form name.

Note: The arguments provided on the command line will take precedence over the values specified in configuration file and the password stored in the configuration file will be treated as having been obfuscated.

Properties File Example

The following is an example of the Properties file used to configure the Bulk Loader Task and the PeopleSoft Feed:

```
#
# These are the connection details of the Identity Manager Server
#

# administrator id and password to be used to carry out the task
user=admin1
password=FPWg3MtYrUnididAMY06LZT/3LPuMtu6O7A+DRzX1Jl=

# server URL
serverUrl=http://nisda03-imr12:8080/idm/TEWS6/feedtst

#
# these are the configuration items for the Identity Manager ObjectsFeeder task
#

# Identity Manager parser to be used for the ObjectsFeeder task
feederParserClass= com.ca.identitymanager.feeder.parser.CSVParser

# The unique identifier attribute name (column name in the CSV file)
uniqueIdentifierAttrName=UID

# The action attribute name (column name in the CSV file)
actionAttrName=action

# The primary object for the ObjectsFeeder task. (This will always be USER")
primaryObject=USER

# The action to admin task mapping
ActionToTaskMapping = create.Create User;modify.Modify User;delete.Delete User

# Is the web service protected by SiteMinder
isProtectedBySiteMinder=false
```

PeopleSoft Feed Error and Response Handling

The PeopleSoft Feed reports to the user on the status of the SOAP request that is sent to the Bulk Loader task. The report will be to a standard out and/or log file. Only network connection, SOAP or TEWS errors are reported. A successful response to the request does not necessarily mean that the Identity Manager task has been processed without error. The task ID of each successfully submitted task will be output to allow cross-referencing with the User Console's View Submitted Tasks (VST) tab and Identity Manager log files.

The Bulk Loader task can be monitored as any other task using VST. Each nested IM task can be checked on this tab.

The XSLT transformation error is handled the same way as the errors mentioned above, for example, the error message is output to the standard out and the log file. When an error is encountered, it will log the error and exit without submitting the task.

PeopleSoft Feed Log Files

The following are examples of logging destinations and a logging configuration file for the PeopleSoft Feed:

- Logging to a standard out (console window) are set to `java.util.logging.Level.INFO` when the command line option `-verbose` is absent. The logging level is set to `java.util.logging.Level.CONFIG` when the option `-verbose` is set.
- Logging to the standard out (console window) will always be available no matter whether logging to a log file is configured or not.
- You can provide a logging configuration file to configure extra logging destination. The configuration file is set by starting the application with:

`java -Djava.util.logging.config.file=configFile MainClass`
- A logging configuration file will be provided to log the message to the file `imbulkloadclient.log` in the logs subdirectory in the application installation folder.

The following is an example of the logging configuration file:

```
# log to a file
handlers= java.util.logging.FileHandler

# global logging level. The valid settings are SEVERE, WARNING, INFO,
# CONFIG, FINE, FINER and FINEST
.level= INFO

# file handler configuration
java.util.logging.FileHandler.pattern = ../logs/imbulkloadclient.log
java.util.logging.FileHandler.append = true
java.util.logging.FileHandler.limit = 50000
java.util.logging.FileHandler.count = 1
java.util.logging.FileHandler.formatter = java.util.logging.SimpleFormatter
```

Axis Library Logging

The Axis library that we use as the stub classes to submit task to the Identity Manager Server has its own logging. A `log4j.properties` is provided in the `/conf` directory and is used to log to the log file `"axis.log"` in `/log` directory.

Appendix A: Endpoint Schema and Structure

This section contains the following topics:

[Endpoint Schema](#) (see page 505)

[Endpoint Structure](#) (see page 510)

This appendix provides an overview of the endpoint schema and structure of the Standard Connectors. The endpoint schema and structure is required when you:

- Use a general purpose LDAP utility to construct batch processes interfacing with the Provisioning Server
- Build, interpret, or modify LDAP Interface File Format (LDIF) files to work with Identity Manager data and combine it with data from other LDAP-enabled applications.

For more information about endpoint schemas and structures, see the **Administrator Guide** and the **Programming Guide for Provisioning**.

Endpoint Schema

An endpoint *schema* consists of the object classes, attributes in object classes, and attribute types. All of this information is necessary when constructing syntactically correct LDAP operations, such as LIST, SEARCH, ADD, MODIFY, and DELETE.

The endpoint schema for your directories is described in the following sections. The schema files discussed in these sections are located under the *PS_HOME*\Data\Endpoint TypeDefinition directory.

For more information about schemas, see the *Provisioning Guide* and the *Programming Guide for Provisioning*.

SchemaAbridged.txt File

The xxxSchemaAbridged.txt file provides a list of each object class and attribute in the schema. For each attribute, only the most commonly used keywords are supplied. Use this file if you are constructing LDAP-compatible files for any of the batch processes.

xxx

Specifies the three or four letter acronym for a connector. Click [here](#) (see page 506) to see the connector names and acronym's.

SchemaUnabridged.txt File

The xxxSchemaUnabridged.txt file provides a complete list of each object class and attribute in the schema and includes all the information provided in the xxxSchemaAbridged.txt file, as well as additional information required when parsing, formatting, and presenting the data received from the your Connector. Use this file if you need more detailed information for the object classes and attributes in the your Connector.

xxx

Specifies the three or four letter acronym for a connector. Click [here](#) (see page 506) to see the connector names and acronym's.

Connector Acronym's

The following list contains the acronym's for the Identity Manager connectors used in this appendix:

Acronym	Connector
N16	Windows NT
NSK	HP NSK
ETC	UNIX
ACF	CA-ACF2
TSS	CA-Top Secret
DBZ	DB2 UDB for z/OS
ORA	Oracle
SQL	MS SQL
ING	Ingres Server
ADS	Active Directory Services

Acronym	Connector
LDA	Generic LDAP
NDS	NDS
E2K	Windows 200x
LND	Lotus Notes/Domino
EIAM	Embedded IAM
ACC	CA-Access Control
PLS	SSO for Advanced Policy Server
UPO	Universal Provisioning
KRB	Kerberos
AS4	OS/400
VMS	OpenVMS
RAC	RACF
DB2	DB2 UDB
FND	Oracle Applications
SAP	SAP
SBL	Siebel
RSA	RSA ACE (SecurID)
PKI	PKI

File Formats

The format of these files is defined using two distinct definitions: object class definitions and attribute definition

Object Class Definitions

The lines that define the object classes are in the following form:

```
CLASS user_friendly_name  
  LDAP ObjectClass Name : ldap_name  
  ExternalName: external_name  
  NamingAttributes: naming_attribute
```

user_friendly_name

Specifies the user-friendly object class name.

ldap_name

Specifies the LDAP name used for defining the schema.

external_name

Specifies the relative distinguished name (RDN) value for containers.

naming_attribute

Specifies the RDN attribute.

Attribute Definitions

Directly beneath the object class definition are several attribute lines. These lines define the attribute types in the object class. Depending on which file you are viewing, the list can vary.

```
ATTRIBUTE (LDAP Name) ldap_object_class_name::ldap_attribute_name  
  User-friendly Name : user_friendly_name  
  Description: Global description  
  ProhibitedCharacters: prohibchars  
  MinLength: minlength  
  MaxLength: maxlength  
  EditType: edittype  
  IsSpaceAllowedIn: spaces  
  IsAsciiOnly: ascii  
  IsMultiValued: multi-valued  
  Case: case
```

ldap_object_class_name

Specifies the LDAP name used for the object class.

ldap_attribute_name

Specifies the LDAP name of the attribute.

user_friendly_name

Specifies the user-friendly name.

description

Specifies the description of the attribute.

prohibchars

Specifies a list of characters prohibited in the attribute.

minlength

Specifies the minimum length of the attribute value.

maxlength

Specifies the maximum length of the attribute value.

edittype

Determines the type of data in LDAP and its characteristics.

spaces

Defines a Boolean value that identifies whether spaces are allowed.

ascii

Defines a Boolean value that determines whether the attribute supports ASCII values.

multi-valued

Defines a Boolean value that determines whether the attribute is multi-valued.

case

Specifies a string that identifies whether the attribute can contain uppercase or lowercase characters. This string can be insensitive, insensitive-upper, insensitive-lower, sensitive, sensitive-upper, or sensitive-lower.

Endpoint Structure

Equally important to the endpoint schema is the hierarchical relationship that exists between the objects in the endpoint. This relationship is expressed through an endpoint structure called the Data Information Tree (DIT). Knowing the hierarchy is essential to constructing syntactically correct endpoint operations.

Distinguished Names

Distinguished names (DNs) identify the objects in a Endpoint Type. They contain a sequence of individual entries that specifies the location of an object in the DIT. That is, the DN is similar to a file system path name.

In Identity Manager, the format of the DN consists of two parts: a base DN and a domain name suffix. The base DN specifies the DN of an object without any domain information. You must specify only the base DN when writing batch processes.

For example, a base DN of an Active Directory Services object is:

```
eTADSAccountName=my_account,eTADSContainerName=Active Dir. Folder,
eTADSDirectoryName=directory_name,
eTNamespaceName=ActiveDirectory,domain_name_suffix
```

The domain name suffix specifies the suffix value of the domain. This parameter is the combination of the domain name RDN, its parent domain RDNs, and the Identity Manager suffix (dc=eta). You must specify the domain name suffix and the base DN when writing LDIF files. For example, if your domain name is chicago, its parent domain name is illinois, and the root domain name is usa, then the domain name suffix for your domain is:

```
dc=chicago,dc=illinois,dc=usa,dc=eta
```

Then, when accessing a logon ID using an Active Directory Services account, the DN would look like this:

```
eTADSAccountName=my_account,eTADSContainerName=Active Dir. Folder,
eTADSDirectoryName=directory_name,
eTNamespaceName=ActiveDirectory,dc=chicago,dc=illinois,dc=usa,dc=eta
```

Connector Objects and DNs

The following sections list the Connector objects and their DNs in hierarchical order:

DBZ Server Objects

The following table lists the DBZ Server objects and their DNs in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=DB2 ZOS Server, domain_name_suffix
eTDBZDirectory	eTDBZDirectoryName=directory_name, eTNamespaceName=DB2 ZOS Server, domain_name_suffix
eTDBZNamespace	eTNamespaceName=DB2 ZOS endpoint type name, eTDBZDirectoryName=directory_name, eTNamespaceName=DB2 ZOS Server, domain_name_suffix

LDAP Object Name	DN of Object Instance
eTDBZAccountContainer	eTDBZAccountContainerName=Accounts, eTDBZDirectoryName= directory_name , eTNamespaceName=DB2 ZOS Server, domain_name_suffix
eTDBZAccount	eTDBZAccountName=DB2 ZOS Account, eTDBZAccountContainerName=Accounts, eTDBZDirectoryName= directory_name , eTNamespaceName=DB2 ZOS Server, domain_name_suffix

DB2 UDB Objects

The following table lists the DB2 UDB objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=DB2 <i>Server, domain_name_suffix</i>
eTDB2Directory	eTDB2DirectoryName= <i>directory_name</i> , eTNamespaceName=DB2 <i>Server, domain_name_suffix</i>
eTDB2AccountContainer	eTDB2AccountContainerName=Accounts, eTDB2DirectoryName= <i>directory_name</i> , eTNamespaceName=DB2 <i>Server, domain_name_suffix</i>
eTDB2Account	eTDB2AccountName=DB2 User, eTDB2AccountContainerName=Accounts, eTDB2DirectoryName= <i>directory_name</i> , eTNamespaceName=DB2 <i>Server, domain_name_suffix</i>
eTDB2GroupContainer	eTDB2GroupContainerName=Groups, eTDB2DirectoryName= <i>directory_name</i> , eTNamespaceName=DB2 <i>Server, domain_name_suffix</i>
eTDB2Group	eTDB2GroupName=DB2Group, eTDB2GroupContainerName=Groups, eTDB2DirectoryName= <i>directory_name</i> , eTNamespaceName=DB2 <i>Server, domain_name_suffix</i>

MS SQL Server Objects

The following table lists the MS SQL Server objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTSQLEDirectory	eTSQLEDirectoryName= <i>directory_name</i> , eTNamespaceName=MS SQL Server, <i>domain_name_suffix</i> datalocation= BOTH (*) editttype=string minlen=1 maxlen=100 description=Directory Name
eTSQLELoginContainer	eTSQLELoginContainerName = MS SQL Logins, eTSQLEDirectoryName= <i>directory_name</i> , eTNamespaceName=MS SQL Server, <i>domain_name_suffix</i> datalocation= BOTH (*) editttype=string maxlen=255 description= MS SQL Server Login Container Name
eTSQLELogin	eTSQLELoginName= <i>MS SQL Login Name</i> eTSQLELoginContainerName=MS SQL Logins, eTSQLEDirectoryName= <i>directory_name</i> , eTNamespaceName=MS SQL Server, <i>domain_name_suffix</i> datalocation= BOTH (*) editttype=string maxlen=128 description= MS SQL Server Login Name
eTSQLEDatabase	eTSQLEDatabaseName = <i>database_name</i> , eTSQLEDirectoryName= <i>directory_name</i> , eTNamespaceName=MS SQL Server, <i>domain_name_suffix</i> datalocation= BOTH (*) editttype=string maxlen=123 description= MS SQL Server Database Name

LDAP Object Name	DN of Object Instance
eTSQUser	eTSQUserName = <i>MS_SQL_User_Name</i> , eSQLDatabaseName= <i>database_name</i> , eSQLDirectoryName= <i>directory_name</i> , eTNamespaceName=MS SQL Server, <i>domain_name_suffix</i> datalocation= BOTH (*) edittype=string maxlen=128 description= MS SQL Server User Name
eSQLRole	eSQLRoleName = <i>MS_SQL_Role_Name</i> , eSQLDatabaseName= <i>database_name</i> , eSQLDirectoryName= <i>directory_name</i> , eTNamespaceName=MS SQL Server, <i>domain_name_suffix</i> datalocation= BOTH (*) edittype=string maxlen=128 description= MS SQL Server Role Name

(*) datalocation= BOTH means that the object is stored in the Namespace server and in the Provisioning Directory.

Ingres Objects

The following table lists the Ingres objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTINGNamespace	eTINGNamespaceName=Ingres Server, <i>domain_name_suffix</i>
eTINGDirectory	eTINGDirectoryName= <i>directory_name</i> , eTINGNamespaceName=Ingres Server, <i>domain_name_suffix</i>
eTINGAccountContainer	eTINGAccountContainerName=Users, eTINGDirectoryName= <i>directory_name</i> , eTINGNamespaceName=Ingres Server, <i>domain_name_suffix</i>
eTINGAccount	eTINGAccountName=Ingres Account, eTINGAccountContainerName=Users, eTINGDirectoryName= <i>directory_name</i> , eTINGNamespaceName=Ingres Server, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTINGProfileContainer	eTINGProfileContainerName=Profiles, eTINGDirectoryName= <i>directory_name</i> , eTINGNamespaceName=Ingres Server, <i>domain_name_suffix</i>
eTINGProfile	eTINGProfileName=Ingres Profile, eTINGProfileContainerName=Profiles, eTINGDirectoryName= <i>directory_name</i> , eTINGNamespaceName=Ingres Server, <i>domain_name_suffix</i>
eTINGRoleContainer	eTINGRoleContainerName=Roles, eTINGDirectoryName= <i>directory_name</i> , eTINGNamespaceName=Ingres Server, <i>domain_name_suffix</i>
eTINGRole	eTINGRoleName=Ingres Role, eTINGRoleContainerName=Roles, eTINGDirectoryName= <i>directory_name</i> , eTINGNamespaceName=Ingres Server, <i>domain_name_suffix</i>
eTINGGroupContainer	eTINGGroupContainerName=Groups, eTINGDirectoryName= <i>directory_name</i> , eTINGNamespaceName=Ingres Server, <i>domain_name_suffix</i>
eTINGGroup	eTINGGroupName=Ingres Group, eTINGGroupContainerName=Groups, eTINGDirectoryName= <i>directory_name</i> , eTINGNamespaceName=Ingres Server, <i>domain_name_suffix</i>

Oracle Objects

The following table lists the Oracle objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=Oracle Server, <i>domain_name_suffix</i>
eTORADirectory	eTORADirectoryName= <i>directory_name</i> , eTNamespaceName=Oracle Server, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTORAAccountContainer	eTORAAccountContainerName=Accounts, eTORADirectoryName= <i>directory_name</i> , eTNamespaceName=Oracle Server, <i>domain_name_suffix</i>
eTORAAccount	eTORAAccountName= <i>account_name</i> , eTORAAccountContainerName=Oracle Accounts, eTORADirectoryName= <i>directory_name</i> , eTNamespaceName=Oracle Server, <i>domain_name_suffix</i>
eTORARoleContainer	eTORARoleContainerName=Roles, eTORADirectoryName= <i>directory_name</i> , eTNamespaceName=Oracle Server, <i>domain_name_suffix</i>
eTORARole	eTORARole= <i>role_name</i> , eTORARoleContainerName=Roles, eTORADirectoryName= <i>directory_name</i> , eTNamespaceName=Oracle Server, <i>domain_name_suffix</i>
eTORAProfileContainer	eTORAProfileContainerName=Profiles, eTORADirectoryName= <i>directory_name</i> , eTNamespaceName=Oracle Server, <i>domain_name_suffix</i>
eTORAProfile	eTORAProfile= <i>profile_name</i> , eTORAProfileContainerName=Profiles, eTORADirectoryName= <i>directory_name</i> , eTNamespaceName=Oracle Server, <i>domain_name_suffix</i>
eTORAProcContainer	eTORAProcContainerName=Procedures, eTORADirectoryName= <i>directory_name</i> , eTNamespaceName=Oracle Server, <i>domain_name_suffix</i>
eTORAProc	eTORAProc= <i>procedure_name</i> , eTORAProcContainerName=Procedures, eTORADirectoryName= <i>directory_name</i> , eTNamespaceName=Oracle Server, <i>domain_name_suffix</i>
eTORAPkgContainer	eTORAPkgContainerName=Packages, eTORADirectoryName= <i>directory_name</i> , eTNamespaceName=Oracle Server, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTORAPkg	eTORAPkg= <i>package_name</i> , eTORAPkgContainerName=Packages, eTORADirectoryName= <i>directory_name</i> , eTNamespaceName=Oracle Server, <i>domain_name_suffix</i>

Oracle Applications Objects

The following table lists the Oracle Applications objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=Oracle Applications, <i>domain_name_suffix</i>
eTFNDDirectory	eTFNDDirectoryName= <i>directory_name</i> , eTNamespaceName=Oracle Applications, <i>domain_name_suffix</i>
eTFNDAccountContainer	eTFNDAccountContainerName=Users, eTFNDDirectoryName= <i>directory_name</i> , eTNamespaceName=Oracle Applications, <i>domain_name_suffix</i>
eTFNDAccount	eTFNDUserName= <i>user_name</i> , eTFNDAccountContainerName=Users, eTFNDDirectoryName= <i>directory_name</i> , eTNamespaceName=Oracle Applications, <i>domain_name_suffix</i>

Windows NT Objects

The following table lists the Windows NT objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=Windows NT, <i>domain_name_suffix</i>
eTN16Directory	eTN16DirectoryName= <i>directory_name</i> , eTNamespaceName=Windows NT, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTN16AccountContainer	eTN16AccountContainerName=Accounts, eTN16DirectoryName= <i>directory_name</i> , eTNamespaceName=Windows <i>NT, domain_name_suffix</i>
eTN16Account	eTN16AccountName=Windows NT Account, eTN16AccountContainerName=Accounts, eTN16DirectoryName= <i>directory_name</i> , eTNamespaceName=Windows <i>NT, domain_name_suffix</i>
eTN16GroupContainer	eTN16GroupContainerName=N16GroupContainer, eTN16DirectoryName= <i>directory_name</i> , eTNamespaceName=Windows <i>NT, domain_name_suffix</i>
eTN16Groups	eTN16GroupName=N16Group, eTN16GroupContainerName=N16GroupContainer, eTN16DirectoryName= <i>directory_name</i> , eTNamespaceName=Windows <i>NT, domain_name_suffix</i>
eTN16SharedFolder Container	eTN16SharedFolderContainerName=Shared Folders, eTN16DirectoryName= <i>directory_name</i> , eTNamespaceName=Windows <i>NT, domain_name_suffix</i>
eTN16SharedFolder	eTN16SharedFolderName=N16SharedFolder, eTN16SharedFolderContainerName=Shared Folders, eTN16DirectoryName= <i>directory_name</i> , eTNamespaceName=Windows <i>NT, domain_name_suffix</i>
eTn16FolderManager	eTN16FolderManagerName=SF_MNGT_name, eTN16DirectoryName= <i>directory_name</i> , eTNamespaceName=Windows <i>NT, domain_name_suffix</i>
eTN16GroupManager	eTN16GroupManagerName=GRP_MNGT_Name, eTN16DirectoryName= <i>directory_name</i> , eTNamespaceName=Windows <i>NT, domain_name_suffix</i>

CA-ACF2 Objects

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=CA-ACF2, <i>domain_name_suffix</i>
eTACFDirectory	eTACFDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-ACF2, <i>domain_name_suffix</i>
eTACFLidContainer	eTACFLidContainerName=Accounts, eTACFDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-ACF2, <i>domain_name_suffix</i>
eTACFLid	eTACFLidName= <i>your_lid</i> ,eTACFLidContainerName=Accounts, eTACFDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-ACF2, <i>domain_name_suffix</i>
eTACFRuleContainer	eTACFRuleContainerName=Rules, eTACFDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-ACF2, <i>domain_name_suffix</i>
eTACFACF2RuleType	ETACFACF2RuleTypeName=ACF2_rule_types, eTACFRuleContainerName=Rules, eTACFDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-ACF2, <i>domain_name_suffix</i>
eTACFACF2RuleKey	eTACFACF2RuleKeyName=\$_key_value, eTACFACF2RuleTypeName=ACF2_rule_types, eTACFRuleContainerName=Rules, eTACFDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-ACF2, <i>domain_name_suffix</i>
eTACFACF2RuleLine	eTACFACF2RuleLineName= <i>highlevel_qualifier_mask</i> , eTACFACF2AssetTypeName= <i>assets_under_\$_key</i> , eTACFACF2RuleKeyName=\$_key_value, eTACFRuleTypeName=ACF2_rule_types, eTACFRuleContainerName=Rules, eTACFDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-ACF2, <i>domain_name_suffix</i>

CA-Top Secret Objects

The following table lists the CA-Top Secret objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSDirectory	eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSAcid	eTTSSAcidName= <i>acid_name</i> , eTTSSAcidContainerName=Accounts, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSFacilityContainer	eTTSSFacilityContainerName =Facilities, eTTSSAcidName= <i>acid_name</i> , eTTSSAcidContainerName=Accounts, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSPermissionContainer	eTTSSPermissionContainerName =Permissions, <i>eTTSSAcidName=acid_name</i> , eTTSSAcidContainerName=Accounts, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSOwnershipContainer	eTTSSOwnershipContainerName =Ownerships, <i>eTTSSAcidName=acid_name</i> , eTTSSAcidContainerName=Accounts, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSProfListContainer	eTTSSProfListContainerName =ProfList, <i>eTTSSAcidName=acid_name</i> , eTTSSAcidContainerName=Accounts, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTTSSAdminFacContainer	eTTSSAdminFacContainerName = AdminFacility, eTTSSAcidName=acid_name, eTTSSAcidContainerName=Accounts, eTTSSDirectoryName=directory_name, eTNamespaceName=CA-Top Secret,domain_name_suffix
eTTSSAdminResourceContainer	eTTSSAdminResourceContainerName = AdminResource, eTTSSAcidName=acid_name, eTTSSAcidContainerName=Accounts, eTTSSDirectoryName=directory_name, eTNamespaceName=CA-Top Secret,domain_name_suffix
eTTSSAdminScopeContainer	eTTSSAdminScpContainerName = AdminScope, eTTSSAcidName=acid_name, eTTSSAcidContainerName=Accounts, eTTSSDirectoryName=directory_name, eTNamespaceName=CA-Top Secret,domain_name_suffix
eTTSSAdminFacility	eTTSSAdminFacilityName = admin_facility_name, eTTSSAdminFacContainerName = AdminFacility, eTTSSAcidName=acid_name, eTTSSAcidContainerName=Accounts, eTTSSDirectoryName=directory_name, eTNamespaceName=CA-Top Secret,domain_name_suffix
eTTSSAdminResource	eTTSSAdminResourceName = admin_resource_name, eTTSSAdminResContainerName = AdminResource, eTTSSAcidName=acid_name, eTTSSAcidContainerName=Accounts, eTTSSDirectoryName=directory_name, eTNamespaceName=CA-Top Secret,domain_name_suffix
eTTSSAdminScope	eTTSSAdminScopeName = admin_scope_name, eTTSSAdminScpContainerName = AdminScope, eTTSSAcidName=acid_name, eTTSSAcidContainerName=Accounts, eTTSSDirectoryName=directory_name, eTNamespaceName=CA-Top Secret,domain_name_suffix
eTTSSAcidContainer	eTTSSAcidContainerName=Accounts, eTTSSendpointName=directory_name, eTNamespaceName=CA-Top Secret,domain_name_suffix

LDAP Object Name	DN of Object Instance
eTTSSDeptContainer	eTTSSDeptContainerName= Departments, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSDept	eTTSSDeptName = <i>dept_name</i> , eTTSSDeptContainerName= Departments, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSDivContainer	eTTSSDivContainerName= Divisions, eTTSSDirectoryName= <i>endpoint_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSDiv	eTTSSDivName= <i>div_name</i> , eTTSSDivContainerName= Divisions, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSZoneContainer	eTTSSZoneContainerName= Zones, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSAcidZone	eTTSSZoneName= <i>zone_name</i> , eTTSSZoneContainerName= Zones, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSGroupContainer	eTTSSGroupContainerName= Groups, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSGroup	eTTSSGroupName = <i>group_name</i> , eTTSSGroupContainerName= Groups, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSFacility	eTTSSFacilityName= <i>facility_name</i> , eTTSSFacilityContainerName=Facilities, eTTSSAcidName= <i>acid_name</i> , eTTSSAcidContainerName=Accounts, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTTSSOwnedName	eTTSSOwnedNameName= <i>detail_name</i> , eTTSSOwnershipName= <i>owned_name</i> , eTTSSOwnershipContainerName=Ownerships, eTTSSAcidName= <i>acid_name</i> , eTTSSAcidContainerName=Accounts, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSOwned	eTTSSOwnershipName= <i>owned_name</i> , eTTSSOwnershipContainerName=Ownerships, eTTSSAcidName= <i>acid_name</i> , eTTSSAcidContainerName=Accounts, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSResName	eTTSSResNameName= <i>permission_detail_name</i> , eTTSSResClassName= <i>permission_name</i> , eTTSSPermissionContainerName=Permissions, eTTSSAcidName= <i>acid_name</i> , eTTSSAcidContainerName=Accounts, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSResClass	eTTSSResClassName= <i>permission_name</i> , eTTSSPermissionContainerName=Permissions, eTTSSAcidName= <i>acid_name</i> , eTTSSAcidContainerName=Accounts, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSProfList	eTTSSProfListName= <i>profile_name</i> , eTTSSProfListContainerName =ProfList, eTTSSAcidName= <i>acid_name</i> , eTTSSAcidContainerName=Accounts, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>
eTTSSProfile	eTTSSProfileName= <i>profile_name</i> , eTTSSProfileContainerName=Profiles, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTTSSProfileContainer	eTTSSProfileContainerName=Profiles, eTTSSDirectoryName= <i>directory_name</i> , eTNamespaceName=CA-Top Secret, <i>domain_name_suffix</i>

RACF Objects

The following table lists the RACF objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=RAC Namespace, <i>domain_name_suffix</i>
eTRACDirectory	eTRACDirectoryName= <i>directory_name</i> , eTNamespaceName=RAC Namespace, <i>domain_name_suffix</i>
eTRACAccount	eTRACAccount= <i>user_id</i> ,eTRACAccountContainerName=Accounts, eTRACDirectoryName= <i>directory_name</i> , eTNamespaceName=RAC Namespace, <i>domain_name_suffix</i>
eTRACAccountContainer	eTRACAccountContainerName=Accounts, eTRACDirectoryName= <i>directory_name</i> , eTNamespaceName=RAC Namespace, <i>domain_name_suffix</i>
eTRACUserResProfile	eTRACPermissionEntry= <i>entry_name</i> , eTRACPermissionClass= <i>class_name</i> , eTRACUserPermissionContainerName=Permissions, eTRACDirectoryName= <i>directory_name</i> , eTNamespaceName=RAC Namespace, <i>domain_name_suffix</i>
eTRACUserPermissionRe sClass	eTRACPermissionClass= <i>class_name</i> , eTRACUserPermissionContainerName=Permissions, eTRACDirectoryName= <i>directory_name</i> , eTNamespaceName=RAC Namespace, <i>domain_name_suffix</i>
eTRACUserPermissionCo ntainer	eTRACUserPermissionContainerName=Permissions, eTRACDirectoryName= <i>directory_name</i> , eTNamespaceName=RAC Namespace, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTRACGroupUser	eTRACGroupUserid= <i>user_id</i> , eTRACgroupid= <i>group_id</i> , eTRACGroupContainer=Groups, eTRACDirectoryName= <i>directory_name</i> , eTNamespaceName=RAC Namespace, <i>domain_name_suffix</i>
eTRACGroup	eTRACgroupid= <i>group_id</i> , eTRACGroupContainer=Groups, eTRACDirectoryName= <i>directory_name</i> , eTNamespaceName=RAC Namespace, <i>domain_name_suffix</i>
eTRACGroupContainer	eTRACGroupContainer=Groups, eTRACDirectoryName= <i>directory_name</i> , eTNamespaceName=RAC Namespace, <i>domain_name_suffix</i>
eTRACResUser	eTRACPermissionUser= <i>user_id</i> , eTRACPermissionEntry=RACF Permission Entry, eTRACPermissionClass= <i>class_name</i> , eTRACPermissionContainerName=Permissions, eTRACDirectoryName= <i>directory_name</i> , eTNamespaceName=RAC Namespace, <i>domain_name_suffix</i>
eTRACResProfile	eTRACPermissionEntry=RACF Permission Entry, eTRACPermissionClass= <i>class_name</i> , eTRACPermissionContainerName=Permissions,
eTRACPermissionResClasses	eTRACPermissionClass= <i>class_name</i> , eTRACPermissionContainerName=Permissions, eTRACDirectoryName= <i>directory_name</i> , eTNamespaceName=RAC Namespace, <i>domain_name_suffix</i>
eTRACPermissionContainer	eTRACPermissionContainerName=Permissions, eTRACDirectoryName= <i>directory_name</i> , eTNamespaceName=RAC Namespace, <i>domain_name_suffix</i>

OS/400 Objects

The following table lists the OS/400 objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=OS400 Namespace, <i>domain_name_suffix</i>
eTAS4Directory	eTAS4DirectoryName= <i>directory_name</i> , eTNamespaceName=OS400 Namespace, <i>domain_name_suffix</i>
eTAS4AccountContainer	eTAS4AccountContainerName=Accounts, eTAS4DirectoryName= <i>directory_name</i> , eTNamespaceName=OS400 Namespace, <i>domain_name_suffix</i>
eTAS4Account	eTAS4UserProfileName= <i>OS/400_account</i> , eTAS4AccountContainerName=Accounts, eTAS4DirectoryName= <i>directory_name</i> , eTNamespaceName=OS400 Namespace, <i>domain_name_suffix</i>
eTAS4GroupContainer	eTAS4GroupContainerName=Groups, eTAS4DirectoryName= <i>directory_name</i> , eTNamespaceName=OS400 Namespace, <i>domain_name_suffix</i>
eTAS4Group	eTAS4GroupName= <i>OS/400_group</i> , eTAS4GroupContainerName=Groups, eTAS4DirectoryName= <i>directory_name</i> , eTNamespaceName=OS400 Namespace, <i>domain_name_suffix</i>

OpenVMS Objects

The following table lists the OpenVMS objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=OpenVMS, <i>domain_name_suffix</i>
eTOpenVMSDirectory	eTVMSDirectoryName= <i>directory_name</i> , eTNamespaceName=OpenVMS, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTOpenVMSAccountContainer	eTVMSAccountContainerName=User Accounts, eTVMSDirectoryName= <i>directory_name</i> , eTNamespaceName=OpenVMS, <i>domain_name_suffix</i>
eTOpenVMSAccount	eTVMSAccountName= <i>account_name</i> , eTVMSAccountContainerName=User Accounts, eTVMSDirectoryName= <i>directory_name</i> , eTNamespaceName=OpenVMS, <i>domain_name_suffix</i>

HP NonStop SAFEGUARD Objects

The following table lists the HP NonStop SAFEGUARD objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName= HP NonStop Safeguard, <i>domain_name_suffix</i>
eTNSKDirectory	eTNSKDirectoryName= <i>directory_name</i> , eTNamespaceName= HP NonStop Safeguard, <i>domain_name_suffix</i>
eTNSKAccountContainer	eTNSKAccountContainerName=Accounts, eTNSKDirectoryName= <i>directory_name</i> , eTNamespaceName= HP NonStop Safeguard, <i>domain_name_suffix</i>
eTNSKAccount	eTNSKAccountName= <i>account name</i> , eTNSKAccountContainerName=Accounts, eTNSKDirectoryName= <i>directory_name</i> , eTNamespaceName= HP NonStop Safeguard, <i>domain_name_suffix</i>
eTNSKGroupContainer	eTNSKGroupContainerName=Groups, eTNSKDirectoryName= <i>directory_name</i> , eTNamespaceName= HP NonStop Safeguard, <i>domain_name_suffix</i>
eTNSKGroup	eTNSKGroupName= <i>group name</i> , eTNSKGroupContainerName=Groups, eTNSKDirectoryName= <i>directory_name</i> , eTNamespaceName= HP NonStop Safeguard, <i>domain_name_suffix</i>

UNIX ETC Objects

The following table lists the UNIX ETC objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=UNIX - <i>etc, domain_name_suffix</i>
eTETCDirectory	eTETCDirectoryName= <i>directory_name</i> , eTNamespaceName=UNIX - <i>etc, domain_name_suffix</i>
eTETCAccountContainer	eTETCAccountContainerName= <i>container_name</i> , eTETCDirectoryName= <i>directory_name</i> , eTNamespaceName=UNIX - <i>etc, domain_name_suffix</i>
eTETCAccount	eTETCAccountName= <i>account_name</i> , eTETCAccountContainerName= <i>container_name</i> , eTETCDirectoryName= <i>directory_name</i> , eTNamespaceName=UNIX - <i>etc, domain_name_suffix</i>
eTETCGroupContainer	eTETCGroupContainerName= <i>group_container_name</i> , eTETCDirectoryName= <i>directory_name</i> , eTNamespaceName=UNIX - <i>etc, domain_name_suffix</i>
eTETCGroup	eTETCGroupName= <i>group_name</i> , eTETCGroupContainerName= <i>group_container_name</i> , eTETCDirectoryName= <i>directory_name</i> , eTNamespaceName=UNIX - <i>etc, domain_name_suffix</i>

UNIX NIS Objects

The following table lists the UNIX NIS objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=UNIX - NIS-NIS plus <i>Domains, domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTNISDirectory	eTNISDirectoryName= <i>directory_name</i> , eTNamespaceName=UNIX - NIS-NIS plus <i>Domains,domain_name_suffix</i>
eTNISAccountContainer	eTNISAccountContainerName= <i>container_name</i> , eTNISDirectoryName= <i>directory_name</i> , eTNamespaceName=UNIX - NIS-NIS plus <i>Domains,domain_name_suffix</i>
eTNISAccount	eTNISAccountName= <i>account_name</i> , eTNISAccountContainerName= <i>container_name</i> , eTNISDirectoryName= <i>directory_name</i> , eTNamespaceName=UNIX - NIS-NIS plus <i>Domains,domain_name_suffix</i>
eTNISGroupContainer	eTNISGroupContainerName= <i>group_container_name</i> , eTNISDirectoryName= <i>directory_name</i> , eTNamespaceName=UNIX - NIS-NIS plus <i>Domains,domain_name_suffix</i>
eTNISGroup	eTNISGroupName= <i>group_name</i> , eTNISGroupContainerName= <i>group_container_name</i> , eTNISDirectoryName= <i>directory_name</i> , eTNamespaceName=UNIX - NIS-NIS plus <i>Domainsdomain_name_suffix</i>
eTNISNetGroupContainer	eTNISNetGroupContainerName= <i>netgroup_container_name</i> , eTNISDirectoryName= <i>directory_name</i> , eTNamespaceName=UNIX - NIS-NIS plus <i>Domains,domain_name_suffix</i>
eTNISNetGroup	eTNISNetGroupName= <i>netgroup_name</i> , eTNISNetGroupContainerName= <i>netgroup_container_name</i> , eTNISDirectoryName= <i>directory_name</i> , eTNamespaceName=UNIX - NIS-NIS plus <i>Domains,domain_name_suffix</i>

Active Directory Services Objects

The following table lists the Active Directory Services objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=ActiveDirectory, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTADSDirectory	eTADSDirectoryName= <i>directory_name</i> , eTNamespaceName=ActiveDirectory, <i>domain_name_suffix</i>
eTADSContainer	eTADSContainerName=Active Dir. Folder, eTADSDirectoryName= <i>directory_name</i> , eTNamespaceName=ActiveDirectory, <i>domain_name_suffix</i>
eTADSAccount	eTADSAccountName= <i>account_name</i> , eTADSContainerName=Active Dir. Folder, eTADSDirectoryName= <i>directory_name</i> , eTNamespaceName=ActiveDirectory, <i>domain_name_suffix</i>
eTADSOrgUnit	eTADSOrgUnitName=Active Dir. Org. Unit, eTADSDirectoryName= <i>directory_name</i> , eTNamespaceName=ActiveDirectory, <i>domain_name_suffix</i>
eTADSGroup	eTADSGroupName=Active Dir. Group, eTADSOrgUnitName=Active Dir. Org. Unit, eTADSDirectoryName= <i>directory_name</i> , eTNamespaceName=ActiveDirectory, <i>domain_name_suffix</i>

LDAP Objects

The following table lists the LDAP objects and their DN's in hierarchical order. Remember that accounts, groups, and containers can appear anywhere under the directory entry and under any container entry.

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=LDAP Endpoint Type, <i>domain_name_suffix</i>
eTLADirectory	eTLADirectoryName= <i>directory_name</i> , eTNamespaceName=LDAP Endpoint Type, <i>domain_name_suffix</i>
eTLDAccount	eTLDAccountName= <i>account_name</i> , [eTLDAccountName= <i>container_name</i> , ...] (zero or more containers) eTLADirectoryName= <i>directory_name</i> , eTNamespaceName=LDAP Endpoint Type, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTLDAContainer	[eTLDAContainerName= <i>container_name</i> , ...] (zero or more containers) eTLDADirectoryName= <i>directory_name</i> , eTNamespaceName=LDAP Endpoint Type, <i>domain_name_suffix</i>
eTLDAGroup	eTLDAGroupName= <i>group_name</i> , [eTLDAContainerName= <i>container_name</i> , ...] (zero or more containers) eTLDADirectoryName= <i>directory_name</i> , eTNamespaceName=LDAP Endpoint Type, <i>domain_name_suffix</i>

NDS Objects

The following table lists the NDS objects and their DN's in hierarchical order:

Note: NDS has optional and mandatory containers that may exist within each other. For this reason, this table does not include every possible DN combination that can exist within an NDS environment.

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
eTNDSTree	eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
eTNDSCountry	eTNDSCountryName= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
eTNSLocality	eTNSLocalityName= <i>locality name</i> , eTNDSCountryName= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNSLocalityName= <i>locality name</i> , eTNSOrganizationName= <i>organization_name</i> , eTNDSCountryName= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
	eTNSDLocalityName= <i>locality_name</i> , eTNSDOrganizationName= <i>organization_name</i> , eTNSDTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNSDOrganizationName= <i>organization_name</i> , eTNSDLocalityName= <i>locality_name</i> , eTNSDOrganizationName= <i>organization_name</i> , eTNSDCountryName= <i>country_name</i> , eTNSDTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNSDOrganizationName= <i>organization_name</i> , eTNSDLocalityName= <i>locality_name</i> , eTNSDOrganizationName= <i>organization_name</i> , eTNSDTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNSDOrgUnitName= <i>organizational_unit</i> eTNSDLocalityName= <i>locality_name</i> , eTNSDOrganizationName= <i>organization_name</i> , eTNSDCountryName= <i>country_name</i> , eTNSDTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNSDOrgUnitName= <i>organizational_unit</i> eTNSDLocalityName= <i>locality_name</i> , eTNSDOrganizationName= <i>organization_name</i> , eTNSDTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNSDLocalityName= <i>locality_name</i> , eTNSDLocalityName= <i>locality_name</i> , eTNSDOrganizationName= <i>organization_name</i> , eTNSDCountryName= <i>country_name</i> , eTNSDTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNSDLocalityName= <i>locality_name</i> , eTNSDLocalityName= <i>locality_name</i> , eTNSDOrganizationName= <i>organization_name</i> , eTNSDTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTNDSEOrganization	eTNDSEOrganizationName= <i>organization_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSEOrganizationName= <i>organization_name</i> , eTNDSCountryName= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSEOrgUnitName= <i>organizational_unit</i> , eTNDSEOrganizationName= <i>organization_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSEOrgUnitName= <i>organizational_unit</i> , eTNDSEOrganizationName= <i>organization_name</i> , eTNDSCountryName= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSEAccountName= <i>account_name</i> , eTNDSEOrganizationName= <i>organization_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSEAccountName= <i>account_name</i> , eTNDSEOrganizationName= <i>organization_name</i> , eTNDSCountryName= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSEGroupName= <i>group_name</i> , eTNDSEOrganizationName= <i>organization_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSEGroupName= <i>group_name</i> , eTNDSEOrganizationName= <i>organization_name</i> , eTNDSCountryName= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>

[illegible]

LDAP Object Name	DN of Object Instance
	eTNDSErgUnitName= <i>organizational_unit_name</i> , eTNDSErgUnitName= <i>organizational_unit</i> , eTNDSErganizationName= <i>organization_name</i> , eTNDSCountryName= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
eTNDSErgUnit	eTNDSErgUnitName= <i>organizational_unit</i> eTNDSErganizationName= <i>organization_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSErgUnitName= <i>organizational_unit</i> , eTNDSErganizationName= <i>organization_name</i> , eTNDSCountry= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSErgUnitName= <i>organizational_unit</i> , eTNDSErgUnitName= <i>locality_name</i> , eTNDSErganizationName= <i>organization_name</i> , eTNDSCountry= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSErgUnitName= <i>organizational_unit</i> , eTNDSErgUnitName= <i>locality_name</i> , eTNDSErganizationName= <i>organization_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSErgUnitName= <i>organizational_unit</i> eTNDSErgUnitName= <i>organizational_unit</i> eTNDSErganizationName= <i>organization_name</i> , eTNDSCountry= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
	eTNDSSAccountName= <i>account_name</i> eTNDSErgUnitName= <i>organizational_unit</i> eTNDSErganizationName= <i>organization_name</i> , eTNDSCountry= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSEGroupNane= <i>group_name</i> eTNDSErgUnitName= <i>organizational_unit</i> eTNDSErganizationName= <i>organization_name</i> , eTNDSCountry= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSELocalityName= <i>locality_name</i> eTNDSErgUnitName= <i>organizational_unit</i> eTNDSErganizationName= <i>organization_name</i> , eTNDSCountry= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
eTNDSEGroup	eTNDSEGroupNane= <i>group_name</i> , eTNDSErganizationName= <i>organization_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSEGroupNane= <i>group_name</i> , eTNDSErganizationName= <i>organization_name</i> , eTCountryName= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
eTNDSEAccount	eTNDSEAccountName= <i>account_name</i> , eTNDSErganizationName= <i>organization_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>
	eTNDSEAccountName= <i>account_name</i> , eTNDSErganizationName= <i>organization_name</i> , eTCountryName= <i>country_name</i> , eTNDSTreeName= <i>directory_name</i> , eTNamespaceName=NDS <i>Servers, domain_name_suffix</i>

Universal Provisioning Objects

The following table lists the Universal Provisioning objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=Universal Provisioning, <i>domain_name_suffix</i>
eTUPODirectory	eTUPODirectoryName= <i>directory_name</i> , eTNamespaceName=Universal Provisioning, <i>domain_name_suffix</i>
eTUPOAccountContainer	eTUPOAccountContainerName=Users, eTUPODirectoryName= <i>directory_name</i> , eTNamespaceName=Universal Provisioning, <i>domain_name_suffix</i>
eTUPOAccount	eTUPOAccountName= <i>user_name</i> , eTUPOAccountContainerName=Users, eTUPODirectoryName= <i>directory_name</i> , eTNamespaceName=Universal Provisioning, <i>domain_name_suffix</i>
eTUPOExitContainer	eTUPOExitContainerName=Program Exits, eTNamespaceName=Universal Provisioning, <i>domain_name_suffix</i>
eTUPOExit	eTUPOExitName= <i>exit_name</i> , eTUPOExitContainerName=Program Exits, eTNamespaceName=Universal Provisioning, dc=cai,dc=etadb

Lotus Notes\Domino Objects

The following table lists the Lotus Notes\Domino objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=Lotus Domino <i>Server, domain_name_suffix</i>
eTLNDDirectory	eTLNDDirectoryName= <i>directory_name</i> , eTNamespaceName=Lotus Domino <i>Server, domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTLNDCountry	eTLNDCountryName=Lotus Domino Country, eTLNDDirectoryName= <i>directory_name</i> , eTNamespaceName=Lotus Domino <i>Server, domain_name_suffix</i>
eTLNDFlatCertifier	eTLNDFlatCertifier=Lotus Domino Flat Certifier, eTLNDDirectoryName= <i>directory_name</i> , eTNamespaceName=Lotus Domino <i>Server, domain_name_suffix</i>
eTLNDAccount	eTLNDAccountName= <i>account_name</i> , eTLNDOrganizationalUnit=Lotus Domino Organizational Unit, eTLNDOrganization=Lotus Domino Organization, eTLNDDirectoryName= <i>directory_name</i> , eTNamespaceName=Lotus Domino <i>Server, domain_name_suffix</i>
eTLNDOrganizationalUnit	eTLNDOrganizationalUnit=Lotus Domino Organizational Unit, eTLNDOrganization=Lotus Domino Organization, eTLNDDirectoryName= <i>directory_name</i> , eTNamespaceName=Lotus Domino <i>Server, domain_name_suffix</i>
eTLNDOrganization	eTLNDOrganization=Lotus Domino Organization, eTLNDDirectoryName= <i>directory_name</i> , eTNamespaceName=Lotus Domino <i>Server, domain_name_suffix</i>
eTLNDGroupContainer	eTLNDGroupContainerName=LND Groups, eTLNDDirectoryName= <i>directory_name</i> , eTNamespaceName=Lotus Domino <i>Server, domain_name_suffix</i>
eTLNDGroup	eTLNDGroupName= <i>group_name</i> , eTLNDGroupContainerName=LND Groups, eTLNDDirectoryName= <i>directory_name</i> , eTNamespaceName=Lotus Domino <i>Server, domain_name_suffix</i>

CA Access Control Objects

The following table lists the CA Access Control objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=Access Control, domain_name_suffix
eTACCDirectory	eTACCDirectoryName=directory_name, eTNamespaceName=Access Control, domain_name_suffix
eTACCAccountContainer	eTACCAccountContainerName=Accounts, eTACCDirectoryName=directory_name, eTNamespaceName=Access Control, domain_name_suffix
eTACCAccount	eTACCAccountName=account_name, eTACCAccountContainerName=Accounts, eTACCDirectoryName=directory_name, eTNamespaceName=Access Control, domain_name_suffix
eTACCGroupContainer	eTACCGroupContainerName=Groups, eTACCDirectoryName=directory_name, eTNamespaceName=Access Control, domain_name_suffix
eTACCGroup	eTACCGroupName=group_name, eTACCGroupContainerName=Groups, eTACCDirectoryName=directory_name, eTNamespaceName=Access Control, domain_name_suffix

PLS Objects

The following table lists the PLS objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace Type	eTNamespaceName=CA SSO WAC, domain_name_suffix
eTPLSDirectory	eTPLSDirectoryName=directory_name, eTNamespaceName=CA SSO WAC, domain_name_suffix

LDAP Object Name	DN of Object Instance
eTPLSUserStore	eTPLSUserStoreName=user store name, eTPLSDirectoryName=directory_name, eTNamespaceName=CA SSO WAC, domain_name_suffix
eTPLSAccount	eTPLSAccountName=account_name, [eTPLSContainerName=container_name, ...](zero or more containers) eTPLSUserStoreName=user store name, eTPLSDirectoryName=directory_name, eTNamespaceName=CA SSO WAC, domain_name_suffix
eTPLSGroup	eTPLSGroupName=group_name, [eTPLSContainerName=container_name, ...](zero or more containers) eTPLSUserStoreName=user store name, eTPLSDirectoryName=directory_name, eTNamespaceName=CA SSO WAC, domain_name_suffix
eTPLSContainer	[eTPLSContainerName=container_name, ...](zero or more containers) eTPLSUserStoreName=user store name, eTPLSDirectoryName=directory_name, eTNamespaceName=CA SSO WAC, domain_name_suffix
eTPLSApplicationContainer	eTPLSApplicationContainerName=Applications, eTPLSDirectoryName=directory_name, eTNamespaceName=CA SSO WAC, domain_name_suffix
eTPLSApplication	eTPLSApplicationName=Application_name, eTPLSApplicationContainerName=Applications, eTPLSDirectoryName=directory_name, eTNamespaceName=CA SSO WAC, domain_name_suffix
eTPLSApplicationGroupCon tainer	eTPLSApplicationGroupContainerName=Applicati on Groups, eTPLSDirectoryName=directory_name, eTNamespaceName=CA SSO WAC, domain_name_suffix

LDAP Object Name	DN of Object Instance
eTPLSApplicationGroup	eTPLSApplicationGroupName=application_group_name, eTPLSApplicationGroupContainerName=Application Groups, eTPLSDirectoryName=directory_name, eTNamespaceName=CA SSO WAC, domain_name_suffix
eTPLSTerminalContainer	eTPLSTerminalContainerName=Terminals eTPLSDirectoryName=directory_name, eTNamespaceName=CA SSO WAC, domain_name_suffix
eTPLSTerminal	eTPLSTerminalName=Terminal_name eTPLSTerminalContainerName=Terminals, eTPLSDirectoryName=directory_name, eTNamespaceName=CA SSO WAC, domain_name_suffix
eTPLSAuthhostContainer	eTPLSAuthhostContainerName=Authentication Hosts, eTPLSDirectoryName=directory_name, eTNamespaceName=CA SSO WAC, domain_name_suffix
eTPLSAuthhost	eTPLSAuthhostName=Authhost_name, eTPLSAuthhostContainerName=Authentication Hosts, eTPLSDirectoryName=directory_name, eTNamespaceName=CA SSO WAC, domain_name_suffix

KRB Objects

The following table lists the KRB objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespace=KRB Namespace,domain_name_suffix
eTKRBDirectory	eTKRBDirectoryName=directory_name, eTNamespace=KRB Namespace,domain_name_suffix
eTKRBAccountContainer	eTKRBAccountContainerName=Accounts, eTKRBDirectoryName=directory_name, eTNamespace=KRB Namespace,domain_name_suffix

LDAP Object Name	DN of Object Instance
eTKRBAccount	eTKRBAccountName= <i>account_name</i> , eTKRBAccountContainerName=Accounts, eTKRBDirectoryName= <i>directory_name</i> , eTNamespace= <i>KRB Namespace, domain_name_suffix</i>
eTKRBAccount	eTKRBAccountName=KRBUser, eTKRBAccountContainerName=KRB Accounts, eTKRBDirectoryName= <i>directory_name</i> , eTNamespaceName= KRB Namespace, <i>domain_name_suffix</i>
eTKRBPASSWORDPolicyContainer	eTKRBPASSWORDPolicyContainerName=KRB Password Policies, eTKRBDirectoryName= <i>directory_name</i> , eTNamespaceName=KRB Namespace, <i>domain_name_suffix</i>
eTKRBPASSWORDPolicy	eTKRBPASSWORDPolicyName=KRBPolicy,eTKRBPASSWORDPolicyC ontainerName=KRB Password Policies, eTKRBDirectoryName= <i>directory_name</i> , eTNamespaceName=KRB Namespace, <i>domain_name_suffix</i>

EEM Objects

The following table lists the EEM objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=EEM, <i>domain_name_suffix</i>
eTEEMDirectory	eTEEMDirectoryName= <i>directory_name</i> , eTNamespaceName=EEM, <i>domain_name_suffix</i>
eTEEMAccountContainer	eTEEMAccountContainerName=EEM Accounts, eTEEMDirectoryName= <i>directory_name</i> , eTNamespaceName=EEM, <i>domain_name_suffix</i>
eTEEMAccount	eTEEMAccountName= <i>account_name</i> , eTEEMAccountContainerName=EEM Accounts, eTEEMDirectoryName= <i>directory_name</i> , eTNamespaceName=EEM, <i>domain_name_suffix</i>
eTEEMGroupContainer	eTEEMGroupContainerName=EEM Groups, eTEEMDirectoryName= <i>directory_name</i> , eTNamespaceName=EEM, <i>domain_name_suffix</i>
eTEEMGroup	eTEEMGroupName=EEM Group, eTEEMGroupContainerName=EEM Groups, eTEEMDirectoryName= <i>directory_name</i> , eTNamespaceName=EEM, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTEEMCalendarContainer	eTEEMCalendarContainerName=EEM Calendars, eTEEMDirectoryName= <i>directory_name</i> , eTNamespaceName=EEM, <i>domain_name_suffix</i>
eTEEMCalendar	eTEEMCalendarName=EEM Calendar, eTEEMCalendarContainerName=EEM Calendars, eTEEMDirectoryName= <i>directory_name</i> , eTNamespaceName=EEM, <i>domain_name_suffix</i>
eTEEMResourceContainer	eTEEMResourceContainerName=EEM Resource Class, eTEEMDirectoryName= <i>directory_name</i> , eTNamespaceName=EEM, <i>domain_name_suffix</i>
eTEEMResource	eTEEMResourceName=EEM Resource Class, eTEEMResourceContainerName=EEM Resource Class, eTEEMDirectoryName= <i>directory_name</i> , eTNamespaceName=EEM, <i>domain_name_suffix</i>

SAP Objects

The following table lists the SAP objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespace=SAP <i>Namespace, domain_name_suffix</i>
eTSAPDirectory	eTSAPDirectoryName= <i>directory_name</i> , eTNamespace=SAP <i>Namespace, domain_name_suffix</i>
eTSAPAccountContainer	eTSAPAccountContainerName=Accounts, eTSAPDirectoryName= <i>directory_name</i> , eTNamespace=SAP <i>Namespace, domain_name_suffix</i>
eTSAPAccount	eTSAPAccountName= <i>account_name</i> , eTSAPAccountContainerName=Accounts, eTSAPDirectoryName= <i>directory_name</i> , eTNamespace=SAP <i>Namespace, domain_name_suffix</i>
eTSAPProfileContainer	eTSAPProfileContainer=Profiles, eTSAPDirectoryName= <i>directory_name</i> , eTNamespace=SAP <i>Namespace, domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTSAPProfile	eTSAPProfile= <i>profile_name</i> , eTSAPProfileContainer=Profiles, eTSAPDirectoryName= <i>directory_name</i> , eTNamespace=SAP <i>Namespace, domain_name_suffix</i>
eTSAPRoleContainer	eTSAPRoleContainer=SAP Roles eTSAPDirectoryName= <i>directory_name</i> eTNamespace=SAP <i>Namespace, domain_name_suffix</i>
eTSAPRole	eTSAPRole= <i>role_name</i> eTSAPRoleContainer=SAP Roles eTSAPDirectoryName= <i>directory_name</i> , eTNamespace=SAP <i>Namespace, domain_name_suffix</i>

Siebel Objects

The following table lists the Siebel objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=Siebel, <i>domain_name_suffix</i>
eTSBLDirectory	eTSBLDirectoryName=Siebel_ <i>directory_name</i> , eTNamespaceName=Siebel, <i>domain_name_suffix</i>
eTSBLUserContainer	eTSBLUserContainerName=Users, eTSBLDirectoryName=Siebel_ <i>directory_name</i> , eTNamespaceName=Siebel, <i>domain_name_suffix</i>
eTSBLUser	eTSBLUserID=Siebel_ <i>user_ID</i> , eTSBLUserContainerName=Users, eTSBLDirectoryName=Siebel_ <i>directory_name</i> , eTNamespaceName=Siebel, <i>domain_name_suffix</i>
eTSBLPositionContainer	eTSBLPositionContainerName=Positions, eTSBLDirectoryName=Siebel_ <i>directory_name</i> , eTNamespaceName=Siebel, <i>domain_name_suffix</i>
eTSBLPosition	eTSBLPositionName=Siebel_ <i>position_name</i> , eTSBLPositionContainerName=Positions, eTSBLDirectoryName=Siebel_ <i>directory_name</i> , eTNamespaceName=Siebel, <i>domain_name_suffix</i>
eTSBLDivision Container	eTSBLDivisionContainerName=Divisions, eTSBLDirectoryName=Siebel_ <i>directory_name</i> , eTNamespaceName=Siebel, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTSBLDivision	eTSBLResponsibilityName=Siebel_division_name, eTSBLDivisionContainerName=Divisions, eTSBLDirectoryName=Siebeldirectory_name, eTNamespaceName=Siebel,domain_name_suffix
eTSBLView Container	eTSBLResponsibilityContainerName=Views, eTSBLDirectoryName=Siebel_directory_name, eTNamespaceName=Siebel,domain_name_suffix
eTSBLView	eTSBLResponsibilityName=Siebel_View_name, eTSBLViewContainerName=Views, eTSBLDirectoryName=Siebeldirectory_name, eTNamespaceName=Siebel,domain_name_suffix
eTSBLResponsibility Container	eTSBLResponsibilityContainerName=Responsibilities , eTSBLDirectoryName=Siebel_directory_name, eTNamespaceName=Siebel,domain_name_suffix
eTSBLResponsibility	eTSBLResponsibilityName=Siebel_responsibility_name, eTSBLResponsibilityContainerName=Responsibilities, eTSBLDirectoryName=Siebeldirectory_name, eTNamespaceName=Siebel,domain_name_suffix
eTSBLLOVContainer	eTSBLLOVContainerName=LOV, eTSBLDirectoryName=Siebel_directory_name, eTNamespaceName=Siebel,domain_name_suffix
eTSBLLOVValue	eTSBLLOVValueName=Siebel_ListOfValues_name, eTSBLUserListContainerName=LOV, eTSBLDirectoryName=Siebel_directory_name, eTNamespaceName=Siebel,domain_name_suffix

RSA Objects

The following table lists the RSA objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespace=RSA Server,domain_name_suffix
eTRSADirectory	eTRSADirectoryName=directory_name, eTNamespace=RSA Server,domain_name_suffix

LDAP Object Name	DN of Object Instance
eTRSAAccountContainer	eTRSAAccountContainerName=RSA Accounts, eTRSADirectoryName= <i>directory_name</i> , eTNamespace=RSA Server, <i>domain_name_suffix</i>
eTRSAAccount	eTRSAAccountName= <i>account_name</i> , eTRSAAccountContainerName=RSA Accounts, eTRSADirectoryName= <i>directory_name</i> , eTNamespace=RSA Server, <i>domain_name_suffix</i>
eTRSAGrpContainer	eTRSAGrpContainerName=RSA Groups, eTRSADirectoryName= <i>directory_name</i> , eTNamespaceName=RSA Server, <i>domain_name_suffix</i>
eTRSAGrp	eTRSAGrpName=Group_Name, eTRSAGrpContainerName=Groups, eTRSADirectoryName= <i>directory_name</i> , eTNamespace=RSA Server, <i>domain_name_suffix</i>
eTRSATokenContainer	eTRSATokenContainerName=RSA Tokens, eTRSADirectoryName= <i>directory_name</i> , eTNamespace=RSA Server, <i>domain_name_suffix</i>
eTRSAToken	eTRSATokenSerialNumber=token_serial_number, eTRSATokenContainerName=RSA Tokens, eTRSADirectoryName= <i>directory_name</i> eTNamespace=RSA Server, <i>domain_name_suffix</i>
eTRSAAgentHostContainer	eTRSAAgentHostContainerName=RSA Agent Hosts, eTRSADirectoryName= <i>directory_name</i> , eTNamespace=RSA Server, <i>domain_name_suffix</i>
eTRSAAgentHost	eTRSAAgentHostName= <i>AgentHost_name</i> , eTRSAAgentHostContainerName=RSA Agent Hosts, eTRSADirectoryName= <i>directory_name</i> , eTNamespaceName=RSA SERVER, <i>domain_name_suffix</i>
eTRSASiteContainer	eTRSASiteContainerName=RSA Sites, eTRSADirectoryName= <i>directory_name</i> , eTNamespaceName=RSA SERVER, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTRSASite	eTRSASiteName= <i>Site_name</i> , eTRSASiteContainerName=RSA Sites, eTRSADirectoryName= <i>directory_name</i> , eTNamespaceName=RSA SERVER, <i>domain_name_suffix</i>

PKI Objects

The following table lists the PKI objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNamespace	eTNamespaceName=PKI <i>domain_name_suffix</i>
eTPKIDirectory	eTPKIDirectoryName= <i>directory_name</i> , eTNamespaceName=PKI <i>domain_name_suffix</i>
eTPKINamespace	eTNamespaceName=PKI endpoint type name, eTPKIDirectoryName= <i>directory_name</i> , eTNamespaceName=PKI, <i>domain_name_suffix</i>
eTPKIAccountContainer	eTPKIAccountContainerName=PKI Accounts, eTPKIDirectoryName= <i>directory_name</i> , eTPKINamespaceName=PKI, <i>domain_name_suffix</i>
eTPKIAccount	eTPKIAccountName= <i>account_name</i> , eTPKIAccountContainerName=PKI Accounts, eTPKIDirectoryName= <i>directory_name</i> , eTNamespaceName= Entrust Authority Security Manager System, eTNamespaceName=PKI, <i>domain_name_suffix</i>
eTPKIGroupContainer	eTPKIGroupContainerName=PKI Groups, eTPKIDirectoryName= <i>directory_name</i> , eTNamespaceName=PKI, <i>domain_name_suffix</i>
eTPKIGroup	eTPKIGroupName=PKI Group, eTPKIGroupContainerName=PKI Groups, eTPKIDirectoryName= <i>directory_name</i> , eTNamespaceName=PKI, <i>domain_name_suffix</i>

Common Objects Tree

The Connector account templates belong to the common objects tree. The following sections lists the connector account template objects and their DN's in hierarchical order:

DBZ Server Common Object Tree

DBZ Server account templates belong to the common objects tree. The following table lists the DBZ Server account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTDBZPolicyContainer	eTDBZPolicyContainerName=DB2 ZOS Account Templates, eTNamespaceName=DB2 ZOS Server, domain_name_suffix
eTDBZPolicy	eTDBZPolicyName=DB2 ZOS Account Template, eTDBZPolicyContainerName=DB2 ZOS Account Templates, eTNamespaceName=DB2 ZOS Server, domain_name_suffix

DB2 UDB Common Objects Tree

DB2 UDB account templates belong to the common objects tree. The following table lists the DB2 UDB account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTDB2Policy	eTDB2PolicyName=policy_name, eTDB2PolicyContainerName=DB2 Policies, eTNamespaceName=CommonObjects, domain_name_suffix
eTDB2PolicyContainer	eTDB2PolicyContainerName=DB2 Policies, eTNamespaceName=CommonObjects, domain_name_suffix

MS SQL Server Common Object Tree

MS SQL Server account templates belong to the common objects tree. The following table lists the MS SQL Server account template objects and their DNs in hierarchical order:

LDAP Object Name	DN of Object Instance
eTSQLPolicyContainer	eTSQLPolicyContainerName= MS SQL Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i> datalocation=DB (*) edittype=string maxlen=255 description= MS SQL Server Policy Container Name
eTSQLPolicy	eTSQLPolicyName= <i>MS SQL Policy name</i> , eTSQLPolicyContainerName=MS SQL Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i> datalocation=DB (*) edittype=string maxlen=50 description= MS SQL Server Policy Name

(*) datalocation =DB means that the object is stored in the Provisioning Directory.

Ingres Common Objects Tree

Ingres account templates belong to the common objects tree. The following table lists the Ingres account template objects and their DNs in hierarchical order:

LDAP Object Name	DN of Object Instance
eTINGPolicy	eTINGPolicyName=Ingres Policy, eTINGPolicyContainerName=Ingres Policies, eTINGNamespaceName=CommonObjects, <i>domain_name_suffix</i>
eTINGPolicyContainer	eTINGPolicyContainerName=Ingres Policies, eTINGNamespaceName=CommonObjects, <i>domain_name_suffix</i>

Oracle Common Objects Tree

Oracle account templates belong to the common objects tree. The following table lists the Oracle account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTORAPolicy	eTORAPolicyName= <i>policy_name</i> , eTORAPolicyContainerName=Oracle Policies, eTNamespaceName=CommonObjects, <i>domain_name_</i> <i>suffix</i>
eTORAPolicyContainer	eTORAPolicyContainerName=Oracle Policies, eTNamespaceName=CommonObjects, <i>domain_name_</i> <i>suffix</i>

Oracle Applications Common Objects Tree

Oracle Applications account templates belong to the common objects tree. The following table lists the Oracle Applications account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
ETFNDPolicy	eTFNDPolicyName= <i>policy_name</i> , eTVMSPolicyContainerName=Oracle Applications Policies, eTNamespaceName=CommonObjects, <i>domain_name_</i> <i>suffix</i>
eTFNDPolicyContainer	eTFNDPolicyContainerName=Oracle Applications Policies, eTNamespaceName=CommonObjects, <i>domain_name_</i> <i>suffix</i>

Windows NT Common Objects Tree

Windows NT account templates belong to the common objects tree. The following table lists the Windows NT account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTN16Policy	eTN16PolicyName= <i>policy_name</i> , eTN16PolicyContainerName=Windows NT Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>
eTN16PolicyContainer	eTN16PolicyContainerName=Windows NT Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>

ACF2 Common Objects Tree

CA-ACF2 account templates belong to the common objects tree. The following table lists the CA-ACF2 account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
ETACFPolicy	eTACFPolicyName= <i>policy_name</i> , eTACFPolicyContainerName=CA-ACF2 Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>
eTACFPolicyContainer	eTACFPolicyContainerName=CA-ACF2 Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>

Top Secret Common Objects Tree

CA-Top Secret account templates belong to the common objects tree. The following table lists the CA-Top Secret account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
ETTSSPolicy	eTTSSPolicyName= <i>policy_name</i> , eTTSSPolicyContainerName=CA-Top Secret Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTTSSPolicyContainer	eTTSSPolicyContainerName=CA-Top Secret Policies, eTNamespaceName=CommonObjects, <i>domain_name_</i> <i>suffix</i>

RACF Common Objects Tree

RACF account templates belong to the common objects tree. The following table lists the RACF account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
ETRACPolicy	eTRACPolicyName= <i>policy_name</i> , eTRACPolicyContainerName=RACF Policies, eTNamespaceName=CommonObjects, <i>domain_name_</i> <i>suffix</i>
eTRACPolicyContainer	eTRACPolicyContainerName=RACF Policies, eTNamespaceName=CommonObjects, <i>domain_name_</i> <i>suffix</i>

OS/400 Common Objects Tree

OS/400 account templates belong to the common objects tree. The following table lists the OS/400 account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
ETAS4Policy	eTAS4PolicyName= <i>policy_name</i> , eTAS4PolicyContainerName=OS/400 Policies, eTNamespaceName=CommonObjects, <i>domain_name_</i> <i>suffix</i>
eTAS4PolicyContainer	eTAS4PolicyContainerName=OS/400 Policies, eTNamespaceName=CommonObjects, <i>domain_name_</i> <i>suffix</i>

OpenvVMS Common Objects Tree

OpenVMS account templates belong to the common objects tree. The following table lists the OpenVMS account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
ETVMSPolicy	eTVMSPolicyName= <i>policy_name</i> , eTVMSPolicyContainerName=OpenVMS Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>
eTVMSPolicyContainer	eTVMSPolicyContainerName=OpenVMS Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>

NSK Common Objects Tree

HP NonStop SAFEGUARD policies belong to the common objects tree. The following table lists the HP NonStop SAFEGUARD policy objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNSKPolicy	eTNSKPolicyName= <i>policy_name</i> , eTNSKPolicyContainerName= HP NonStop Safeguard Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>
eTNSKPolicyContainer	eTNSKPolicyContainerName= HP NonStop Safeguard Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>

UNIX ETC Common Objects Tree

UNIX ETC account templates belong to the common objects tree. The following table lists the UNIX ETC account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTETCPolicy	eTETCPolicyName= <i>policy_name</i> , eTETCPolicyContainerName=UNIX - etc Policies, eTNamespaceName=CommonObjects, <i>domain_name_</i> <i>suffix</i>
eTETCPolicyContainer	eTETCPolicyContainerName=UNIX - etc Policies, eTNamespaceName=CommonObjects, <i>domain_name_</i> <i>suffix</i>

UNIX NIS Common Objects Tree

UNIX NIS account templates belong to the common objects tree. The following table lists the UNIX account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNISPolicy	eTNISPolicyName= <i>policy_name</i> , eTNISPolicyContainerName=UNIX - NIS-NIS plus Domains Policies, eTNamespaceName=CommonObjects, <i>domain_name_</i> <i>suffix</i>
eTNISPolicyContainer	eTNISPolicyContainerName=UNIX - NIS-NIS plus Domains Policies, eTNamespaceName=CommonObjects, <i>domain_name_</i> <i>suffix</i>

ADS Common Objects Tree

Active Directory Services account templates belong to the common objects tree. The following table lists the Active Directory Services account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
ETADSPolicy	eTADSPolicyName= <i>policy_name</i> , eTADSPolicyContainerName=Active Directory Policies, eTNamespaceName=CommonObjects, <i>domain_name</i> <i>_suffix</i>
ETADSPolicyContainer	eTADSPolicyContainerName=Active Directory Policies, eTNamespaceName=CommonObjects, <i>domain_name</i> <i>_suffix</i>

Generic LDAP Common Objects Tree

LDAP account templates belong to the common objects tree. The following table lists the LDAP account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
eLDAPPolicy	eLDAPPolicyName= <i>policy_name</i> , eLDAPPolicyContainerName=LDAP Policies, eTNamespaceName=CommonObjects, <i>domain_name</i> <i>_suffix</i>
eLDAPPolicyContainer	eLDAPPolicyContainerName=LDAP Policies, eTNamespaceName=CommonObjects, <i>domain_name</i> <i>_suffix</i>

NDS Common Objects Tree

NDS account templates belong to the common objects tree. The following table lists the NDS account template objects and their DNs in hierarchical order:

LDAP Object Name	DN of Object Instance
eTNDSPolicy	eTNDSPolicyName= <i>policy_name</i> , eTNDSPolicyContainerName=NDS Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>
eTNDSPolicyContainer	eTNDSPolicyContainerName=NDS Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>

Universal Provisioning Common Objects Tree

Universal Provisioning account templates belong to the common objects tree. The following table lists the Universal Provisioning account template objects and their DNs in hierarchical order:

LDAP Object Name	DN of Object Instance
eTUPOPPolicy	eTUPOPPolicyName= <i>policy_name</i> , eTUPOPPolicyContainerName=Universal Provisioning Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>
eTUPOPPolicyContainer	eTUPOPPolicyContainerName=Universal Provisioning Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>

LND Common Objects Tree

Lotus Notes/Domino account templates belong to the common objects tree. The following table lists the Lotus Notes/Domino account template objects and their DNs in hierarchical order:

LDAP Object Name	DN of Object Instance
eTLNDPolicy	eTLNDPolicyName= <i>policy_name</i> , eTLNDPolicyContainerName=LND Policies, eTNamespaceName=CommonObjects, <i>domain_name_s</i> <i>uffix</i>
eTLNDPolicyContainer	eTLNDPolicyContainerName=LND Policies, eTNamespaceName=CommonObjects, <i>domain_name_s</i> <i>uffix</i>

Access Control Common Objects Tree

CA Access Control account templates belong to the common objects tree. The following table lists the CA Access Control account template objects and their DNs in hierarchical order:

LDAP Object Name	DN of Object Instance
eTACCPolicy	eTACCPolicyName= <i>policy_name</i> , eTACCPolicyContainerName=Access Control Policies, eTNamespaceName=CommonObjects, <i>domain_name_s</i> <i>uffix</i>
eTACCPolicyContainer	eTACCPolicyContainerName=Access Control Policies, eTNamespaceName=CommonObjects, <i>domain_name_s</i> <i>uffix</i>

SSO for Advanced Policy Server Common Objects Tree

PLS account templates belong to the common objects tree. The following table lists the PLS account template objects and their DNs in hierarchical order:

LDAP Object Name	DN of Object Instance
eTPLSPolicy	eTPLSPolicyName= <i>policy_name</i> , eTPLSPolicyContainerName=CA SSO WAC Policies, eTNamespaceName=CommonObjects, <i>domain_name_suf</i> <i>fix</i>

LDAP Object Name	DN of Object Instance
eTPLSPolicyContainer	eTPLSPolicyContainerName=CA SSO WAC Policies, eTNamespaceName=CommonObjects, domain_name_suffix

Kerberos Common Objects Tree

KRB account templates belong to the common objects tree. The following table lists the KRB account template objects and their DNs in hierarchical order:

LDAP Object Name	DN of Object Instance
eTKRBPOLICY	eTKRBPOLICYName= <i>policy_name</i> , eKRBPolicyContainerName=KRB policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>
eTKRBPOLICYContainer	eTKRBPOLICYContainerName=KRB Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>

Embedded Entitlements Manager Common Objects Tree

EEM account templates belong to the common objects tree. The following table lists the EEM account template objects and their DNs in hierarchical order:

LDAP Object Name	DN of Object Instance
eTEEMPOLICY	eTEEMPOLICYName= <i>policy_name</i> , eTEEMPOLICYContainerName=EEM Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>
eTEEMPOLICYContainer	eTEEMPOLICYContainerName=EEM Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>
eTEEMACCESSPOLICY	eTEEMACCESSPOLICYName= <i>access_policy_name</i> , eTEEMACCESSPOLICYContainerName=EEM Access Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>

LDAP Object Name	DN of Object Instance
eTEEMAccessPolicyContainer	eTEEMAccessPolicyContainerName=EEM Access Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>

SAP Common Objects Tree

SAP account templates belong to the common objects tree. The following table lists the SAP account template objects and their DNs in hierarchical order:

LDAP Object Name	DN of Object Instance
eTSAPPolicy	eTSAPPolicyName= <i>policy_name</i> , eTSAPPolicyContainerName=SAP Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>
eTSAPPolicyContainer	eTSAPPolicyContainerName=SAP Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>

Siebel Common Objects Tree

Siebel account templates belong to the common objects tree. The following table lists the Siebel account template objects and their DNs in hierarchical order:

LDAP Object Name	DN of Object Instance
eTSBLPolicy	eTSBLPolicyName= <i>policy_name</i> , eTSBLPolicyContainerName=Siebel Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>
eTSBLPolicyContainer	eTSBLPolicyContainerName=Siebel Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>

RSA Common Objects Tree

RSA account templates belong to the common objects tree. The following table lists the RSA account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
ETRSAPolicy	eTRSPolicyName= <i>policy_name</i> , eTRSPolicyContainerName=RSA Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>
eTRSPolicyContainer	eTRSPolicyContainerName=RSA Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>

PKI Common Objects Tree

PKI account templates belong to the common objects tree. The following table lists the PKI account template objects and their DN's in hierarchical order:

LDAP Object Name	DN of Object Instance
ETPKIPolicy	eTPKIPolicyName= <i>policy_name</i> , eTPKIPolicyContainerName=PKI Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>
eTPKIPolicyContainer	eTPKIPolicyContainerName=PKI Policies, eTNamespaceName=CommonObjects, <i>domain_name_suffix</i>

Object User-Friendly Names

The following sections list the LDAP object names for the connectors and their user-friendly names:

DBZ Object User Friendly Names

The following table lists the DBZ object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTDBZAccount	DB2 ZOS Account	DB2 ZOS User
eTDBZAccountContainer	DB2 ZOS Account Container	DB2 ZOS User container

eTDBZDirectory	DB2 ZOS Directory	Directory name
eTDBZPolicy	DB2 ZOS Policy	Policy
eTDBZPolicyContainer	DB2 ZOS PolicyContainer	Policy container
eTNamespace	Namespace	Namespace name

DB2 UDB Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names alphabetically:

LDAP Object Name	User-Friendly Name	Description
eTDB2Account	DB2Account	DB2 UDB User
eTDB2AccountContainer	DB2AccountContainer	DB2 UDB User container
eTDB2Directory	DB2Directory	Directory name
eTDB2Group	DB2Group	DB2 UDB Group
eTDB2GroupContainer	DB2GroupContainer	DB2 UDB Group container
eTDB2Policy	DB2Policy	Policy
eTDB2PolicyContainer	DB2PolicyContainer	Policy container
eTNamespace	Namespace	Policy

MS SQL Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTSQLDirectory	SQLDirectory	Directory
eTSQLLogin	SQLLogin	MS SQL Login
eTSQLLoginContainer	SQLLoginContainer	MS SQL Login Container
eTSQLUser	SQLUser	MS SQL User
eTSQLRole	SQLRole	MS SQL Role
eTSQLDatabase	SQLDatabase	MS SQL Database
eTSQLPolicy	SQLPolicy	MS SQL Policy
eTSQLPolicyContainer	SQLPolicyContainer	MS SQL Policy Container

Ingres Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTNamespace	Namespace	Identity Manager directory container
eTINGAccount	INGAccount	Account
eTINGAccountContainer	INGAccountContainer	Account container
eTINGProfile	INGProfile	Profiles
eTINGProfileContainer	INGProfileContainer	Profile container
eTINGRole	INGRole	Roles
eTINGRoleContainer	INGRoleContainer	Role container
eTINGGroup	INGGroup	Groups
eTINGGroupContainer	INGGroupContainer	Group container
eTINGPolicy	INGPolicy	Policy
eTINGPolicyContainer	INGPolicyContainer	Policy container

Oracle Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTORAAccount	ORAAccount	Account
eTORAAccountContainer	ORAAccountContainer	Account container
eTORADirectory	ORADirectory	Directory name
eTORAPkgContainer	ORAPkgContainer	Package container
eTORAPkg	ORAPkg	Package
eTORAProcContainer	ORAProcContainer	Procedure container
eTORAProc	ORAProc	Procedure
eTORAProfile	ORAProfile	Oracle profile
eTORAProfileContainer	ORAProfileContainer	Profile container
eTORARole	ORARole	Oracle role

LDAP Object Name	User-Friendly Name	Description
eTORARoleContainer	ORARoleContainer	Role container
eTORAPolicy	ORAPolicy	Policy
eTORAPolicyContainer	ORAPolicyContainer	Policy container
eTNamespace	Namespace	Namespace name

Oracle Applications Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTFNDAccount	FNDAccount	Account
eTFNDAccountContainer	FNDAccountContainer	Account container
eTFNDDirectory	FNDDirectory	Directory name
eTFNDPolicy	FNDPolicy	Policy
eTFNDPolicyContainer	FNDPolicyContainer	Policy container
eTNamespace	Namespace	Namespace name

Windows NT Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTN16Account	N16Account	Account
eTN16AccountContainer	N16AccountContainer	Account container
eTN16Directory	N16Directory	Directory name
eTN16FolderManager	N16FolderManager	Windows NT directory storage browser
eTN16Group	N16Group	Group
eTN16GroupContainer	N16GroupContainer	Group container

LDAP Object Name	User-Friendly Name	Description
eTN16GroupManager	N16GroupManager	Retrieves user list included in a group
eTN16Policy	N16Policy	Policy
eTN16PolicyContainer	N16PolicyContainer	Policy container
eTN16SharedFolder	N16SharedFolder	Shared folder
eTN16SharedFolderContainer	N16SharedFolderContainer	Shared folder container
eTNamespace	Namespace	Namespace name

ACF2 Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTACFACF2RuleKey	ACFRuleKey	Rule key
eTACFACF2RuleLine	ACFRuleLine	Rule line
eTACFACF2RuleType	ACFRuleType	Rule type
ETACFDDirectory	ACFDDirectory	Directory name
ETACFLid	ACFAccount	Logon ID
ETACFLidContainer	ACFLidContainer	Accounts container
ETACFRuleContainer	ACFRuleContainer	Rules container
ETNamespace	Namespace	Namespace name

Top Secret Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTNamespace	Namespace	Namespace name
eTTSSAcid	TSSAcid	Acid
eTTSSAcidContainer	TSSAcidContainer	Acid container
eTTSSFacilityContainer	TSSFacilityContainer	Facility Container

LDAP Object Name	User-Friendly Name	Description
eTTSSPermissionContainer	TSSPermissionContainer	Permission Container
eTTSSOwnershipContainer	TSSOwnershipContainer	Ownership Container
eTTSSProfListContainer	TSSProfListContainer	Profile List Container
eTTSSAdminFacContainer	TSSAdminFacContainer	Admin Facility Container
eTTSSAdminResContainer	TSSAdminResContainer	Admin Resource Container
eTTSSAdminScpContainer	TSSAdminScpContainer	Admin Scope Container
eTTSSAdminFacility	TSSAdminFacility	Admin Facility
eTTSSAdminResource	TSSAdminResource	Admin Resource
eTTSSAdminScope	TSSAdminScope	Admin Scope
eTTSSDeptContainer	DeptContainer	Department Container
eTTSSDept	TSSDept	Department
eTTSSDivContainer	DivContainer	Division Container
eTTSSDiv	TSSDiv	Division
eTTSSZoneContainer	ZoneContainer	Zone Container
eTTSSAcidZone	TSSZone	Zone
eTTSSGroupContainer	GroupContainer	Group Container
eTTSSGroup	TSSGroup	Group
eTTSSFacility	TSSFacility	Facility
eTTSSDirectory	TSSDirectory	Directory
eTTSSOwned	TSSOwned	Owned object
eTTSSPolicy	TSSPolicy	Policy
eTTSSPolicyContainer	TSSPolicyContainer	Policy container
eTTSSProfile	TSSProfile	Profile
eTTSSProfList	TSSProfList	Profile List
eTTSSProfileContainer	TSSProfileContainer	Profile container
eTTSSResClass	TSSResClass	Resource class

LDAP Object Name	User-Friendly Name	Description
eTTSSResName	TSSResName	Resource name

RACF Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTNamespace	Namespace	Namespace name
eTRACAccount	RACAccount	Account
eTRACAccountContainer	RACAccountContainer	Account container
eTRACDirectory	RACDirectory	Directory
eTRACGroup	RACGroup	Group
eTRACGroupContainer	eTRACGroupContainer	Group container
eTRACGroupUser	RACGroupUser	Groupuser
eTRACPermissionContainer	RACPermissionContainer	Permission container
eTRACPermissionResourceClass	RACPermissionResourceClass	Permission resource class
eTRACResProfile	RACResProfile	Resource profile
eTRACResUser	RACResUser	Resource user
eTRACUserPermissionContainer	RACUserPermissionContainer	User permission container
eTRACUserPermissionResourceClass	RACUserPermissionResourceClass	User permission resource class
eTRACUserResProfile	RACUserResProfile	User Resource profile
eTRACPolicy	RACPolicy	Policy
eTRACPolicyContainer	RACPolicyContainer	Policy container

OS/400 Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTAS4Account	AS4Account	Account
eTAS4AccountContainer	AS4AccountContainer	Account container
eTAS4Directory	AS4Directory	Directory name
eTAS4Group	AS4Group	Group
eTAS4GroupContainer	AS4GroupContainer	Group container
eTAS4Policy	AS4Policy	Policy
eTAS4PolicyContainer	AS4PolicyContainer	Policy container
eTNamespace	Namespace	Namespace name

OpenVMS Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTOpenVMSAccount	OpenVMSAccount	Account
ETOpenVMSAccountContainer	OpenVMSAccountContainer	Account container
eTOpenVMSDirectory	OpenVMSDirectory	Directory name
ETVMSPolicy	OpenVMSPolicy	Policy
eTOpenVMSPolicyContainer	OpenVMSPolicyContainer	Policy container
ETNamespace	Namespace	Namespace name

NSK Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTNSKAccount	NSKAccount	Account
eTNSKAccountContainer	NSKAccountContainer	Account container
eTNSKDirectory	NSKDirectory	Directory name

LDAP Object Name	User-Friendly Name	Description
eTNSKGroup	NSKGroup	Group
eTNSKGroupContainer	NSKGroupContainer	Group container
eTNSKPolicy	NSKPolicy	Policy
eTNSKPolicyContainer	NSKPolicyContainer	Policy container
eTNamespace	Namespace	Namespace name

UNIX ETC Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTETCAccount	ETCAccount	Account
eTETCAccountContainer	ETCAccountContainer	Account container
eTETCGroup	ETCGroup	Group
eTETCGroupContainer	ETCGroupContainer	Group container
eTETCPolicy	ETCPolicy	Policy
eTETCPolicyContainer	ETCPolicyContainer	Policy container
eTNamespace	Namespace	Namespace name

UNIX NIS Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names alphabetically:

LDAP Object Name	User-Friendly Name	Description
eTNISAccount	NISAccount	Account
eTNISAccountContainer	NISAccountContainer	Account container
eTNISGroup	NISGroup	Group
eTNISGroupContainer	NISGroupContainer	Group container
eTNISNetGroup	NISNetGroup	NetGroup
eTNISNetGroupContainer	NISNetGroupContainer	NetGroup Container
eTNISPolicy	NISPolicy	Policy
eTNISPolicyContainer	NISPolicyContainer	Policy container

LDAP Object Name	User-Friendly Name	Description
eTNamespace	Namespace	Namespace name

ADS Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
ETADSAccount	ADSAccount	Account
ETADSContainer	ADSContainer	Account container
ETADSDirectory	ADSDirectory	Directory name
ETADSGroup	ADSGroup	Group
ETADSOrgUnit	ADSOrgUnit	Organizational unit, a container for groups
ETADSPolicy	ADSPolicy	Policy
ETADSPolicyContainer	ADSPolicyContainer	Policy container
ETNamespace	Namespace	Namespace name

Generic LDAP Object User-Friendly Names

The following table lists alphabetically the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTLDAAccount	LDAAccount	Account
eTLDAAccountContainer	LDAContainer	Container (for accounts)
eTLDADirectory	LDADirectory	Directory name
eTLDAGroup	LDAGroup	Group
eTLDAPolicy	LDAPPolicy	Policy
eTLDAPolicyContainer	LDAPPolicyContainer	Policy container
eTNamespace	Namespace	Namespace name

NDS Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTNSAccount	NDSAccount	Account
eTNSCountry	NDSCountry	Country
eTNSGroup	NDSGroup	Group (of accounts)
eTNSLocality	NDSLocality	Locality (within a country)
eTNSOrgUnit	NDSOrgUnit	Organizational unit
eTNSOrganization	NDSOrganization	Organization
eTNSPolicy	NDSPolicy	Policy
eTNSPolicyContainer	NDSPolicyContainer	Policy container
eTNamespace	Namespace	Namespace name
eTNDSTree	NDSTree	Tree (directory) name

Universal Provisioning Object User-Friendly Names

The following table lists the object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTUPOAccount	UPOAccount	Account
eTUPOAccountContainer	UPOAccountContainer	Account container
eTUPODirectory	UPODirectory	Directory name
eTUPOPolicy	UPOPolicy	Policy
eTUPOPolicyContainer	UPOPolicyContainer	Policy container
eTNamespace	Namespace	Namespace name
eTUPOExitContainer	UPOExitContainer	Exit container
eTUPOExit	UPO Exit	Exit

Lotus Notes/Domino Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTLNDAccount	LNDAccount	Account
eTLNDCountry	LNDCountry	Country name
eTLNDDirectory	LNDDirectory	Directory name
eTLNDFlatCertifier	LNDFlatCertifier	Flat certifier
eTLNDGroup	LNDGroup	Group
eTLNDGroupContainer	LNDGroupContainer	Group container
eTLNDOrganizationalUnit	LNDOrganizationalUnit	Organizational unit name
eTLNDOrganization	LNDOrganization	Organization
eTLNDPolicy	LNDPolicy	Policy
eTLNDPolicyContainer	LNDPolicyContainer	Policy container
eTNamespace	Namespace	Namespace name

Access Control Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTACCAccount	ACCAccount	Account
eTACCAccountContainer	ACCAccountContainer	Account container
eTACCDirectory	ACCDirectory	Directory name
eTACCGroup	ACCGroup	Group
eTACCGroupContainer	ACCGroupContainer	Group container
eTACCPolicy	ACCPolicy	Policy
eTACCPolicyContainer	ACCPolicyContainer	Policy container
eTNamespace	Namespace	Namespace name

PLS Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTPLSAccount	PLSAccount	Account
eTPLSContainer	PLSContainer	Container
eTPLSUserStore	PLSUserStore	User store name
eTPLSDirectory	PLSDirectory	Directory name
eTPLSGroup	PLSGroup	Group
eTPLSPolicy	PLSPolicy	Policy
eTPLSPolicyContainer	PLSPolicyContainer	Policy container
eTPLSApplicationContainer	PLSApplicationContainer	Application container
eTPLSApplication	PLSApplication	Application
eTPLSApplicationGroupContainer	PLSApplicationGroupContainer	Application group container
eTPLSApplicationGroup	PLSApplicationGroup	Application group
eTPLSTerminalContainer	PLSTerminalContainer	Terminal Container
eTPLSTerminal	PLSTerminal	Terminal
eTPLSAuthhostContainer	PLSAuthhostContainer	Authhost Container
eTPLSAuthhost	PLSAuthhost	Authhost
eTNamespace	Namespace	Namespace name

Note: Information about the CA Single Sign-On for Advanced Policy Server policy and policy container can be found in the common objects schema. For details, see the *Programming Guide for Provisioning*.

KRB Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTKRBAccount	KRBAccount	KRB User
eTKRBAccountContainer	KRBAccountContainer	KRB User container
eTKRBPasswordPolicyContainer	KRBPasswordPolicyContainer	KRB Password Policy Container

LDAP Object Name	User-Friendly Name	Description
eTKRBPASSWORDPolicy	KRBPASSWORDPolicy	KRB Password Policy
eTKRBDIRECTORY	KRBDIRECTORY	Directory Name
eTKRBPOLICY	KRBPOLICY	Policy
eTKRBPOLICYContainer	KRBPOLICYContainer	Policy Container
eNamespace	Namespace	Namespace name

EEM Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTNamespace	Namespace	Namespace name
eTEEMAccount	EEMAccount	Account
eTEEMAccountContainer	EEMAccountContainer	Account container
eTEEMGroup	EEMGroup	Group
eTEEMGroupContainer	EEMGroupContainer	Group container
eTEEMCalendar	EEMCalendar	Calendar
eTEEMCalendarContainer	EEMCalendarContainer	Calendar container
eTEEMResource	EEMResource	Resource
eTEEMResourceContainer	EEMResourceContainer	Resource container
eTEEMAccessPolicy	EEMAccessPolicy	Access policy
eTEEMAccessPolicyContainer	EEMAccessPolicyContainer	Access policy container
eTEEMPOLICY	EEMPOLICY	Policy
eTEEMPOLICYContainer	EEMPOLICYContainer	Policy container

Note: Information about the EEM policy and account template container can be found in the common objects schema. For details, see the Provisioning Guide for P.

SAP Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTSAPAccountContainer	SAPAccountContainer	Account container
eTSAPAccount	SAPAccount	SAP account
eTSAPProfileContainer	SAPProfileContainer	SAP profile container
eTSAPProfile	SAPProfile	SAP profile
eTSAPRoleContainer	SAPRoleContainer	SAP role container
eTSAPRole	SAPRole	SAP role
eTSAPPolicyContainer	SAPPolicyContainer	SAP policy container
eTSAPPolicy	SAPPolicy	SAP policy

Siebel Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTNamespace	Namespace	Namespace name
eTSBLDirectory	SBLDirectory	Siebel directory
eTSBLUserContainer	SBLUserContainer	User account container
eTSBLUser	SBLUser	User account
eTSBLPositionContainer	SBLPositionContainer	Position container
eTSBLPosition	SBLPosition	Position
eTSBLDivisionContainer	SBLDivisionContainer	Division container
eTSBLDivision	SBLDivision	Division
eTSBLViewContainer	SBLViewContainer	View container
eTSBLView	SBLView	View
eTSBLResponsibilityContainer	SBLResponsibilityContainer	Responsibility container
eTSBLResponsibility	SBLResponsibility	Responsibility
eTSBLLOVContainer	SBLLOVContainer	List of Values container

LDAP Object Name	User-Friendly Name	Description
eTSBPolicyContainer	SBLPolicyContainer	Policy

RSA Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTNamespace	Namespace	Namespace name
eTRSAAccount	RSAAccount	Account
eTRSAAccountContainer	RSAAccountContainer	Account container
eTRSAAgentHost	RSAAgentHost	Agent Host
eTRSAAgentHostContainer	RSAAgentHostContainer	Agent Host container
eTRSAGrp	RSAGrp	Group
eTRSAGrpContainer	RSAGrpContainer	Group container
eTRSAPolicy	RSAPolicy	Policy
eTRSAPolicyContainer	RSAPolicyContainer	Policy container
eTRSASite	RSASite	Site
eTRSASiteContainer	RSASiteContainer	Site container
eTRSAToken	RSAToken	Token
eTRSATokenContainer	RSATokenContainer	Token container

PKI Object User-Friendly Names

The following table lists the LDAP object names and their user-friendly names:

LDAP Object Name	User-Friendly Name	Description
eTPKIAccount	PKIAccount	Account
eTPKIAccountContainer	PKIContainer	Account container
eTPKIDirectory	PKIDirectory	Directory name
eTPKIPolicy	PKIPolicy	Policy
eTPKIPolicyContainer	PKIPolicyContainer	Policy container
eTNamespace	Namespace	Namespace name

Appendix B: Procedures

This appendix covers the basic tasks that can be performed using your connector.

This section contains the following topics:

[Endpoint Procedures](#) (see page 577)

[Account Template Procedures](#) (see page 578)

[Account Procedures](#) (see page 579)

[Group Procedures](#) (see page 582)

Endpoint Procedures

This section describes the tasks that can be performed on endpoints using the Provisioning Manager.

Registering Endpoints

To register an endpoint, perform the following procedure:

From the Endpoint Type task view

1. Select the endpoint type from the Object Type drop-down list box that you want to register.
2. Click New.
3. Complete the Endpoint property sheet for your endpoint.
4. Click OK.

Modifying or Viewing Endpoints

To modify or view an endpoint, perform the following procedure.

From the Endpoints Type task view

1. Select the endpoint type in the Object Type drop-down list box and search for the endpoint that you want.
2. Right-click the endpoint in the list view and choose Properties from the pop-up menu (You can also double-click the endpoint to bring up the Properties in the right frame).
3. Modify or view the properties in the Endpoint property sheet.
4. Click Apply/OK.

Account Template Procedures

This section describes the tasks that can be performed on endpoints.

Creating Account Templates

To create account templates, perform this procedure.

1. Click the Roles task button and select the account template type in the Object Type drop-down list box for your connector.
2. Click New.
3. Complete the Account Template property sheet for your connector and click OK.

A new account template has been created.

Modifying or Viewing Account Templates

To modify or view an account template's properties, perform this procedure.

1. Click the Roles task button and select the account template type for the connector you want to view or modify.
2. Click Search for the account template that you want.
3. Right-click the account template in the list view and choose Properties from the pop-up menu. You can also double-click the object to display its properties.
4. Modify or view the properties in the Account Template property sheet and click Apply.

The account template has been viewed and modified.

Using the Default Account Template

Identity Manager provides a default account template for each installed connector. This account template is called the xxxDefaultPolicy, where xxx is replaced by an endpoint type-specific prefix. The default account template contains the minimum security levels needed to access an endpoint in an endpoint type.

You can use the default endpoint type account template as a model when creating account templates for a provisioning role.

Account Procedures

This section describes the tasks that can be performed on accounts.

Creating Accounts

To create accounts, perform this procedure.

1. Click the Endpoints task button and select the endpoint type in the Object Type drop-down list box.
2. Click Search to search for the endpoint on which you want to create an account.

3. Right-click the endpoint in the list view and choose Content from the pop-up menu.

The Endpoint Content dialog appears.

4. Select a container in the Container Tree box and the kind of object (account, group, and so forth) in the Object Type box and click New.
5. Complete the Account property sheet and click OK.

An account has been created.

Adding Group Memberships to Existing Accounts

To add group memberships to an existing account, perform this procedure.

1. Click the Endpoints task button and select the endpoint type in the Object Type drop-down list box.
2. Click Search to search for the endpoint that you want.
3. Right-click the endpoint in the list view and choose Content from the pop-up menu.

The Endpoint Content dialog appears.

4. Select a container in the Container Tree box and the kind of object (account, group, and so forth) in the Object Type box and click Search.

The accounts for your connector appear in the list view.

5. Click Done.
6. Right-click the account in the list view and choose Copy Object(s) from the pop-up menu.

The object you selected is copied to the clipboard.

7. You can paste an account onto a group. To do this, right-click the group and choose Paste Object(s) from the pop-up menu.

Modifying or Viewing Account Properties

To modify or view the properties of an account, perform this procedure.

1. Click the Endpoints task button and select the endpoint type in the Object Type drop-down list box.
2. Click Search to search for the endpoint that you want.
3. Right-click the endpoint in the list view and choose Content from the pop-up menu.

The Endpoint Content dialog appears.

4. Select Accounts in the Container Tree box and then click Search.

The accounts for your endpoint appear in the list view.

5. Click Done.
6. Right-click an account in the list view and choose Properties from the pop-up menu. You can also double-click the object to display its properties.
7. Modify or view the properties in the Account property sheet for your endpoint and click Apply.

The account for your endpoint has been viewed and modified.

Checking Account Synchronization

To check if an account is compliant with its account template(s), perform the following procedure:

1. Click the Endpoints task button and select the endpoint type in the Object Type drop-down list box.
2. Click Search for the endpoint that you want.
3. Right-click on the endpoint in the list view and choose Content from the pop-up menu.

The Endpoint Content dialog appears.

4. Select Accounts in the Container Tree.

The accounts appear in the list view.

5. Click Search.

The accounts appear in the list view.

6. Click Done.
7. Right-click on the account in the list view and choose Check Account Synchronization from the pop-up menu.

Deleting Accounts

To delete an account, perform this procedure.

1. Click the Endpoints task button and select an endpoint type in the Object Type drop-down list box.
2. Click Search to search for the endpoint that you want.
3. Right-click the endpoint in the list view and choose Content from the pop-up menu.

The Endpoint Content dialog appears.

4. Select Accounts in the Container Tree box and then click Search.

The accounts for your endpoint appear in the list view.

5. Click Done.
6. Right-click an account in the list view and choose Delete from the pop-up menu.
7. Click Yes in the confirmation dialog.

The account has been deleted.

Group Procedures

This section describes the tasks that can be performed on groups when they are supported in the endpoint type.

Creating Groups

To create groups, perform this procedure.

1. Click the Endpoints task button and select an endpoint type in the Object Type drop-down list box.
2. Click Search to search for the endpoint that you want.
3. Right-click the endpoint in the list view and choose Content from the pop-up menu.

The Endpoint Content dialog appears.

4. Select a container in the Container Tree box and the kind of object (account, group, and so forth) in the Object Type box and click New.
5. Complete the Group property sheet for your endpoint and click OK.

A new group has been created.

Deleting Groups

To delete a group, perform this procedure.

1. Click the Endpoints task button and select an endpoint type in the Object Type drop-down list box.
2. Click Search to search for the endpoint that you want.
3. Right-click the endpoint in the list view and choose Content from the pop-up menu.

The Endpoint Content dialog appears.

4. Select Groups in the Container Tree box and then click Search.

The groups for your endpoint appear in the list view.

5. Click Done.
6. Right-click a group in the list view and choose Delete from the pop-up menu.
7. Click Yes in the confirmation dialog.

The selected group has been deleted.

Index

1

- 1. Use the Certsrv Web Page from the Active Directory Server • 182

2

- 2. Manually Export and Import • 182

A

- ACC Account Templates • 322
- ACC Connector Multi-Threading Support • 315
- ACC Etautil Conventions • 325
- ACC Support for FIPS and IPv6 • 320
- Access and Resource Rules and Rule Keys • 89
- Access Control Common Objects Tree • 554
- Access Control Object User-Friendly Names • 567
- Account Custom Operation (Rename, Recertify, Move In Hierarchy) • 308
- Account Management - Map a Certificate to an Account Object • 200
- Account Modify • 480
- Account Password Management • 456
- Account Password Management in CUA Environment • 459
- Account Procedures • 575
- Account Properties • 488
- Account Properties Tab • 386
- Account Resume • 480
- Account Screen Specifications • 92
- Account Search • 480
- Account Suspend • 480
- Account Template Creation with etautil • 31
- Account Template Procedures • 574
- Accounts • 30
- ACF2 Accounts • 88
- ACF2 Common Objects Tree • 549
- ACF2 Object User-Friendly Names • 560
- ACF2 Program Exits • 95
- ACF2 Provisioning Roles and Account Templates • 88
- ACF2 Support for FIPS and IPv6 • 85
- Acquire a CA ACF2 System • 86
- Acquire a CA Single Sign-On Server • 343
- Acquire a CA Top Secret System • 104
- Acquire a DB2 UDB Database • 42
- Acquire a DBZ Database • 36
- Acquire a Lotus Notes/Domino Server • 294
- Acquire a New Endpoint • 363
- Acquire a RACF Endpoint • 129
- Acquire a SAP Endpoint • 451
- Acquire a Siebel Server • 463
- Acquire a UNIX Server • 159
- Acquire an ADS Server • 197
- Acquire an EEM Server Machine • 441
- Acquire an Entrust Authority Security Manager System • 486
- Acquire an HP NonStop SAFEGUARD Machine • 437
- Acquire an Ingres Database • 51
- Acquire an MS SQL Server • 47
- Acquire an NDS Tree • 239
- Acquire an OpenVMS System • 428
- Acquire an Oracle Application System • 63
- Acquire an Oracle System • 56
- Acquire an OS/400 Machine • 407
- Acquire an RSA ACE/Server Machine • 474
- Acquire Endpoints • 29
- Acquire or Remove a New Endpoint • 36
- Acquire the CA Access Control Server • 321
- Activate the CAM and CAFT Encryption for Windows NT • 71
- Activating CAM and CAFT Encryption • 270
- Activating Syslog Messages • 172
- Activating Trace Messages • 173
- Active Directory Services Connector • 176
- Active Directory Services Objects • 528
- Add a KRB Account Template • 371
- Add a KRB Password Policy • 376
- Add a Principal • 367
- Add New Endpoint Request • 43
- Add Operation • 18
- Add the SendMail Exit to a Universal Provisioning Endpoint • 258
- Adding an Exit Reference for Monitor • 264
- Adding Group Memberships to Existing Accounts • 576
- Additional NDS Install Requirements • 238
- Administer Secondary Authids • 91
- Administer the Ingres Server • 49
- Administration Process (Adminp) • 292

- ADS Account Templates • 275
- ADS Common Objects Tree • 552
- ADS Default Account Template • 198
- ADS Defaults • 185
- ADS Etautil Conventions • 208
- ADS Object User-Friendly Names • 565
- ADS Support for FIPS and IPv6 • 195
- ADS_MANAGE_GROUPS • 191
- ADSI Option • 190
- Alternative Languages Support for both Organization and Organizational Unit Certifiers • 301
- Architecture • 332
- Archive Database Data Collection • 305
- As a CUA Engine Processing Mode • 458
- Assign the Server Certificate to Your OS/400 Applications • 400
- Assigning Account Templates to Provisioning Roles • 253
- Associating ACC Account Templates with Endpoints • 325
- Attribute Definitions • 233, 509
- Attribute Mapping • 303
- Authenticating to the Identity Manager Server • 499
- Authentication Methods • 351
- Authentication Process • 274
- Authorizing Access to the CA Access Control Server • 319
- Authorizing Account Template Administration • 126
- Automated Administration of Secondary Authids • 93
- Automatically Start the Agent • 422
- Availability Requirements for Categories and Security Labels • 330
- Availability Requirements for Groups • 329
- Available Groups in Account Template Property Sheet • 443
- Axis Library Logging • 504

B

- Best Practices • 118
- Business Application Connectors • 445

C

- C++ to Java Connectors Migration • 28
- CA Access Control Connector • 314
- CA Access Control Connector for UNIX • 316

- CA Access Control Objects • 537
- CA ACF2 Connector • 83
- CA Product References • iii
- CA SSO Connector for Advanced Policy Server • 337
- CA Top Secret Connector • 101
- CA-ACF2 Objects • 518
- Capability Attributes • 124
- CA-Top Secret Objects • 519
- Certificates • 491
- Change a User Role • 489
- Change Administrator Passwords • 365
- Change Control • 89
- Change the Active Directory Search Limit • 206
- Change the Certificate Type • 489
- Changed Passwords are Expired • 452
- Changing Connection Settings • 416
- Changing Passwords Using Windows Tools • 333
- Changing the Encryption Method for the CA Access Control Connector • 316
- Check the Policy Setting • 72, 158
- Check Your CA LDAP Server for z/OS Configuration Information • 88, 107
- Checking Account Synchronization • 577
- Checking Your CA LDAP Server for z/OS Configuration Information • 131
- Command Line Options • 495
- Common Communication Interface (CCI) Installation for HP NonStop Safeguard • 435
- Common Connector Operations • 17
- Common Objects Tree • 545
- Common Program Exits • 247
- Comparing PMDB to Local Database • 332
- Computers • 204
- Conditions for the execution of the EXIT Commands • 163
- Configuration and Usage • 100, 117
- Configuration File • 80, 162
- Configuration for ACF2 • 90
- Configure ODBC • 50
- Configure ODBC on Your Oracle Client • 61
- Configure the Agent • 422
- Configure the CAM and CAFT Service for Windows NT • 69
- Configure the CAM and CAFT Service--VMS • 423
- Configure the Java Connector Server • 401
- Configure the JDBC URL • 45

- Configure the Lotus Notes/Domino Connector • 286
- Configure the pam.conf File • 170
- Configure the pam_CA_etc.conf File • 167
- Configure the Virtual Node • 49
- Configure the Windows Service • 46
- Configure Your OS/400 System • 395
- Configure Your Windows Servers Using SSL • 177
- Configuring • 68
- Configuring a CA Access Control UNIX or Windows Server • 317
- Configuring CA Access Control UNIX on the C++ Connector Server System • 317
- Configuring the CA Single Sign-On Server • 338
- Configuring the Exchange 200x Remote Agent Component • 267
- Configuring the PeopleSoft Feed • 501
- Configuring the UNIX Password Synchronization Feature • 166
- Connect to a Child-Domain • 185
- Connect Using SSL • 395
- Connecting to a CUA Child • 459
- Connecting to a CUA Master • 459
- Connecting to the Nearest Domain Controller • 204
- Connector Acronym's • 506
- Connector and Agent Installation • 21
- Connector and C++ Connector Server Installation • 23
- Connector Objects and DNs • 511
- Connector Only Installation • 25
- Connector Overview • 13
- Connector Types • 15
- Connector.XML File • 410
- Connectors • 13
- Connectors Installed Using the Connector and Agent Installation • 22
- Connectors Installed Using the Connector and C++ Connector Server Installation • 25
- Connectors Installed Using the Connector Only Installation • 26
- Connectors Using the Java Connector Server Installation • 28
- Connector-Specific Features • 35, 41, 47, 51, 55, 62, 74, 85, 103, 128, 159, 196, 216, 239, 246, 273, 294, 320, 342, 360, 407, 427, 436, 440, 451, 462, 473, 484

- Contact CA • iii
- Conventions • 97, 416
- Correlate ADS Extended Attributes • 195
- Correlate on Shortname • 303
- Create a New User and SAP Role with Minimum Rights for Administration • 454
- Create a PKI Environment • 484
- Create a Profile for an Existing User • 490
- Create a UPO Endpoint • 244
- Create Account Templates • 43, 64, 429
- Create an ADS Account Template • 199
- Create DBZ Accounts • 39
- Create PKI Accounts • 488
- Creating a SendMail Program Exit • 258
- Creating a Service for Your Oracle Client • 60
- Creating Account Templates • 574
- Creating Accounts • 575
- Creating Custom ADS Attributes • 214
- Creating E-mail Addresses for Groups • 279
- Creating Groups • 578
- Creating LDAP User Entries • 218
- Creating the System Administrator's Account • 318
- CUA Distribution Settings • 460
- Custom Attribute Support • 299
- Custom Menu on Accounts and Contacts • 278
- Custom Menu on Group • 278
- Custom Settings Tab • 384

D

- Data Passed to Program Exits • 247
- Database Connectors • 33
- Database Roles • 48
- Database Users • 48
- DB2 Limitation • 34
- DB2 Provisioning Roles and Account Templates • 42
- DB2 UDB Common Objects Tree • 546
- DB2 UDB for z/OS Connector • 34
- DB2 UDB Groups • 43
- DB2 UDB Installation • 40
- DB2 UDB Object User-Friendly Names • 557
- DB2 UDB Objects • 512
- DB2 UDB Support for FIPS and IPv6 • 41
- DB2 z/OS Support for FIPS and IPv6 • 34
- DBZ Account Templates • 37
- DBZ Accounts • 38
- DBZ Endpoint • 35
- DBZ Installation Prerequisites • 34

- DBZ Object User Friendly Names • 557
- DBZ Server Common Object Tree • 546
- DBZ Server Objects • 511
- DBZ User Property Sheet • 39
- Default Primary Group on Endpoint Property Sheet • 160
- Define a CAFTHOST Symbolic Name • 422
- Define UPO Exits Using Manager • 254
- Definition File Names • 230
- Delete a KRB Account Template • 374
- Delete a KRB Password Policy • 378
- Delete a Principal • 369
- Delete an ADS Account Template • 199
- Delete Operation • 18
- Deleting Account Members from Groups • 415
- Deleting Accounts • 578
- Deleting Groups • 579
- Delta Dump • 497
- Departments • 108
- Directory Services Connectors • 175
- Disable a Token • 477
- Display Parameters • 237
- Distinguished Names • 510
- Distributed Connector • 67
- Distribution Lists • 279
- Distribution to Security • 203
- Divisions • 108
- DJX Support • 298
- DLL Methods • 255
- Domain Local to Universal • 202
- DYN GUI • 31
- Dynamic Connectors • 17

E

- E2K Etautil Conventions • 279
- E2K Support for FIPS and IPv6 • 273
- EEM Access Policies • 442
- EEM Account Templates • 442
- EEM Accounts • 442
- EEM Calendars • 443
- EEM Groups • 443
- EEM Installation • 440
- EEM Object User-Friendly Names • 569
- EEM Objects • 540
- EEM Resource Classes • 443
- EEM Support for FIPS and IPv6 • 440
- Embedded Entitlements Manager Common Objects Tree • 555
- Embedded Entitlements Manager Connector • 439
- Enable a Token • 477
- Enable Application Password Propagation • 341
- Enable the Use of SSL to Manage ADS • 180
- Enabling the Administrator's Account • 319
- Encryption Type and Salt Pairs Dialog • 391
- Endpoint Connections • 487
- Endpoint Overview • 427
- Endpoint Procedures • 573
- Endpoint Schema • 505
- Endpoint Schema and Structure • 505
- Endpoint Structure • 510
- Endpoints • 29
- Enhanced Filtering • 219
- Entrust PKI Connector • 481
- Entrust PKI Connector Installation Prerequisites • 482
- Entrust PKI Connector Limitations • 482
- Entrust PKI Installation • 481
- Entrust PKI Post Installation Requirements • 483
- Error when Changing Password on Native LDAP System • 218
- Errors • 94
- Etautil Script Considerations • 293
- ETAUTIL Tool • 207
- Example Files • 215
- Exchange 200x Log Files • 266
- Exchange Accounts • 277
- Exit Commands • 79
- Exit Types • 247
- Exits • 247
- Exits Unique to the Managed Mode • 251
- Exits Unique to the Non-Managed Mode • 250
- Exploration Results in Managed Mode • 252
- Exploration Results in Non-Managed Mode • 252
- Explore and Correlate Endpoints • 29
- Explore and Correlate Principals • 366
- Explore Non-Dialog Accounts Without Correlation • 456
- Extend Definition File Parser • 231
- Extend the ADS Schema • 193
- Extending Custom Attributes • 221
- Extending the Generic LDAP Connector • 222
- Extension File Header • 232

F

- Facilities • 109
- Failover • 186
- Failover Retry Interval • 189
- Features • 115
- Feed Connector • 493
- File Class Administration • 322
- File Formats • 507
- Finding Capability and Initial Attributes • 19
- Firewall Configuration • 350
- Flat Connectors • 16
- For Windows 2000 • 269
- For Windows NT • 269
- Force Logging • 192
- Full Dump • 497

G

- Generic LDAP Common Objects Tree • 552
- Generic LDAP Connector • 216
- Generic LDAP Object User-Friendly Names • 565
- Global and Application Endpoints are Managed Separately • 442
- Global to Universal • 202
- Grace Interval on Logon Tab of SAP Account Template Property Sheet • 455
- Grant Access to the Provisioning Server Host • 149
- Group Management - Changing Group Scope • 202
- Group Management - Changing Group Type • 202
- Group Procedures • 578
- Groups • 30, 108
- Groupware Connectors • 265

H

- Hierarchical Connectors • 16
- Host Access Management Connectors • 313
- How Exploration and Correlation Works • 120
- How Managed Objects are Referred to in the Java LND Connector • 296
- How PKI Endpoints are Authenticated • 485
- How the Password Synchronization Agent Works • 402
- How the PeopleSoft Feed Works • 494
- How to Acquire an LDAP Server • 218
- How to Acquire the Windows NT Machine • 75

- How to Add the Exit Reference to the UPO Endpoint • 248
- How to Configure and Start the OpenVMS Agent on your OpenVMS System • 422
- How to Configure the CAM and CAFT Service • 473
- How to Configure the CAM and CAFT Service for Windows NT • 70
- How to Configure the Oracle Applications Connector • 59
- How to Configure UPO • 244
- How to Configure Your CA ACF2 System • 84
- How to Configure Your CA Top Secret System • 101
- How to Configure your Machines • 394
- How to Configure Your RACF System • 127
- How to Explore the UPO Endpoint • 252
- How to Extend the Generic LDAP Connector • 223
- How to Install the OpenVMS Agent • 419
- How to Install the SAP Connector with the Java Connector Server • 447
- How to Manage Mailboxes • 266
- How to Manage the CAM and CAFT Service • 423
- How to Restart Automatically the CAM Service • 155
- How to Restrict CAFT Commands • 156
- How to Secure Your Information (Optional) • 394
- How to Set Up the Java CS Host to be a Member of the Target Realm • 352
- How to Synchronize an Account from an Account Template • 44
- How to Write Auxiliary Object Class Definition Files • 228
- How UNIX PAM Works • 164
- How You Acquire and Manage Kerberos Endpoints • 361
- How you set Up Keytab Authentication Using the Host Principal • 352
- HP NonStop SAFEGUARD Connector • 431
- HP NonStop SAFEGUARD Objects • 525

I

- IBM DB2 UDB Connector • 40
- Ignore Group Insufficient Rights Error • 192
- Import the Certificate • 182
- Import the Certificate Authority • 397

- Import the Root Certification Authority • 179, 183
- Import the Server Certificate • 399
- Individual Entrust Profile Approach • 485
- Ingres Accounts • 52
- Ingres Common Objects Tree • 547
- Ingres Connector • 48
- Ingres Database Limitations • 50
- Ingres Directory Tab Authentication Attributes • 52
- Ingres Etautil Conventions • 53
- Ingres Groups • 53
- Ingres Object User-Friendly Names • 558
- Ingres Objects • 514
- Ingres Profiles • 52
- Ingres Roles • 53
- Ingres Support for FIPS and IPv6 • 50
- Install and Configure Programs on OS/400 • 394
- Install and Configure the CAM Service for Irix and NCR • 150
- Install and Configure the TSS Connector • 101
- Install and Deploy the Connector • 349
- Install the CAM and CAFT Encryption Key • 157
- Install the CAM and CAFT Service for Windows NT • 69
- Install the Identity Manager UNIX Remote Agent • 145
- Install the Ingres Client and Ingres Connector • 49
- Install the LND Connector with JCS • 286
- Install the OpenVMS Agent From a Windows Machine • 420
- Install the OpenVMS Agent From an OpenVMS System • 421
- Install the Password Synchronization Agent • 402
- Install the PeopleSoft Feed • 493
- Install the Provisioning Server Certificate • 404
- Install the UNIX PAM Feature • 164
- Install UPO • 243
- Install Without a Microsoft Certificate Authority • 276
- Installation Options • 141
- Installation Prerequisites • 348
- Installation Requirements and Scenarios • 21
- Installation Requirements for Windows • 41
- Installing Filtering Rules for the Policy Model Database (PMDB) • 320

- Installing the HP NonStop SAFEGUARD Agent • 431
- Installing the Provisioning Agent for Windows Local Users and Groups with setup.exe • 69
- Installing the Provisioning Windows NT Remote Agent with msiexec • 68
- Interactive Installation • 138
- Issue the Certificate • 181

J

- Java Connector Server Installation • 27

K

- Kerberos Common Objects Tree • 554
- Kerberos Connector • 345
- Kerberos Connector Installation Options • 348
- Kerberos Connector Limitations • 345
- Kerberos Default Account Template • 369
- Kerberos Installation and Deployment • 348
- Keytab and Cross-realm Paths Setup • 351
- KRB Account Property Sheet • 385
- KRB Account Template Property Sheet • 388
- KRB Directory Tab • 380
- KRB Endpoint Property Sheet • 379
- KRB Endpoints Tab • 390
- KRB Etautil Conventions • 360
- KRB Object User-Friendly Names • 568
- KRB Objects • 539

L

- LDAP Accounts • 220
- LDAP Containers • 221
- LDAP Default Account Templates • 220
- LDAP Directory Services (LDS) • 97, 114
- LDAP Groups • 221
- LDAP Objects • 217, 529
- LDAP Support for FIPS and IPv6 • 216
- LDAPMODRDN Command • 207
- LDS Features • 99
- LND Account Templates • 304
- LND Accounts • 308
- LND Common Objects Tree • 553
- LND Etautil Conventions • 309
- LND Installation • 280
- LND Java Implementation Considerations • 281
- LND Migration • 284
- LND Support for FIPS and IPv6 • 293
- Locations for Storing Certifier IDs • 298
- Locations for Storing User IDs • 298

Logging • 135
LOGINAPPL Class Administration • 324
Lotus Notes/Domino Connector • 280
Lotus Notes/Domino Object User-Friendly
Names • 567
Lotus Notes\Domino Objects • 536

M

Mainframe Connectors • 83
Manage Oracle Applications User Accounts • 65
Manage the CAM and CAFT Service for
Windows NT • 73
Manage the CAM Service • 153
Managed Objects • 296
Management of Alternate Names on LND
Accounts • 308
Managing Exchange Users • 274
Managing Exits • 253
Managing Passwords • 163
Managing SAP Central User Administration
(CUA) Environments • 457
Managing the CAM and CAFT Service for
Microsoft Exchange 200x • 272
Managing the Connectors • 29
Mandatory Properties for etautil Add Command
• 160
Mapping Configuration from Windows • 334
Mark a Pending Operation as Completed with
etautil Utility • 260
Mark a Pending Operation as Completed with
Provisioning Manager • 260
Mark a Pending Operation as Completed with
the LDAP Protocol Directly • 261
Marking a Pending Operation as Completed •
260
Member Of • 491
Microsoft Best Practices for Group
Memberships • 203
Microsoft Exchange 200x Connector • 265
Microsoft Exchange Distribution Lists and ADS
Policies Using Strong Synchronization • 276
Middleware Connectors • 439
Midrange Connectors • 393
Migrate Passwords • 285
Modify a KRB Account Template • 373
Modify a KRB Password Policy • 377
Modify a Principal • 368
Modify an Endpoint • 364
Modify Home Server • 308

Modify Operation • 18
Modify SAP Account Template Properties • 453
Modify the schema.ext File • 194
Modifying or Viewing Account Properties • 577
Modifying or Viewing Account Templates • 575
Modifying or Viewing Endpoints • 574
Monitor Service • 262
Mount the CD-ROM • 139
Move Functions • 206
Move Using ETAUTIL • 208
MS SQL Configuration • 45
MS SQL Conventions • 48
MS SQL Migration Steps • 46
MS SQL Object User-Friendly Names • 558
MS SQL Server Common Object Tree • 547
MS SQL Server Connector • 45
MS SQL Server Objects • 513
MS SQL Support for FIPS and IPv6 • 46

N

Naming Limitations • 346
NDS Accounts • 240
NDS Common Objects Tree • 553
NDS Connector • 238
NDS Containers • 241
NDS Default Account Templates • 240
NDS Groups • 240
NDS Object User-Friendly Names • 566
NDS Objects • 529
NDS Support for FIPS and IPv6 • 238
NDS Trees • 239
New ADS Accounts • 200
NIS Netgroups • 160
Non-Managed and Managed Modes Exits • 249
Non-Managed Mode Operations • 259
Non-Managed System Connector • 243
NSK Accounts • 438
NSK Common Objects Tree • 551
NSK Groups • 438
NSK Object User-Friendly Names • 563
NSK Support for FIPS and IPv6 • 436

O

Object Class Definitions • 508
Object User-Friendly Names • 557
OpenVMS Connector • 418
OpenVMS Disk Space and Environment
Requirements • 418
OpenVMS Installation and Configuration • 418

- OpenVMS Object User-Friendly Names • 563
- OpenVMS Objects • 525
- OpenVMS Sample Installation Log File • 424
- OpenVMS Support for FIPS and IPv6 • 427
- OpenvVMS Common Objects Tree • 550
- Operations • 490
- Oracle Account Templates • 57
- Oracle Accounts • 57
- Oracle Applications Common Objects Tree • 548
- Oracle Applications Connector • 58
- Oracle Applications Installation and Configurations • 59
- Oracle Applications Limitations • 61
- Oracle Applications Object User-Friendly Names • 559
- Oracle Applications Objects • 517
- Oracle Applications Prerequisite • 61
- Oracle Applications Support for FIPS and IPv6 • 62
- Oracle Applications User Provisioning Roles and Account Templates • 64
- Oracle Common Objects Tree • 548
- Oracle Configuration • 54
- Oracle Connector • 53
- Oracle Etautil Conventions • 58
- Oracle Limitations • 55
- Oracle Migration Steps • 55
- Oracle Object User-Friendly Names • 559
- Oracle Objects • 515
- Oracle Packages • 57
- Oracle Procedures • 57
- Oracle Profiles • 58
- Oracle Roles • 57
- Oracle Support for FIPS and IPv6 • 55
- Order of Program Exits in the Process Flow • 253
- Organizations and Organizational Units Handling Extended • 302
- OS/400 Cascading Delete • 409
- OS/400 Common Objects Tree • 550
- OS/400 Connector • 393
- OS/400 Directory Entry Names • 415
- OS/400 Groups • 415
- OS/400 Installation • 394
- OS/400 Migration Steps • 394
- OS/400 Native Program Exits • 417
- OS/400 Object User-Friendly Names • 563
- OS/400 Objects • 524

- OS/400 Password Agent Parameter Must be Set Correctly • 406
- OS/400 Provisioning Roles and Account Templates • 408
- OS/400 Security Requirements • 415
- OS/400 Support for FIPS and IPv6 • 406
- Ownerships • 109

P

- Paged Searches • 205
- Passing Account Attributes • 212
- Passing Global User Attributes • 213
- Password Account Template Properties Tab • 389
- Password Change Propagate Mode • 475
- Password Limitations • 347
- Password Management in a CUA Environment • 460
- Password Policy Naming Limitations • 347
- Password Synchronization • 307, 330
- PeopleSoft Feed • 493
- PeopleSoft Feed Error and Response Handling • 503
- PeopleSoft Feed Localization • 499
- PeopleSoft Feed Log Files • 504
- Permissions • 109, 132
- PKI Account Templates • 487
- PKI Accounts • 487
- PKI Common Objects Tree • 556
- PKI Object User-Friendly Names • 571
- PKI Objects • 545
- PKI Provisioning Roles and Account Templates • 487
- PKI Support for FIPS and IPv6 • 484
- PLS Authorization Hosts • 344
- PLS Control Application Groups • 344
- PLS Control Applications • 344
- PLS Object User-Friendly Names • 568
- PLS Objects • 538
- PLS Support for FIPS and IPv6 • 342
- PLS Terminal • 344
- Policy Default Values • 408
- Policy Synchronization • 92
- policy_setting Options • 158
- Polling Frequency • 263
- Prepare the System • 395
- Principal Naming Limitations • 347
- Printers • 204
- Procedures • 573

- Product Interchange Format (PIF) Installation Procedures • 138
- Profile Lists • 109
- Profiles • 108
- Profiles Tab • 385
- PROGRAM Class Administration • 323
- Program Exit Objects • 248
- Program Exits • 208
- Program Exits (Common or Native) • 360
- Properties File Example • 502
- Properties Tab • 382
- Provisioning Manager • 206
- Provisioning Roles and Account Templates • 30, 429
- Proxy Administration Support • 96, 112, 134
- Proxy Configuration • 95, 111, 134
- Proxy Profile Approach • 485
- Public Key Infrastructure Connector • 481

R

- RACF Accounts • 131
- RACF Common Objects Tree • 550
- RACF Connector • 126
- RACF Conventions • 133
- RACF Groups • 132
- RACF Object User-Friendly Names • 562
- RACF Objects • 523
- RACF Program Exits • 133
- RACF Provisioning Roles and Account Templates • 131
- RACF Support for FIPS and IPv6 • 128
- Recommended Patch Levels • 314
- Reconfiguring the Password Synchronization Component • 331
- Recover User Operation • 491
- Register ACF2 Endpoints on a Solaris System • 88
- Register ACF2 Endpoints on a Windows System • 87
- Register CA Top Secret Endpoints on Solaris • 107
- Register CA Top Secret Endpoints on Windows • 106
- Register RACF Endpoints on Solaris Machines • 131
- Register RACF Endpoints on Windows Machines • 130
- Registering Endpoints • 573
- REGKEY Class Administration • 325

- Re-Initialize an ADS Endpoint • 198
- Relocate Accounts • 201
- Relocate to Sync with Account Template • 207
- Remote Access Connector • 471
- Remove an Account from the CUA Master System • 460
- Removing ACC Endpoints from Account Templates • 330
- Removing an Account from the UPO Endpoint • 261
- Rename Accounts • 78
- Rename Operation • 18
- Request a Server Certificate for Your System • 399
- Request a Server Certificate from the CA • 398
- Request the Certificate • 181
- Required Oracle Administrator Account Privileges • 55, 61
- Requirements for the User Commands • 81
- Requirements for Using UNIX Password Synchronization • 164
- Resetting the Encryption Key for the CA Access Control Connector Back to the Default Key • 315
- Resource Management • 204
- Restart the CAM Service • 154
- Roles and Policies • 344
- RSA Common Objects Tree • 556
- RSA Endpoint Property Sheet • 474
- RSA Groups • 475
- RSA Installation • 472
- RSA Limitations • 472
- RSA Object User-Friendly Names • 571
- RSA Objects • 543
- RSA Post Installation Requirements • 472
- RSA Support for FIPS and IPv6 • 473
- RSA Tokens Property Sheet • 475
- Rule Lines • 90
- Run the Agent Processes • 434
- Runtime Environment Settings • 315

S

- Sample .CPP file • 209
- Sample etautl Commands • 310
- Sample Scenario Using Additional Groups • 327
- Sample Scenario Using Groups • 326
- SAP (CUA) Management • 457
- SAP Accounts • 455

SAP Common Objects Tree • 555
SAP Connector • 446
SAP Conventions • 457
SAP Installation • 446
SAP JCO Library Installation on SUN Solaris • 448
SAP JCO Library Installation on Windows NT, Windows 2000, and Windows 2003 • 449
SAP Migration Steps • 450
SAP Object User-Friendly Names • 570
SAP Objects • 541
SAP Prerequisites • 446
SAP Profiles • 455
SAP Provisioning Roles and Account Templates • 452
SAP Support for FIPS and IPv6 • 450
Scenario for Connector and Agent Installation • 22
Scheduling a Load • 498
SchemaAbridged.txt File • 506
SchemaUnabridged.txt File • 506
Search Operation • 18
Search Screen Notes • 92
Secondary Authids • 90
Securing Provisioning Server Communication to the CA LDAP Server • 85, 102, 127
Security to Distribution • 203
Select the Certificate Location • 397
Select the Installation Script • 140
Selecting the Character Set on the Directory Property Sheet • 160
SendMail Program Exit • 255
Service Installation • 262
Service Level Agreement (SLA) • 248
Service Removal • 262
Set Emergency Mode Off • 478
Set Emergency Mode On • 478
Set New PIN Mode • 478
Set Replacement Mode • 479
Set Up Keytab Authentication Using a Custom Keytab and a Principal Other than the Java CS Host Principal • 358
Set Up Keytab Authentication Using a Custom Keytab and the Java CS Host Principal • 355
Set Up Keytab Authentication Using the Default Keytab and a Principal Other than the Java CS Host Principal • 356

Set Up Keytab Authentication Using the Java CS Host Principal if Keytab File Does Not Exist • 353
Set Up Keytab Authentication Using the Java CS Host Principal if Keytab File Exists • 354
Set Up Principal and Password Authentication • 359
Set Up the Domino Server • 290
Setting Live Communication Server 2003 Attributes for ADS • 278
Setting the Encryption Key for the CA Access Control Connector • 315
Setting Up Role-based Administration • 122
Shared Folders • 79, 205
Siebel Common Objects Tree • 556
Siebel Connector • 461
Siebel Endpoint Property Sheet • 464
Siebel Installation • 462
Siebel Internal Division Property Sheet • 469
Siebel LOV Property Sheet • 469
Siebel Object User-Friendly Names • 570
Siebel Objects • 542
Siebel Organization Property Sheet • 468
Siebel Position Property Sheet • 468
Siebel Requirements • 462
Siebel Responsibility Property Sheet • 467
Siebel Support for FIPS and IPv6 • 462
Siebel User Property Sheet • 466
Siebel View Property Sheet • 468
Significant Fields in the Endpoint Tab of the Endpoint Property Sheet • 76
Silent Installation • 143
Silent Installation Notes • 144
Size Limit Exceeded • 79
Special Characters for ADS • 200
Specifying Datastore Names in an ADS Policy • 275
SSL Configuration • 404
SSL Prerequisites • 285
SSL Support • 500
SSO for Advanced Policy Server Common Objects Tree • 554
Start the CAM and CAFT Service • 423
Start the CAM and CAFT Service for Windows NT • 73
Start the CAM/CAFT Service • 272
Start the Installation Wizard • 141
Starting the Selang Command Interpreter • 317

Static Connectors • 16
Step 1. Configure the CAM and CAFT Service • 268
Step 1. Create Extension Definition Files • 223
Step 1. Install the CA LDAP Server • 84, 102
Step 1. Install the CA LDAP Server for z/OS • 127
Step 1. Install the High Encryption Pack • 177
Step 1. Start the Selang Command Interpreter • 338
Step 1. Transfer the Files to the HP NonStop SAFEGUARD System • 432
Step 1. Transfer the Necessary Files • 150
Step 1. Verify Your Access Privileges • 287
Step 2. Build LDAP Connector Plug-ins • 225
Step 2. Configure Your System • 433
Step 2. Create the System Administrator's Account • 338
Step 2. Enable the Administration Process (Adminp) • 287
Step 2. Install and Configure a Certificate Authority • 178
Step 2. Install the CAM Service • 151
Step 2. Review the CA LDAP Server • 127
Step 2. Review the CA LDAP Server for z/OS Configuration Options • 84, 102
Step 2. Update the CAM and CAFT Service Logon Account • 268
Step 2a. Log on Anonymously (Optional) • 178
Step 3. Add Encryption Keys to the Server ID • 288
Step 3. Add Extended LDAP Connector Objects to the Provisioning Directory • 226
Step 3. Configure the CAM Service • 152
Step 3. Enable the Administrator's Account • 339
Step 3. Set Up ADS, Certificate Authority, and Identity Manager • 178
Step 3a. Set Up on a Single Computer • 179
Step 3b. Set Up on a Dual Computer System • 179
Step 3c. Set Up Multiple Systems • 180
Step 4. Encrypt Local Databases • 289
Step 4. Install the UNIX Agent • 152
Step 4. Update the Schema to Include Extended Attributes • 227
Step 4. Verify the C++ Connector Server Service Logon ID • 184
Step 5. Restart the Services • 227

Step 5. Verify the Installation • 153
Stop the CAM and CAFT Service • 424
Stop the CAM and CAFT Service for Windows NT • 74
Stop the CAM Service • 154
Stop the CAM/CAFT Service • 273
Streaming Search Results • 408
Supported Use Cases • 497
Surrogate Class Administration • 324
Suspend/Resume • 307
Suspending and Resuming RSA Accounts • 480
Synchronize Accounts with Account Templates • 374
Synchronize an Account from an Account Template • 37
Synchronize BDC Systems • 78
System Databases • 297

T

TERMINAL Class Administration • 323
Terminal Server Attributes Management for Accounts • 77
Terminal Services • 203
Test the PAM Framework • 174
Testing the Configuration File • 173
Testing the LDAP/TLS Connection • 173
Testing the Password Synchronization • 174
The Command Line Interface • 211
The Default ADS Account Template • 199
The Dynamic Link Library (DLL) Interface • 209
The Provisioning Manager • 29
The RSA ACE (SecurID) Connector • 471
Token Management • 475
Token Operations Tab • 477
Tokens Profile Tab • 476
Tools for Managing Data in Endpoints • 360
Top Secret Common Objects Tree • 549
Top Secret Object User-Friendly Names • 561
Top Secret Support for FIPS and IPv6 • 102
Troubleshooting UNIX Password Synchronization • 172
Trust Relationships • 79
TSO Alias Support • 94, 109, 132
TSS Accounts • 108
TSS Conventions • 113
TSS Program Exits • 110
TSS Provisioning Roles and Account Templates • 107

U

- Understanding Account Templates • 122
- Understanding Failover • 186
- Understanding Synchronization • 123
- Understanding the Explore and Correlate Function • 119
- Un-Install the Password Synchronization Agent • 405
- Universal Provisioning Common Objects Tree • 553
- Universal Provisioning Connector • 243
- Universal Provisioning Object User-Friendly Names • 566
- Universal Provisioning Objects • 535
- Universal to Domain Local • 202
- Universal to Global • 202
- UNIX Connector • 137
- UNIX Etautil Conventions • 161
- UNIX ETC Common Objects Tree • 551
- UNIX ETC Object User-Friendly Names • 564
- UNIX ETC Objects • 526
- UNIX Groups • 160
- UNIX NIS Common Objects Tree • 551
- UNIX NIS Object User-Friendly Names • 564
- UNIX NIS Objects • 527
- UNIX Password Synchronization • 163
- UNIX Support for FIPS and IPv6 • 158
- Unsupported kadmin Options • 346
- Upgrading the Provisioning Server • 68
- UPO and UPO Monitoring Server Post Installation Step • 244
- UPO Endpoint Property Sheet • 257
- UPO Exits • 253
- UPO Installation Requirements • 243
- UPO Operational Modes • 246
- UPO Program Exit Property Sheet • 256
- UPO Program Exits • 247
- UPO Running in Managed Mode Account Creation Issue • 246
- UPO Support for FIPS and IPv6 • 245
- User Credentials • 263
- User Groups • 202
- User License Contractual Type • 456
- Using Account Templates • 124
- Using Failover • 187, 340
- Using Identity Manager Exit Commands • 161
- Using Model ACIDs • 126
- Using PeopleSoft Feed • 498
- Using Profiles with Account Templates • 125

- Using Program Exits When Creating ADS Objects • 208
- Using the Default Account Template • 575
- Using the Init Utility • 155
- Using the RC Utility • 156
- Using the XSLT Template • 498
- Using z/OS Security Administration • 118

V

- View a KRB Password Policy • 375
- View All Principals • 366
- View KRB Account Templates • 370
- View the CAM and CAFT Service • 423
- View the CAM and CAFT Service for Windows NT • 73
- View the CAM Service • 153
- View the CAM/CAFT Process • 272
- VMS Program Exits • 430

W

- When to Perform Exploration and Correlation • 121
- Windows NT Common Objects Tree • 548
- Windows NT Connector • 67
- Windows NT Groups • 78
- Windows NT Object User-Friendly Names • 560
- Windows NT Objects • 517
- Windows NT Support for FIPS and IPv6 • 74

Z

- Zones • 108