Oracle® Identity Manager

Connector Guide for CA Top Secret Advanced

9.1.0.9.0
F21529-10
March 2023

Oracle Identity Manager Connector Guide for CA Top Secret Advanced, 9.1.0.9.0

F21529-10

Primary Authors: Maya Chakrapani, Mike Howlett

Contributors: Amol Datar, Vaidyanath Laturkar, Nilesh Nikalje

Contents

Preface

What's New in the Oracle Identity Manager Advanced Connector for CA Top Secret?

1 About the CA Top Secret Advanced Connector

List of Figures

List of Tables

Preface

This guide describes the connector that is used to integrate Oracle Identity Manager with CA Top Secret.

Audience

This guide is intended for resource administrators and target system integration teams. Installation of the connector components on the mainframe requires experience with CA Top Secret and various z/OS technologies and components, including TCP/IP, QSAM (flat files), and z/OS libraries.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

For information about installing and using Oracle Identity Governance 12.2.1.3.0, visit the following Oracle Help Center page:

`https://docs.oracle.com/en/middleware/idm/identity-governance/12.2.1.3/index.html`

For information about installing and using Oracle Identity Manager 11.1.2.3, visit the following Oracle Help Center page:

`http://docs.oracle.com/cd/E52734_01/index.html`

For information about Oracle Identity Governance Connectors 12.2.1.3.0 documentation, visit the following Oracle Help Center page:

`https://docs.oracle.com/en/middleware/idm/identity-governance-connectors/12.2.1.3/index.html`

For information about Oracle Identity Manager Connectors 11.1.1 documentation, visit the following Oracle Help Center page:

`http://docs.oracle.com/cd/E22999_01/index.htm`

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

What's New in the Oracle Identity Manager Advanced Connector for CA Top Secret?

These are the updates made to the software and documentation for release 9.1.0.6.0 of the Oracle Identity Manager Advanced Connector for CA Top Secret.

The updates discussed in this chapter are divided into the following categories:

- Software Updates

  This section describes updates made to the connector software.

- Documentation-Specific Updates

  These include major changes made to the connector documentation. These changes are not related to software updates.

Software Updates

These are the updates made to the connector software.

- Software Updates in Release 9.1.0.9.0
- Software Updates in Release 9.1.0.8.1
- Software Updates in Release 9.1.0.8.0
- Software Updates in Release 9.1.0.7.0
- Software Updates in Release 9.1.0.6.0
- Software Updates in Release 9.1.0.5.0
- Software Updates in Release 9.1.0.4.0
- Software Updates in Release 9.1.0.3.0
- Software Updates in Release 9.1.0.2.0
- Software Updates in Release 9.1.0.1.0
- Software Updates in Release 9.1.0.0.0

Software Updates in Release 9.1.0.9.0

The following are software updates in release 9.1.0.9.0:

Resolved Issues in Release 9.1.0.9.0

The following table lists the issues resolved in release 9.1.0.9.0:

Bug Number	Issue	Resolution
32498921	CVE-2021-26117: APACHE ACTIVEMQ UPDATE TO AT LEAST 5.16.1 OR 5.15.14.	This issue has been resolved.
32054805	CVE-2019-10086: APACHE COMMONS BEANUTILS UPDATE TO AT LEAST 1.9.4.	This issue has been resolved.
31974483	CVE-2020-5421: SPRING FRAMEWORK UPDATE TO AT LEAST 5.2.9, 5.1.18, 5.0.19, OR 4.3.29.	This issue has been resolved.

Software Updates in Release 9.1.0.8.1

The following are software updates in release 9.1.0.8.1:

Resolved Issues in Release 9.1.0.8.1

The following table lists the issues resolved in release 9.1.0.8.1:

Bug Number	Issue	Resolution
32663510	TSS 9108 w/IPV6 - UnSuccessful Connection to LDAP Using IPV6 hostname.	This issue has been resolved.

Software Updates in Release 9.1.0.8.0

The following are software updates in release 9.1.0.8.0:

Resolved Issues in Release 9.1.0.8.0

The following table lists the issues resolved in release 9.1.0.8.0:

Bug Number	Issue	Resolution
31778959	Support PhraseOnly for TSS.	This issue has been resolved.
31046304	IPV6 support for TSS.	This issue has been resolved.
32408771	Wrong command is being used for remove INSTDATA operation from OIM to Top Secret	This issue has been resolved.
31935863	TSS 9.1.0.4 - Recon Timezone Issue	This issue has been resolved.

Software Updates in Release 9.1.0.7.0

The following are software updates in release 9.1.0.7.0:

Resolved Issues in Release 9.1.0.7.0

The following table lists the issues resolved in release 9.1.0.7.0:

Bug Number	Issue	Resolution
31935863	Timestamp attribute in logs were shown in GMT timezone, and not in Oracle Identity Manager timezone.	This issue has been resolved.
31748336	EOF Exception encountered while constructing entryDN (`uid=$#02b002,ou=tops,ou=People,dc=system,dc=backend`) for IDs that have special characters like `#` followed by a numeric character.	This issue has been resolved.

Software Updates in Release 9.1.0.6.0

The following are software updates in release 9.1.0.6.0:

Resolved Issues in Release 9.1.0.6.0

The following table lists the issues resolved in release 9.1.0.6.0:

Bug Number	Issue	Resolution
31538898	When the `revokePsuspendUsers` property in the *`LDAP_INSTALL_DIR`*`/conf/tops.properties` file was set to `false`, user accounts with the PSUSPEND attribute were disabled in Oracle Identity Manager.	This issue has been resolved. When the `revokePsuspendUsers` property is set to `false`, user accounts with the PSUSPEND attribute are now being displayed as revoked accounts in Oracle Identity Manager.
31569978	While reconciling data via batch / CFILE, inconsistency in treatment of the `revoke` attribute is seen.	This issue has been resolved. If there is an `<attrs>` element in our `CFILE <record>`, but does not have an `ASUSPEND <attr>`, then we default to `revoke=n`. This includes a totally empty `<attrs>` element or `<attrs>` with other elements such as Console or Audit. However, if there is no `<attrs>` element in our `CFILE <record>` at all (which happens when there is no 0700 entry in the raw CFILE), then do not default to `revoke=n`. Instead, leave the revoke attribute as-is / blank.

Bug Number	Issue	Resolution
31647086	When an LDAP client, such as Oracle Identity Manager disconnects from the IDF Gateway unexpectedly, an error is logged by the IDF Gateway.	This issue has been resolved. While the log level and content is technically accurate, in all observed cases, the signal-to-noise ratio of this message causes an undue burden on our support staff When logging an instance of ClosedChannelException, use the DEBUG log level rather than the ERROR log level.

Software Updates in Release 9.1.0.5.0

The following are software updates in release 9.1.0.5.0:

- Support for New Oracle Identity Governance Release
- Logging Mechanism Enhanced
- Resolved Issues in Release 9.1.0.5.0

Support for New Oracle Identity Governance Release

From this release onward, you can install and use the connector with Oracle IdentityGovernance 12c PS4 (12.2.1.4.0).

See Table 1-1 for the full list of certified Oracle Identity Governance releases.

Logging Mechanism Enhanced

From this release onward, depending on the log level you set, the connector provides detailed information for any event, including reasons for an event failure.

Resolved Issues in Release 9.1.0.5.0

The following table lists the issues resolved in release 9.1.0.5.0:

Bug Number	Issue	Resolution
31113886	The Rename event is the target system was executed correctly. Rename event failed whenever you tried to rename the 8th character in a string. For example, renaming user ID `JSMITH25` to `JSMITH29` failed, however renaming `JSMITH25` to `JSMITH55` succeeded.	This issue has been resolved.

Bug Number	Issue	Resolution
30910256	When you delete accounts from the target system, information about these deleted accounts were fetched into the LDAP Gateway. Subsequently, when you ran the Top Secret Reconcile Users to Internal LDAP scheduled job (CFILE job), information about the deleted accounts were not reconciled into Oracle Identity Manager.	This issue has been resolved. The deleted records are now fetched into Oracle Identity Manager and are displayed as either Revoked or Deleted, depending on the operation performed on the target system.

Software Updates in Release 9.1.0.4.0

The following are software updates in release 9.1.0.4.0:

Resolved Issues in Release 9.1.0.4.0

The following table lists the issues resolved in release 9.1.0.4.0:

Bug Number	Issue	Resolution
30860763	The Attributes field in the LDAP gateway is a multivalued field. When you run the Top Secret Reconcile Users to Internal LDAP scheduled job (CFILE job), the Attributes field containing multiple values showed only a single value.	This issue has been resolved.
30897544	When using the EXPORT_MON parameter of the Pioneer control file, the progress message displayed on the number of records processed was incorrect. For example, suppose there are 750 records and you set EXPORT_MON=YES, REC=200. Then, Pioneer displayed a progress message for every 200 records in 3 iterations. And in the fourth iteration, Pioneer display that it processed 200 records, instead of 150.	This issue has been resolved. The number of records retrieved in the final iteration is now displayed correctly.

Bug Number	Issue	Resolution

30910322	When you ran the Top Secret Reconcile User to LDAP to Internal LDAP, the Netview attribute for some user accounts were not reconciled.	To ensure the Netview attributes for all users are reconciled, in the LDAP_INSTALL_DIR/conf/parser-grammars/tops/tops_recon_FetchAllUserData.xml file, search for and replace the **`Line id="fdtData"`** line with the following:
		`<Line id="netviewControl" required="true" enabled="yes" sig="^4011([A-Za-z0-9]{0,3})(\S)?\s+([\s\S\-]{18})NETVCTL\s+CONTROL\s+(?&lt;netviewControl&gt;[\s\S\-]{0,249})" />`
		`        <Line id="netviewConsname" required="true" enabled="yes" sig="^4011([A-Za-z0-9]{0,3})(\S)?\s+([\s\S\-]{18})NETVCONSCONSNAME(?&lt;netviewConsname&gt;[\s\S\-]{0,249})" />`
		`        <Line id="netviewInitCmd" required="true" enabled="yes" sig="^4011([A-Za-z0-9]{0,3})(\S)?\s+([\s\S\-]{18})NETVIC\s+INIT CMD\s+(?&lt;netviewInitCmd&gt;[\s\S\-]{0,249})" />`
		`        <Line id="netviewMsgrecvr" required="true" enabled="yes" sig="^4011([A-Za-z0-9]{0,3})(\S)?\s+([\s\S\-]{18})NETVMSGRMSGRECVR\s+(?&lt;netviewMsgrecvr&gt;[\s\S\-]{0,249})" />`

Bug Number	Issue	Resolution

		`<Line id="netviewOpclass" required="true" enabled="yes" sig="^4011([A-Za-z0-9]{0,3})(\S)?\s+([\s\S\-]{18})NETVOPCLOPCLASS\s+(?&lt;netviewOpclass&gt;[\s\S\-]{0,249})" />`
		`<Line id="netviewDomains" required="true" enabled="yes" sig="^4011([A-Za-z0-9]{0,3})(\S)?\s+([\s\S\-]{18})NETVDMNSDOMAINS\s+(?&lt;netviewDomains&gt;[\s\S\-]{0,249})" />`
		`<Line id="netviewNgmfadmn" required="true" enabled="yes" sig="^4011([A-Za-z0-9]{0,3})(\S)?\s+([\s\S\-]{18})NETVNGMFNGMFADMN\s+(?&lt;netviewNgmfadmn&gt;[\s\S\-]{0,249})" />`

Software Updates in Release 9.1.0.3.0

The following are software updates in release 9.1.0.3.0:

Custom Reconciliation Exit

You can customize the default reconciliation exit (TSSINSTX) to meet any special requirements in your environment. The connector installation package includes several sample files that enable you to write and call your own logic for the reconciliation exit.

See Customizing the Reconciliation Exit (TSSINSTX) for more information about working with custom reconciliation exit routines.

Software Updates in Release 9.1.0.2.0

The following are software updates in release 9.1.0.2.0:

- Transformation of LDAP Gateway Attributes
- Running Multiple Instances of the LDAP Gateway on the Same Host

- Support for Filtering

Transformation of LDAP Gateway Attributes

By including transformation rules within the *LDAP_INSTALL_DIR*/conf/customer-configuration.properties file, you can configure the LDAP gateway to transform the gateway attributes in search results.

See Configuring Transformation of the LDAP Gateway Attributes for more information on the transformation rules to include and its format.

Running Multiple Instances of the LDAP Gateway on the Same Host

From this release onward, you can run multiple instances of the LDAP Gateway on the same host.

See Configuring Multiple Instances of the LDAP Gateway for more information on configuring and running multiple gateway instances in your environment.

Support for Filtering

The "Top Secret Reconcile All Users" and "Top Secret Reconcile LDAP Users to OIM" scheduled tasks have been updated to include a filter attribute. You can use this attribute to retrieve user records that match a given filter criteria. See Top Secret Reconcile All Users and Top Secret Reconcile LDAP Users to OIM for more information about the filter attribute.

Software Updates in Release 9.1.0.1.0

The following are software updates in release 9.1.0.1.0:

Addition of a New Parameter in the Voyager Control File

The FAST_SHUTDOWN_NUM parameter has been introduced in the Voyager control file.

During peak loads there can be thousands of events written to sub pool and Voyager processes those events sequentially. You can use the FAST_SHUTDOWN_NUM parameter to implement the batching feature, which processes the events in batches. You specify the number of events in a 'batch' in the FAST_SHUTDOWN_NUM parameter. Voyager checks for any operator command after processing each batch instead of checking it after processing all events at once. This helps in fast shutdown capability for Voyager during maintenance cycles.

See Configuring the Reconciliation Agent for more information about the FAST_SHUTDOWN_NUM parameter.

Software Updates in Release 9.1.0.0.0

The following are software updates in release 9.1.0.0.0:

- Support for New Oracle Identity Governance Release
- Support for New Target System Version
- Detailed Audit Logs
- Support for High Availability and Disaster Recovery in the LDAP Gateway
- Support for ADMIN and DEADMIN Keywords

- Support for New Diagnostic Tool
- Support for the MOVE Function
- Enhancement to the Scheduled Tasks for Lookup Field Synchronization
- Support for Passphrases
- Enhancement to the IT Resource Definition

Support for New Oracle Identity Governance Release

From this release onward, the connector can be installed and used on Oracle Identity Governance release12.2.1.3.0. Be sure to download and apply the 28682376 and 29133050 mandatory patches from My Oracle Support.

Support for New Target System Version

From this release onward, the you can install and use the connector with CA Top Secret R15 or R16 running on IBM z/OS version 2.2 or 2.3.

Detailed Audit Logs

From this release onward, the connector provides a LOGGERX module that you can configure for detailed debug level log information on the Pioneer and Voyager agents. This detailed logging provides additional auditing and monitoring capabilities for your target system. In addition, you can choose to print or suppress log messages.

See Configuring Logging for more information.

Support for High Availability and Disaster Recovery in the LDAP Gateway

From this release onward, the LDAP gateway supports high availability and disaster recovery when you use OpenDS as the backend.

Support for ADMIN and DEADMIN Keywords

From this release onward, the connector provides support for provisioning and reconciliation of all Admin multivalued attributes.

See Adding Admin Multivalued Attributes for Provisioning and Reconciliation for more information.

Support for New Diagnostic Tool

From this release onward, a new diagnostic tool for TSS Agents, ENVINFO, is available for use as described in Understanding and Using the ENVINFO Diagnostic Tool.

Support for the MOVE Function

The provisioning and reconciliation ability of Oracle Identity Manager has been enhanced to achieve the expected functionality of the MOVE keyword. For example, for provisioning operations:

- Attributes DEPTACID, DIVACID, and ZONEACID will be used for MOVE with Type operation.
- Attributes DEPARTMENT, DIVISION, and ZONE will be used for MOVE without Type operation.

Enhancement to the Scheduled Tasks for Lookup Field Synchronization

The "Top Secret Find All Groups" and "Top Secret Find All Profiles" scheduled tasks for lookup field synchronization have been enhanced to include the following three new parameters:

- SearchBaseDN
- AttrsToReturn
- DescTemplate

See Scheduled Tasks for Lookup Field Synchronization for descriptions of these parameters.

Support for Passphrases

From this release onward, the connector provides support for passphrase security, in addition to password.

A new field for passphrase has been added to the OIM User process form, which lets you provision an account by using passphrases. In addition, you can modify the passphrase of an account along with other fields on the process form.

For reconciliation, the phraseExpire and phraseExpireInterval attributes have been added to the "Top Secret Reconcile All Users" and "Top Secret Reconcile All LDAP Users" scheduled tasks.

Enhancement to the IT Resource Definition

The IT resource definition has been enhanced to include a new parameter named "auditTemplate" for passing audit statements. If you do not specify any value for this parameter, then the connector will not post audit comments for any process task that is initiated from Oracle Identity Manager.

See Configuring the Connector for Audit Comments for information about setting up the connector for displaying audit information.

Documentation-Specific Updates

These are the updates made to the connector documentation.

- Documentation-Specific Updates in Release 9.1.0.9.0
- Documentation-Specific Updates in Release 9.1.0.8.0
- Documentation-Specific Updates in Release 9.1.0.7.0
- Documentation-Specific Updates in Releases 9.1.0.4.0 through 9.1.0.6.0
- Documentation-Specific Updates in Release 9.1.0.3.0
- Documentation-Specific Updates in Release 9.1.0.2.0
- Documentation-Specific Updates in Release 9.1.0.1.0
- Documentation-Specific Updates in Release 9.1.0.0.0

Documentation-Specific Updates in Release 9.1.0.9.0

There are no documentation-specific updates in this release.

Documentation-Specific Updates in Release 9.1.0.8.0

A new parameter called IP has been added to Table 4-4.

A new parameter called IP has been added to Table 4-5.

The parameter "LDAP Time Zone" in Table 5-6 has been amended to use the Timezone database name value.

The attribute "PHRASEONLY" has been added to Table 1-4

Documentation-Specific Updates in Release 9.1.0.7.0

There are no documentation-specific updates in this release.

Documentation-Specific Updates in Releases 9.1.0.4.0 through 9.1.0.6.0

The following documentation-specific updates have been made in revision "05" of the guide:

- The idfConnectTimeoutMS" and "idfReadTimeoutMS" rows of Table 3-1 has been updated.
- Troubleshooting Information has been updated.
- The "Oracle Identity Governance or Oracle Identity Manager" row of Table 1-1 has been updated to include support for Oracle Identity Governance release 12c PS4 (12.2.1.4.0).

Documentation-Specific Updates in Release 9.1.0.3.0

The following documentation-specific update has been made in revision "04" of the guide:

Activating Reconciliation Exits has been created.

Documentation-Specific Updates in Release 9.1.0.2.0

The following documentation-specific update has been made in revision "03" of the guide:

Configuring Memory Pool Settings has been added.

The following documentation-specific updates have been made in revision "02" of the guide:

- Table 2-2 has been updated to include the agentMetaRecon and agentCachingRecon properties.
- The name of the file and its location for managing LDAP Gateway logging operations has been updated in Enabling Logging for the LDAP Gateway.
- The "topsecret-agent-recon.log" row as been removed from Table 3-2 as it is no longer available.
- The following topics have been updated to clarify the encryption requirement for the connector:
  - The "Infrastructure requirement for the message transport layer between Oracle Identity Manager and the mainframe environment" row of Table 1-1
  - Description of the Message Transport Layer component in About the Connector Components
  - Encrypted Communication Between the Target System and Oracle Identity Manager
  - The "Message Transport Layer" row of Installation Requirements for Agents
- CFILE Reconciliation Process has been updated.

- Table 3-1 and Table 5-3 have been updated to include the "Secondary IT resource" attribute.
- The "phraseExpire" and "phraseExpireInterval" attributes have been removed Table 5-3 as they are not present in the scheduled task.
- Table 5-4 has been updated to include the "UID Case" attribute.

Documentation-Specific Updates in Release 9.1.0.1.0

There are no documentation-specific updates in this release.

Documentation-Specific Updates in Release 9.1.0.0.0

This is the first release of the connector in this release track. Therefore, there are no documentation-specific updates in this release.

1 About the CA Top Secret Advanced Connector

The CA Top Secret Advanced connector integrates Oracle Identity Manager with a CA Top Secret target system. This connector lets you use CA Top Secret as a managed (target) resource of identity data for Oracle Identity Manager.

This chapter contains the following topics:

Introduction to the Connector

Oracle Identity Manager automates access rights management, security, and provisioning of IT resources. Oracle Identity Manager connectors are used to integrate Oracle Identity Manager with external, identity-aware applications. The advanced connector for CA Top Secret provides a native interface between CA Top Secret installed on an IBM z/OS mainframe and Oracle Identity Manager. The connector functions as a trusted virtual administrator on the target system, performing tasks related to creating and managing users. In the account management (target resource) mode of the connector, information about users (ACIDs) created or modified directly on the target system can be reconciled into Oracle Identity Manager. In addition, you can use Oracle Identity Manager to perform provisioning operations on the target system.

Certified Components

These are the software components and their versions required for installing and using the connector.

Table 1-1 Certified Components

Item	Requirement
Oracle Identity Manager or Oracle Identity Governance	You can use one of the following releases of Oracle Identity Manager or Oracle Identity Governance: • Oracle Identity Governance 12*c* PS4 (12.2.1.4.0) • Oracle Identity Governance 12*c* (12.2.1.3.0) with the 28682376 and 29133050 mandatory patches installed. You can download the mandatory patches from My Oracle Support. • Oracle Identity Manager 11*g* release 2 PS3 (11.1.2.3.0)
JDK	The JDK version can be one of the following: • For Oracle Identity Governance release 12.2.1.3.0 or later, use JDK 1.8.0_131+ . • For Oracle Identity Manager release 11.1.2.*x* or later, use JDK 1.6 update 31 or later.
Target systems	CA Top Secret R15 or R16 running on IBM z/OS version 2.2, 2.3, 2.4, or 2.5
Infrastructure requirement for the message transport layer between Oracle Identity Manager and the mainframe environment	TCP/IP
Target system user account for reconciliation and provisioning operations	CA Top Secret authorized user account with System Administrators privileges During installation of the mainframe agents, the <HLQ>.JCLLIB.TOPSDEF job automatically creates user accounts (ACIDs) with System Administrators privileges.
Pioneer and Voyager	Pioneer and Voyager are written in single thread LE Cobol. They were developed to run above the 16M line. Options that can adversely affect these STCs are LE run options: ALL31(OFF) instead of ON STACK(,,,BELOW,,) instead of STACK(,,,ANYWHERE,,)
LDAP Gateway	The computer hosting the LDAP Gateway must run the following software: • Operating system: Microsoft Windows Server 2012, or Red Hat Enterprise Linux 7 (64-bit) • Oracle Java JRE 1.8 or 1.7

Certified Languages

The connector supports the following languages:

- Arabic
- Chinese (Simplified)
- Chinese (Traditional)
- Danish
- English
- French
- German

- Italian
- Japanese
- Korean
- Portuguese (Brazilian)
- Spanish

Connector Architecture

Connectors require certain architecture consisting of Gateways and Provisioning Agent.

This section contains the following topics:

- About the Connector Components
- Connector Operations

About the Connector Components

The CA Top Secret Advanced connector contains the following components:

- **LDAP Gateway**: The LDAP Gateway receives instructions from Oracle Identity Manager in the same way as any LDAP version 3 identity store. These LDAP commands are then converted into native commands for CA Top Secret and sent to the Provisioning Agent. The response, which is also native to CA Top Secret, is parsed into an LDAP-format response and returned to Oracle Identity Manager.

  During reconciliation, the LDAP Gateway receives event notification, converts the events to LDAP format, and then forwards them to Oracle Identity Manager, or events can be stored in the LDAP Gateway internal store and pulled into Oracle Identity Manager by a scheduled task.

- **Provisioning Agent (Pioneer)**: The Pioneer Provisioning Agent is a mainframe component. It receives native mainframe CA Top Secret identity and authorization change events from the LDAP Gateway. These events are processed against the CA Top Secret authentication repository, in which all provisioning updates from the LDAP Gateway are stored. The response is parsed and returned to the LDAP Gateway.

  > **Note:**
  >
  > At some places in this guide, the Provisioning Agent is referred to as **Pioneer.**

- **Reconciliation Agent (Voyager):** The Reconciliation Agent captures mainframe events by using a Top Secret exit, which is a program run after events in CA Top Secret are processed. These events include the ones generated at TSO logins, the command prompt, batch jobs, and other native events. These events are stored in the subpool cache area that is established by a supplied, standard z/OS procedure (STARTUP). The Reconciliation Agent captures these events, transforms them into LDAPv3 protocol notification messages, and then sends them to Oracle Identity Manager through the LDAP Gateway.

> **Note:**
>
> At some places in this guide, the Reconciliation Agent is referred to as **Voyager.**

- **Message Transport Layer**: This connector supports a message transport layer by using the TCP/IP protocol, which is functionally similar to proprietary message transport layer protocols. In addition, the connector provides AES encryption for messages sent and received through the transport layer.

  The AES encryption is performed using 128-bit cryptographic keys. In addition, Encryption and Decryption programs are supplied in the Distribution Load Library. The encryption or decryption does not require any network software or hardware.

Connector Operations

This section provides an overview of the following processes involving the CA Top Secret Connector.

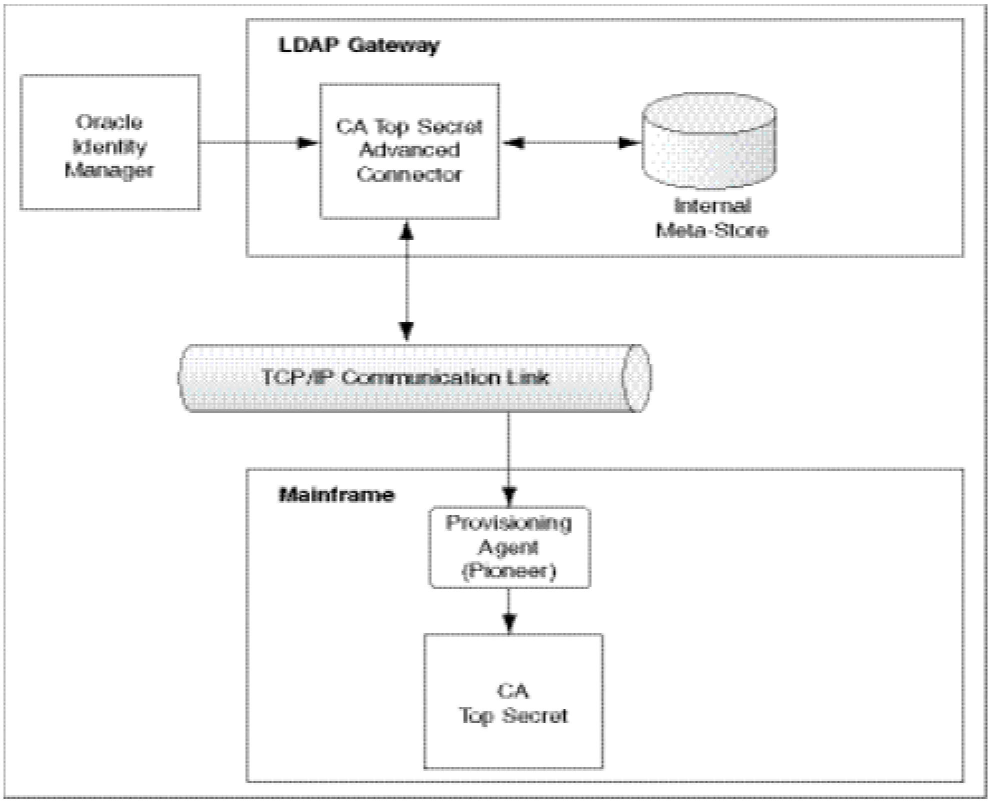
This section contains these topics:

Full Reconciliation Process

Full reconciliation involves fetching existing user data from the mainframe to Oracle Identity Manager. This user data is converted into accounts or resources for OIM Users.

Figure 1-1 shows the flow of data during full reconciliation.

Figure 1-1 Full Reconciliation Process for CA Top Secret Connector


LDAP Gateway
Oracle Identity Manager
CA Top Secret Advanced Connector
Internal Meta-Store
TCP/IP Communication Link
Mainframe
Provisioning Agent (Pioneer)
CA Top Secret

The following is a summary of the full reconciliation process:

> **Note:**
>
> The detailed procedure is explained later in this guide.

1. Set values for the attributes of the Reconcile All Users scheduled task and run it. The task sends a search request to the LDAP Gateway.
2. The LDAP Gateway encrypts the search request and then sends it to the Provisioning Agent on the mainframe.
3. The Provisioning Agent encrypts user profile data received from CA Top Secret and then passes this data to the LDAP Gateway.
4. The LDAP Gateway decrypts the user profile data. If the user profile data does not include any changes when compared to the OIM user's existing resource data, then the event is ignored and reconciliation continues with the next user on the target system. If the user profile data includes a change, then the LDAP Gateway passes the data on to Oracle Identity Manager.
5. This user profile data is converted into accounts or resources for OIM Users.

CFILE Reconciliation Process

CFILE reconciliation involves fetching existing user data in the form of a TSSCFILE extract from the mainframe to Oracle Identity Manager. This user data is converted into accounts or resources for OIM Users.

Instead of reconciling directly from the target system to OIM (which can be slow on large systems), the LDAP gateway offers an internal LDAP store that can be populated with target system users by using a single transaction to the mainframe. Oracle Identity Manager then reconciles user data from the LDAP store instead of the target system.

Reconciling user, profile, and facility data from an extract file requires the following procedure:

> **Note:**
>
> The detailed procedure is explained later in this guide.

1. Generate an extract file of the user data by executing the TSSCFILE job on the CA Top Secret system.

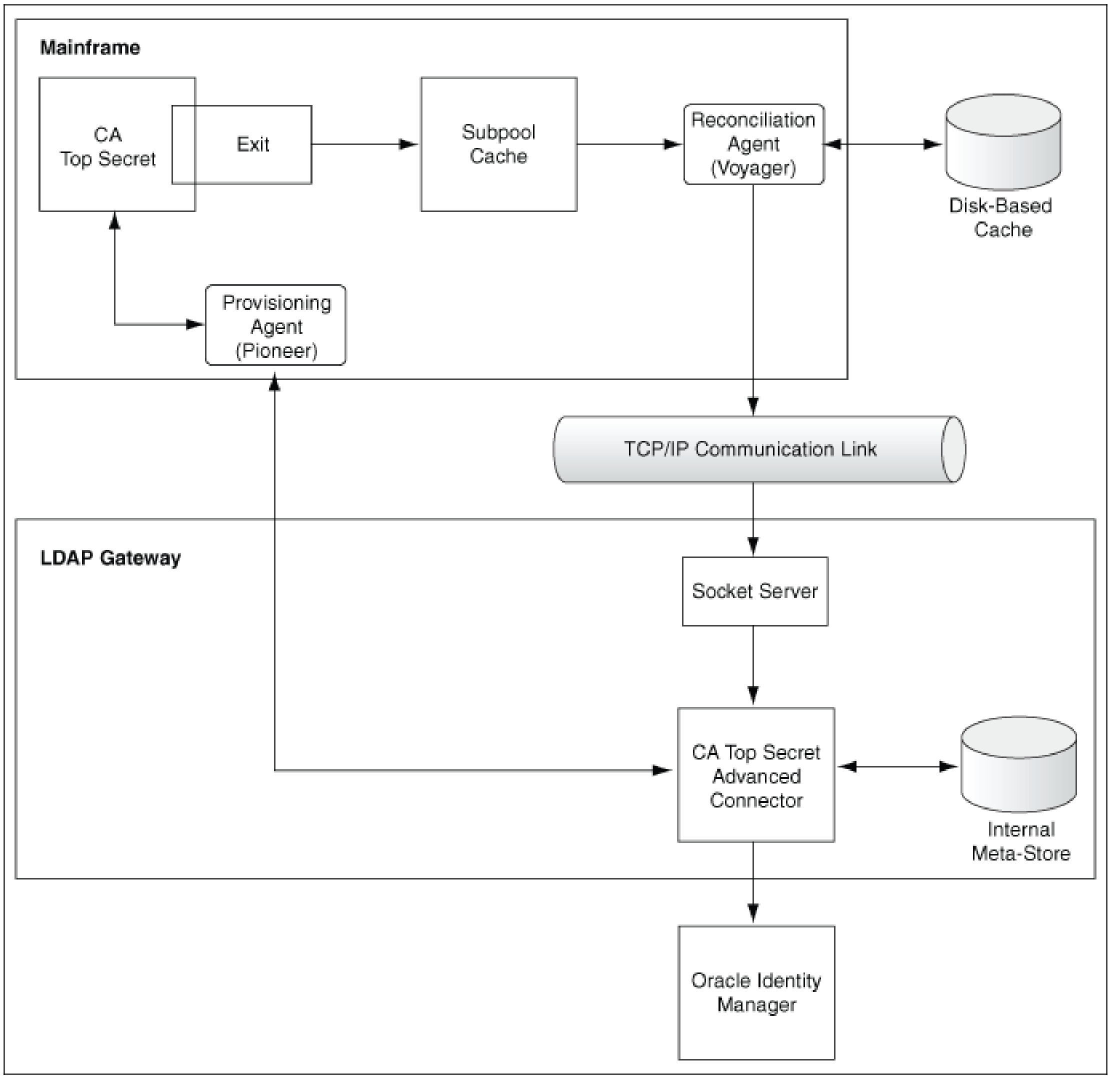
   Note that the TSSCFILE job is supplied as part of <hlq>.JCLLIB in the connector installation package.
2. Set values for the attributes of the Top Secret Reconcile Users to Internal LDAP scheduled task and run it. The task sends a request to the LDAP gateway to retrieve the extract file from Pioneer.
3. The Provisioning Agent receives the request from LDAP Gateway and reads the data from the extract dataset.
4. The Provisioning Agent encrypts the user data and passes it to the LDAP Gateway.
5. The LDAP Gateway decrypts the user profile data. The data is stored in the LDAP Gateway's internal data-store.
6. Set values for the attributes defined in the TSS Reconcile LDAP Users to OIM scheduled task.
7. Run the scheduled task. The next step depends on the setting in the IT resource as mentioned below:
   1. If you set the "Last Modified Time Stamp" property to `0`, then all the user profile data is retrieved from the LDAP internal store.
   2. If you configure the "Last Modified Time Stamp" property with a timestamp, then only the user profile data updated since the timestamp is retrieved from the LDAP internal store.
8. The next step depends on the user data as mentioned below:
   1. If the user profile data does not include any changes when compared to the OIM user's existing resource data, then the event is ignored and reconciliation continues with the next retrieved user.

b. If the user profile data includes a change, then the LDAP Gateway passes the data on to Oracle Identity Manager. The user profile data is converted into accounts or resources for OIM Users.

Incremental (Real-Time) Reconciliation Process

Incremental or real-time reconciliation is initiated by the exit that works in conjunction with the Reconciliation Agent. Figure 1-2 shows the flow of data during this form of reconciliation.

Figure 1-2 Reconciliation Process for CA Top Secret Connector



Performing (Real-Time) Reconciliation

If you want to perform a (Real Time) reconciliation the following is a summary of this process:

1. Incremental reconciliation begins when a user is created or, updated on CA Top Secret. This event might take place either directly on the mainframe or in response to a provisioning operation on Oracle Identity Manager.

2. TSSINSTX is a standard CA Top Secret exit. This exit is used in conjunction with the Reconciliation Agent. The exit detects the event and sends a message containing user data to Subpool 231 (cache).
3. The Reconciliation Agent polls Subpool 231. When it finds the message in the subpool, it reads the message into its buffer. This frees up the subpool.
4. The Reconciliation Agent opens up a connection with the LDAP Gateway, and then sends the message to the gateway over TCP/IP.

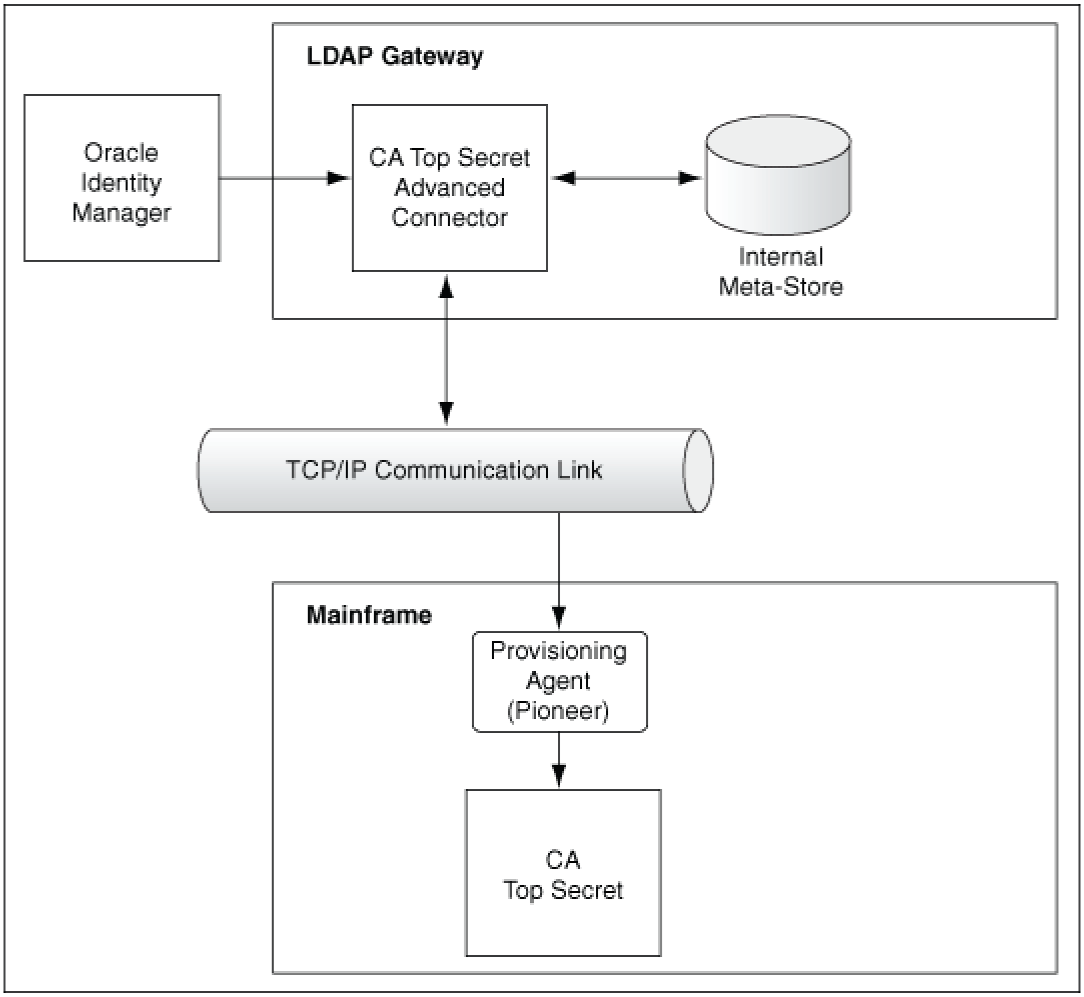
> **Note:**
>
> Messages sent to the LDAP Gateway are encrypted using AES-128 encryption.

5. The LDAP Gateway decrypts the user profile data. If the user profile data does not include any changes when compared to the OIM user's existing resource data, then the event is ignored and reconciliation continues with the next user on the target system. If the user profile data includes a change, then the LDAP Gateway can store the data internally for use by a scheduled task, or it can pass the data on to Oracle Identity Manager.
6. Oracle Identity Manager processes the message and creates or updates either the corresponding CA Top Secret resource or the OIM User.

Provisioning Process

Figure 1-3 shows the flow of data during provisioning.

Figure 1-3 Provisioning Process


LDAP Gateway
Oracle Identity Manager
CA Top Secret Advanced Connector
Internal Meta-Store
TCP/IP Communication Link
Mainframe
Provisioning Agent (Pioneer)
CA Top Secret

The following is a summary of the provisioning process:

1. Provisioning data submitted from Oracle Identity Self Service is sent to the LDAP Gateway.
2. The LDAP Gateway converts the provisioning data into mainframe commands, encrypts the commands, and converts the message from ASCII to EBCDIC.
3. The Provisioning Agent executes the commands and runs them on the mainframe and within the Pioneer STC (Started Task) using the RACF API (IRRSEQ00).
4. The Provisioning Agent converts the RACF API output to ASCII and encrypts the message prior to sending it back to the LDAP Gateway.
5. The outcome of the operation on the mainframe is displayed on Identity Self Service. A more detailed message is recorded in the connector log file.

Connector Features

The features of the connector include support for full and incremental reconciliation, encrypted communication, and high availability.

The following are the features of the connector:

- Support for Target Resource Reconciliation
- Full and Incremental Reconciliation
- Encrypted Communication Between the Target System and Oracle Identity Manager
- High Availability Feature of the Connector

Support for Target Resource Reconciliation

You can configure the connector as a target resource of Oracle Identity Manager.

Full and Incremental Reconciliation

After you deploy the connector, you can perform full reconciliation to bring all existing user data from the target system to Oracle Identity Manager. After the first full reconciliation run, change-based or incremental reconciliation is automatically enabled and active. Incremental reconciliation is a real-time process.

User changes on the target system are directly sent to Oracle Identity Manager or stored in the LDAP Gateway internal store.

You can perform a full reconciliation run at any time. See Performing Full Reconciliation for more information.

Encrypted Communication Between the Target System and Oracle Identity Manager

AES-128 encryption is used to encrypt data that is exchanged between the LDAP Gateway, and the Reconciliation and Provisioning Agents on the Mainframe. This encryption is taken care by the Mainframe agents.

High Availability Feature of the Connector

The following are component-failure scenarios and the response of the connector to each scenario.

- **Scenario 1: The Reconciliation Agent is running and the LDAP Gateway stops responding**
  1. The Reconciliation Agent stops sending messages (event data) to the LDAP Gateway.
  2. Messages that are not sent are stored in the subpool cache.
  3. When the LDAP Gateway is brought back online, the Reconciliation Agent reads data from the subpool cache and then sends messages to the LDAP Gateway.
- **Scenario 2: The LDAP Gateway is running and the Reconciliation Agent stops responding**
  1. Event data is sent to the subpool cache.

2. When the Reconciliation Agent is brought back online, it reads data from the subpool cache and then sends messages to the LDAP Gateway.

> **Note:**
>
> During SHUTDOWN, there is a possibility that events that had been sent to the LDAP might be saved and re-sent again once the Agent is brought back online. This is to ensure no data lose and this process will re-list the event data to provide the most current view.

- **Scenario 3: The LDAP Gateway is running and the mainframe stops responding**
  1. Messages that are in the subpool cache are written to disk.
  2. When the mainframe is brought back online, event data written to disk is again stored in the subpool cache.
  3. The Reconciliation Agent reads data from the subpool cache and then sends messages to the LDAP Gateway.

> **Note:**
>
> During SHUTDOWN, there is a possibility that events that had been sent to the LDAP might be saved and re-sent again once the Agent is brought back online. This is to ensure no data lose and this process will re-list the event data to provide the most current view.

- **Scenario 4: The LDAP Gateway is running and the Provisioning Agent or mainframe stops responding**

  The process task that sends provisioning data to the LDAP Gateway retries the task.
- **Scenario 5: The subpool is stopped by an administrator**

  If the subpool is stopped by an administrator, then it shuts down the Reconciliation Agent, thereby destroying any messages that are not transmitted. However, the messages in the AES-encrypted file are not affected and can be recovered.

Connector Objects Used During Reconciliation and Provisioning

These are the connector objects that used during reconciliation and provisioning operations.

- Supported Functions for Target Resource Reconciliation
- Supported Functions for Provisioning
- User Attributes for Target Resource Reconciliation and Provisioning
- PROFILE Attributes for Target Resource Reconciliation and Provisioning
- GROUP Attributes for Target Resource Reconciliation and Provisioning
- SOURCE Attributes for Provisioning
- FACILITY Attributes for Target Resource Reconciliation and Provisioning
- DATASET Attributes for Provisioning

- Provisioning GENCERT Operations
- Provisioning GENREQ Operations
- Reconciliation Rules
- Viewing the Reconciliation Rule
- Reconciliation Action Rules
- Viewing the Reconciliation Action Rules

Supported Functions for Target Resource Reconciliation

The connector supports reconciliation of user data from these events.

- Create user
- Modify user
- Password Interval changes
- Change password
- Reset password
- Disable user
- Delete user
- Enable user
- Group Membership Changes
- Create group data
- Modify group data
- Delete groups
- Audit information

Supported Functions for Provisioning

These are the provisioning functions that the connector supports.

Table 1-2   Supported Functions for Provisioning

Function	Description
Create user	Adds new users in CA Top Secret
Modify user	Modifies user information in CA Top Secret
Change password	Changes user passwords in CA Top Secret in response to password changes made in Oracle Identity Manager through user self-service
Reset password	Resets user passwords in CA Top Secret The passwords are reset by the administrator.
Disable user	Disables users in CA Top Secret
Enable user	Enables users in CA Top Secret
Delete users	Removes users from CA Top Secret
Authenticate users	Validates user's LoginId and Password

Table 1-2 (Cont.) Supported Functions for Provisioning

Function	Description
Create profiles	Adds new profiles to CA Top Secret
Modify profiles	Modifies profiles information in CA Top Secret
Delete profiles	Removes profiles from CA-Top Secret
Define resources	Define new resources to CA Top Secret
Alter resources	Modify resources in CA Top Secret
Delete resources	Remove resources from CA Top Secret
Search All Users	Retrieves all users with current data from CA-Top Secret
Search All Profiles	Retrieves all profiles with current data from CA-Top Secret
Search All Departments	Retrieves all departments with current data from CA-Top Secret
Search All Resources	Retrieves all resources with current data from CA-Top Secret
Search All Datasets	Retrieves all datasets with current data from CA-Top Secret
Grant users access to data sets and general resources	Permits users access to a CA Top Secret dataset or resource
Grant users access to privileges (TSO)	Provides TSO login access to users or other privileges
Assign profile membership	Add or remove user from profiles in CA Top Secret
Grant user TSO attributes	Provides TSO information
Grant User NETVIEW attributes	Provides NETVIEW information
Grant User CICS attributes	Provides CICS information
Grant User Custom Segment attributes	Provides custom segment support for user-defined information
Grant User OMVS attributes	Provides OMVS information

User Attributes for Target Resource Reconciliation and Provisioning

The CA Top Secret connector uses three categories of attributes: mapped, unmapped, and custom.

Mapped and unmapped attributes are supported in the LDAP Gateway, but unmapped attributes are not shipped with preconfigured OIM metadata such as form fields, process tasks, or reconciliation mappings.

Custom attributes require additional configuration steps in the LDAP Gateway. See Adding Custom Fields for Target Resource Reconciliation through Adding Custom Fields for Provisioning for more information.

Table 1-3 lists the major differences between attribute types.

Table 1-3 Attribute Characteristics of CA Top Secret Connector

Attribute Type	Out-of-the-box OIM Metadata Support	Out-of-the-box LDAP Support	Additional LDAP Configuration Required
Mapped	Yes	Yes	No
Unmapped	No	Yes	No

Table 1-3   (Cont.) Attribute Characteristics of CA Top Secret Connector

Attribute Type	Out-of-the-box OIM Metadata Support	Out-of-the-box LDAP Support	Additional LDAP Configuration Required
Custom	No	No	Yes

Table 1-4 lists mapped attribute mappings between CA Top Secret and Oracle Identity Manager. The OnBoardUser and ModifyTopsUser adapters are used for Create User and Modify User provisioning operations, respectively.

Table 1-4   Mapped User Attributes for Target Resource Reconciliation and Provisioning of CA Top Secret Connector

Process Form Field	CA Top Secret Attribute Display Name	Description
USER_ID	USER	Login ID of the user
FULL_NAME	NAME	Full name of the user
Password	PASSWORD	Password
department	DEPARTMENT	Default department of the user **Note:** Provisioning is done using "department" attribute but reconciliation brings department's full name in "department" attribute and the acid value is brought in DEPTACID.
deptacid	DEPARTMENT	Default department of the user **Note:** Provisioning is done using "department" attribute but reconciliation brings department's full name in "department" attribute and the acid value is brought in DEPTACID.
instdata	DATA	Installation-defined data of the user
createdate	CREATED	Date user was created
passwordExpire	EXPIRES	Expire the user's password
passwordExpireInterval	INTERVAL	Number of days the user's password remains valid
suspendUntilDate	SUSPENDED DATE	Future date on which the user will be prevented from accessing the system
divacid	DIVISION	Default division for the user **Note:** Provisioning is done using "division" attribute but reconciliation brings division's full name in "division" attribute and the acid value is brought in "divacid."
division	DIVISION	Default division for the user **Note:** Provisioning is done using "division" attribute but reconciliation brings division's full name in "division" attribute and the acid value is brought in "divacid."
lastmodificationdate	LAST MOD	Last time the user connected
tsocommand	COMMAND	Command to be run during TSO/E logon
tsodest	DEST	Default SYSOUT destination

Table 1-4 (Cont.) Mapped User Attributes for Target Resource Reconciliation and Provisioning of CA Top Secret Connector

Process Form Field	CA Top Secret Attribute Display Name	Description
tsounit	UNIT	Default unit name for allocations
tsoudata	USERDATA	Site-defined data field for a TSO user
tsolacct	ACCTNUM	Default TSO account number on the TSO/E logon panel
tsohclass	HOLDCLASS	Default hold class
tsojclass	JOBCLASS	Default job class
tsomsize	MAXSIZE	Maximum region size the user can request at logon
tsomclass	MSGCLASS	Default message class
tsolproc	PROC	Default logon procedure on the TSO/E logon panel
tsolsize	SIZE	Minimum region size if not requested at logon
tsoopt	OPT	TSO options, such as MAIL and NOTICES
tsosclass	SYSOUTCLASS	Default SYSOUT class
zone	ZONE	Display name of default zone for the user
zoneAcid	ZONE ACID	Default zone for the user
PASSPHRASE	PASSPHRASE	Password phrase
PHRASEONLY	PHRASEONLY	Valid values are `true` or `false` The LDAP gateway and Mainframe TOPS Pioneer agent will allow provisioning and reconcile of the PHRASEONLY attribute
PHRASE_EXPIRE	PHRASEEXPIRE	Number of days before a password phrase expires
PHRASE_EXPIRE_INTERVAL	PHRASE EXPIRE INTERVAL	Passphrase expiration interval

The Top Secret connector supports provisioning and reconciliation of additional attributes that are not included on the main process form or preconfigured with process tasks and reconciliation mappings.

Table 1-5 lists unmapped the attribute mappings between CA Top Secret and Oracle Identity Manager. The adpModifyTopsUser adapter is used for Modify User provisioning operations, respectively.

Table 1-5 Unmapped User Attributes for Target Resource Reconciliation and Provisioning of CA Top Secret Connector

LDAP Gateway Name	CA Top Secret Attribute	Description	Supported Operations
lu62#appl	#APPL	LU 6.2 #APPL	Both
lu62#entity	#ENTITY	LU 6.2 #ENTITY	Both
lu62bc1chain	BC1CHAIN	LU 6.2 BC1CHAIN	Both
lu62bc2chain	BC2CHAIN	LU 6.2 BC2CHAIN	Both
lu62set1disp	SET1DISP	LU 6.2 SET1DISP	Both

Table 1-5 (Cont.) Unmapped User Attributes for Target Resource Reconciliation and Provisioning of CA Top Secret Connector

LDAP Gateway Name	CA Top Secret Attribute	Description	Supported Operations
lu62set2disp	SET2DISP	LU 6.2 SET2DISP	Both
waaccnt	WAACCNT	APPC SYSOUT ACCT NUMBER	Both
waaddr1	WAADDR1	APPC SYSOUT ADDRESS 1	Both
waaddr2	WAADDR2	APPC SYSOUT ADDRESS 2	Both
waaddr3	WAADDR3	APPC SYSOUT ADDRESS 3	Both
waaddr4	WAADDR4	APPC SYSOUT ADDRESS 4	Both
wabldg	WABLDG	APPC SYSOUT BUILDING	Both
wadept	WADEPT	APPC SYSOUT DEPARTMENT	Both
waname	WANAME	APPC SYSOUT NAME	Both
waroom	WAROOM	APPC SYSOUT ROOM	Both
tsodefprfg	TSODEFPRFG	DEFAULT PERFORMANCE GROUP	Both
tsompw	TSOMPW	MULTIPLE PASSWORDS	Both **NOTE:** In reconciliation, the attribute is stored as "attributes" with value of "TSOMPW".
tsoacct	TSOACCT	SECURE TSO LOGON ACCOUNT CODES	Provisioning Only
tsoauth	TSOAUTH	SECURE TSO USER ATTRIBUTES	Provisioning Only
tsoprfg	TSOPRFG	SECURE TSO PERFORMANCE GROUPS	Provisioning Only
tsoproc	TSOPROC	SECURE TSOP LOGON PROCS	Provisioning Only
defaultGroup	DFLTGRP	OMVS DEFAULT GROUP	Both
omvsProgram	OMVSPGM	OMVS PROGRAM	Both
omvsUid	UID	OMVS USER ID	Both
omvsHome	HOME	OMVS HOME SUBDIRECTORY	Both
omvsGid	GID	OMVS GROUP ID	Both
omvsAssize	ASSIZE	OMVS MAX ADDRESS SPACE SIZE	Both

Table 1-5 (Cont.) Unmapped User Attributes for Target Resource Reconciliation and Provisioning of CA Top Secret Connector

LDAP Gateway Name	CA Top Secret Attribute	Description	Supported Operations
omvsMmaparea	MMAPAREA	OMVS MAX DATASPACE PAGES	Both
omvsOecputm	OECPUTM	OMVS MAX CPU TIME	Both
omvsoeflep	OEFILEP	OMVS MAX FILES PER PROCESS	Reconciliation Only
omvsProcuser	PROCUSER	OMVS MAX PROCESSES	Both
omvsThreads	THREADS	OMVS MAX PTHREADS CREATED	Both
netviewMsgrecvr	MSGRECVR	NETVIEW RECEIVE UNSOLICITED MESSAGES	Both
netviewInitcmd	IC	NETVIEW INITIAL COMMAND	Both
netviewControl	CTL	NETVIEW SECURITY CHECK TYPE	Both
netviewOpclass	OPCLASS	NETVIEW SCOPE CLASS	Both
netviewDomains	DOMAINS	NETVIEW CROSS-DOMAIN SESSIONS	Both
netviewNgmfadmn	NGMFADMN	NETVIEW GRAPHICAL DISPLAY ADMIN	Both
netviewConsName	CONSNAME	NETVIEW EXTENDED CONSOLE NAME	Both
cicsOpclass	OPCLASS	CICS OPERATOR CLASSES	Both
cicsOpident	OPIDENT	CICS OPERATOR IDENTIFICATION VALUE	Both
cicsOpprty	OPPRTY	CICS OPERATOR PRIORITY	Both
cicsSctykey	SCTYKEY	CICS SECURITY KEYS	Both
cicsSitran	SITRAN	CICS TRANSACTION FOLLOWING FACILITY SIGN-IN	Both **Note:** To provision cicsSitran, you must map the process task to the adpModifySitranTopsUser adapter instead of adpModifyTopsUs.

Table 1-5 (Cont.) Unmapped User Attributes for Target Resource Reconciliation and Provisioning of CA Top Secret Connector

LDAP Gateway Name	CA Top Secret Attribute	Description	Supported Operations
cicsSitranFacility	SITRAN FACILITY	CICS FACILITY ASSOCIATED WITH TRANSACTION	Both **Note:** To provision cicsSitranFacility, you must map the process task to the adpModifySitranTopsUser adapter instead of adpModifyTopsUser.
misc1	MISC1	ADMIN MISC	Reconciliation Only
misc2	MISC2	ADMIN MISC	Reconciliation Only
misc3	MISC3	ADMIN MISC	Reconciliation Only
misc4	MISC4	ADMIN MISC	Reconciliation Only
misc5	MISC5	ADMIN MISC	Reconciliation Only
misc7	MISC7	ADMIN MISC	Reconciliation Only
misc8	MISC8	ADMIN MISC	Reconciliation Only
misc9	MISC9	ADMIN MISC	Reconciliation Only

PROFILE Attributes for Target Resource Reconciliation and Provisioning

The connector supports reconciliation and provisioning of the PROFILE multivalued attribute. For any particular user, a child form is used to hold values of the PROFILE attributes listed in the table.

The AddUserToProfile and RemoveUserFromProfile adapters are used for PROFILE provisioning operations. Table 1-6 lists PROFILE attribute mappings between CA Top Secret and Oracle Identity Manager.

Table 1-6 PROFILE Attribute Mappings for CA Top Secret Connector

Child Form Field	CA Top Secret Attribute	Description
UD_TSSPROF_ID	PROFILE	Profile ID

GROUP Attributes for Target Resource Reconciliation and Provisioning

The connector supports reconciliation and provisioning of the GROUP multivalued attribute. For any particular user, a child form is used to hold values of the GROUP attributes listed in the table.

The AddUserToGroup and RemoveUserFromGroup adapters are used for GROUP provisioning operations.

Table 1-7 lists GROUP attribute mappings between CA Top Secret and Oracle Identity Manager.

Table 1-7 GROUP Attribute Mappings for CA Top Secret Connector

Child Form Field	CA Top Secret Attribute	Description
UD_TSSGROUP_ID	GROUP	Group ID

SOURCE Attributes for Provisioning

The connector supports provisioning of the SOURCE multivalued attribute. For any particular user, a child form is used to hold values of the SOURCE attributes listed in the table.

The AddUserToSource and RemoveUserFromSource adapters are used for SOURCE provisioning operations. Table 1-8 lists SOURCE attribute mappings between CA Top Secret and Oracle Identity Manager.

Table 1-8 SOURCE Attribute Mappings for CA Top Secret Connector

Child Form Field	CA Top Secret Attribute	Description
UD_TSSSOURC_ID	SOURCE	Source ID

FACILITY Attributes for Target Resource Reconciliation and Provisioning

The connector supports reconciliation and provisioning of the FACILITY multivalued attribute. For any particular user, a child form is used to hold values of the FACILITY attributes listed in the table.

The AddUserToFacility and RemoveUserFromFacility adapters are used for FACILITY provisioning operations. Table 1-9 lists FACILITY attribute mappings between CA Top Secret and Oracle Identity Manager.

Table 1-9 FACILITY Attribute Mappings for CA Top Secret Connector

Child Form Field	CA Top Secret Attribute	Description
UD_TSSFAC_ID	FACILITY	Facility ID

DATASET Attributes for Provisioning

The connector supports provisioning of the DATASET multivalued attribute. For any particular user, a child form is used to hold values of the DATASET attributes listed in the table.

The AddUserToDataset and RemoveUserFromDataset adapters are used for DATASET provisioning operations. Table 1-10 lists DATASET attribute mappings between CA Top Secret and Oracle Identity Manager.

Table 1-10   DATASET Attribute Mappings for CA Top Secret Connector

Child Form Field	CA Top Secret Attribute	Description
DATASET_ID	DATASET	Dataset ID
DATASET_ACCESS	ACCESS	Users level of access to the dataset

Provisioning GENCERT Operations

The connector supports provisioning operations for the TSS GENCERT command, however a pre-configured child form, process task, and adapter are not included in with the release. To provision GENCERT actions, the OIM administrator will need to create an adapter and map it to the GenerateCertificate function in the topsecret-provisioning-adapter.jar file.

The following is the function header for GenerateCertificate:

```
public String generateCertificate(String idfUserId, String digicert, String
dcdsn, String keysize, String keyusage,String nbdate, String nbtime, String
nadate, String natime, String lablcert, String altname, String subjects, String
signwith, String icsf, String dsa, String pcicc)
```

For boolean attributes such as ICSF or DSA, the administrator should map these values as literal String values equal to either true or false.

Provisioning GENREQ Operations

The connector supports provisioning operations for the TSS GENREQ command, however a preconfigured child form, process task, and adapter are not included in with the release. To provision GENREQ actions, the OIM administrator will need to create an adapter and map it to the GenerateCertificateRequest function in the topsecret-provisioning-adapter.jar file.

The following is the function header for GenerateCertificateRequest:

```
public String generateCertificateRequest(String idfUserId, String digicert,
String dcdsn, String lablcert)
```

Reconciliation Rules

During target resource reconciliation, Oracle Identity Manager tries to match each user fetched from CA Top Secret with existing CA Top Secret resources provisioned to OIM Users. This is known as process matching. A reconciliation rule is applied for process matching.

If a process match is found, then changes made to the user on the target system are copied to the resource on Oracle Identity Manager. If no match is found, then Oracle Identity Manager tries to match the user against existing OIM Users. This is known as entity matching. The reconciliation rule is applied during this process. If an entity match is found, then a CA Top Secret resource is provisioned to the OIM User. Data for the newly provisioned resource is copied from the user.

Rule name: IdfReconUserRule

Rule element: User Login Equals uid

In this rule element:

- User Login is the User ID field on the process form and the OIM User form.
- uid is the USER attribute on CA Top Secret.

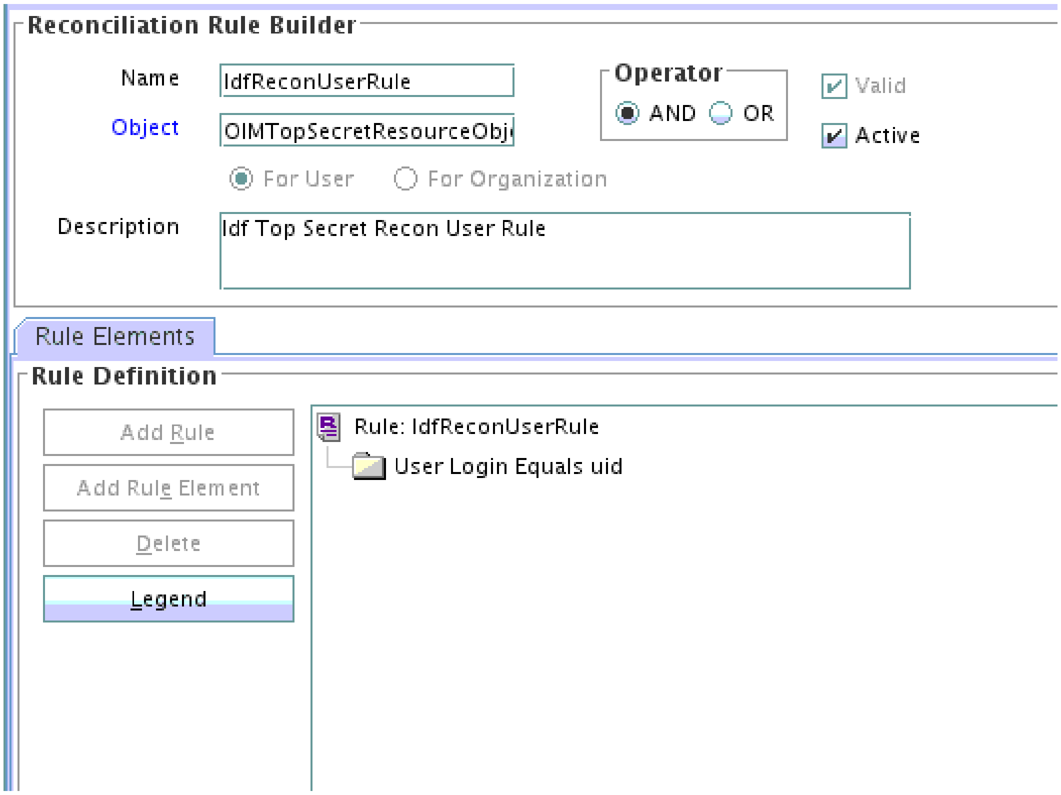
Viewing the Reconciliation Rule

You can view the reconciliation rule for this connector from Development Tools in Oracle Identity Manager Design Console.

After you deploy the connector, you can view this reconciliation rule by performing the following steps:

1. On the Design Console, expand **Development Tools** and then double-click **Reconciliation Rules**.
2. Search for and open the **IdfReconUserRule** rule. Figure 1-4 shows this rule.

Figure 1-4 Reconciliation Rule



Reconciliation Rule Builder

Name: IdfReconUserRule

Object: OIMTopSecretResourceObj

Operator: ● AND ○ OR

☑ Valid

☑ Active

● For User ○ For Organization

Description: Idf Top Secret Recon User Rule

Rule Elements

Rule Definition

Rule: IdfReconUserRule
- User Login Equals uid

Reconciliation Action Rules

Reconciliation action rules specify actions that must be taken depending on whether or not matching CA Top Secret resources or OIM Users are found when the reconciliation rule is applied.

Table 1-11 lists the reconciliation action rules.

Table 1-11 Reconciliation Action Rules for CA Top Secret Connector

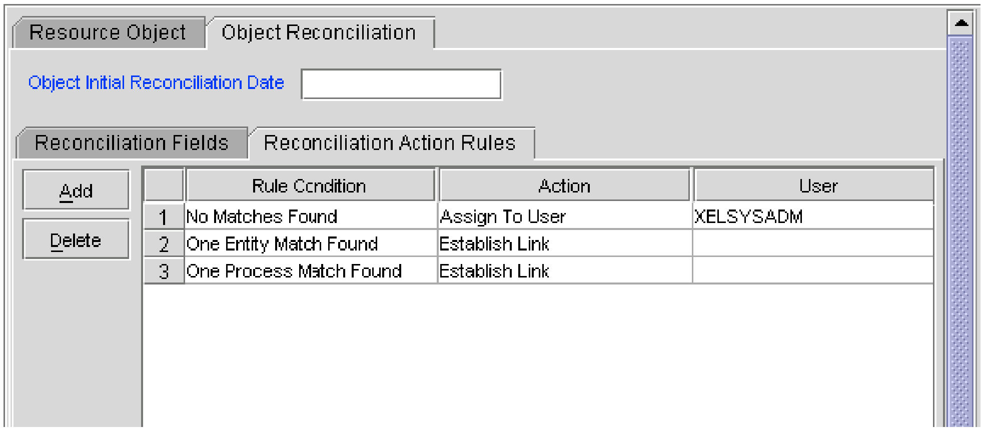
Rule Condition	Action
No Matches Found	None
One Entity Match Found	Establish Link
One Process Match Found	Establish Link

Viewing the Reconciliation Action Rules

After you deploy the connector, you can view the reconciliation action rules for target resource reconciliation by performing these steps.

1. On the Design Console, expand **Resource Management** and then double-click **Resource Objects**.
2. Search for and open the **OIMTopSecretResourceObject** resource object.
3. Click the **Object Reconciliation** tab, and then click the **Reconciliation Action Rules** tab. The Reconciliation Action Rules tab displays the action rules defined for this connector. Figure 1-5 shows the reconciliation action rule for target resource reconciliation.

Figure 1-5 Reconciliation Action Rules for CA Top Secret Connector



	Rule Condition	Action	User
1	No Matches Found	Assign To User	XELSYSADM
2	One Entity Match Found	Establish Link	
3	One Process Match Found	Establish Link	

2

Installing and Configuring the LDAP Gateway

The LDAP Gateway acts as the intermediary between Oracle Identity Manager and the connector components on the mainframe. You can install the LDAP Gateway either on a Microsoft Windows or RHEL Linux platform.

- Hardware Requirements for Installing the LDAP Gateway
- Installing the LDAP Gateway
- Upgrading the LDAP Gateway
- Configuring the LDAP Gateway
- Configuring the Windows Service for the LDAP Gateway
- Configuring Transformation of the LDAP Gateway Attributes
- Configuring Multiple Instances of the LDAP Gateway
- Encrypting Data
- Understanding the Caching Layer
- Configuring Scheduled Reconciliation
- About Parsing Grammar Protocol 1.0

Hardware Requirements for Installing the LDAP Gateway

These are the recommended hardware requirements that are designed to give you optimal system performance from the LDAP gateway.

Table 2-1 Hardware Requirements for Installing the LDAP Gateway

Requirement Type	Processor	RAM	Hard Disk	Network Interface
Minimum hardware requirement	2 GHz single-core processor	4 GB RAM	10GB hard disk drive	1
Recommended hardware requirement	2 GHz multicore processor	16 GB RAM	50GB hard disk drive	1

Installing the LDAP Gateway

You can install the LDAP Gateway on Windows and Linux platforms.

See Hardware Requirements for Installing the LDAP Gateway and the "LDAP Gateway" row of Certified Components to ensure that the computer on which you want to install the LDAP Gateway meets the recommended specifications.

To install the LDAP Gateway:

1. Download and save the connector installation package to any directory on the computer that will host the LDAP Gateway. You can download the connector installation package from the OTN website at http://www.oracle.com/technetwork/middleware/id-mgmt/downloads/connectors-101674.html
2. Extract the contents of the connector installation package to any directory on the computer. This creates a directory named *CONNECTOR_NAME-RELEASE_NUMBER*.
3. Extract the contents of the etc/LDAP Gateway/IDF_LDAP_GATEWAY_vX.X.X.zip file from the connector installation package to a temporary directory on the computer hosting the LDAP gateway.
4. Depending on the operating system computer on which you want to install the LDAP Gateway, run one of the following files:
   - Microsoft Windows: IDFLDAPGateway-X-windows-oracle-vX.X.X.exe
   - Linux: IDFLDAPGateway-X-linux-x64-oracle-vX.X.X.run
5. On the Setup - LDAP Gateway screen, click **Next** to proceed with installation.
6. On the License Agreement screen, select **I accept the agreement** if you agree with the terms of the agreement, and then click **Next**.
7. On the Installation Location screen, specify the location where the LDAP Gateway must be installed.
   - For Linux:
     When you install the gateway as a normal user, the default location is inside the Home folder `(home/ubuntu/IDFLDAPGateway-X)`.

     When you install the gateway as a sudo or root user, the default location is `/opt/IDFLDAPGateway-X`.
   - For Microsoft Windows, the default location is Program files `(…\ProgramFiles (x86)\IDFLDAPGateway-X)`
8. Click **Next** to proceed.
9. On the License File screen, browse to the location containing the license.lic file, select it and then click **Next**. For the license.lic file please contact the Oracle team.

   The Ready to Install window is displayed.
10. Click **Next** to proceed.

    The Installing screen with a progress indicator bar for the installation is displayed.
11. On the Completing the LDAP Gateway Setup Wizard screen, select **View Readme File** if you want to read the enhancements made to the gateway. Click **Finish** to complete the installation process.

Upgrading the LDAP Gateway

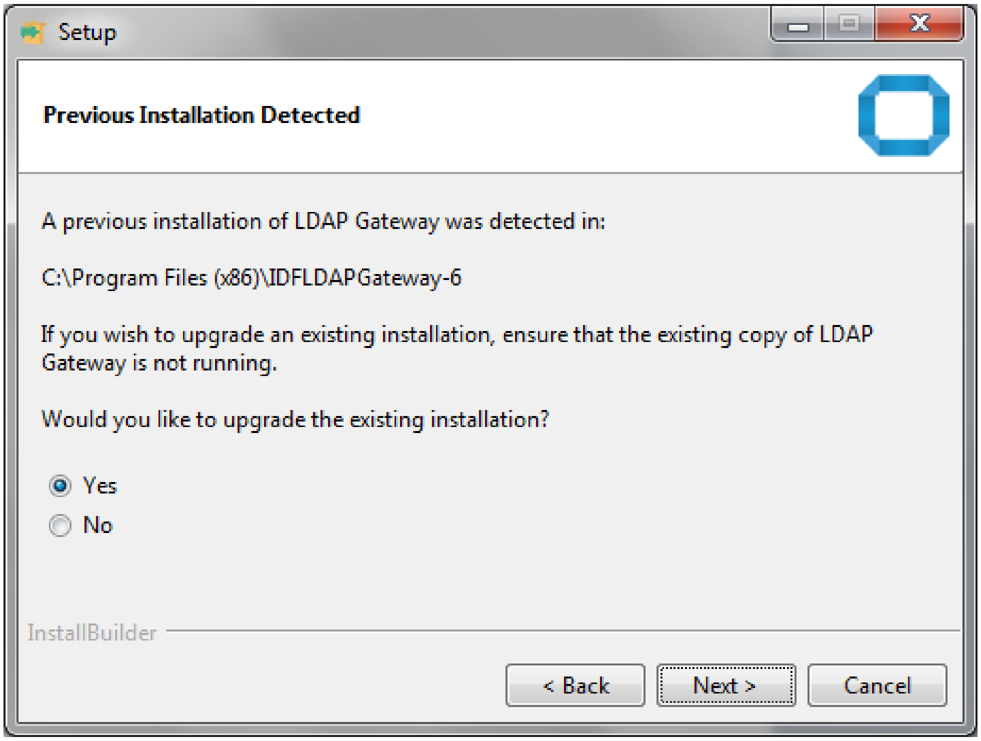
If you already have an earlier version of the LDAP Gateway (for example, version 5.*x*), then you can upgrade it to the latest version 6.x by running the LDAP gateway installer.

> **Note:**
>
> Before you begin the upgrade procedure:
>
> - On the computer hosting the gateway, stop the running instance of the gateway. If you are using a Microsoft Windows Service to run the gateway, then uninstall the Windows service.
> - In the target system environment, shut down any agents (for example, Pioneer or Voyager) that may be running.
> - Disable any cron jobs.

To upgrade the LDAP Gateway, do the following:

1. Download and save the connector installation package to any directory on the computer that will host the LDAP Gateway. You can download the connector installation package from the OTN website at http://www.oracle.com/technetwork/middleware/id-mgmt/downloads/connectors-101674.html.
2. Extract the contents of the connector installation package to any directory on the computer. This creates a directory named *CONNECTOR_NAME-RELEASE_NUMBER*.
3. Extract the contents of the etc/LDAP Gateway/IDF_LDAP_GATEWAY_v6.4.0.zip file from the connector installation package to a temporary directory on the computer hosting the LDAP gateway.
4. Depending on the operating system of the computer on which the LDAP gateway is installed, run one of the following files:
   - For Linux: IDFLDAPGateway-6-linux-x64-v6.4.0.run
   - For Microsoft Windows: IDFLDAPGateway-6-windows-v6.4.0.exe
5. On the Setup - LDAP Gateway screen, click **Next** to proceed with upgrade.
6. On the License Agreement screen, select **I accept the agreement** if you agree with the terms of the agreement, and then click **Next**.

   The installer detects the earlier installation of the gateway as shown in the following image:



A previous installation of LDAP Gateway was detected in:

C:\Program Files (x86)\IDFLDAPGateway-6

If you wish to upgrade an existing installation, ensure that the existing copy of LDAP Gateway is not running.

Would you like to upgrade the existing installation?

● Yes
○ No

7. On the Previous Installation Detected screen, when you are prompted whether you want to upgrade the existing installation, select one of the following options:
   - select **Yes** if you want to upgrade, and click **Next** to proceed. Then, on the Ready to Install screen, click **Next** to proceed with the upgrade.
   - Select **No** if you want to perform a fresh installation, and then click **Next** to proceed.

   > **Note:**
   >
   > To upgrade from version 5.x to 6.x, you need to provide the location of the existing installation folder location and the path of the valid license file. If the installation folder location is same, then the installer detects and creates a backup of the entire folder of the previous version with a suffix pre- and a timestamp. This can be verified at the installation location. The backup of the entire folder happens only once when you are upgrading from version 5.x to version 6.x. For example, if you already have a Gateway version 5.3 installed on your system, and you want to install Gateway version 6, then a backup folder for the files of 5.3 is created at the installation location.

   The Ready to Install window is displayed.

8. If you selected **No** on the Previous Installation Detected screen, then on the Installation Directory screen, specify the location where the gateway must be installed.
   a. For Linux:

When you install the gateway as a normal user, the default location is inside the Home folder `(home/ubuntu/IDFLDAPGateway-6)`.

When you install the gateway as a sudo or root user, the default location is `/opt/IDFLDAPGateway-6`.

b. For Microsoft Windows, the default location is Program files `(…\ProgramFiles (x86)\IDFLDAPGateway-6)`

> **Note:**
>
> If the installation directory points to a location containing an existing gateway, that gateway is automatically upgraded during the installation process.

9. Click **Next**. In the Upgrade Previous Install dialog box, click **Yes** to confirm that you want to upgrade your existing installation of the gateway.
10. In the Ready to Install screen, click **Next** to proceed with the upgrade.

    The Installing screen with a progress indicator bar for the installation is displayed.
11. On the Completing the LDAP Gateway Setup Wizard screen, select **View Readme File** if you want to read the enhancements made to the gateway. Click **Finish** to complete the upgrade process.

Configuring the LDAP Gateway

Configure the LDAP gateway to connector to the target system and access the data.

The following topics describe the procedure to configure the LDAP Gateway:

> **Note:**
>
> The following procedures are for a fresh installation only. If you already have a running setup or if you want to upgrade, then you do not have to perform these procedures.

- Setting Connection Properties
- Creating the Connector Configuration
- Configuring the LDAP Gateway for Multiple Installations of the Target System
- Overriding the Default System Configuration

Setting Connection Properties

The *`LDAP_INSTALL_DIR`*`/conf` directory contains the tops.properties.example file that contains sample entries and is used as the basis for configuring the gateway.

The tops.properties.example file is only a sample file. Therefore, create a separate properties file (for example, tops.properties) in the *`LDAP_INSTALL_DIR`*`/conf` location by creating a copy of the tops.properties.example file. Use this properties file to specify connection information that the gateway uses to connect to your target system. To do so:

1. In the *LDAP_INSTALL_DIR*/conf directory, create a copy of the *LDAP_INSTALL_DIR*/conf/tops.properties.example file and rename it to for example, tops.properties.

   > **Note:**
   >
   > If you are configuring the gateway for multiple instances of the target system, then you must create a copy of the *LDAP_INSTALL_DIR*/conf/tops.properties.example file and rename it for each target system instance. Ensure that the names of the renamed files are not the same.

2. In a text editor, open the tops.properties file for editing and set values for properties such as host, port, user credentials and so on to point to your environment.

   The following table describes these properties.

Table 2-2 Properties in the tops.properties File

Property	Description
agentPort	Enter the port number on the LDAP gateway host computer that you are going to reserve for messages sent from the mainframe by the Reconciliation Agent, Voyager. The LDAP gateway will receive real-time reconciliation messages using this port. This value should match the value of the PORT parameter in the Voyager agent control file.
agentMetaRecon	This property specifies whether Voyager must reconcile user data from the target system into the legacy meta store that is located in the LDAP gateway. The reconciled data is stored in the OU=tops subtree of the OU=People tree that is located in the system backend (DC=System,DC=Backend). Enter true to reconcile user data into the legacy meta store. Otherwise, enter false. The default value of this property is true. **Note:** If you upgraded the LDAP gateway from release 5.x to 6.x, then this property is not available in the tops.properties file by default. If you want to use this property, then you must add it to the tops.properties file manually.

Table 2-2 (Cont.) Properties in the tops.properties File

<table>
<tr><th>Property</th><th>Description</th></tr>
<tr><td>agentCachingRecon</td><td>This property specifies whether Voyager must reconcile user data from the target system into the caching store that is located in the LDAP gateway. The reconciled data is stored in the <code>OU=people</code> subtree of the <code>OU=tops1</code> tree that is located in the system backend (<code>DC=System,DC=Backend</code>).
Enter <code>true</code> to reconcile user data into the caching store. Otherwise, enter <code>false</code>.
The default value of this property is <code>true</code>.
<b>Note:</b><ul><li>If you upgraded the LDAP gateway from release 5.x to 6.x, then this property is not available in the tops.properties file by default. If you want to use this property, then you must add it to the tops.properties file manually.</li><li>If you set the value of this property to <code>true</code>, then ensure that the caching layer is enabled. If the caching layer is not enabled, then data is reconciled into the legacy meta store instead of the caching store. See Understanding the Caching Layer for information on how to enable the caching layer.</li></ul></td></tr>
<tr><td>configDNames</td><td>This property holds the name of any custom target system attributes that should be included when the LDAP gateway parses a user profile from the target system (typically performed during reconciliation). If you are using a target system attribute that is not supported out-of-the-box, then add the name of that attribute to the value of the configDNames property. The name should match the format of the attribute name when executing a LIST command on the target system and you must include the Spaces and = for each attribute. This step is mentioned in the following sections:<ul><li>Adding Custom Fields for Target Resource Reconciliation</li><li>Adding Custom Fields for Provisioning</li></ul>For example, if you defined two Top Secret fields named PST15, and VEND, then you would enter:<pre># CONFIG DISPLAY NAMES
configDNames =VEND =|PST15 =|</pre>To enter multiple custom attributes, separate each entry with a vertical bar.</td></tr>
</table>

Table 2-2 (Cont.) Properties in the tops.properties File

Property	Description

configAttrs	This property holds the name of any custom target system attributes that should be included when the LDAP gateway parses a user profile from the target system (typically performed during reconciliation). If you are using a target system attribute that is not supported out-of-the-box, then add the name of that attribute to the value of the configAttrs property. The name should match the format of the attribute name when executing a LIST command on the target system but without the data from above in the configDNames and this is that will match your LDAP and OIM attribute name when configuring. This step is mentioned in the following sections:
	• Adding Custom Fields for Target Resource Reconciliation
	• Adding Custom Fields for Provisioning
	For example, if you define three Top Secret fields named $PST15, PST VRO 16, and VEND ID, then you would enter:
	`# CONFIG ATTRIBUTES`
	`configAttrs=$PST15|VEND|`
	To enter multiple custom attributes, separate each entry with a vertical bar.

configDatasets	If you create a custom dataset on the target system, then add the name of that dataset type to the value of the configDatasets property.
	For example:
	`# CONFIG DATASETS`
	`configDatasets=$RAFT`
	To enter multiple custom dataset names, separate each entry with a vertical bar.

customDataset	This property is used to store custom dataset names when issuing a WHOHAS command to the Top Secret system. If more than one custom dataset exists, separate each entry with a vertical bar ('|') character.
	Custom datasets that are added to this property will be included when the dataset lookup synchronization task ('Top Secret Find All Datasets') is run in Oracle Identity Manager.
	For example:
	`# CUSTOM DATASETS FOR WHOHAS COMMAND`
	`_customDataset_$XRAFT\|$RAFT|`

Table 2-2 (Cont.) Properties in the tops.properties File

Property	Description
host	Enter the host name or IP address of the computer that must connect to Pioneer. For example, `_host_=localhost`.
port	Enter the number of the port on the Mainframe that you are going to reserve for Pioneer. The LDAP gateway will send provisioning messages to this port. This value should match the PORT parameter specified in the Pioneer provisioning agent STC. For example, `_port_=5790`.
stcID	This property is not supported from 9.0.4.18 and later releases of this connector. This property allows the real-time agent to ignore events that have been submitted to the target system by the Pioneer STC (such as by request from Oracle Identity Manager). Enter the name given to the Pioneer STARTED TASK.
domainOu	This property stores users in the specified subtree under the ou=People tree of the internal LDAP store. This entry needs to be unique and specific for each system if multiple systems are used within one LDAP gateway. Default setting is `domainOu=tops`
internalEnt	This property allows the real-time agent to store user data in the LDAP gateway internal store. Values: `[true|false]`

Table 2-2 (Cont.) Properties in the tops.properties File

Property	Description

ignoreChar	Use this property to specify characters in PROFILE names that must be ignored when retrieving user data from the LDAP gateway. For example:
	Suppose User1 is a member of the TESTGRP1 profile until 01-Jan-2020. When a LIST function is called for User1, the output for the PROFILES section is `*TESTGRP1`.
	If you set the value of the ignoreChar property to *, then the LDAP gateway ignores the asterisk character in the name of the profile. In other words, when the LIST function is called, the output for the PROFILES section is `TESTGRP1`.
	You can set multiple characters to be ignored as the value of the ignoreChar property. Do not use any blank character or space as the delimiter for the set of characters that you specify. For example, if you want both the asterisk character and the dollar sign ($) to be ignored, then enter the value as shown:
	`ignoreChar=*$`
	Suppose User2 is a member of the TESTGRP1, *TESTGRP2, and *TESTGRP$ profiles. If a LIST function is run for User2, then the following profiles are listed:
	`TESTGRP1, TESTGRP2, TESTGRP`

processFailedXML	This property is used by the Top Secret Reconcile Users to Internal LDAP scheduled task and determines whether the LDAP gateway will attempt to parse any failed XML entries. Default value: `true`
isStreamingUsers	This property is used by the Top Secret Reconcile Users to Internal LDAP scheduled task. If you set the value of this property to `true`, the LDAP gateway will process the CFILE data from the mainframe. If you set the value of this property to `false`, the LDAP gateway will not process any CFILE data. Default value: `true`

Table 2-2 (Cont.) Properties in the tops.properties File

Property	Description

revokePsuspendUsers	Use this property to specify whether users with the PSUSPEND attribute should be flagged as revoked when parsing a LIST USER result message.
	• Set `true` as the value if you want the user to be disabled in Oracle Identity Manager as the outcome of a LIST USER reconciliation operation.
	• Set `false` as the value if you want the PSUSPEND attribute to not factor into the user's Oracle Identity Manager Status setting as the outcome of a LIST USER reconciliation operation.
	For example:
	`# REVOKE OIM USERS WITH PSUSPEND`
	`revokePsuspendUsers=true`

secretKeyValue	Enter the secret key that the LDAP gateway must use to connect to the Mainframe.
includeData	This property is used when retrieving a list of all users on the Top Secret system. If you set the value of this property to true, for each ACID in TSS, the LDAP gateway will return both the ACID and the user data. If you set the value of this property to false, for each ACID in TSS, the LDAP gateway will only the ACID. Default value: `false`
resumeOnReset	This property is used when resetting a user's password. If you set the value of this property to true, the user will be enabled during a reset password operation. If you set the value of this property to false, the user will not be enabled during a reset password operation. Default value: `true`
trimOmvsUid	This property is used with the omvsUid attribute. If you set the value of this property to true, the LDAP gateway will trim leading zeros, "0", from the omvsUid value. If you set the value of this property to false, the LDAP gateway will not trim any leading zeroes from the omvsUid value. Default value: `true`

Table 2-2 (Cont.) Properties in the tops.properties File

Property	Description
trimNum	This property is used with the trimOmvsUid property and specifies the number of leading zeroes to trim from a user's omvsUid attribute. Default value: `2`
newOmvsUidAttr	This property specifies the new name to use for the omvsUid property. Default value: `OmvsUidEmplNumber`
usePwdComplexLength	This property is used to control the length of passwords. If you set the value of this property to true, the LDAP gateway will use the properties file password length settings. If you set the value of this property to false, the LDAP gateway will use the standard password length. Default value: `true`
idMinLength	This property specifies the minimum ACID length in characters. Default value: `1`
idMaxLength	This property specifies the maximum ACID length in characters. Default value: `8`
pwdMinLength	This property specifies the minimum password length for an ACID. Default value: `1`
pwdMaxLength	This property specifies the maximum password length for an ACID. Default value: `8`
minDays	This password specifies the minimum number of days that must pass before a password can be changed. Default value: `0`
mainframeCodePage	This property specifies the mainframe code page in use on the mainframe. Default value: `CP857`
luMulti	This property is used with LU6.2 attributes. If you set the value of this property to true, the LDAP gateway will process LU6.2 attributes as multivalued attributes. If you set the value of this property to false, the LDAP gateway will process LU6.2 attributes as single-valued attributes. Default value: `true`

Table 2-2 (Cont.) Properties in the tops.properties File

Property	Description
luMultiSep	This property is used with LU6.2 attributes and specifies the separator character used for multivalued attributes. Default value: `|`
userTypes	This property is used to specify all the types of users that must be retrieved and added to the ou=people container when a full-blown search is performed. You must list all the user types separated by a vertical bar (|). For example, USER|DCA|VCA|SCA|LSCA|ZCA|. Default value: `USER|`
sslEnabled	Set the value of this property to `true` if the agent supports SSL messaging. Otherwise, specify `false`. Default value: `false`
tlsVersion	This property is used to specify the TLS version that is enabled for the agent. You can set the value of this property to `TLSv1.1` or `TLSv1.2` and can be extended to `TLSv1.3` in future if required. Default value: `TLSv1.2`
type isencrypted timeout authretries requestorId CPF CPF-WAIT _adminAttrs_ _adminDNames_	These properties are no longer used in Oracle installations. Do not modify their values.

3. If you want to include custom segment as a part of the TSS LIST command set, then set a value for the `_configDatasets_` property.

   Use the following components to set a value for the `_configDatasets_` property:

   - Use fn to represent the first name.
   - Use sp to represent the space character.
   - Use ln to represent the last name.
   - Use a comma (,) to represent the comma.
   - Use a period (.) to represent the period.
   - Use the vertical bar (|) as the separator for the other components.
4. Save and close the file.

Creating the Connector Configuration

To allow the gateway to work with the target system, you must create and configure the customer-configuration.properties file for the type of connector and its related parameters for the operations.

> **Note:**
>
> In this guide, *LDAP_INSTALL_DIR* is the standard term used to refer to the directory in which the gateway has been installed. For example, for a Microsoft Windows host machine, the default installation directory for the gateway is . .\Program Files (x86)\IDFLDAPGateway-6\.

The `LDAP_INSTALL_DIR/conf` directory has the customer-configuration.properties.example file that contains sample configuration entries and is used as the basis for creating the connector configuration. As the customer-configuration.properties.example file is only a sample file, you must create a separate properties file (for example, customer-configuration.properties) in the `LDAP_INSTALL_DIR/conf` directory to include and configure entries specific to your connector from the customer-configuration.properties.example file.

1. Create an empty customer-configuration.properties text file in the `LDAP_INSTALL_DIR/conf` directory.

   > **Note:**
   >
   > If you have upgraded the gateway, then skip this step as the customer-configuration.properties file already exists and contains all the connector configurations present in the beans.xml file.

2. In the `LDAP_INSTALL_DIR/conf` directory, locate and open the customer-configuration.properties.example file.

   The customer-configuration.properties.example file contains sample definitions (configuration properties) in various sections for each connector that the LDAP Gateway can be used with.

3. Search for and copy the following snippet from the customer-configuration.properties.example file, and paste it into the customer-configuration.properties file located in the `LDAP_INSTALL_DIR/conf` directory.

   ```
   cnctr.tops.class=com.identityforge.idfserver.backend.tops.TopsModule
   cnctr.tops.tops1.schema=schemas
   cnctr.tops.tops1.suffix=dc=tops,dc=com
   cnctr.tops.tops1.adminUserDN=cn=idfTopsAdmin,dc=tops,dc=com
   cnctr.tops.tops1.adminUserPassword=idfTopsPwd
   cnctr.tops.tops1.altAdminUserDN=cn=oimTopsAdmin,dc=tops,dc=com
   cnctr.tops.tops1.altAdminUserPassword=oimTopsPwd
   cnctr.tops.tops1.configLocation=../conf/tops.properties
   cnctr.tops.tops1.allowAnonymous=false
   cnctr.tops.tops1.metaBackend=ldapds
   ```

```
cnctr.tops.tops1.agent=true
cnctr.tops.tops1.customSchemaLocation=
cnctr.tops.tops1.people.multiCallAttributes=userpassword,attributes,uid,us
erpassword|userpassword,userpassword|passwordexpire|passwordexpiredays
# Simple equality filters using the following attributes will be passed
through to the target
cnctr.tops.tops1.cachingAllowedTargetFilterAttributes=uid,objectClass,alld
ata
```

4. In the customer-configuration.properties file, rename the connector qualifier for the newly pasted entries to match the name of the connection properties file that you created in Setting Connection Properties. Suppose you created a properties file named topsecret.properties, then rename all instances of `tops` in the newly pasted configuration entries to `topsecret`. For example, in the `cnctr.`**`tops`**`.tops1.suffix=dc=tops,dc=com` property, rename tops to `topsecret`. So the entry will now be `cnctr.`**`topsecret`**`.tops1.suffix=dc=tops,dc=com`
5. Similarly, rename the instance ID for all the configuration properties. For example, in the `cnctr.tops.tops1.schema=schemas` property, rename tops1 to `tops2`.
6. Edit the value of the `cnctr.tops.tops1.configLocation=` property to point to the connection properties file that you created in Setting Connection Properties. For example, if you created a file named topsecret.properties, then replace `cnctr.tops.tops1.configLocation= ../conf/tops.properties` with `cnctr.tops.tops1.configLocation= ../conf/`**`topsecret.properties`**
7. Change the default system administrator credentials that the gateways uses to connect to the target system as follows:
   a. Locate the following properties:

      ```
      cnctr.tops.tops1.adminUserDN=cn=idfTopsAdmin,dc=tops,dc=com
      cnctr.tops.tops1.adminUserPassword=idfTopsPwd
      ```

   b. Set new values for the adminUserDN and adminUserPassword properties and note them down. You must enter the same values for the idfPrincipalDn and idfPrincipalPwd parameters of the IT resource.

      > **Note:**
      >
      > - By default, all sensitive data is automatically encrypted when you start the gateway.
      > - For the adminUserDN property:
      >   – It is mandatory to that you use `cn` as the RDN identifier.
      >   – If you put spaces after the commas in the DN, then you must match that when using that ID to connect to the gateway. For example, if the required format is `cn=adminId,dc=tops,dc=com`, then `dc=tops,dc=com` must match the suffix property.

8. Save and close the file.
9. Restart the gateway for the changes to take effect.

Configuring the LDAP Gateway for Multiple Installations of the Target System

You can instantiate the same type of connector multiple times to represent multiple different endpoints of the same target system. This is in addition to the gateway supporting the ability to run connectors for various target systems within a single gateway instance.

If you have already configured a single instance of the connector for one target system installation and want to configure an additional instance, then:

1. For each target system installation in your environment, create a properties file in the *LDAP_INSTALL_DIR*/conf directory by creating a copy of the *LDAP_INSTALL_DIR*/conf/tops.properties file. Then, edit the newly created properties file to specify all connection properties.
2. Open the customer-configuration.properties.example file located in the *LDAP_INSTALL_DIR*/conf directory, copy the following configuration properties specific to your connector and paste it into the *LDAP_INSTALL_DIR*/conf/customer-configuration.properties file, below the existing set of configuration properties.

   ```
   cnctr.tops.class=com.identityforge.idfserver.backend.tops.TopsModule
   cnctr.tops.tops1.schema=schemas
   cnctr.tops.tops1.suffix=dc=tops,dc=com
   cnctr.tops.tops1.adminUserDN=cn=idfTopsAdmin,dc=tops,dc=com
   cnctr.tops.tops1.adminUserPassword=idfTopsPwd
   cnctr.tops.tops1.altAdminUserDN=cn=oimTopsAdmin,dc=tops,dc=com
   cnctr.tops.tops1.altAdminUserPassword=oimTopsPwd
   cnctr.tops.tops1.configLocation=../conf/tops.properties
   cnctr.tops.tops1.allowAnonymous=false
   cnctr.tops.tops1.metaBackend=ldapds
   cnctr.tops.tops1.agent=true
   cnctr.tops.tops1.customSchemaLocation=
   cnctr.tops.tops1.people.multiCallAttributes=userpassword,attributes,
   uid,userpassword|userpassword,userpassword|passwordexpire|
   passwordexpiredays
   # Simple equality filters using the following attributes will be
   passed through to the target
   cnctr.tops.tops1.cachingAllowedTargetFilterAttributes=uid,objectClas
   s,alldata
   ```

   Close the customer-configuration.properties.example file.
3. In the *LDAP_INSTALL_DIR*/conf/customer-configuration.properties file, rename the instance ID for all the newly pasted configuration properties. For example, in the cnctr.tops.tops1.schema=schemas property, replace tops1 with tops2.

> **Note:**
>
> Ensure that the connector name qualifier in the configuration properties matches the one that you specified while performing the procedure described in Creating the Connector Configuration. For example, in the `cnctr.tops.tops1.suffix=dc=tops,dc=com` configuration property, if you renamed tops to `topsecret`, then you must do the same for all newly added configuration properties here.

4. Modify the following properties:
   - `adminUserPassword` - change the default value for security reasons.
   - `suffix` - Enter the unique baseDN that you want to use in OIM. The default value is `dc=tops,dc=com`. You can change the default value to a baseDN of your choice.
   - `adminUserDN` - Enter the full DN of an administrative user account that is allowed to use the connector for reconciliation and provisioning operations. Note that the DN suffix must match the value that you set for `suffix` property.
   - `altAdminUserDN` - Enter the full DN of the alternative administrative user account that is allowed to use the connector for reconciliation and provisioning operations. Note that the DN suffix must match the value that you set for `suffix` property.
   - `configLocation` - Enter the location of the property file (created in Step 1) for the instance of the target system. For example, . . conf/topsecret10.properties. If the intent is to point these two connectors to different target systems, then the configLocation property should point to a different connector properties file (created in Step 1) for each target system instance. The new properties file can be a copy of the original properties file with changes in the necessary properties to point to the new system.
5. Save and close the `customer-configuration.properties` file and then restart the gateway for the changes to take effect.

Overriding the Default System Configuration

You can override the default system configuration by modifying the `*LDAP_INSTALL_DIR*/conf/customer-configuration.properties` file.

To change the default system properties, locate that property in the `configuration.properties` file (located in the `conf/` folder) and copy it to `customer-configuration.properties` file and provide a new value.

> **Note:**
>
> Not all properties can be modified and must be done in consultation with Support.

By default, all system configurations are stored in the `*LDAP_INSTALL_DIR*/conf/configuration.properties` file. If required, you can override any of these system configurations by copying relevant properties from the `*LDAP_INSTALL_DIR*/conf/configuration.properties` file to the `*LDAP_INSTALL_DIR*/conf/customer-configuration.properties` file, and then providing a new value.

> **Note:**
>
> Do not edit *LDAP_INSTALL_DIR*/conf/configuration.properties file directly as it will be overwritten when you upgrade the gateway.

There can be several reasons when you want to override the default system configuration. For example, you may want to change the default passwords for the system backend persistence store or change the listening port when the default collides with another service or when the policies of the company require using a different port.

- To change the default system backend passwords, add the following properties to the *LDAP_INSTALL_DIR*/conf/customer-configuration.properties file:

  ```
  cnctr.proxy.ldapds.adminUserPassword=<admin-password>
  cnctr.proxy.ldapds.altAdminUserPassword=<alt-admin-password>
  ```

  In the preceding lines, replace ***<admin-password>*** with the password for accessing the system backend. Similarly, replace ***<alt-admin-password>*** with the alternative password for accessing the system backend (dc=system,dc=backend)

  Not all properties can be modified and must be done in consultation with Support.

- To change the default port, add the following properties to the *LDAP_INSTALL_DIR*/conf/customer-configuration.properties file:

  ```
  system.port=6389
  system.ssl_port=7389
  ```

  In the preceding lines, replace 6386 with the desired listening port for LDAP. Similarly, replace 7389 with the desired listening port for LDAPS.

Configuring the Windows Service for the LDAP Gateway

In a Windows environment, the LDAP Gateway can also be installed as a Windows Service. The Windows Service for the LDAP Gateway is installed using the IdentityForge batch file (IDF-Win-Service) that is included in the installation media.

- Installing and Configuring the Windows Service for the LDAP Gateway
- Uninstalling the Windows Service for the LDAP Gateway
- Configuring Memory Pool Settings

Installing and Configuring the Windows Service for the LDAP Gateway

You can install the Windows Service by running the `IDF-Win-Service install` command.

To install the Windows service, switch to the *LDAP_INSTALL_DIR*/win_service directory in a command window and then run the `IDF-Win-Service install` command. If you encounter any issues with the installation, then uncomment the `CG_PATH_TO_JVM` variable in the*LDAP_INSTALL_DIR*/win_service/IDF-Win-

Service.bat file and ensure that the path is accurate. The following is the code snippet from the *LDAP_INSTALL_DIR*/win_service/IDF-Win-Service.bat file that you need to uncomment:

```
rem -- 7. Set this if you want to use a different JVM than the one
configured in your registry, or if it is not configured in the windows
registry
rem set CG_PATH_TO_JVM=C:\Program Files\Java\jre7\bin\server\jvm.dll
```

If you need to modify the Windows service settings, then it is recommended to first uninstall the service, make the modifications, and then reinstall the service until it installs and runs correctly.

After installing the service, you can start, stop, or restart it anytime by using the Windows Services console. Alternatively, run the following command to start the service:

```
> net start IdentityForgeService
```

Run the following command to stop the service:

```
> net stop IdentityForgeService
```

Uninstalling the Windows Service for the LDAP Gateway

Uninstall the Windows service for the LDAP Gateway by running the IDF-Win-Service remove command.

To uninstall the Windows service, switch to the *LDAP_INSTALL_DIR*/win_service directory in a command window and then run the IDF-Win-Service remove command.

Configuring Memory Pool Settings

You can configure the memory pool size for the Windows service by setting values for the CG_JVMMS and CG_JVMMX variables in the *LDAP_INSTALL_DIR*/win_service/IDF-Win-Service.bat file.

By default, the CG_JVMMS and CG_JVMMX variables are set to 1024 MB and 2048 MB, respectively. If the LDAP gateway processes a large number of records, then you might encounter the "Out of memory" exception. In such a scenario, you can allocate higher memory for your Windows service by increasing the values of the CG_JVMMS and CG_JVMMX variables.

To do so:

1. Stop the LDAP gateway Windows service and then uninstall it.
2. In a text editor, open the *LDAP_INSTALL_DIR*/win_service/IDF-Win-Service.bat file for editing.
3. Set the JVM minimum and maximum values by modifying values for the following lines:

   ```
   rem Initial memory pool size in MB.
   set CG_JVMMS=1024
   ```

```
rem Maximum memory pool size in MB.
set CG_JVMMX=2048
```

> **Note:**
>
> When you receive the "Out of memory" exception, start with increasing the minimum and maximum values to 2048 and 4096, respectively. If the number of records is greater than 40k, then use higher minimum and maximum values.

4. In the `LDAP_INSTALL_DIR/conf/log4j.properties` file, set the gateway debug level to `ERROR` as follows:

   ```
   rootLogger.level = ERROR
   ```

5. Install the LDAP gateway Windows service.
6. Start the LDAP gateway through the Windows service.

Configuring Transformation of the LDAP Gateway Attributes

You can configure transformation of LDAP Gateway attributes in search results by adding relevant entries to the `LDAP_INSTALL_DIR/conf/customer-configuration.properties` file.

You must include the transformation rule within the `LDAP_INSTALL_DIR/conf/customer-configuration.properties` file as an inline Jtwig template. For more information about Jtwig templates, see http://jtwig.org/documentation.

For example, you can add a transformation rule to render the value of the `sn` attribute in the People OU in uppercase. To do so, you must add the following line in the `LDAP_INSTALL_DIR/conf/customer-configuration.properties` file:

```
cnctr.tops.tops1.transformation.People.read.sn.template.inline={{sn|
upper}}
```

This entry will render all the letters in the `sn` attribute in uppercase.

To configure transformation of LDAP gateway attributes:

1. In a text editor, open the customer-configuration.properties file located in the `LDAP_INSTALL_DIR/conf` directory.
2. Add the transformation rule in the following format:

   ```
   cnctr.CONNECTOR_QUALIFIER.INSTANCE_ID.transformation.OU.read.ATTR_NA
   ME.template.inline=JTwig_TEMPLATE
   ```

   In this format, replace:

   - *CONNECTOR_QUALIFIER* with the name of the connection properties file that you created in Setting Connection Properties.
   - *INSTANCE_ID* with the instance ID for your target.

- *OU* with the organizational unit against which the connector must perform transformation. The supported OU values are `People`, `Groups`, `Resources`, and `Datasets`.
- *ATTR_NAME* with the name of the LDAP Gateway attribute in which the transformed value must be stored.
- *Jtwig_TEMPLATE* with the Jtwig template for transformation.

3. Save and close the file.

Configuring Multiple Instances of the LDAP Gateway

You can configure and run multiple instances of the LDAP Gateway on the same host by entering unique port values for each instance of the LDAP Gateway.

To do so, install and configure the LDAP Gateway for each instance that you want to run. While installing the LDAP Gateway, ensure that the installation directory is different for each instance of the gateway.

Then, update the default values for each property listed in Table 2-3 so that the value is unique for each instance of the LDAP Gateway that is installed on the host. Suppose you are using the default values in the property files for instance 1, then for instance 2, replace the default value with a unique value for the property. For example, for instance 2, change the default value `6398` of the system `system.port` property in the*`LDAP_INSTALL_DIR`*`/conf/customer-configuration.properties` file to a unique value such as `8389`.

Table 2-3 Property Values To Be Updated for Running Multiple Instances of the LDAP Gateway

Property Name and Location	Property Description	Default Value
The `system.port` property in the*`LDAP_INSTALL_DIR`*`/conf/customer-configuration.properties` file	Gateway listening port (LDAP)	`6389`
The `system.ssl_port` property value in the *`LDAP_INSTALL_DIR`*`/conf/customer-configuration.properties` file	Gateway listening port (LDAPS)	`7389`
The `ds-cfg-listen-port` property under `dn: cn=LDAP Connection Handler,cn=Connection Handlers,cn=config` in the *`LDAP_INSTALL_DIR`*`/dsroot/config/config.ldif` file	OpenDJ listening port (LDAP)	`1389`
Set the value of the `ldap.port` property in the *`LDAP_INSTALL_DIR`*`/conf/ldapds.properties` file to the value set for the `ds-cfg-listen-port` property (in the preceding row)	Gateway config to read OpenDJ	`1389`

Table 2-3 (Cont.) Property Values To Be Updated for Running Multiple Instances of the LDAP Gateway

Property Name and Location	Property Description	Default Value
The `ds-cfg-listen-port` property under `dn: cn=LDAPS Connection Handler,cn=Connection Handlers,cn=config` in the *`LDAP_INSTALL_DIR`*`/dsroot/config/config.ldif` file	OpenDJ listening port (LDAPS)	`1636`
The `ds-cfg-listen-port` property under `dn: cn=Administration Connector,cn=config` in the *`LDAP_INSTALL_DIR`*`/dsroot/config/config.ldif` file	OpenDJ Administration port	`4444`
The `ds-cfg-replication-server` property under `dn: cn=localhost,cn=domains,cn=Multimaster Synchronization,cn=Synchronization Providers,cn=config` in the *`LDAP_INSTALL_DIR`*`/dsroot/config/config.ldif` file	OpenDJ Replication Server	`localhost:8989`
The `ds-cfg-replication-port` property value under `dn: cn=replication server,cn=Multimaster Synchronization,cn=Synchronization Providers,cn=config` in the *`LDAP_INSTALL_DIR`*`/dsroot/config/config.ldif` file	OpenDJ Replication Server Port	`8989`

Encrypting Data

Learn about encryption performed by the LDAP gateway and how to configure it.

- Understanding Encryption
- Configuring Encryption

Understanding Encryption

The *`LDAP_INSTALL_DIR`*`/conf/encryption.properties` file allows the ability to configure what properties, associated with the connector, must the LDAP Gateway manage as encrypted values.

The *`LDAP_INSTALL_DIR`*`/conf/encryption.properties` file is a common file containing properties of various modules that need to be securely protected. Use this file to define and encrypt any property located in the following files:

- connection properties file (created in Setting Connection Properties)
- *LDAP_INSTALL_DIR*/conf/customer-configuration.properties

When the LDAP gateway starts, it uses the encryption.properties file to examine the properties that it must represent in encrypted format.

For example, when the LDAP gateway starts, it reads the following entry from the encryption.properties file:

```
file.customer-configuration=adminUserPassword,altAdminUserPassword
```

This entry implies that there exists a properties file called customer-configuration.properties that contains sensitive properties adminUserPassword and altAdminPassword. The LDAP gateway searches for the customer-configuration.properties file, and if found, replaces any clear-text values for the adminUserPassword and altAdminPassword properties with an encrypted version.

Similarly, at start up, the LDAP gateway also reads the following entry from the encryption.properties file:

```
class.TopsModule=_secretKeyValue_
```

This entry implies that there exists a connector called TopsModule and its associated properties file (the one created in Setting Connection Properties) contains the sensitive property _secretKeyValue_ . The LDAP gateway searches for this properties file and replaces the clear-text value for the _secretKeyValue_ property with an encrypted value.

Encrypted values within property files are always represented using the ENC(*ENCRYPTED_STRING*) format. To add or replace an existing encrypted value with a new value, replace the entire encryption string if present (including the ENC(*ENCRYPTED_STRING*)) with a new clear-text value, and then restart the gateway. Once the gateway restarts, the newly added clear-text value goes through an encryption process with the result being written back out to the property file replacing the original clear-text value.

During the encryption process, the encryption framework that the gateway uses automatically detects the highest level of encryption possible by examining the version of the Java Virtual Machine running, along with any additional encryption libraries that may have been installed alongside the JVM. By default, Java 1.8 supports 128-bit AES encryption and Java 1.7 supports 40-bit AES encryption. You can install additional encryption libraries by BouncyCastle into the JVM allowing for up to 256-bit AES encryption.

The encryption process in the LDAP gateway also allows for automatic migration of encryption values from a lower bit strength to a higher strength as it becomes available. For example, if the gateway is initially deployed on a system running Java 1.7 with 40-bit AES and that system is upgraded to Java 1.8 running 128-bit AES, then upon the next restart of the gateway, all encrypted values remaining at the 40-bit AES level are automatically re-encrypted at the higher 128-bit and stored back out in the property files. This process eliminates the need to manually replace the values in every property file in order to take advantage of the higher bit strength.

The gateway uses the private key located in the *LDAP_INSTALL_DIR*/conf/idf.properties file for all the encryption and decryption that it performs. The idf.properties file is created in the conf directory when the LDAP gateway is started for the first time. It is recommended that access to this file is restricted.

> **Note:**
>
> Once the gateway is deployed and started for the first time, the value of the autogenerated encryption key in the idf.properties file should not be changed. However, you can change the file name and its location. For example, to store the idf.properties file to a more secure location, the default location (where the gateway resides) can be overwritten and defined as `system.idfprops.filepath=`*`ABSOLUTE_PATH_OF_THE_NEW_FILE`* in the customer-configuration.properties file.

Configuring Encryption

You can configure encryption by editing the `encryption.properties` file located in the *`LDAP_INSTALL_DIR`*`/conf/` directory.

By default, the LDAP gateway encrypts the values of:

- the `adminUserPassword` and `altAdminPassword` properties in the *`LDAP_INSTALL_DIR`*`/conf/customer-configuration.properties` file.
- the `_secretKeyValue_` property in the connection properties file (created in Setting Connection Properties).

If you want to encrypt additional properties in the `customer-configuration.properties` file, then you must include them as a comma-separated list in the following property of the `encryption.properties` file:

```
file.customer-configuration=adminUserPassword,altAdminUserPassword
```

For example, if you want to encrypt the `schema` and `suffix` properties of the `customer-configuration.properties` file, then include them in the`file.customer-configuation` property of the `encryption.properties` file as follows:

```
file.customer-
configuration=adminUserPassword,altAdminUserPassword,schema,suffix
```

If you want to encrypt additional properties in the connection properties file, then include them as a comma-separated list in the following property of the `encryption.properties` file:

```
class.TopsModule=_secretKeyValue_
```

For example, if you want to encrypt the `_host_` and `_port_` properties of the connection properties file, then include them in the `class.TopsModule=_secretKeyValue_` property of the `encryption.properties` file as follows:

```
class.TopsModule=_secretKeyValue_,_host_,_port_
```

If you want to change the values any encrypted properties, then remove the `ENC` along with the value and then add the new value.

For example, if the value of the `adminUserPassword` property in the `customer-configuration.properties` file is encrypted, then from the `adminUserPassword=ENC(t8+B0TbafPKyFFf0KoTlAmde82aRnwtf)` value, remove

`ENC(t8+B0TbafPKyFFf0KoTlAmde82aRnwtf)` and replace it with the new value, without the prefix ENC. Whenever the gateway is restarted, it automatically overwrites the clear-text value with its encrypted counterpart.

Understanding the Caching Layer

The LDAP gateway features an optional and configurable caching layer, which is a temporary storage area where frequently accessed data is stored for rapid access.

An expiration policy defines the time dependency for the cached resource. For example, the `cachingMaxAge` parameter specifies the maximum time in minutes when the data is not in sync with the target system. You can pair the caching layer with an incremental reconciliation (to maintain the most recently updated data in the caching layer. This improves the performance of the LDAP gateway. In addition, the caching layer opens the LDAP gateway for more advanced features defined by the LDAPv3 RFC.

Benefits of Using the Caching Layer

Using the caching layer provides the following benefits:

- Faster search operations (when the cache is primed)
- A unified Base DN for both provisioning and reconciliation data

When paired with an embedded directory server, the caching layer offers these additional benefits:

- The ability to perform advanced LDAP search filters against the gateway.
- The ability to query an RFC compliant ChangeLog for delta reconciliation.

> **Note:**
>
> In an environment where the items noted above may not be required, you can disable the caching layer.

Considerations for Using the Caching Layer

The LDAP gateway can suffer a performance penalty when all of the following conditions are met:

- There is no data in the cache, or the cache is stale based on the configuration.
- An LDAP search operation is performed to retrieve the children of an Organizational Unit. For example, the contents of `ou=People`. Such an LDAP search operation returns only DNs (along with RDN components).
- The connector only returns *key* information when returning a list of objects.
- The `cachingIterateBehavior` property in the `*LDAP_INSTALL_DIR*/conf/configuration.properties` file remains set to the default of `AUTO` and not overwritten within the `customer-configuration.properties` file.
  In such a scenario, an LDAP search operation initially retrieves the list of results, containing only DN and RDN values. The caching layer then iterates through each result, fetching and caching the details from the target system. Finally, the full set of results are returned to Oracle Identity Manager.

To avoid this scenario, it is recommended that you use the caching layer in combination with scheduled reconciliation. With reconciliation setup and the staleness settings configured properly the above conditions will not be met.

How to Enable or Disable the Caching Layer?

The caching layer is enabled by default. To override this default setting or disable the caching layer, copy the `cnctr.coreBean.nexus.cachingEnabled` property from the *`LDAP_INSTALL_DIR`*`/conf/configuration.properties` file to the *`LDAP_INSTALL_DIR`*`/conf/customer-configuration.properties` file and then set its value to `false`.

You can enable the caching layer by setting the value of the `cnctr.coreBean.nexus.cachingEnabled` property in the *`LDAP_INSTALL_DIR`*`/conf/customer-configuration.properties` file to `true` .

Configuring Scheduled Reconciliation

Scheduled reconciliation allows for establishing a periodic synchronization between the Identity Store associated with the LDAP Gateway and that represented by your target system reachable by way of the connector.

The Scheduled Recon Utility (provided by *`LDAP_INSTALL_DIR`*`/dist/scheduled-recon.jar`) is a tool that ships with the IdentityForge LDAP Gateway. It provides the ability to perform a full recon against a configurable target system, placing the results in the internal identity store of the gateway. This utility provides a basic scheduling service for kicking off the built-in batched reconciliation of the connector on a configurable interval.

An example properties file, scheduled-recon.properties.example file that defines the reconciliation setup and behavior is available in the *`LDAP_INSTALL_DIR`*`/conf` folder. Use this file to configure scheduled reconciliation.

1. In the *`LDAP_INSTALL_DIR`*`/conf` directory, create a copy of the *`LDAP_INSTALL_DIR`*`/conf/scheduled-recon.example` file and rename it `scheduled-recon.properties`.
2. If required, open the *`LDAP_INSTALL_DIR`*`/conf/scheduled-recon.properties` file in a text editor and configure it to meet your requirements.
3. Run the *`LDAP_INSTALL_DIR`*`/bin/run-recon.bat` file to start the scheduled recon utility.

   You can run this batch file with the following options

Argument	Description
`-h`	Use this argument for help.

Argument	Description
`-loglevel <level>`	Use this argument to define the logging level. Possible values are: • `severe` • `warning` • `info` • `fine` • `finer` • `finest` Default value is `warning`.
`-logfile <filepath>`	Use this argument to specify the path to the log file.
`-p <properties filepath>`	Use this argument to specify the path to the scheduled-recon.properties file.

The following is the basic command structure for executing this batch file:

```
…\ldapgateway6\bin>run-recon.bat -loglevel "warning" -logfile <location of
the log file> -p "D:\ldapgateway6\conf\scheduled-recon.properties"
```

About Parsing Grammar Protocol 1.0

Grammar is necessary for properly parsing user and group listings that come into the gateway from the mainframe agent during search requests and reconciliation events.

The grammar represents line-by-line parsing instructions that convert the semi-structured textual data into LDAP attributes and their respective values. Each line (ending in CRLF) of the listing received from the agent can be represented by an individual grammar definition and specified in the grammar file. The grammar file is present in the `<conf/parser-grammar/>` folder.

Grammar files with the default grammar are present in the *LDAP_INSTALL_DIR*`/conf/parser-grammars/tops` directory. It parses user and group listings that come into the gateway from the mainframe agent during search requests and reconciliation events.

For example, following is the user listing for CA Top Secret:

```
ACCESSORID = 0518AA NAME = 0518AA
TYPE = USER SIZE = 1024 BYTES
DEPT ACID = DEPTX DEPARTMENT = LARGE-DEPT-TEST
CREATED = 05/19/15 16:50 LAST MOD = 07/29/15 22:02
----------- SEGMENT CICS
OPCLASS = 09
OPIDENT = ABC
OPPRTY = 010
SCTYKEY = 001,005,007,009,011,097
SITRAN = F FACILITY = *ALL*
----------- SEGMENT LU6.2
#APPL = APTRA_VISION_USER|ATM_CONFIG_USER|CS_REVIEW
SET1DISP = AB|AC|AD|AE|
------------ SEGMENT NETVIEW
DOMAINS = 09
INIT CMD = 09
----------- SEGMENT OMVS
```

```
ROOT = 09
----------- SEGMENT WORKATTR
ACCOUNT = 123
ADDRESS1 = ADDR 1
ADDRESS2 = ADDR 2
ADDRESS3 = ADDR 3
ADDRESS4 = ADDR 4
BUILDING = BLDG 17
DEPARTMENT = DEPART 1
NAME = NAME
ROOM NUMBER= ROOM B1
XA DATASET = ADBFLE OWNER(00123 21322 )
ACCESS = READ,WRITE
XA DATASET = TDBFLE OWNER(00123 54232 )
ACCESS = READ,DELETE,WRITE
XA DATASET = TDBFLE OWNER(00123 )
PASSWORD = EXPIRES = 06/18/15 INTERVAL = 030
```

Using the above listing, if you want to parse out the OPCLASS value from the listing and assign it to an LDAP attribute called “opcls”, then you can construct the following <Line> element in the grammar file:

```
<Line id=”opclassVal” enabled=”yes” sig=”[ ]*OPCLASS = (?
&lt;opcls&gt;.*)"/>
```

The signature attribute (sig) in the Line element above is a regex that represents the rules for pulling out the value and assigning it to an LDAP attribute. Regex named groups are used as the convention for assigning the discovered values to LDAP attributes exposed through the connector.

The following table lists the attributes of a line element. The allowed values for these attributes are **yes**or **no**.

String	Mandatory?	Definition
id	Yes	Unique ID that is given to the line definition. Used primarily for internal referencing purposes, such as with the 'dependson' attribute. Values allowed: `any`
enabled	No	Specifies whether the line is eligible for participating in the parsing process. Use this flag to override files (turn off lines). Default value: yes
signature	Yes	Defines the rules for what values are to be extracted for each line of the listing and which LDAP attributes should be assigned the values.
required	No	Defines whether an attribute is required or not. Default to: `yes`

String	Mandatory?	Definition
multiline_sig	No	An optional regex expression to define the signature of a follow-on line that could represent whether the value was wrapped around two additional lines in the document. Values allowed: Any valid regex containing attribute matching key and attribute name. Defaults to: `empty value`
repeats	No	Represents whether the line can show up multiple times in the document. If set to `no`, then once the line is found, this Line definition is not evaluated again for the rest of the document. Defaults to: `No`
overflow	No	Represents whether data for an associated attribute can overflow to the next line. In case of an overflow, the final value of an attribute is derived by concatenating all values. Defaults to: `No`
multivalue_parser	No	An optional regex expression that defines how the found values are to be parsed out and turned into a multivalued list, such as using '(\S+)' to parse values that are space delimited. Values Allowed: Any valid regex Defaults to: `empty value`
applyCompositeRef	no	An optional comma-separated list of composite attributes to be built immediately after processing the line. Each value in the comma-separated list must correspond to the "id" attribute of a CompositeAttribute definition.
defaultvalue	No	Defines the default value for an attribute. If this line does not match with any line of input, then this default value will be assigned to attribute.

Customizing Grammar Rules

The grammar file with the default grammar is present in the conf folder. It parses user and group listings that come into the gateway from the mainframe agent during search requests and reconciliation events. You can apply new grammar rules to append to or override rules that come out of the box. To define new grammar rules or override the existing rules, create a grammar file `parser-grammars.cust` file in the `<conf/parser-grammar/>` folder.

You can apply new grammar rules to append to or override rules that are available by default in the *LDAP_INSTALL_DIR*/conf/parser-grammars/tops directory.

To define new grammar rules or override the existing rules, you must create a custom grammar file (for example, `tops_FindAllUsers.cust`) in the *LDAP_INSTALL_DIR*/conf/parser-grammars/tops directory.

> **Note:**
>
> - If the Id of the existing attribute matches with the attribute in the grammar line, it overrides the existing grammar definition.
> - If the Id of the existing attribute does not match with the attribute in the grammar line, it creates a new grammar definition.

Key Considerations

- The `parser-grammars.cust` grammar file must be at the same location where the default grammar files are located (*LDAP_INSTALL_DIR*/conf/parser-grammars/tops).
- The name of the grammar file must be the same except the `cust` extension.
- The name of the grammar file must be the same except the `cust` extension. For example, if you need to customize the grammar for the *LDAP_INSTALL_DIR*/conf/parser-grammars/tops/tops_FindAllUsers.xml file, then create a custom grammar file *LDAP_INSTALL_DIR*/conf/parser-grammars/tops/tops_FindAllPeople.cust.
- For the grammar definitions to override, the ID attribute from both the files should match.

Nomenclature of the parsing grammar files

Each grammar file is named for the type of operation and listing it is responsible for parsing.

For example, for CA Top Secret, use the following for user extraction:

- `tops_FindAllUsers.xml` – fetches the IDs of all users.
- `tops_FindUserById.xml` – fetches all the details of a single user (for the given ID).

Overriding default existing grammar definitions

The grammar definitions specified in the grammar file parser-grammars.cust override the default grammar definitions specified in the property files. To enable overriding of the particular line, the ID attribute in the custom provided attribute should match with the default grammar definition.

The grammar definitions specified in the custom grammar file override the default grammar definitions. To enable overriding of a particular line, the ID attribute in the custom provided attribute should match with the default grammar definition.

For example, if the default grammar definition in the property file and the definition specified in the custom grammar file is as shown in the following lines, then the definition is disabled and the line is not parsed.

```
<Protocol><Lines>
<Line id="elId" enabled="yes" sig="[ ]*ELID[ ]*=[ ]*(?<ELID>.*)"/>
</Lines>/Lines>
```

```
<Protocol><Lines>
<Line id="elId" enabled="no" sig="[ ]*ELID[ ]*=[ ]*(?<ELID>.*)"/>
</Lines></Protocol>
```

New grammar definitions

You can specify new grammar definitions in the custom grammar file that you create. For example, the following grammar definition is used to get values of `DEPT_ACID=001 DEPT_NAME=hr`.

```
<Protocol><Lines>
="deptAcid" enabled="yes" sig="[ ]*DEPT_ACID[ ]*=[ ]*(?&lt;deptacid&gt;.*?)
[ ]*DEPT_NAME[ ]*=[ ]*(?&lt;department&gt;.*)" />
</Lines></Protocol>
```

3
Deploying the CA Top Secret Connector in Oracle Identity Manager

The LDAP Gateway acts as the intermediary between Oracle Identity Manager and the connector components on the mainframe. The following sections of this chapter describe the procedure to deploy some components of the connector, including the LDAP Gateway, on the Oracle Identity Manager host computer:

> **Note:**
>
> The procedure to deploy the mainframe components of the connector is described in the next chapter.

- Running the Connector Installer
- Configuring the IT Resource
- Configuring Oracle Identity Manager
- Localizing Field Labels in UI Forms
- Clearing Content Related to Connector Resource Bundles from the Server Cache
- Enabling Logging
- Configuring the Connector for Audit Comments

Running the Connector Installer

When you run the Connector Installer, it automatically copies the connector files to directories in Oracle Identity Manager, imports connector XML files, and compiles adapters used for provisioning.

To run the Connector Installer:

1. Copy the contents of the connector installation media into the following directory: `OIM_HOME/server/ConnectorDefaultDirectory`

   > **Note:**
   >
   > If this is the first time you are running the Connector Installer for deploying the connector bundle in a Connector Server, then place the bundle in the connector server bundle directory.

2. Log in to Oracle Identity System Administration.
3. In the left pane, under System Management, click **Manage Connector**.
4. In the Manage Connector page, click **Install**.

5. From the Connector List list, select **CA Top Secret Advanced *RELEASE_NUMBER***. This list displays the names and release numbers of connectors whose installation files you copy into the default connector installation directory: `OIM_HOME/server/ConnectorDefaultDirectory`

   If you have copied the installation files into a different directory, then:

   a. In the **Alternative Directory** field, enter the full path and name of that directory.

   b. To repopulate the list of connectors in the Connector List list, click **Refresh**.

   c. From the Connector List list, select **CA Top Secret Advanced *RELEASE_NUMBER***.

6. Click **Load**.

7. To start the installation process, click **Continue**. The following tasks are performed in sequence:

   a. Configuration of connector libraries

   b. Import of the connector XML files (by using the Deployment Manager)

   c. Compilation of adapters

   On successful completion of a task, a check mark appears for the task. If a task fails, then an X mark and a message stating the reason for failure are displayed. If a task fails, then make the required correction and perform one of the following steps:

   a. To retry the installation, click **Retry**.

   b. To cancel the installation and restart the installation process, click **Cancel**, and then repeat Steps 1 through 8.

8. If all three tasks of the connector installation process are successful, then a message indicating successful installation appears.

   In addition, a list of the steps that you must perform after the installation appears. These steps are as follows:

   a. Configuring the IT resource for the connector. The procedure to configure the IT resource is described later in this guide.

   b. Configuring the scheduled tasks that are created when you installed the connector. The procedure to configure these scheduled tasks is described later in this guide.

   When you run the Connector Installer, it copies the connector files and external code files to destination directories on the Oracle Identity Manager host computer. These files are listed in Files and Directories in the CA Top Secret Connector Package.

Configuring the IT Resource

The IT resource for the target system contains connection information about the target system. Oracle Identity Manager uses this information during provisioning and reconciliation. The IT resource for this connector is automatically created when you run the Connector Installer, and you must specify values for the parameters of the IT resource.

You must specify values for the parameters of the TopSecretResource IT resource as follows:

1. Log in to the Oracle Identity System Administration.
2. In the left pane, under Configuration, click **IT Resource.**
3. In the **IT Resource Name** field on the Manage IT Resource page, enter `TopSecretResource` and then click **Search.**
4. Click the edit icon for the IT resource.
5. From the list at the top of the page, select **Details and Parameters.**
6. Specify values for the parameters of the IT resource. Table 3-1 describes each parameter.

Table 3-1 IT Resource Parameters for CA Top Secret Connector

Parameter	Description
AtMap User	This parameter holds the name of the lookup definition containing attribute mappings that are used for provisioning. Value: `AtMap.TOPS` **Note:** You must not change the value of this parameter.
auditTemplate	This parameter is required for audit statements to be passed on along with all TSS commands. If you do not specify a value for this parameter, then the connector will not post audit comments for any process task that is initiated from Oracle Identity Manager. Sample value: `/* Operation initiated by {{auditcomment}} through OIM */` See Configuring the Connector for Audit Comments for detailed information on value to be specified for this parameter.
idfBackendDn	Enter the user ID that the connector will use to connect to the LDAP Gateway backend. Sample value: `cn=Directory Manager,dc=system,dc=backend`
idfBackendPassword	Enter the password of the user ID that the connector will use to connect to the LDAP Gateway backend. You also set this password in the configuration.properties file of the LDAP Gateway. **Note:** Do not enter an encrypted value.
idfbackendContext	Enter the root context for LDAP Gateway backend. Sample Value: `dc=system,dc=backend`
idfConnectTimeoutMS	Enter an integer value that specifies the number of milliseconds after which an attempt to establish a connection between the LDAP gateway and Oracle Identity Manager times out. If you do not enter a value for this parameter, then the connector uses a default time out of `300000` ms (that is, 5 minutes). **Note:** If the number of records to be retrieved are high, ensure to adjust or increase the timeout value accordingly.
idfPrincipalDn	Set a user ID for an account that the connector will use to connect to the LDAP Gateway. Format: `cn=`*`USER_ID`*`,dc=tops,dc=com` Sample value: `cn=idfTopsAdmin,dc=tops,dc=com`

Table 3-1 (Cont.) IT Resource Parameters for CA Top Secret Connector

Parameter	Description
idfPrincipalPwd	Set a password for the account that the connector will use to connect to the LDAP Gateway. You also set this password in the files listed in the description of the idfPrincipalDn parameter. **Note:** Do not enter an encrypted value.
idfReadTimeoutMS	Enter an integer value that specifies the number of milliseconds after which an attempt to read data from the target system times out. If you do not enter a value for this parameter, then the connector uses a default time out of `1800000` ms (that is, 30 minutes). **Note:** If the number of records to be retrieved are high, ensure to adjust or increase the timeout value accordingly.
idfRootContext	This parameter holds the root context for CA Top Secret. Value: `dc=tops,dc=com` **Note:** You must not change the value of this parameter.
idfServerHost	This parameter holds the host name or IP address of the computer on which you install the LDAP Gateway. For this release of the connector, you install the LDAP Gateway on the Oracle Identity Manager host computer. Default value: `localhost` **Note:** Do not change the value of this parameter unless you have installed the LDAP Gateway on a different machine from the Oracle Identity Manager host computer.
idfServerPort	Enter the number of the port for connecting to the LDAP Gateway. Sample value: `5389`
idfSsl	This parameter determines whether the LDAP Gateway will use SSL to connect to the target system. Enter `true` if using SSL. Otherwise, enter `false.` Sample value: `true`
idfTrustStore	This parameter holds the directory location of the trust store containing the SSL certificate. This parameter is optional, and should only be entered when using SSL authentication. This must be the full path to the directory location. Sample value: `/app/home/ldapgateway/conf/idf.jks`
idfTrustStorePassword	This parameter holds the password for the SSL trust store. This parameter is optional, and should only be entered when using SSL authentication.
idfTrustStoreType	This parameter holds the trust store type for the SSL trust store. This parameter is optional, and should only be entered when using SSL authentication. Sample value: `jks`

Table 3-1 (Cont.) IT Resource Parameters for CA Top Secret Connector

Parameter	Description
Last Modified Time Stamp	The most recent start time of the Reconcile LDAP Users reconciliation scheduled task is stored in this parameter. See Top Secret Reconcile LDAP Users to OIM for more information about his scheduled task. The format of the value stored in this parameter is as follows: MM/dd/yy hh:mm:ss a In this format: MM is the month of the year. dd is the day of the month. yy is the year. hh is the hour in am/pm (01-12). mm is the minute in the hour. ss is the second in the minute. a is the marker for AM or PM. Sample value: `05/07/10 02:46:52 PM` The default value is 0. The reconciliation task will perform full LDAP user reconciliation when the value is 0. If the value is a non-zero, standard time-stamp value in the format given above, then incremental reconciliation is performed. Only records that have been created or modified after the specified time stamp are brought to Oracle Identity Manager for reconciliation. **Note:** When required, you can manually enter a time-stamp value in the specified format.
Secondary IT resource	If you created a secondary IT resource for reconciliation or provisioning, then enter its name.

7. To save the values, click **Update.**

Configuring Oracle Identity Manager

You must create additional metadata such as a UI form and an application instance. In addition, you must run entitlement and catalog synchronization jobs. These procedures are described in the following sections:

- Creating and Activating a Sandbox
- Creating a New UI Form
- Creating an Application Instance
- Publishing a Sandbox
- Harvesting Entitlements and Sync Catalog
- Updating an Existing Application Instance with a New Form

Creating and Activating a Sandbox

Create and activate a sandbox as follows:

1. On the upper navigation bar, click **Sandboxes.** The Manage Sandboxes page is displayed.
2. On the toolbar, click **Create Sandbox.** The Create Sandbox dialog box is displayed.

3. In the Sandbox Name field, enter a name for the sandbox. This is a mandatory field.
4. In the Sandbox Description field, enter a description of the sandbox. This is an optional field.
5. Click **Save and Close.** A message is displayed with the sandbox name and creation label.
6. Click **OK.** The sandbox is displayed in the Available Sandboxes section of the Manage Sandboxes page.
7. From the table showing the available sandboxes in the Manage Sandboxes page, select the newly created sandbox that you want to activate.
8. On the toolbar, click **Activate Sandbox.**

   The sandbox is activated.

Creating a New UI Form

Create a new UI form as follows:

1. In the left pane, under Configuration, click **Form Designer.**
2. Under Search Results, click **Create.**
3. Select the resource type for which you want to create the form, for example, **OIMTopSecretResourceObject.**
4. Enter a form name and click **Create.**

Creating an Application Instance

Create an application instance as follows:

1. In the System Administration page, under Configuration in the left pane, click **Application Instances.**
2. Under Search Results, click **Create.**
3. Enter appropriate values for the fields displayed on the Attributes form and click **Save.**
4. In the Form drop-down list, select the newly created form and click **Apply.**
5. Publish the application instance to an organization to make the application instance available for requesting and subsequent provisioning to users. See the "Managing Organizations Associated With Application Instances" section in *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Manager* for detailed instructions.

Publishing a Sandbox

To publish the sandbox that you created in Creating and Activating a Sandbox:

1. Close all the open tabs and pages.
2. In the upper-right corner of the page, click the Sandboxes link.

   The Manage Sandboxes page is displayed.

3. From the table showing the available sandboxes in the Manage Sandboxes page, select the sandbox that you created in Creating and Activating a Sandbox.
4. On the toolbar, click **Publish Sandbox.** A message is displayed asking for confirmation.
5. Click **Yes** to confirm. The sandbox is published and the customizations it contained are merged with the main line.

Harvesting Entitlements and Sync Catalog

To harvest entitlements and sync catalog:

1. Run the scheduled jobs for lookup field synchronization. See Scheduled Tasks for Lookup Field Synchronization for more information about these scheduled jobs.
2. Run the Entitlement List scheduled job to populate Entitlement Assignment schema from child process form table.
3. Run the Catalog Synchronization Job scheduled job.

> **See Also:**
>
> Predefined Scheduled Tasks in *Oracle Fusion Middleware Administering Oracle Identity Governance* for a description of the Entitlement List and Catalog Synchronization Job scheduled jobs

Updating an Existing Application Instance with a New Form

For any changes you do in the Form Designer, you must create a new UI form and update the changes in an application instance. To update an existing application instance with a new form:

1. Create a sandbox and activate it as described in Creating and Activating a Sandbox.
2. Create a new UI form for the resource as described in Creating a New UI Form.
3. Open the existing application instance.
4. In the **Form** field, select the new UI form that you created.
5. Save the application instance.
6. Publish the sandbox as described in Publishing a Sandbox.

Localizing Field Labels in UI Forms

You can localize UI form field labels by using the resource bundle corresponding to the language you want to use. The resource bundles are available in the connector installation package.

Perform the following steps to localize field labels that you add to in UI forms:

1. Log in to Oracle Enterprise Manager.
2. In the left pane, expand **Application Deployments** and then select **oracle.iam.console.identity.sysadmin.ear.**
3. In the right pane, from the Application Deployment list, select **MDS Configuration.**

4. On the MDS Configuration page, click **Export** and save the archive to the local computer.
5. Extract the contents of the archive, and open the following file in a text editor:

   SAVED_LOCATION\xliffBundles\oracle\iam\ui\runtime\BizEditorBundle.xlf
6. Edit the BizEditorBundle.xlf file as follows:
   a. Search for the following text:

      ```
      <file source-language="en"
      original="/xliffBundles/oracle/iam/ui/runtime/BizEditorBundle.xlf"
      datatype="x-oracle-adf">
      ```

   b. Replace with the following text:

      ```
      <file source-language="en" target-language="LANG_CODE"
      original="/xliffBundles/oracle/iam/ui/runtime/BizEditorBundle.xlf"
      datatype="x-oracle-adf">
      ```

      In this text, replace *LANG_CODE* with the code of the language that you want to localize the form field labels. The following is a sample value for localizing the form field labels in Japanese:

      ```
      <file source-language="en" target-language="ja"
      original="/xliffBundles/oracle/iam/ui/runtime/BizEditorBundle.xlf"
      datatype="x-oracle-adf">
      ```

   c. Search for the application instance code. The original code will be in the following format:

      ```
      <trans-unit id="$
      {adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResourceBundl
      e']
      ['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity.userEO.
      UD_<Field_Name>__c_description']}">
      <source><Field_Label></source>
      <target/>
      </trans-unit>
      <trans-unit
      id="sessiondef.oracle.iam.ui.runtime.form.model.<UI_Form_Name>.entity.
      <UI_Form_Name>EO.UD_<Field_Name>__c_LABEL">
      <source><Field_Label></source>
      <target/>
      </trans-unit>
      ```

      For example, the following sample code show the update that should be made for the FULL NAME field on a UI form named TopSecretUserFormv1:

      ```
      <trans-unit id="$
      {adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResourceBundl
      e']
      ['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity.userEO.
      UD_IDF_TOPS_CN__c_description']}">
      <source>FULL NAME</source>
      <target/>
      </trans-unit>
      <trans-unit
      id="sessiondef.oracle.iam.ui.runtime.form.model.TopSecretUserFormv1.entit
      y.TopSecretUserFormv1EO.UD_IDF_TOPS_CN__c_LABEL">
      <source>FULL NAME</source>
      <target/>
      </trans-unit>
      ```

d. Open the resource file from the /resources directory in the connector installation media, for example TopSecret-Adv_ja.properties, and get the value of the attribute from the file, for example global.udf.UD_IDF_TOPS_CN=\u6C0F\u540D.

e. Replace the original code shown in Step 6.c with the following:

```
<trans-unit id="$
{adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResourceBundle']
['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity.userEO.UD_<Fi
eld_Name>__c_description']}">
<source>< global.udf.UD_Field_Name></source>
<target/>enter Unicode values here</target>
</trans-unit>
<trans-unit
id="sessiondef.oracle.iam.ui.runtime.form.model.<UI_Form_Name>.entity.
<UI_Form_Name>EO.UD_<Field_Name>__c_LABEL">
<source><Field_Label></source>
<target/>enter Unicode values here</target>
</trans-unit>
```

As an example, the code for FULL_NAME field translation would be:

```
<trans-unit id="$
{adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResourceBundle']
['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity.userEO.UD_IDF
_TOPS_CN__c_description']}">


<source>FULL_NAME</source>
<target>\u6C0F\u540D</target>
</trans-unit>
<trans-unit
id="sessiondef.oracle.iam.ui.runtime.form.model.TopSecretUserFormv1.entity.TopS
ecretUserFormv1EO.UD_IDF_TOPS_CN__c_LABEL">
<source>FULL_NAME</source>
<target>\u6C0F\u540D</target>
</trans-unit>
```

f. Repeat Steps 6.6.c through 6.6.e for all attributes of the process form.

g. Save the file as BizEditorBundle_LANG_CODE.xlf. In this file name, replace *LANG_CODE* with the code of the language to which you are localizing. Sample file name: `BizEditorBundle_ja.xlf`.

7. Repackage the ZIP file and import it into MDS.

8. Log out of and log in to Oracle Identity Manager.

Clearing Content Related to Connector Resource Bundles from the Server Cache

When you deploy the connector, the resource bundles are copied from the resources directory on the installation media into Oracle Identity Manager database. Whenever you add a new resource bundle to the connectorResources directory or make a change in an existing resource bundle, you must clear content related to connector resource bundles from the server cache.

To clear content related to connector resource bundles from the server cache:

1. In a command window, switch to the *OIM_HOME*/server/bin directory.

> **Note:**
>
> You must perform Step 1 before you perform Step 2. Otherwise, an exception is thrown if you run the command described in Step 2 as follows:
>
> ```
> OIM_HOME/server/bin/SCRIPT_FILE_NAME
> ```

2. Enter one of the following commands:

   > **Note:**
   >
   > You can use the PurgeCache utility to purge the cache for any content category. Run `PurgeCache.bat` *`CATEGORY_NAME`* on Microsoft Windows or `PurgeCache.sh` *`CATEGORY_NAME`* on UNIX. The *`CATEGORY_NAME`* argument represents the name of the content category that must be purged.
   >
   > For example, the following commands purge Metadata entries from the server cache:
   >
   > ```
   > PurgeCache.bat MetaData
   > ```
   >
   > ```
   > PurgeCache.sh MetaData
   > ```

   - On Microsoft Windows: `PurgeCache.bat All`
   - On UNIX: `PurgeCache.sh All`

   When prompted, enter the user name and password of an account belonging to the SYSTEM ADMINISTRATORS group. In addition, you are prompted to enter the service URL in the following format:

   ```
   t3://OIM_HOST_NAME:OIM_PORT_NUMBER
   ```

   In this format:

   - Replace *`OIM_HOST_NAME`* with the host name or IP address of the Oracle Identity Manager host computer.
   - Replace *`OIM_PORT_NUMBER`* with the port on which Oracle Identity Manager is listening.

Enabling Logging

The CA Top Secret connector supports two forms of logging, namely LDAP gateway-level logging and Oracle Identity Manager-level logging.

This section discusses the following topics:

- Enabling Logging for the LDAP Gateway
- Event Logging in Oracle Identity Manager

Enabling Logging for the LDAP Gateway

LDAP Gateway logging operations are managed by the log4j2.properties file, which is located in the *LDAP_INSTALL_DIR*/conf/ directory.

In the log4j2.properties file, edit the rootLogger log level:

```
rootLogger.level = INFO
```

The following is a list of log levels that can be used:

- ALL

  This level enables logging for all events.
- DEBUG

  This level enables logging of information about fine-grained events that are useful for debugging.
- INFO

  This level enables logging of messages that highlight the progress of the application at a coarse-grained level.
- WARN

  This level enables logging of information about potentially harmful situations.
- ERROR

  This level enables logging of information about error events that might allow the application to continue running.
- FATAL

  This level enables logging of information about very severe error events that could cause the application to stop functioning.
- OFF

  This level disables logging for all events.

Multiple log files are available for use with the connector. Table 3-2 lists the name, location, and contents of each LDAP gateway log file.

Table 3-2 Log Files and their Contents for CA Top Secret Connector

Log File	Description
nohup.out	This log file contains the console window output from the LDAP Gateway. This file is primarily used in conjunction with the run.sh script (instead of the run.bat file) **Location:** .../ldapgateway/bin/
idfserver.log.0	This log file contains provisioning and reconciliation logging messages from the LDAP Gateway and is the primary log file used by the gateway component. **Location:** .../ldapgateway/logs/

Event Logging in Oracle Identity Manager

Oracle Identity Manager uses Oracle Java Diagnostic Logging (OJDL) for logging. OJDL is based on java.util.logger.

This section contains the following topics:

Understanding the Log Levels

Configuring Logging in Oracle Identity Manager

Understanding the Log Levels

To specify the type of event for which you want logging to take place, you can set the log level to one of the following:

- SEVERE.intValue()+100

  This level enables logging of information about fatal errors.
- SEVERE

  This level enables logging of information about errors that might allow Oracle Identity Manager to continue running.
- WARNING

  This level enables logging of information about potentially harmful situations.
- INFO

  This level enables logging of messages that highlight the progress of the application.
- CONFIG

  This level enables logging of information about fine-grained events that are useful for debugging.
- FINE, FINER, FINEST

  These levels enable logging of information about fine-grained events, where FINEST logs information about all events.

Log Levels in Oracle Identity Manager

These log levels are mapped to ODL message type and level combinations as shown in Table 3-3.

Table 3-3 Log Levels and ODL Message Type:Level Combinations

Log Level	ODL Message Type:Level
SEVERE.intValue()+100	INCIDENT_ERROR:1
SEVERE	ERROR:1
WARNING	WARNING:1
INFO	NOTIFICATION:1
CONFIG	NOTIFICATION:16

Table 3-3 (Cont.) Log Levels and ODL Message Type:Level Combinations

Log Level	ODL Message Type:Level
FINE	TRACE:1
FINER	TRACE:16
FINEST	TRACE:32

Configuring Logging in Oracle Identity Manager

OIM level logging operations are managed by the logging.xml file, which is located in following directory:

DOMAIN_NAME/config/fmwconfig/servers/*SERVER_NAME*/

Loggers are used to configure logging operations for the connector's OIM functions. To configure loggers:

1. In the text editor, open the *DOMAIN_NAME*/config/fmwconfig/servers/*SERVER_NAME*/logging.xml file.
2. Locate the logger you want to configure. If adding a logger for the first time, you must create the logger definition. Table 3-4 lists the Oracle Identity Manager loggers for this connector.

Table 3-4 Logger Parameters

Logger	Description
COM.IDENTITYFORGE.IDFTOPSUSEROPERATIONS	Logs events related to provisioning operations from Oracle Identity Manager to the LDAP gateway, such as user creation and modification events.
COM.IDENTITYFORGE.UTIL.TOPS.IDFLDAPOPERATIONS	Logs events related to basic LDAP functions, including connecting to and disconnecting from the LDAP gateway.
COM.IDENTITYFORGE.TOPS.TASKS.FINDALLDATASETSTASK	Logs events related to the Find All Datasets scheduled task.
COM.IDENTITYFORGE.TOPS.TASKS.FINDALLFACILITIESTASK	Logs events related to the Find All Facilities scheduled task.
COM.IDENTITYFORGE.TOPS.TASKS.FINDALLGROUPSTASK	Logs events related to the Find All Groups scheduled task.
COM.IDENTITYFORGE.TOPS.TASKS.FINDALLPROFILESTASK	Logs events related to the Find All Profiles scheduled task.
COM.IDENTITYFORGE.TOPS.TASKS.FINDALLSOURCESTASK	Logs events related to the Find All Sources scheduled task.
COM.IDENTITYFORGE.TOPS.TASKS.RECONCILEALLLDAPUSERSTASK	Logs events related to the Reconcile All LDAP Users scheduled task.
COM.IDENTITYFORGE.TOPS.TASKS.RECONCILEUSERSTOINTERNALLDAPTASK	Logs events related to the CFILE extract from TSS to initialize users to the internal LDAP, reconcile users to internal LDAP scheduled task.

Table 3-4 (Cont.) Logger Parameters

Logger	Description
COM.IDENTITYFORGE.TOPS.TASKS.RECONCILEALLUSERSTASK	Logs events related to the Reconcile All Users scheduled task
COM.IDENTITYFORGE.TOPS.TASKS.RECONCILEDELETEDUSERSTOOIMTASK	Logs events related to the Reconcile Deleted Users to OIM scheduled task.

3. Define the <logger> element and its handlers. You can use the standard odl-handler as the log handler, or write your own.

   The following is an example of a logger definition for the Reconcile All Users scheduled task:

   ```
   <logger name="COM.IDENTITYFORGE.TOPS.TASKS.RECONCILEALLUSERSTASK"
   level='TRACE:32'>
   <handler name='odl-handler'/>
   </logger>
   ```

4. Save the changes and close the file.
5. Restart the Oracle Identity Manager server for the changes to take effect.

Log statements will be written to the path that is defined in the log handler that you assigned in the logger definition. For example, in the above logger definition for the Reconcile All Users scheduled task (in step3), the handler is odl-handler, which has the following default output file path:

```
${domain.home}/servers/${weblogic.Name}/logs/${weblogic.Name}-
diagnostic.log'
```

Configuring the Connector for Audit Comments

If you want to configure the connector to pass on all TSS command comments for audit purposes, then you must specify a value for auditTemplate parameter of the IT resource.

The value of this parameter must be in the following format for the connector to construct the required audit statement:
`/*`*`MY_AUDIT_TEXT`* `{{auditcomment}}` *`MY_AUDIT_TEXT`*`*/`

Sample value: `/* Operation initiated by {{auditcomment}} through OIM */`

In this format::

- The value must be begin with `/*` and end with `*/`
- `{{auditcomment}}` must be included in this exact manner. At run time, the connector replaces `{{auditcomment}}` with a dynamic value that is obtained from the **Desc** field of the `auditInfo` parameter that is present in the method signature of the adapter task.
- The text surrounding `{{auditcomment}}` can be any text of your choice for audit.

The connector already includes the `auditInfo` parameter for some of the commonly used provisioning adapters such as **ModifyUserAttrTops**. In such a scenario, you only need to search for the adapter task corresponding to the provisioning operation for which you want the connector to pass on audit statements. Then, edit the adapter task to locate the `auditInfo` method parameter and update its **Desc** field to include the audit text that meets your requirements. This value replaces `{{auditcomment}}` in the audit template to build the audit comment to be passed with the TSS command.

In scenarios where the adapter task does not include the `auditInfo` parameter (for example, **RemoveTopsUserFromSources**), you need to manually create a new adapter task for audit (for example, RemoveTopsUserFromSourceWithAudit) selecting the relevant constructor and method signatures, and then adding the `auditInfo` method parameter.

The following is the procedure for updating the default description of the `auditInfo` parameter to include an audit message that meets your requirements:

1. Log in to Oracle Identity Manager Design Console.
2. Expand **Development Tools**, and then double-click **Adapter Factory**.
3. Search for and open the adapter corresponding to the provisioning operation for which you want audit statements to be included. For example, if you want the connector to include audit statements for modify user provisioning operations, then search for and open the **ModifyUserAttrTops** adapter.
4. On the Adapter Tasks tab, double-click the corresponding adapter task (for example, **modifyTopsUserWithAuditGeneric**) for editing.

   The Edit Adapter Factory Task Parameters window is displayed.
5. In the Application Method Parameters region, expand Method, double-click the parameter for auditInfo, and then in the **Desc** field, enter the audit text that must be passed to Adapter Factory Task Parameters {{auditcomment}} in the audit template.
6. Save your updates and close the window.

> **See Also:**
>
> Using the Adapter Factory in the *Oracle® Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance* guide for detailed information about creating and modifying adapter tasks

4

Installing and Configuring the Agents of the CA Top Secret Connector on the Mainframe

Install the Provisioning Agent - Pioneer and Reconciliation Agent - Voyager of the CA Top Secret connector on the mainframe. These agents communicate with the LDAP Gateway during connector operations.

Installation Requirements for Agents

These are the software and environmental setting requirements for installing the Provisioning Agent - Pioneer and Reconciliation Agent - Voyager.

Verifying Installation Requirement

Ensure that the mainframe system on which you intend to install Pioneer and Voyager meet the following requirements:

Table 4-1 Installation Requirements for Agents

Item	Requirement
Operating System	IBM z/OS 2.2, 2.3
Message Transport Layer	TCP/IP
CA Top Secret Identity Repository	Verify that the current patch for z/OS is installed.
Target system user account for the Provisioning Agent - Pioneer and Reconciliation Agent - Voyager	CA Top Secret-authorized user account with System Administrators privileges.

Note:

Both the Voyager and Pioneer Agents must have CA Top Secret ACIDs defined on the CA Top Secret database. These ACIDs must have at least the permissions of the System Administrators group on the mainframe. These user accounts have permissions above those of ordinary administrators on the mainframe, which include Read, Write, Execute, and Modify privileges. Voyager and Pioneer use Language Environment. The following are the recommended Language Environment runtime options that avoid issues when installing Voyager and Pioneer:

- ALL31(ON)
- HEAP(32768,32768,ANYWHERE,KEEP,8192,4096)
- STACK(131072,131072,ANYWHERE,KEEP,524288,524288)

Environmental Settings and Other Requirements

Ensure that the following requirements are met on the mainframe:

- Voyager and Pioneer each require approximately a 2-megabyte region to work. In addition, a subpool is created to contain the reconciliation changes for Voyager to access and send to the LDAP gateway. The subpool is in ECSA and is generally a small, temporary staging area for reconciliation requests. If there is an outage, Voyager saves the encryped messages from the subpool to the //CACHESAV ddname in the Voyager STC. When Voyager is restarted and the subpool is rebuilt, the CACHESAV file is read and the messages are reloaded into the subpool. Once the LDAP gateway connects, the subpool data is sent to the LDAP.
- A CA Top Secret ACID profile is required to start both Voyager and Pioneer. An IBM type userid such as START2 or START2 can be used to perform this function. The Voyager Agent operates by using the Installation Exit, TSSINSTX, CA Top Secret. The IDF – TSSINSTX is passive. It does not change any z/OS storage. The storage area for collected CA Top Secret events is created by using STARTUP and only referenced by the TSSINSTX exit and is fully re-entrant.
- Once the TSSINSTX module is enabled either by using the TSS control file or Operator command, the TSS events are queued into the subpool. If STARTUP has not been executed, then these TSS events or messages will be lost. You can recover these TSS events ot messages by performing a full import reconciliation process.
- The TSSINSTX caching mechanism uses Storage Tokens and is safe. No operating system integrity can or will be lost with its usage. The storage is obtained by using the STORAGE OBTAIN macros and is in the ECSA. After storage is obtained, the storage token anchors are inserted. These are checked for by Voyager. If they are not present, then Voyager issues a message and shuts down.

Maintaining a specific password format is an example of the objective for which you use custom exits.

The IDF modified TSSINSTX has multiple exit points to capture CA Top Secret events. The exit points are:

- Pre-Init

- TSS command
- Post-Init
- New Password Verification
- Action (exit)
- Site Via RACHECK

When the exit is enabled, it will collect TSS events and cache them in a storage subpool. In addition, TSSINSTX calls IDFCACHE, which is the CA Top Secret caching module.

> **Note:**
>
> As the systems programmer, you must do an IPL after a system component is changed or modified.

Installing the Mainframe Agents

The CA Top Secret connector is shipped with a pair of agents, one for provisioning (Pioneer) and one for real-time reconciliation (Vogayer). If real-time reconciliation is not required, then install and start only the provisioning agent.

1. On the computer hosting the mainframe, extract the contents of the `TOPSECRET-<TIMESTAMP>-<VERSION>.zip` file located in the connector installation media.

   The following XMIT files are extracted:

   - `CLISTLIB.XMIT`
   - `JCLLIB.XMIT`
   - `LINKLIB.XMIT`
   - `PARMLIB.XMIT`
   - `PROCLIB.XMIT`

2. Transmit the extracted XMIT files to z/OS by using the following specifications:

   - `RECFM=FB`
   - `LRECL=80`
   - `BLKSIZE=3120`
   - `DSORG=PS`

   For example, you can use 3270 or FTP to transfer the files.

   The following datasets will exist on z/OS:

   - `<`*`HLQ`*`>.CLISTLIB.XMIT`
   - `<`*`HLQ`*`>.JCLLIB.XMIT`
   - `<`*`HLQ`*`>.LINKLIB.XMIT`
   - `<`*`HLQ`*`>.PARMLIB.XMIT`
   - `<`*`HLQ`*`>.PROCLIB.XMIT`

> **Note:**
>
> In the preceding list, *<HLQ>* is the high-level-qualifier used when transmitting the files to z/OS.

3. For each of the XMIT files that have been transmitted, execute the following command at the TSO prompt: `TSO RECEIVE INDA('<HLQ>.<FILE>.XMIT')` .

   When prompted to specify restore parameters, enter `DA('<HLQ>.<FILE>')` .

   For example, if the high-level qualifier is IDF and the file is `CLISTLIB.XMIT`, then execute the following command:

   `TSO RECEIVE INDA('IDF.CLISTLIB.XMIT')`
   When prompted, respond with: `DA('IDF.CLISTLIB')`

   The following datasets will exist on z/OS:

   - `<HLQ>.CLISTLIB`
   - `<HLQ>.JCLLIB`
   - `<HLQ>.LINKLIB`
   - `<HLQ>.PARMLIB`
   - `<HLQ>.PROCLIB`

   > **Note:**
   >
   > In the preceding list, *<HLQ>* is the high-level-qualifier used when receiving the previously transmitted files.

4. Edit each of the following installed job streams to replace any placeholders in them with actual values.

   - `<HLQ>.CLISTLIB.ENVINFO`
   - `<HLQ>.JCLLIB.CREATDSN`
   - `<HLQ>.JCLLIB.IEBCOPYL`
   - `<HLQ>.JCLLIB.IEBCOPYP`
   - `<HLQ>.JCLLIB.IEBCPYPR`
   - `<HLQ>.JCLLIB.KEYMODR`
   - `<HLQ>.PARMLIB.PROGID`

     > **Note:**
     >
     > In the preceding job stream, update the ++vol++ placeholder with the VOLUME from where you have received LINKLIB.

   - `<HLQ>.PROCLIB.PIONEER`
   - `<HLQ>.PROCLIB.STARTUP`

- <*HLQ*>.PROCLIB.STARTUP
- <*HLQ*>.PROCLIB.VOYAGER
- <*HLQ*>.PROCLIB.WRAPUP
- <*HLQ*>.JCLLIB.IEBCPYCL
- <*HLQ*>.JCLLIB.LOADDSN
- <*HLQ*>.JCLLIB.CREATEXP
- <*HLQ*>.JCLLIB.IDCAMSC
- <*HLQ*>.JCLLIB.IEBCPYRX
- <*HLQ*>.JCLLIB.PSAMCTL1
- <*HLQ*>.JCLLIB.REXXCL
- <*HLQ*>.JCLLIB.TOPSDEF
- <*HLQ*>.JCLLIB.TSSCFLE

> **Note:**
>
> In the preceding list, <*HLQ*> is the high-level-qualifier used when receiving the previously transmitted files.

The following table lists the installation placeholders found in job streams, their description, and example.

Table 4-2 Installation Placeholders

Placeholder	Description	Example
++hlq++	The high-level qualifier where the mainframe agent is to be installed. You must include all the multiple segments, if any.	IDF.PROD
++hlq1++	The top-most segment of the high-level qualifier where the mainframe agent is to be installed	IDF
++vol++	The volume where the mainframe agent is to be installed.	SDWRK1
++lpalib++	The DSN of the data set that contains customized lpalibs. Customize based on the z/OS environment.	USER.LPALIB
++parmdtr++	The name of the PARMLIB XMIT that was trasmitted to z/OS (without the .XMIT).	<*HLQ*>.PARMLIB
++parmlib++	The DSN of the data set that contains customized parmlibs. Customize based on z/OS environment.	USER.PARMLIB

Table 4-2 (Cont.) Installation Placeholders

Placeholder	Description	Example
++procdtr++	The name of the PROCLIB XMIT that was trasmitted to z/OS (without the .XMIT).	<HLQ>.PROCLIB
++proclib++	The DSN of the data set that contains customized proclibs. Customize based on z/OS environment.	USER.PROCLIB
++linkdtr++	The name of the LINKLIB XMIT that was trasmitted to z/OS (without the .XMIT).	<*HLQ*>.LINKLIB
++linklib++	The DSN where the LINKLIB XMIT that was received.	<*HLQ*>.LINKLIB
++rexxdtr++	The name of the CLISTLIB XMIT that was trasmitted to z/OS (without the .XMIT).	<*HLQ*>.CLISTLIB
++rexxlib++	The DSN where the CLISTLIB XMIT that was received.	<*HLQ*>.CLISTLIB
++pionprms++	The DSN of the control (configuration) file for the provisioning agent.	PIONEER.CONTROL.FILE
++voyprms++	The DSN of the control (configuration) file for the reconciliation agent.	VOYAGER.CONTROL.FILE
++pionlog++	The DSN of the control log (configuration) file for the LOGGERX feature of provisioning agent.	PIONEER.CONTROL.LOG
++voyglog++	The DSN of the control log (configuration) file for the LOGGERX feature of reconciliation agent.	VOYAGER.CONTROL.LOG
++pstcuserid++	The ACID of the user to be created for running the provisioning agent STC.	PIONEER
++vstcuserid++	The ACID of the user to be created for running the reconciliation agent STC.	VOYAGER
++pstcnm++	The name / description for the provisioning agent STC.	'PIONEER STARTED TASK'
++vstcnm++	The name / description for the reconciliation agent STC.	'VOYAGER STARTED TASK'
++pstcuid++	The OMVS UID assigned to the provisioning agent STC. Customize based on z/OS environment.	80
++vstcuid++	The OMVS UID assigned to the reconciliation agent STC. Customize based on z/OS environment.	90

Table 4-2 (Cont.) Installation Placeholders

Placeholder	Description	Example
`++stcgrp++`	The group assigned to the provisioning and reconciliation agent STCs. Ensure the group has UID(0) or BPX.SUPERUSER assigned. Customize based on z/OS environment.	`OMVSGRP`
`++secgrp++`	The Secure ID user default group.	`IDFSGRP`
`++secuid++`	The Secure ID user ACID.	`IDFAGNT`
`++secidnm++`	The Secure ID name.	`SECURE_ID`
`++cailink++`	The CA Linklist Library DSN. Customize based on Top Secret environment.	`CAI.CAKOLINK`

For example, in the following snippet from `CREATEDSN`, replace the placeholders `++hlq++` and `++vol++` with values such as `IDF.PROD` and `SDWRK1`:

```
//*
//S1       SET  PHLQ=++hlq++.PIONEER
//S2       SET  VHLQ=++hlq++.VOYAGER
//S3       SET  PVOL=++vol++
//S4       SET  VVOL=++vol++
//*
```

The following snippet displays the placeholders replaced with values:

```
//*
//S1       SET  PHLQ=IDF.PROD.PIONEER
//S2       SET  VHLQ=IDF.PROD.VOYAGER
//S3       SET  PVOL=SDWRK1
//S4       SET  VVOL=SDWRK1
//S5       SET THLQ=IDF.PROD
//*
```

5. Execute each of the following job streams in the order as shown in the following table to complete installation.

Table 4-3 Job Streams to Execute

Job Stream	Description
`<HLQ>.JCLLIB.IEBCOPYP`	Copies PARMLIB members to user PARMLIB.
`<HLQ>.JCLLIB.IEBCPYPR`	Copies PROCLIB members to user PROCLIB.
`<HLQ>.JCLLIB.IEBCPYCL`	Copies Rexx execs to user Rexx library.
`<HLQ>.JCLLIB.IEBCOPYL`	Copies exit routines to use LPA library.
`<HLQ>.JCLLIB.CREATDSN`	Allocates run time data sets, deleting the data sets first if they already exist.
`<HLQ>.JCLLIB.CREATEXP`	Allocates run time EXPORTIN data set.

Table 4-3 (Cont.) Job Streams to Execute

Job Stream	Description
`<HLQ>.JCLLIB.LOADDSN`	Copies PIONEER & VOYAGER configuration (control) files.
`<HLQ>.JCLLIB.TOPSDEL`	Deletes pre-existing users accounts and privileges on these user accounts required to execute agent STCs.
`<HLQ>.JCLLIB.TOPSDEF`	Defines users and permissions required to run the mainframe agent STCs.

The installation of the provisioning and reconciliation agents, Pioneer and Voyager, is complete. At this point, you can optionally remove the XMIT datasets that were originally transmitted to z/OS.

Configuring the Mainframe Agents

After installing Pioneer and Voyager, you must configure the mainframe agents to receive requests from and send responses to the LDAP gateway.

This section discusses the following topics:

- Configuring the Provisioning Agent
- Configuring the Reconciliation Agent

Configuring the Provisioning Agent

You must configure the provisioning agent to receive requests from the LDAP gateway, which originates from Oracle Identity Manager.

Edit the `<HLQ>.PIONEER.CONTROL.FILE` file to configure the behavior of the provisioning agent. Here, *`<HLQ>`* is the high-level-qualifier that you specified while installing the agents.

Table 4-4 Parameters of the Pioneer Control File

Parameter	Value	Description
TCPN	`TCPIP`	The name of the TCP/IP STC where the agent is executing.
IPAD	`0.0.0.0`	Do not change.
PORT	`9999`	The TCP/IP port that the agent will listen on.
CRLF	`Y` or `N`	If this flag is set to `Y`, then mainframe sends a response with carriage line feed. You must set the value of this parameter to `Y` for version 6+ of the LDAP Gateway. Set to `N` for version 5.
ESIZE	`16`	This is the only valid value. This parameter is for the AES128 encryption and decryption.

Table 4-4 (Cont.) Parameters of the Pioneer Control File

Parameter	Value	Description
POST_PROC_ALIAS	`T` or `F`	If you set the value of this parameter to `T`, then all LDAP **Alias** requests are processed. If you set it to `F`, then all LDAP **Alias** requests are rejected.
RWAIT	`0` or `999` (in seconds)	Enter the number of seconds the agent must wait before executing the jobs submitted by the batch recon.
JWAIT	`0` or `999` (in seconds)	Enter the number of seconds the agent must wait before executing the IDCAMS jobs.
QUEUE_DSN	`IDF.SEARCH`	Max 44 character DSN used with RWAIT for recons. This DSN does not need allocated or deleted.
EXPORT_MON	`NO` or `YES, REC=X`	Monitor XML imports displaying a message every X ACIDS.
IP	`V4` or `V6`	IP version to be used for communication between LDAP gateway and PIONEER agent. Value `V4` would be used as an IPv4 based IP address or hostname for communication. (e.g. In tops.properties value for *host*=192.168.100.0) Value `V6` would be used as an IPv6 based IP address or hostname for communication. (e.g. In tops.properties value for *host*=FE80::A0:A001:A0:A0A0%tap0) Default value is `V4`, when not specified in `<HLQ>.PIONEER.CONTROL.FILE`.
DEBUG	`Y` or `N`	This parameter is deprecated.
IDLEMSG	`Y` or `N`	This parameter is deprecated.
DEBUGOUT	`SYSOUT, CLASS(X)`	This parameter is deprecated.
SPIN_CLASS	`X`	This parameter is deprecated.
AUDIT_LOG	`YES` or `NO`	This parameter is deprecated.

Postprocessing Procedure for the Provisioning Agent

If the provisioning agent requires post processing for it to run, then you must add additional statements to the Pioneer control file as follows:

```
C=CREATE,M=TESTA,L=TEST.TESTA
C=ADDTO,M=TESTB,L=TEST.TESTB
C=REMOVE,M=TESTC,L=TEST.TESTC
Control file ( //PARMFLE ) explanations:
```

By default, the post-processing submits member (M=) from PDS library (L=) for every CREATE, ADDTO, REMOVE done on TSS. The post-processing takes place on every command added to the Pioneer control file. This library is dynamically allocated to Pioneer and later freed. If no post-processing is required, then do not code the C= for the TSS command. For example, `C=CREATE …… C=ADDTO ….`.

Pioneer post-processes the TSS commands received from the LDAP for CREATE and ADDTO. By default, the following parameters are passed to only a clist:

- CREATE - ACID
- ADDTO - ACID and KEYWORD
- REMOVE - ACID and KEYWORD

The REXX clist should have the following line to accept the parameters:

```
/* rexx sample clist */
Arg p1 p2
```

The Library specified with L= parameter and the member with M= parameter should contain batch JCL to execute REXX Clist.

The following is a sample job using the high-level qualifier of Pioneer:

```
//REXXCLST JOB SYSTEMS,MSGLEVEL=(1,1),MSGCLASS=X,CLASS=A,PRTY=8,
// NOTIFY=&SYSUID,REGION=0K
//STEP0 EXEC PGM=IKJEFT01,DYNAMNBR=50
//SYSTSPRT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPROC DD DISP=SHR,DSN=PIONEER.CLIST.LIBRARY
//SYSTERM DD DUMMY
//SYSTSIN DD *
/*
```

For postprocessing the commands (CREATE/ADDTO/REMOVE etc) mentioned in the control file, Pioneer adds: `%clistname P1 P2`

Where clistname is the value specified with M= parameter in the control file for the corresponding command.

> **Note:**
>
> The JCL member name specified with the M= parameter in the control file and the corresponding REXX/Clist member name needs to be the same.

Configuring the Reconciliation Agent

You must configure the reconciliation agent to send incremental responses to the LDAP gateway.

Edit the `<HLQ>.VOYAGER.CONTROL.FILE` file to configure the behavior of the reconciliation agent. `<HLQ>` is the high-level qualifier that you specified while installing the agents.

> **Note:**
>
> Voyager keeps reading supbool 231 for any reconciliation events and processes them. Therefore, for efficient use of allocated subpool 231, it is recommended to have Voyager up and running, failing which condition of event loss may occur.

Table 4-5 Parameters of the Voyager Control File

Parameter	Value	Description
TCPN	`TCPIP`	The name of the TCP/IP STC where the agent is executing.
IPAD	`999.999.999.999` or `ldap.example.com`	LDAP destination IP address or hostname (up to 40 characters).
PORT	`9999`	LDAP destination port that is listening to the incoming agent messages.
CRLF	`Y` or `N`	If this flag is set to `Y`, then mainframe sends a response with carriage line feed. You must set the value of this parameter to `Y` for version 6+ of the LDAP Gateway. Set to `N` for version 5.
ESIZE	`16`	This is the only valid value. This parameter is for the AES128 encryption and decryption.
CACHE_DELAY	`0` to `999`	This is the number of seconds that Voyager waits before issuing a write socket to the LDAP Gateway.
VOYAGER_ID	`VOYAGER`	This value will be included in the LDAP logs for diagnostic
CONNECT_RETRY	`999`	The number of times to retry when the LDAP connection is down.
CONNECT_INTV	`10`	The number of seconds between retries when the LDAP connection is down.

Table 4-5 (Cont.) Parameters of the Voyager Control File

Parameter	Value	Description
FAST_SHUTDOWN_NUM	Any 3-digit numeric value	A 3-digit numeric value representing a **batch**. **Note:** If you enter `0` or `1` as the value of this parameter, then this value is automatically defaulted to `100`. Voyager uses this 3-digit numeric value to process the records prior to checking operator's shutdown command. An explicit check for the shutdown command (/F VOYAGER,SHUTDOWN) is made only after processing each batch (of FAST_SHUTDOWN_NUM number of events). Whenever you enter the shutdown command, Voyager saves any remaining events (including events from subpool 231) to the disk ("cache save file") file for processing them later and shuts the process down. If there is no shutdown command, then Voyager processes the next 'batch' of events. Re-polling (reading from subpool 231) will continue to happen after all events are processed (when events are less than 100 or after processing 100 events each).
IP	`V4` or `V6`	IP version to be used for communication between LDAP gateway and VOYAGER agent. Value `V4` would be used as an IPv4 based IP address or hostname for communication. (e.g. IPAD entry in `VOYAGER.CONTROL.FILE` = 192.168.100.10) Value `V6` would be used as an IPv6 based IP address or hostname for communication. (e.g. IPAD entry in `VOYAGER.CONTROL.FILE` = fe80::74c3:eeff:fe1e:60fd) Default value is `V4`, when not specified in `<HLQ>.VOYAGER.CONTROL.FILE`.

Table 4-5 (Cont.) Parameters of the Voyager Control File

Parameter	Value	Description
PIONEER_DELETE_MSGS	Not applicable	The parameter is deprecated.
RECOVERY_INTERVAL	Not applicable	The parameter is deprecated.
DNS_RECOVERY_INTERVAL	Not applicable	The parameter is deprecated.
DEBUG	`Y` or `N`	This parameter is deprecated.
DEBUGOUT	`SYSOUT, CLASS (X)`	This parameter is deprecated.
CONNECT_MSGS	`Y` or `N`	This parameter is deprecated.
MSGID01	`NO` or `YES,IDMV602E,X`	This parameter is deprecated.

Configuring Logging

You can configure logging for both Pioneer and Voyager by editing the <HLQ>.PIONEER.CONTROL.LOG and <HLQ>.VOYAGER.CONTROL.LOG files, respectively, and setting values for various log parameters based on your requirement. For example, you can have complete control over the messages that you want to print or suppress and also the device over which the message must be printed. A separate control file is designed and used to control the functionality of logging through LOGGERX.

Logging Parameters

LOGGERX requires initial parameters setup for operating. This is achieved by using a control file (different from the control file for Pioneer). The parameters of this control file described in the following table.

Table 4-6 Logging Parameters

Parameter	Accepted Value	Description
LOGGERX_MSGID01	`NO` or `YES,IDMV602E,X`	If you want to suppress the IDMV602E recovery message, then set the value of this parameter to `NO`. If you want to display the IDMV602E recovery message, then set the value of the parameter to YES in the following format: `YES,IDMV602E,`*`X`* In this format, replace *X* with any number between `0` through `99`, which specifies the number of times the recovery message IDMV602E must be displayed. For example, `YES,IDMV602E,6`. **Note:** This parameter is applicable only to the <HLQ>.VOYAGER.CONTROL.LOG file.

Table 4-6 (Cont.) Logging Parameters

Parameter	Accepted Value	Description
LOGGERX_SYSOUT_CLASS	`A` through `Z`	The value in this parameter determines the class where the SYSOUT messages must be rolled to. For example, if you set the value of this parameter to `A`, then all SYSOUT messages will be directed to class A. If you do not specify a value for this parameter, then by default, all SYSOUT messages are rolled to class A.
LOGGERX_LEVEL_ROUTING	*MSG_TYPE:DEVICE* In this format, replace: • *MSG_TYPE* with types of messages such as `INFO`, `WARN`, `ERR`, or `DBG`. • *DEVICE* with any combination of `SYSOUT`, `CONSOLE`, `FILE`, or `NONE` by using a vertical bar (`|`) as the delimiter.	This parameter controls the message logging based on message type. The value of this parameter must contain the message type and the devices on which it is to be printed. For example, if you set the value of this parameter to `INFO:SYSOUT|CONSOLE`, then it means that all Informational messages will be written on to SPOOL/SYSOUT and the mainframe operator console. The same is applicable for message types – WARN(Warning), EROR(Error) and DEBG(DEBUGOUT).
LOGGERX_XXXX where XXXX can be either INFO, WARN, EROR, DEBG, AUDT, or PARM	`SYSOUT`	Use this parameter to specify SYSOUT when the value of *DEVICE* in the LOGGERX_LEVEL_ROUTING parameter is `FILE`. When the value is passed as SYSOUT, the file is created in the SPOOL as part of job output. For example, consider that the value of the LOGGERX_LEVEL_ROUTING parameter is set to `WARN:FILE`. In such a case, the entry `LOGFILE_WARN=SYSOUT` means that the job output will contain a file by the name WARNOUT that will contain warning messages.

Table 4-6 (Cont.) Logging Parameters

Parameter	Accepted Value	Description
LOGGERX_MSG_ROUTING	*MSGID:DEVICE* In this format, replace: • *MSGID* with the message ID corresponding to a message text. • *DEVICE* with any combination of `SYSOUT`, `CONSOLE`, `FILE`, or `NONE` by using the vertical bar (`|`) as the delimiter.	Use this parameter to redirect messages to a different device or suppress individual message based on message IDs. This parameter overrides the message levels set in the LOGGERX_LEVEL_ROUTING parameter. For example, the entries LOGGERX_MSG_ROUTING=IDFRPI001:NONE and LOGGERX_MSG_ROUTING=IDFRPI002:FILE combined with LOGGERX_LEVEL_ROUTING=INFO:CONSOLE mean that all Informational messages will go out on CONSOLE except, IDFRPI001(suppressed) and IDFRPI002(written on a file). You can provide 999 message IDs for each agent. In other words, you can choose to override, suppress, or redirect any number of messages. For a comprehensive list of message IDs and the corresponding message text, see Pioneer and Voyager Messages.
LOGGERX_FILE_MSG	`SYSOUT`	This parameter is used when `FILE` is specified as the Device type in the LOGGERX_MSG_ROUTING parameter to route all message ID- specific messages to MSGOUT in the spool. This parameter accepts a value of `SYSOUT`. When the value is passed as SYSOUT, the file is (MSGOUT) created in the SPOOL as part of job output. For example, the entry LOGFILE_MSG=SYSOUT means that the job output will contain a file by the name MSGOUT that contains messages corresponding to the message ID provided in the value of the LOGGERX_MSG_ROUTING parameter with the destination device as FILE.

Table 4-6 (Cont.) Logging Parameters

Parameter	Accepted Value	Description
LOGGERX_DEBUG	`Y` or `N`	This parameter is deprecated in v6.0.0 and later versions of the Mainframe agents.
LOGGERX_SPIN_CLASS	`X`	This parameter is deprecated in v6.0.0 and later versions of the Mainframe agents.
LOGGERX_AUDIT_LOG	`YES` or `NO`	This parameter is deprecated in v6.0.0 and later versions of the Mainframe agents.
LOGGERX_CONNECT_MSGS	`Y` or `N`	This parameter is deprecated in v6.0.0 and later versions of the Mainframe agents.

Important Use Case of the Log File

1. LOGGERX_LEVEL_ROUTING=INFO:FILE
   - LOGGERX_LEVEL_ROUTING=AUDT:FILE
   - LOGGERX_LEVEL_ROUTING=WARN:FILE
   - LOGGERX_LEVEL_ROUTING=ERR:FILE
   - LOGGERX_LEVEL_ROUTING=DBG:FILE
   - LOGGGERX_FILE_WARN=SYSOUT
   - LOGGGERX_FILE_INFO=SYSOUT
   - LOGGGERX_FILE_AUDT=SYSOUT
   - LOGGGERX_FILE_DEBG=SYSOUT
   - LOGGGERX_FILE_EROR=SYSOUT

   The above combination results in all INFO, AUDT, WARN, ERR, and DBG messages written onto INFOOUT, AUDOUT, WARNOUT, ERROROUT and DEBUGOUT, respectively, in spool/Sysout.

2. LOGGERX_LEVEL_ROUTING=INFO:FILE|SYSOUT
   - LOGGERX_LEVEL_ROUTING=AUDT:FILE|SYSOUT
   - LOGGERX_LEVEL_ROUTING=WARN:FILE|SYSOUT
   - LOGGERX_LEVEL_ROUTING=ERR:FILE|SYSOUT
   - LOGGERX_LEVEL_ROUTING=DBG:FILE|SYSOUT
   - LOGGGERX_FILE_WARN=SYSOUT
   - LOGGGERX_FILE_INFO= SYSOUT
   - LOGGGERX_FILE_AUDT=SYSOUT
   - LOGGGERX_FILE_DEBG= SYSOUT
   - LOGGGERX_FILE_EROR= SYSOUT

   The above combination results in all INFO, AUDT, WARN, ERR, DBG messages written onto INFOOUT, AUDOUT, WARNOUT, ERROROUT, and DEBUGOUT, respectively, in spool and all the messages will also be written onto SYSOUT file in job output.

3. LOGGERX_LEVEL_ROUTING=INFO:FILE|SYSOUT|CONSOLE
    - LOGGERX_LEVEL_ROUTING=AUDT:FILE|SYSOUT|CONSOLE
    - LOGGERX_LEVEL_ROUTING=WARN:FILE|SYSOUT|CONSOLE
    - LOGGERX_LEVEL_ROUTING=ERR:FILE|SYSOUT|CONSOLE
    - LOGGERX_LEVEL_ROUTING=DBG:FILE|SYSOUT|CONSOLE
    - LOGGGERX_FILE_WARN=SYSOUT
    - LOGGGERX_FILE_INFO= SYSOUT
    - LOGGGERX_FILE_AUDT= SYSOUT
    - LOGGGERX_FILE_DEBG= SYSOUT
    - LOGGGERX_FILE_EROR= SYSOUT

    The above combination results in all INFO, AUDT, WARN, ERR, and DBG messages written onto INFOOUT, AUDOUT, WARNOUT, ERROROUT and DEBUGOUT, respectively, in spool and all the messages will also be written onto SYSOUT file in job output and on the mainframe operator console.

4. LOGGERX_LEVEL_ROUTING=INFO:NONE|SYSOUT|CONSOLE
    - LOGGERX_LEVEL_ROUTING=AUDT:NONE
    - LOGGERX_LEVEL_ROUTING=WARN:NONE|SYSOUT|CONSOLE
    - LOGGERX_LEVEL_ROUTING=ERR:NONE|SYSOUT|CONSOLE
    - LOGGERX_LEVEL_ROUTING=DBG:NONE|SYSOUT|CONSOLE
    - LOGGGERX_FILE_WARN=SYSOUT
    - LOGGGERX_FILE_INFO= SYSOUT
    - LOGGGERX_FILE_AUDT=SYSOUT
    - LOGGGERX_FILE_DEBG= SYSOUT
    - LOGGGERX_FILE_EROR= SYSOUT
    - LOGGERX_MSG_ROUTING=IDMP000I :CONSOLE
    - LOGGERX_MSG_ROUTING=IDMP010I :CONSOLE
    - LOGGERX_MSG_ROUTING=IDMP300I :CONSOLE
    - LOGGERX_MSG_ROUTING=IDMP001E:CONSOLE

    The above combinations results in all INFO, AUDT, WARN, ERR, and DBG messages being suppressed. Since NONE is specified it does not matter if other devices are specified too, the messages will be suppressed. However, as LOGGERX_MSG_ROUTING is also specified, the messages IDs IDMP000I, IDMP010I, IDMP300I, and IDMP001E are not suppressed and are displayed on the CONSOLE. This establishes that at any point of time, the LOGGERX_MSG_ROUTING parameter has a higher priority in deciding the message’s output device, than its corresponding LEVEL ROUTING

Note:

In the sample control log files, for **Parm** message output, logging is routed based on message IDs IDMP400I, IDMP401E, and IDMV400I. These are set to route to 'SYSOUT' device and needs to maintain to get the PARMOUT dataset created in SPOOL.

Customizing the Reconciliation Exit (TSSINSTX)

Learn about working with custom reconciliation exit routines.

Note:

If you have made changes to the standard TSSINSTX exit routine provided by CA Top Secret, then you must perform the procedure described in this topic. Skip this topic if you are using the default TSSINSTX exit.

- Understanding the Sample Exit
- Calling Custom Exits

Understanding the Sample Exit

You can customize the default TSSINSTX exit to meet any special requirements in your environment.

The `<HLQ>.JCLLIB` dataset includes several sample files such as a sample reconciliation exit (TSSINSTX) and a custom exit (CUSTINSX). Use the CUSTINSX file to include your custom logic for the reconciliation exit. Use the sample reconciliation exit (TSSINSTX) to call the CUSTINSX file that includes your custom logic.

The source in the sample reconciliation exit (TSSINSTX) includes a call to IDMWORKS' modified version of TSSINSTX (IDFINSTX) in EXIT0 (before exiting from the TSSINSTX exit).

The following is the sample source as seen under the label 'EXIT0':

```
* IDMWORKS Modification to call Real-time exit
* Starts at Label EXIT0 for 9 lines
*
EXIT0 DS 0H COMMON EXIT POINT
*** CODE BELOW ADDED TO CALL IDFINSTX (IDMWORKS' TSSINSTX) ***
SLR R15,R15
LR R1,R9 COPY PARMLIST ADDR TO R1
LR R11,R13 COPY WORKAREA ADDR TO R11
LA R13,WORKAREA
L R15,=V(IDFINSTX) LOAD ADDR OF CUSTOMER EXIT
BALR R14,R15 CALL IT
LR R13,R11
*** CODE ABOVE ADDED TO CALL IDFINSTX (IDMWORKS' TSSINSTX) ***
```

In addition, the sample reconciliation exit (TSSINSTX) source contains the label `CUSTEXIT`, which calls the module CUSTINSX (your custom version of TSSINSTX). The call part of the sample code is commented by specifying `*` in column 1 on each row as shown below:

```
*** CODE BELOW ADDED TO CALL CUSTOMER'S MODIFIED EXIT ***
**** UNCOMMENT BELOW CODE TO CALL CUSTOMIZED EXIT(CUSTINSX) ****
*CUSTEXIT DS 0H
* LR R1,R9
* LR R11,R13
* LA R13,WORKAREA
* L R15,=V(CUSTINSX)
* BALR R14,R15
* LR R13,R11
* B EXIT0
**** UNCOMMENT ABOVE CODE TO CALL CUSTOMIZED EXIT(CUSTINSX) ****
*** CODE ABOVE ADDED TO CALL CUSTOMER'S MODIFIED EXIT ***
```

Uncomment the code to update it as per your requirements.

Calling Custom Exits

You can call a custom TSSINSTX exit, for example CUSTINSX, from an IDF supplied TSSINSTX exit.

To do so:

1. In a text editor, open the sample reconciliation exit (TSSINSTX) file for editing. This file is located in the `<`*`HLQ`*`>.JCLLIB` dataset.
2. Locate the CUSTEXIT section and remove `*` from column 1 to uncomment the code block as specified in the comments. Ensure that the CUSTEXIT label is starting at column 1.
3. Add or uncomment (by removing `*` from column 1) the branch instruction to label CUSTEXIT (B CUSTEXIT) from the EXIT entry points where you have your changes.

   For example, in the sample TSSINSTX exit, the "B CUSTEXIT" exit is commented under the PREINIT, POSTINIT, PASSWORD, and COMMAND exit entry-points as shown in below snippet for the PREINIT entry point:

   ```
   PREINIT DS 0H
   *---------------------------------------------------------------------*
   * Customer user code here
   *---------------------------------------------------------------------*
   * USER CODE GOES HERE TO INTERPRET INITIATION
   * THIS IS INVOKED PRIOR TO OBTAINING ACID SECURITY RECORD FROM TSS
   * ONLY JOBNAME, ACID, TERMINAL, PASSWORD(S), INSTDATA, MODE,
   * MAY BE CHANGED
   *** CODE BELOW TO CALL CUSTOM CODE (UN-COMMENT TO DEMO IT) ***
   *** REFER TO PDF ADMIN GUIDE (Documentation) BEFORE MAKING CHANGES ***
   * WTO 'TSSINSTX PREINIT',ROUTCDE=11
   * B CUSTEXIT
   *** CODE ABOVE TO CALL CUSTOM CODE (UN-COMMENT TO DEMO IT) ***
   B EXIT0
   EJECT
   ```

4. Refer to the sample CUSTINSX source supplied in the `<HLQ>.JCLLIB` dataset.
5. Add the custom code in the CUSTINSX exit specific to the desired entry points. You can set return code to greater than zero for any undesired command or function so that in TSSINSTX, when checked for return code from CUSTINSX, further processing can be skipped. Use the general purpose register R15 for storing the return code and populate it with a value to set the return code value in CUSTINSX.

   For example, the following sample sets the return code to 8:

   ```
   EXIT8 LA R15,8 Set Return code to 8
   ```
6. Assemble and link-edit the CUSTINSX and TSSINSTX files (supplied in the `<HLQ>.JCLLIB` dataset and with a call to CUSTINSX uncommented). Thus, TSSINSTX calls CUSTINSX (your custom version of TSSINSTX) followed by IDMWORKS' TSSINSTX (IDFINSTX). A sample JCL (ASMJCL) file to assemble and link-edit CUSTINSX and TSSINSTX is also supplied in `<HLQ>.JCLLIB`. Change placeholders such as `++linklib++` and `++hlq++` with the site-specific Load library and Source dataset high-level qualifier, respectively and SUBMIT the job. Ensure the job completes with MAX-CC of 0 or 4.
7. Activate the reconciliation exit as described in Activating Reconciliation Exits. While doing so, ensure to copy the fresh version of TSSINSTX located in the CA Top Secret loadlib, usually CAI.CAKOLINK, from the site-specific Load library specified in ASMJCL.

Activating and Deactivating Reconciliation Exits

To make use of real-time reconciliation and the reconciliation agent, you must activate system exits for capturing and reacting to changes in the target system.

- Activating Reconciliation Exits
- Deactivating Reconciliation Exits

Activating Reconciliation Exits

Real-time reconciliation requires the activation of the TSSINSTX exit. The TSSINSTX exit captures commands passively and then passes them to a caching module.

The TSSINSTX exit that is available in the installation Loadlib library works by calling the IDFINSTX module. Before you activate the exit, you must copy the TSSINSTX and IDFCACHE modules into the Loadlib that is available in the installation Linklist. Then, refresh the Linklist ensuring caution during times of high system activity.

To activate the reconciliation exit:

1. Copy the TSSINSTX and IDFCACHE modules from the `<HLQ>.LINKLIB` dataset into the CA Top Secret loadlib, usually `CAI.CAKOLINK`, that is available in the installation Linklist.
2. Refresh the Linklist as follows:
   a. Verify that the TSSINSTX exit is inactive by running the `F TSS,EXIT(OFF)` command from the z/OS operator interface.

      The `OKAY` response is displayed on the master console.

   b. From the z/OS Master console, run the `F LLA,REFRESH` command.

The `LIBRARY LOOKASIDE REFRESHED` message is displayed on the master console.

3. Run the `F TSS,EXIT(ON)` command and wait for the `OKAY` response to be displayed.

Deactivating Reconciliation Exits

Deactivate the system exits to disable the reconciliation of real-time changes to the target system.

To do so, run the following command from the z/OS operator interface:

```
F TSS,EXIT(OFF)
```

Operator Interface for Mainframe Agents

Both provisioning and reconciliation agents have an operator interface, and you can control the agents by passing commands through the interface.

The following topics are discussed in this section:

- Provisioning Agent Commands
- Reconciliation Agent Commands

Provisioning Agent Commands

Pass the Pioneer provisioning agent commands through the operator interface to control Pioneer.

Table 4-7 Provisioning Agent Commands

Command	Description
`T PROG=ID`	APF authorizes `<HLQ>.LINKLIB` - required to start the agent.
`S PIONEER`	Starts the agent.
`F PIONEER,SHUTDOWN`	Shuts down the agent.
`F PIONEER,STATUS`	Sends a status request to the agent.
`F PIONEER,DEBUG=Y`	Enables debug-level (detailed) log output.
`F PIONEER,DEBUG=N`	Disables debug-level (detailed) log output.

> **Note:**
>
> This interface through the z/OS modify command is a *single-threaded* system. Commands are queued and may take a few seconds before the agent acknowledges them.

Reconciliation Agent Commands

Pass the Voyager reconciliation agent through the operator interface to control Voyager.

Table 4-8 Reconciliation Agent Commands

Command	Description
`T PROG=ID`	APF authorizes `<HLQ>.LINKLIB` - *required to start the agent*.
`S STARTUP`	Allocates the subpool used to store reconciliation events - required for real-time reconciliation.
`F TSS,EXIT(ON)`	Activates system exits - required for real-time reconciliation as described in Activating Reconciliation Exits.
`S VOYAGER`	Starts the agent.
`F VOYAGER,SHUTDOWN`	Shuts down the agent.
`F VOYAGER,STATUS`	Sends a status request to the agent.
`F VOYAGER,DEBUG=Y`	Enables debug-level (detailed) log output.
`F VOYAGER,DEBUG=N`	Disables debug-level (detailed) log output.
`F VOYAGER,IPAD=999.999.999.999,PORT=9999`	Changes the IP address and port of the target LDAP Gateway.

Note:

The interface through the z/OS modify command is a *single-threaded* system. Commands are queued and take a few seconds before the agent acknowledges them.

5
Using the CA Top Secret Connector

You can use the CA Top Secret connector for performing reconciliation and provisioning operations after configuring it to meet your requirements.

The procedure to use the CA Top Secret Connector can be divided into the following topics:

- Guidelines on Using the Connector
- Scheduled Tasks for Lookup Field Synchronization
- Scheduled Task for Managing User's Access to Sources
- Configuring Reconciliation
- Configuring Account Status Reconciliation
- Configuring Reconciliation Jobs
- Performing Provisioning Operations

Guidelines on Using the Connector

These are the guidelines to apply while using the connector.

- The subpool and the LDAP Gateway must be started before starting the Reconciliation Agent. If the LDAP Gateway is not available when the Reconciliation Agent is started, then an error is generated with RETCODE=-01 and ERRORNO=61.
- The Top Secret connector LDAP gateway encrypts ASCII data transmitting the encrypted message to the mainframe. The mainframe decrypts this message, as the in bound message is in ASCII format, it is translated to EBCDIC for mainframe processing. As a result, any task that requires non-ASCII data transfer fails. In addition, there is no provision in the connector to indicate that the task has failed or that an error has occurred on the mainframe. To avoid errors of this type, you must exercise caution when providing inputs to the connector for the target system, especially when using a regional language interface. (See bug 18268599 for related information)
- Passwords used on the mainframe must conform to stringent rules related to passwords on mainframes. These passwords are also subject to restrictions imposed by corporate policies and rules about mainframe passwords. Keep in mind these requirements when you create or modify target system accounts through provisioning operations on Oracle Identity Manager.
- When using the connector with multiple LPAR(s), ensure to install both the Mainframe agents on each LPAR. The LDAP gateway can then be used to connect to multiple systems using a different Naming Context. See Configuring the LDAP Gateway for Multiple Installations of the Target System for more information.

Scheduled Tasks for Lookup Field Synchronization

The scheduled tasks for lookup field synchronization populate lookup tables with facility, dataset, group, or profiles IDs that can be assigned during the user provisioning process.

The following are the scheduled tasks for lookup field synchronization:

- Top Secret Find All Facilities
- Top Secret Find All Datasets
- Top Secret Find All Profiles
- Top Secret Find All Groups

When you configure these scheduled tasks, they run at specified intervals and fetch a listing of all facility, dataset, group, or profiles IDs on the target system for reconciliation. Table 5-1 describes the attributes of the scheduled task.

Table 5-1 Attributes of the Find All Facilities, Find All Datasets, Find All Profiles and Find All Groups Scheduled Tasks

Attribute	Description
IT Resource	Enter the name of the IT resource that was configured for the target system. Sample value: `TopSecretResource`
Resource Object	Enter the name of the resource object against which provisioning runs must be performed. Sample value: `OIMTopSecretResourceObject`
Lookup Code Name	Enter the name of the lookup code where OIM will store the results of the scheduled task. Sample value: `Lookup.profileNames` **Note:** The value supplied for the Lookup Code Name should match the value set in the properties of the Lookup Field in the corresponding Top Secret child table form.
Recon Type	Enter `Append` or `Replace`. This attribute determines whether the values from the target system will be appended to the current lookup, or replace the existing values in the lookup. If set to `Replace`, the existing lookup will be deleted. Sample value: `Replace`
SearchBaseDN	This parameter is available only in the Top Secret Find All Groups and Top Secret Find All Profiles scheduled tasks. Enter the container in which the search for groups and profile IDs must be performed during reconciliation and loaded into Oracle Identity Manager. Sample value: `ou=tops,ou=groups,dc=system,dc=backend` The preceding sample value implies that the connector loads groups from the LDAP system backend (dc=system, dc=backend) into Oracle Identity Manager. **Note:** If you do not enter a value for this attribute, then the connector loads groups into Oracle Identity Manager directly from the target system (by using the Pioneer route).

Table 5-1 (Cont.) Attributes of the Find All Facilities, Find All Datasets, Find All Profiles and Find All Groups Scheduled Tasks

Attribute	Description
AttrsToReturn	Enter a comma-separated list of object attributes that the connector must retrieve from LDAP. For example, enter a comma-separated list of group attributes that the connector must fetch from LDAP and load into Oracle Identity Manager. **Note:** The connector ignores this attribute if the SearchBaseDN attribute is empty. Also since groups are loaded into Oracle Identity Manager as a lookup, only two attributes are required. You must specify one for lookup value and one for lookup description. Sample value: `cn, displayname`
DescTemplate	By default, when lookup reconciliation is performed, the lookup description is same as the lookup value in the lookup window. Therefore, if required, use the DescTemplate attribute to specify the attribute whose value must be used as the lookup description and displayed in the lookup window. For example, consider that for one of the groups that is being fetched, the values of the `cn` and `displayName` attributes are `FINGRPIN` and `Finance Group in India`, respectively. Now suppose you set the value of the DescTemplate attribute to `cn`, then the lookup description that is displayed in the lookup window is `FINGRPIN`. However, if you set the value of the DescTemplate attribute to `displayName`, then the lookup description is `Finance Group in India`. If the lookup description has to be a combination of multiple attributes values, then enter multiple attribute names separated by a space character. For example, enter `{{cn}} {{displayname}}`. **Note:** This attribute value will be ignored if the value of the SearchBaseDN attribute is empty.
R2	Enter whether the version of Oracle Identity Manager in use is 11.1.2.*x*. Sample value: `true`

Scheduled Task for Managing User's Access to Sources

The Lookup.SourceNames lookup definition is created in Oracle Identity Manager when you deploy the connector and is used to add and remove a user's access to a source on the mainframe.

This connector includes a scheduled task to automatically populate the lookup field used for storing Top Secret source IDs.

> **Note:**
>
> The Find All Sources scheduled task does not query the target system for data. Instead, the scheduled task automatically populates the lookup field with "itResourceKey~sourceName" pairs based on the IT Resource and Find All Sources scheduled task property values.

Scheduled Tasks for Lookup Field Synchronization describes the properties of the Find All Sources scheduled task.

Table 5-2 Attributes of the Find All Sources Scheduled Task for CA Top Secret

Attribute	Description
IT Resource	Enter the name of the IT resource that was configured for the target system. Sample value: `TopSecretResource`
Resource Object	Enter the name of the resource object against which provisioning runs must be performed. Sample value: `OIMTopSecretResourceObject`
Sources List	Enter a comma-separated list of Top Secret sources. Sample value: `TSO,R5`
Lookup Code Name	Enter the name of the lookup code where Oracle Identity Manager will store the source entries. Sample value: `Lookup.SourceNames`
Recon Type	Enter `Append` or `Replace`. This attribute determines whether "IT resource key~sourceName" pairs will be appended to the current lookup, or replace the existing values in the lookup. If set to `Replace`, the existing lookup will be deleted. Sample value: `Replace`
R2	Enter whether the version of Oracle Identity Manager in use is 11.1.2.*x*. Sample value: `true`

However, you can also manually add additional values. To add additional sources for provisioning and reconciliation perform the following steps:

1. Log in to Oracle Identity Manager Design Console.
2. Expand **Administration** and then double-click **Lookup Definition**.
3. Search for the **Lookup.SourceNames** lookup definition, and then click **Add**.
4. In the Code Key column, enter the name of the source. Enter the same value in the Decode column. The following is a sample entry:
   - Code Key: R5
   - Decode: R5
5. Click the **Save** icon.

Configuring Reconciliation

The CA Top Secret Advanced connector supports both incremental reconciliation (sometimes referred to as real-time reconciliation) and full reconciliation. This section discusses the following topics related to configuring reconciliation:

- Performing Full Reconciliation
- Reconciliation Scheduled Tasks
- Guidelines for Configuring Filtered Reconciliation to Multiple Resource Objects

Performing Full Reconciliation

Full reconciliation involves reconciling all existing user records from the target system into Oracle Identity Manager.

After you deploy the connector, you must first perform full reconciliation. When you run the Connector Installer, a scheduled job for user reconciliation (Top Secret Reconcile All Users) is automatically created in Oracle Identity Manager.

To perform full reconciliation, you must run the Top Secret Reconcile All Users scheduled job. See Top Secret Reconcile All Users for information about the scheduled job attributes.

Reconciliation Scheduled Tasks

When you run the Connector Installer, these reconciliation scheduled tasks are automatically created in Oracle Identity Manager.

- Top Secret Reconcile All Users
- Top Secret Reconcile Deleted Users to OIM
- Top Secret Reconcile Users to Internal LDAP
- Top Secret Reconcile LDAP Users to OIM

Top Secret Reconcile All Users

Use the Top Secret Reconcile All Users scheduled task to reconcile user data in the target resource (account management) mode of the connector. This scheduled task runs at specified intervals and fetches create or modify events on the target system for reconciliation.

Table 5-3 describes the attributes of the scheduled task.

Table 5-3 Attributes of the Top Secret Reconcile All Users Scheduled Task

Attribute	Description
IT Resource	Enter the name of the IT resource that was configured for the target system. Sample value: `TopSecretResource`

Table 5-3 (Cont.) Attributes of the Top Secret Reconcile All Users Scheduled Task

Attribute	Description
filter	Enter a filter criteria to search for and retrieve user records that match the given filter criteria. You can use any target system attribute to create the filter criterion. The filter criterion that you enter must be a valid filter according to RFC2254. The filter can be either simple or complex. A simple filter uses only a single attribute whereas a complex filter is a combination of two or more attributes. Sample value for a simple filter: `(revoke=n)` Sample value for a complex filter: `(|(commandflag=UPDATE)(deptacid=OMVSDEPT))` This complex filter searches for and retrieves all user records whose commandflag attribute value is `UPDATE` or department ACID is `OMVSDEPT`. **Note:** If you specify a complex filter, then ensure that you have enabled the caching layer of the LDAP Gateway as described in Understanding the Caching Layer. If the caching layer is disabled, then the connector considers only the simple filter `(uid=<userid>)`.
Resource Object	Enter the name of the resource object against which reconciliation runs must be performed. Sample value: `OIMTopSecretResourceObject`
MultiAttrsWithoutITRKey	Enter a comma-separated list of multivalued attributes that connector must reconcile without the *ITRKEY~* prefix.*ITRKEY~* is the numeric code assigned to each IT resource in Oracle Identity Manager. Sample value: `adminresouce, admindataset`
MultiValuedAttributes	Enter a comma-separated list of multivalued attributes that you want to reconcile. Do not include a space after each comma. Sample value: `profiles,sources,groupIds,facilities`
SingleValueAttributes	Enter a comma-separated list of single-valued attributes that you want to reconcile. Do not include a space after each comma. Do not include attributes already listed in the MultiValueAttributes field. Sample value: `uid,owner,defaultGroup,waddr1,tsoMaxSize` **Note**: By default, Oracle Identity Manager's design form only allows entering up to 150 characters in a text field. To increase this limit, change the value of the TSA_VALUE column in Oracle Identity Manager database.
UID Case	Enter either `upper` or `lower` to specify the case for the UID attribute.
UsersList	Enter a comma-separated list of UIDs that you want to reconcile from the target system. If this property is left blank, all users on the target system will be reconciled. Sample value: `userQA01,georgeb,marthaj,RST0354`
R2	Enter whether the version of Oracle Identity Manager in use is 11.1.2.*x*. Sample value: `true`
Secondary IT resource	If you created a secondary IT resource for reconciliation or provisioning, then enter its name.

Top Secret Reconcile Deleted Users to OIM

The Top Secret Reconcile Deleted Users to OIM scheduled task is used to reconcile data about deleted users in the target resource (account management) mode of the connector.

When you configure this scheduled task, it runs at specified intervals and fetches a list of users on the target system. These user names are then compared with provisioned users in Oracle Identity Manager. Any user profiles that exist within Oracle Identity Manager, but not in the target system, are deleted from Oracle Identity Manager.

Table 5-4 describes the attributes of the scheduled task.

Table 5-4 Attributes of the Top Secret Reconcile Deleted Users to Oracle Identity Manager Scheduled Task

Attribute	Description
IT Resource	Enter the name of the IT resource that was configured for the target system. Sample value:`TopSecretResource`
Resource Object	Enter the name of the resource object against which the delete reconciliation runs must be performed. Sample value: `OIMTopSecretResourceObject`
Recon Matching Rule Attributes	Enter a comma-separated list of attributes used in the matching rule. If the IT resource is used, enter "IT". Sample value: `UID,IT`
UID Case	Enter either `upper` or `lower` to specify the case for the UID attribute.

Top Secret Reconcile Users to Internal LDAP

The Top Secret Reconcile Users to Internal LDAP scheduled task is used to process the CFILE extract from the target system to the internal LDAP store. When you configure this scheduled task, it runs at specified intervals and fetches a list of users and their profiles on the target system. Each of these users is then reconciled to the internal LDAP store. No reconciliation to Oracle Identity Manager is performed.

Table 5-5 describes the attributes of the scheduled task.

Table 5-5 Attributes of the Top Secret Reconcile Users to Internal LDAP Scheduled Task

Attribute	Description
IT Resource	Enter the name of the IT resource that was configured for the target system. Sample value: `TopSecretResource`
Domain OU	Enter the name of the internally-configured directory in the LDAP internal store where the contents of event changes will be stored.Sample value: `tops`

Top Secret Reconcile LDAP Users to OIM

Use the Top Secret Reconcile LDAP Users to OIM scheduled task to reconcile users from the internal LDAP store to Oracle Identity Manager. When you configure this scheduled task, it runs at specified intervals and fetches a list of users within the internal LDAP store and reconciles these users to Oracle Identity Manager.

When changes occur on the target system, the Voyager agent passes the change event to the LDAPv3 Virtual Directory, and each event is stored in the internal "meta" directory. This can be used to run normal LDAPv3 searches by filtering the lastModificationDate and commandFlag LDAP attributes. The commandFlag LDAP attribute is used internally by Topsecret Reconcile All LDAP Users task to perform delete reconciliation from the backend.

Table 5-6 describes the attributes of the scheduled task.

Table 5-6 Attributes of the Top Secret Reconcile LDAP Users to OIM Scheduled Task

Attribute	Description
IT Resource	Enter the name of the IT resource that was configured for the target system. Sample value: `TopSecretResource`
Resource Object	Enter the name of the resource object against which the delete reconciliation runs must be performed. Sample value: `OIMTopSecretResourceObject`
Domain OU	Enter the name of the internally-configured directory in the LDAP internal store where the contents of event changes will be stored.Sample value: `tops`
filter	Enter a filter criteria to search for and retrieve user records that match the given filter criteria. You can use any target system attribute to create the filter criterion. The filter criterion that you enter must be a valid filter according to RFC2254. The filter can be either simple or complex. A simple filter uses only a single attribute whereas a complex filter is a combination of two or more attributes. Sample value for a simple filter: `(revoke=n)` Sample value for a complex filter: `(&(commandflag=UPDATE)(attributes=ASUSPEND))` This complex filter searches for and retrieves all user records whose commandflag attribute value is `UPDATE` and attribute is `ASUSPEND`.
MultiValuedAttributes	Enter a comma-separated list of multivalued attributes that you want to reconcile. Do not include a space after each comma. Sample value: `profiles,sources,facilities,groupIds`
MultiAttrsWithoutITRKey	Enter a comma-separated list of multivalued attributes that connector must reconcile without the *ITRKEY~* prefix.*ITRKEY~* is the numeric code assigned to each IT resource in Oracle Identity Manager. Sample value: `adminresouce, admindataset`

Table 5-6 (Cont.) Attributes of the Top Secret Reconcile LDAP Users to OIM Scheduled Task

Attribute	Description
SingleValueAttributes	Enter a comma-separated list of single-valued attributes that you want to reconcile. Do not include a space after each comma. Do not include attributes already listed in the MultiValueAttributes field. Sample value: `uid,owner,defaultGroup,waddr1,tsoMaxSize` **Note:** By default, Oracle Identity Manager's design form only allows entering up to 150 characters in a text field. To increase this limit, change the value of the TSA_VALUE column in the Oracle Identity Manager database.
LDAP Time Zone	Enter the full timezone database name value. Do not use the abbreviated timezone value. To find out the timezone database value refer to List of tz database time zones. Sample value: `America/New York`
UID Case	Enter `upper` or `lower` to specify whether the user ID must be displayed in uppercase or lowercase.
R2	If you are using Oracle Identity Manager release 11.1.2.*x*, then enter `true`. Otherwise, enter `false`.
Secondary IT resouce	If you created a secondary IT resource for reconciliation or provisioning, then enter its name.

Guidelines for Configuring Filtered Reconciliation to Multiple Resource Objects

Some organizations use multiple resource objects to represent multiple user types in their system. The Resource Object property of the Top Secret Reconcile All Users scheduled task is used to specify the resource object used during reconciliation, and you can enter more than one resource object in the value of the Resource Object attribute. Further, you can include CA Top Secret attribute-value pairs to filter records for each resource object.

> **See Also:**
>
> Top Secret Reconcile All Users for information about the Top Secret Reconcile All Users scheduled task

The following is a sample format of the value for the Resource Object attribute:

```
(ATTRIBUTE1:VALUE1)RESOURCE_OBJECT1,RESOURCE_OBJECT2
```

As shown by RESOURCE_OBJECT2 in the sample format, specifying a filter attribute is optional, but if more than one resource object is specified, you must specify a filter for each additional resource object. If you do not specify a filter attribute, then all records are reconciled to the first resource object in the list. Further, the filters are checked in order, so the resource object without a filter attribute should be included last in the list.

Filter attributes should be surrounded by parentheses.

Apply the following guidelines while specifying a value for the Resource Object attribute:

- The names of the resource objects must be the same as the names that you specified while creating the resource objects in the Oracle Identity Manager Design Console.
- The CA Top Secret attribute names must be the same as the names used in the LDAP Gateway configuration files.
- The value must be a regular expression as defined in the java.util.regex Java package. Note that the find() API call of the regex matcher is used rather than the matches() API call. This means that a substring matching rule can be specified in the pattern, rather than requiring the entire string matching rule.

  Further, substring matching is case-sensitive. A "(tso)" filter will not match a user with the user ID "TSOUSER1".
- Multiple values can be matched. Use a vertical bar (|) for a separator as shown in the following example:

  (*ATTRIBUTE*:*VALUE1*|*VALUE2*|*VALUE3*)*RESOURCE_OBJECT*
- Multiple filters can be applied to the attribute and to the same resource object. For example:

  (*ATTRIBUTE1:VALUE1*)&(*ATTRIBUTE2*:*VALUE2*)*RESOURCE_OBJECT*

The following is a sample value for the Resource Object attribute:

```
(tsoProc:X)TSSR01,(instdata:value1|value2|value3)TopSecretResourceObject2,
(tso)TopSecretResourceObject24000,Resource
```

In this sample value:

- (tsoProc:X)TSSRO1 represents a user with X as the attribute value for the TSO Proc segment. Records that meet this criterion are reconciled with the TSSRO1 resource object.
- (instdata:value1|value2|value3)TopSecretResourceObject2 represents a user with value1, value2, or value3 as their INSTDATA attribute value. Records that meet this criterion are reconciled with the TopSecretResourceObject2 resource object.
- (tso)TopSecretResourceObject24000 represents a user with TSO privileges. A TSO attribute value is not specified. Records that meet this criterion are reconciled with the TopSecretResourceObject24000 resource object.
- All other records are reconciled with the resource object.

Configuring Account Status Reconciliation

When a user is disabled or enabled on the target system, the status of the user can be reconciled into Oracle Identity Manager.

> **Note:**
>
> This section describes an optional procedure. Perform this procedure only if you want reconciliation of user status changes on CA Top Secret.

To configure reconciliation of user status changes made on CA Top Secret:

1. In the *LDAP_INSTALL_DIR*/VOYAGER_ID.properties file, add the Status attribute to the reconAttrs property.
2. If using scheduled task reconciliation, in the Top Secret Reconcile All Users scheduled task, add the Status attribute to the SingleValueAttributes property list.
3. In the Design Console:
   - In the **OIMTopSecretResourceObject** resource object, create a reconciliation field to represent the Status attribute.
   - In the **OIMTopsProvisioningProcess** process definition, map the field for the Status field to the OIM_OBJECT_STATUS field.

Scheduled Tasks for CA Top Secret Connector

Table Table 5-7 lists the scheduled tasks that you must configure.

Table 5-7 Scheduled Tasks for Lookup Field Synchronization and Reconciliation for CA Top Secret

Scheduled Task	Description
Top Secret Find All Groups	This scheduled task is used to synchronize the values of group lookup fields between Oracle Identity Manager and the target system. For information about this scheduled task and its attributes, see Scheduled Tasks for Lookup Field Synchronization.
TopSecret Find All Facilities	This scheduled task is used to synchronize the values of facilities lookup fields between Oracle Identity Manager and the target system. For information about this scheduled task and its attributes, see Scheduled Tasks for Lookup Field Synchronization.
Top Secret Find All Datasets	This scheduled task is used to synchronize the values of dataset lookup fields between Oracle Identity Manager and the target system. For information about this scheduled task and its attributes, see Scheduled Tasks for Lookup Field Synchronization.
Top Secret Find All Profiles	This scheduled task is used to synchronize the values of profiles lookup fields between Oracle Identity Manager and the target system. For information about this scheduled task and its attributes, see Scheduled Tasks for Lookup Field Synchronization.

Table 5-7 (Cont.) Scheduled Tasks for Lookup Field Synchronization and Reconciliation for CA Top Secret

Scheduled Task	Description
Top Secret Find All Sources	This scheduled task is used to synchronize the values of source lookup fields in Oracle Identity Manager. For information about this scheduled task and its attributes, see Scheduled Tasks for Lookup Field Synchronization.
Top Secret Reconcile All Users	This scheduled task is used to fetch user data during target resource reconciliation. For information about this scheduled task and its attributes, see Top Secret Reconcile All Users.
Top Secret Reconcile Deleted Users to OIM	This scheduled task is used to fetch data about deleted users during target resource reconciliation. During a reconciliation run, for each deleted user account on the target system, the Top Secret User resource is revoked for the corresponding OIM User. For information about this scheduled task and its attributes, see Top Secret Reconcile Deleted Users to OIM.
Top Secret Reconcile Users to Internal LDAP	This scheduled task is used to reconcile users from the target system to the internal LDAP store. For information about this scheduled task and its attributes, see Top Secret Reconcile Users to Internal LDAP.
Top Secret Reconcile All LDAP Users	This scheduled task is used to reconcile users from the internal LDAP store to Oracle Identity Manager. For information about this scheduled task and its attributes, see Top Secret Reconcile LDAP Users to OIM.

Configuring Reconciliation Jobs

Configure reconciliation jobs to perform reconciliation runs that check for new information on your target system periodically and replicates the data in Oracle Identity Governance.

You can apply this procedure to configure the reconciliation jobs for users and entitlements.

To configure a reconciliation job:

1. Log in to Identity System Administration.
2. In the left pane, under System Management, click **Scheduler**.
3. Search for and open the scheduled job as follows:
   a. In the Search field, enter the name of the scheduled job as the search criterion. Alternatively, you can click **Advanced Search** and specify the search criterion.
   b. In the search results table on the left pane, click the scheduled job in the Job Name column.
4. On the Job Details tab, you can modify the parameters of the scheduled task:
   - **Retries**: Enter an integer value in this field. This number represents the number of times the scheduler tries to start the job before assigning the Stopped status to the job.
   - **Schedule Type**: Depending on the frequency at which you want the job to run, select the appropriate schedule type. See Creating Jobs in *Oracle Fusion Middleware Administering Oracle Identity Governance*.

   In addition to modifying the job details, you can enable or disable a job.

5. On the **Job Details** tab, in the Parameters region, specify values for the attributes of the scheduled task.

   > **Note:**
   >
   > Values (either default or user-defined) must be assigned to all the attributes. If even a single attribute value is left empty, then reconciliation is not performed.

6. Click **Apply** to save the changes.

   > **Note:**
   >
   > You can use the Scheduler Status page in Identity System Administration to either start, stop, or reinitialize the scheduler.

Performing Provisioning Operations

You create a new user in Identity Self Service by using the Create User page. You provision or request for accounts on the Accounts tab of the User Details page.

To perform provisioning operations in Oracle Identity Governance:

1. Log in to Identity Self Service.
2. Create a user as follows:
   a. In Identity Self Service, click **Manage**. The Home tab displays the different Manage option. Click **Users**. The Manage Users page is displayed.
   b. From the Actions menu, select **Create**. Alternatively, you can click **Create** on the toolbar. The Create User page is displayed with input fields for user profile attributes.
   c. Enter details of the user in the Create User page.
3. On the Account tab, click **Request Accounts**.
4. In the Catalog page, search for and add to cart the application instance for the connector that you configured earlier, and then click **Checkout**.
5. Specify value for fields in the application form and then click **Ready to Submit**.
6. Click **Submit**.

> **See Also:**
>
> Creating a User in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance* for details about the fields on the Create User page

6

Extending the Functionality of the CA Top Secret Connector

The following optional procedures that you can perform to extend the functionality of the connector for addressing your business requirements can be divided into the following stages:

- Adding Custom Fields for Target Resource Reconciliation
- Adding Custom Multivalued Fields for Reconciliation
- Adding Custom Fields for Provisioning
- Removing Attributes Mapped for Target Resource Reconciliation
- Adding Admin Multivalued Attributes for Provisioning and Reconciliation
- Configuring the Connector for Provisioning to Multiple Installations of the Target System
- Configuring the Generation of Single-Use Passwords for the Reset Password Operation
- Customizing Log File Locations
- Handling Pioneer Error Messaging Exceptions in the Gateway

Adding Custom Fields for Target Resource Reconciliation

> **Note:**
>
> You must ensure that new attributes you add for reconciliation contain only string-format data. Binary attributes must not be brought into Oracle Identity Manager natively.

By default, the attributes listed in Table 1-4 are mapped for reconciliation between Oracle Identity Manager and the target system. If required, you can add new attributes for target resource reconciliation.

To add a custom field for reconciliation, you must first update the connector reconciliation component you are using, and then update Oracle Identity Manager. This section discusses the following topics:

- Adding Custom Fields for Full Reconciliation for CA Top Secret Connector
- Adding Custom Fields to Oracle Identity Manager

Adding Custom Fields for Full Reconciliation for CA Top Secret Connector

You can add custom fields for full reconciliation by specifying a value for the SingleValueAttributes attribute of the Top Secret Reconcile All Users scheduled task. See Performing Full Reconciliation for more information.

To add a custom field for scheduled task reconciliation:

1. Log in to Oracle Identity System Administration.
2. In the left pane, under System Management, click **Scheduler.**
3. Search for and open the **Top Secret Reconcile All Users** scheduled task as follows:
   a. In the left pane, in the Search field, enter `Top Secret Reconcile All Users` as the search criterion. Alternatively, you can click **Advanced Search** and specify the search criterion.
   b. In the search results table on the left pane, click the scheduled job in the Job Name column.
4. Add the custom field to the list of attributes in the `SingleValueAttributes` scheduled task attribute.
5. Click **Apply.**

Adding Custom Fields to Oracle Identity Manager

After adding the custom field to either the VOYAGER_ID.properties file (if using real-time reconciliation) or the Top Secret Reconcile All users scheduled task (if using scheduled task reconciliation), you must add the custom field to the Oracle Identity Manager components.

To update Oracle Identity Manager with the custom field:

1. Log in to the Oracle Identity Manager Design Console.
2. Add the custom field to the list of reconciliation fields in the resource object as follows:
   a. Expand **Resource Management** and then double-click **Resource Objects.**
   b. Search for and open the **OIMTopSecretResourceObject** resource object.
   c. On the Object Reconciliation tab, click **Add Field.**
   d. In the Add Reconciliation Field dialog box, enter the details of the field.

      For example, if you are adding a Top Secret attribute called "Description", then enter `Description` in the Field Name field and select **String** from the Field Type list.
   e. Click **Save** and close the dialog box.
   f. Click **Create Reconciliation Profile.** This copies changes made to the resource object into MDS.
   g. Click **Save.**
3. Add the custom field on the process form as follows:
   a. Expand **Development Tools** and then double-click **Form Designer.**
   b. Search for and open the **UD_IDF_TOPS** process form.
   c. Click **Create New Version,** and then click **Add.**
   d. Enter the details of the field.

For example, if you are adding the Description field, then enter `UD_IDF_TOPS_DESCRIPTION` in the Name field, and then enter the rest of the details of this field.

   e. Click **Save** and then click **Make Version Active.**

4. Create a reconciliation field mapping for the custom field in the provisioning process as follows:

   a. Expand **Process Management** and then double-click **Process Definition.**

   b. Search for and open the **OIMTopsProvisioningProcess** process definition.

   c. On the Reconciliation Field Mappings tab of the provisioning process, click **Add Field Map.**

   d. In the Add Reconciliation Field Mapping dialog box, from the Field Name field, select the value for the field that you want to add.For example, from the Field Name field, select **Description.**

   e. Double-click the **Process Data field**, and then select **UD_IDF_TOPS_DESCRIPTION.**

   f. Click **Save** and close the dialog box.

   g. Click **Save.**

5. If you are using Oracle Identity Manager release 11.1.2.*x*, then create a new UI form and attach it to the application instance to make this new attribute visible. See Creating a New UI Form and Updating an Existing Application Instance with a New Form for the procedures.

6. If you are adding a custom attribute or custom dataset, then set values for the _configAttrs_, _configDNames and _configDatasets_ properties in the tops.properties file.

Adding Custom Multivalued Fields for Reconciliation

To add a custom multivalued field to reconciliation, you must first update the IDF reconciliation component you are using, and then update Oracle Identity Manager.

- Adding Custom Multivalued Fields for Full Reconciliation
- Adding Custom Multivalued Fields to Oracle Identity Manager for CA Top Secret Connector

Adding Custom Multivalued Fields for Full Reconciliation

You can add custom multivalued fields for full reconciliation by specifying a value for the multiValuedAttributes property of the Top Secret Reconcile All Users reconciliation scheduled task. See Top Secret Reconcile All Users for more information.

To add a custom field for scheduled task reconciliation:

1. Log in to Oracle Identity System Administration.
2. In the left pane, under System Management, click **Scheduler.**
3. Search for and open the **Top Secret Reconcile All Users** as follows:

a. On the left pane, in the Search field, enter **Top Secret Reconcile All Users** as the search criterion. Alternatively, you can click **Advanced Search** and specify the search criterion.

b. In the search results table on the left pane, click the **scheduled job** in the Job Name column.

4. Add the custom field to the list of attributes in the `MultiValuedAttributes property.`

5. Click **Apply.**

Adding Custom Multivalued Fields to Oracle Identity Manager for CA Top Secret Connector

After adding the custom multivalued field to either the VOYAGER_ID.properties file (if using real-time reconciliation) or the Top Secret Reconcile All users scheduled task (if using scheduled task reconciliation), you must add the custom multivalued field to the Oracle Identity Manager components.To update Oracle Identity Manager with the multivalued field:

1. Log in to the Oracle Identity Manager Design Console.

2. Create a form for the multivalued field as follows:

   a. Expand **Development Tools** and double-click **Form Designer.**

   b. Create a form by specifying a table name and description, and then click **Save.**

   c. Click **Add** and enter the details of the field.

   d. Click **Save** and then click **Make Version Active.** Figure 6-1 shows the multivalued field added on a new form.

Figure 6-1 Multivalued Field Added on a New Form for CA Top Secret Connector

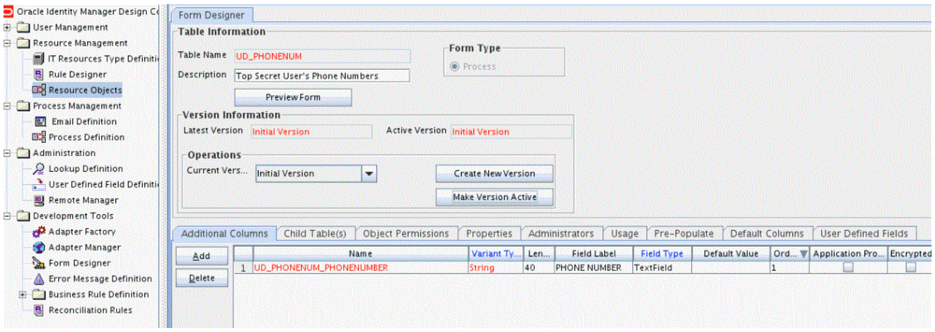


Table Information

Table Name: UD_PHONENUM

Description: Top Secret User's Phone Numbers

Form Type: ● Process

Version Information

Latest Version: Initial Version

Active Version: Initial Version

Operations

Current Vers...: Initial Version

	Name	Variant Ty...	Len...	Field Label	Field Type	Default Value	Ord...	Application Pro...	Encrypted
1	UD_PHONENUM_PHONENUMBER	String	40	PHONE NUMBER	TextField		1	□	□

3. Add the form created for the multivalued field as a child form of the process form as follows:

   a. Search for and open the **UD_IDF_TOPS** process form.

   b. Click **Create New Version.**

   c. Click the **Child Table(s) tab.**

   d. Click **Assign.**

e. In the Assign Child Tables dialog box, select the newly created child form, click the **right arrow,** and then click **OK.**

f. Click **Save** and then click **Make Version Active.** Figure 6-2 shows the child form added to the process form.

Figure 6-2 Child Form Added to the Process Form for CA Top Secret Connector

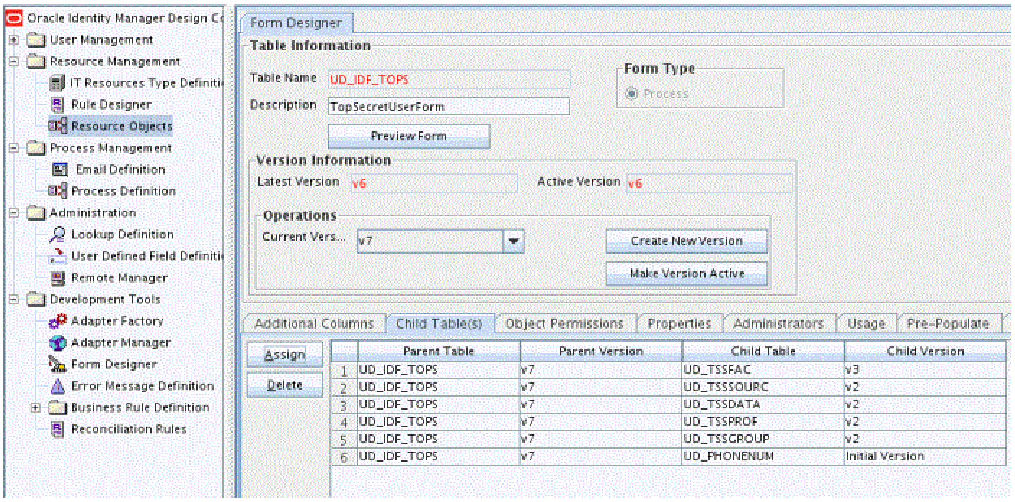


Table Information

Table Name: UD_IDF_TOPS

Description: TopSecretUserForm

Form Type: ● Process

Version Information

Latest Version: v6

Active Version: v6

Operations

Current Vers...: v7

	Parent Table	Parent Version	Child Table	Child Version
1	UD_IDF_TOPS	v7	UD_TSSFAC	v3
2	UD_IDF_TOPS	v7	UD_TSSSOURC	v2
3	UD_IDF_TOPS	v7	UD_TSSDATA	v2
4	UD_IDF_TOPS	v7	UD_TSSPROF	v2
5	UD_IDF_TOPS	v7	UD_TSSGROUP	v2
6	UD_IDF_TOPS	v7	UD_PHONENUM	Initial Version

4. Add the new multivalued field to the list of reconciliation fields in the resource object as follows:

a. Expand **Resource Management** and then double-click **Resource Objects.**

b. Search for and open the **OIMTopsResourceObject** resource object.

c. On the Object Reconciliation tab, click **Add Field.**

d. In the Add Reconciliation Field dialog box, enter the details of the field.

For example, enter `phoneNumber` in the Field Name field and select **Multi-Valued Attribute** from the Field Type list.

e. Click **Save** and close the dialog box.

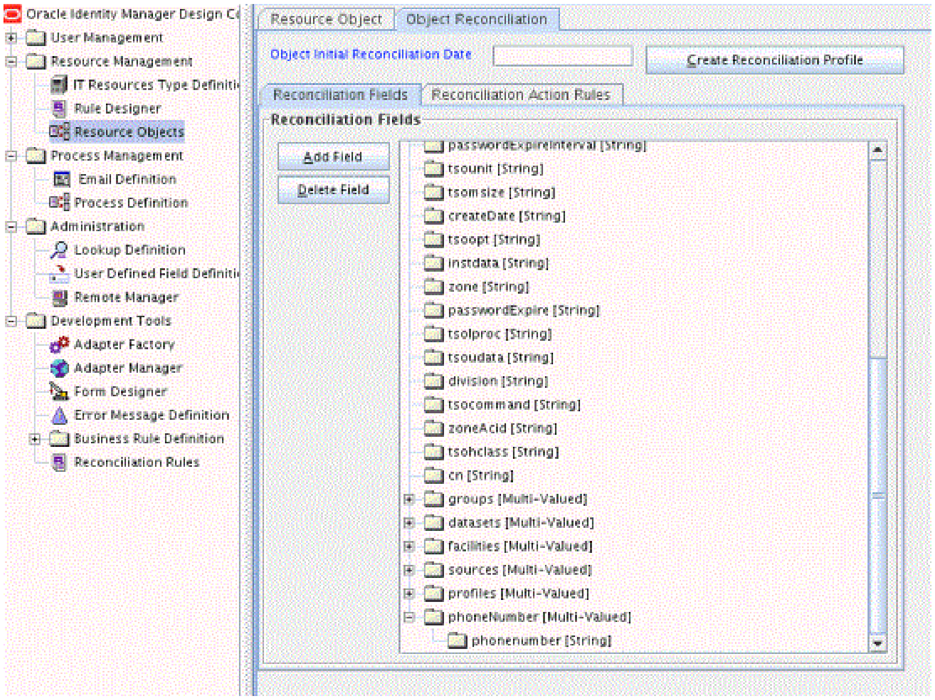
f. Right-click the newly created field and select **Define Property Fields.**

g. In the Add Reconciliation Fields dialog box, enter the details of the newly created field.

For example, enter `phonenumber` in the Field Name field and select **String** from the Field Type list.

h. Click **Save** and then close the dialog box. Figure 6-3 shows the new reconciliation field added in the resource object.

Figure 6-3   New Reconciliation Field Added in the Resource Object for CA Top Secret Connector



i. Click **Create Reconciliation Profile.** This copies changes made to the resource object into MDS.

5. Create an entry for the field in the `AtMap.Tops` lookup definition, as follows:

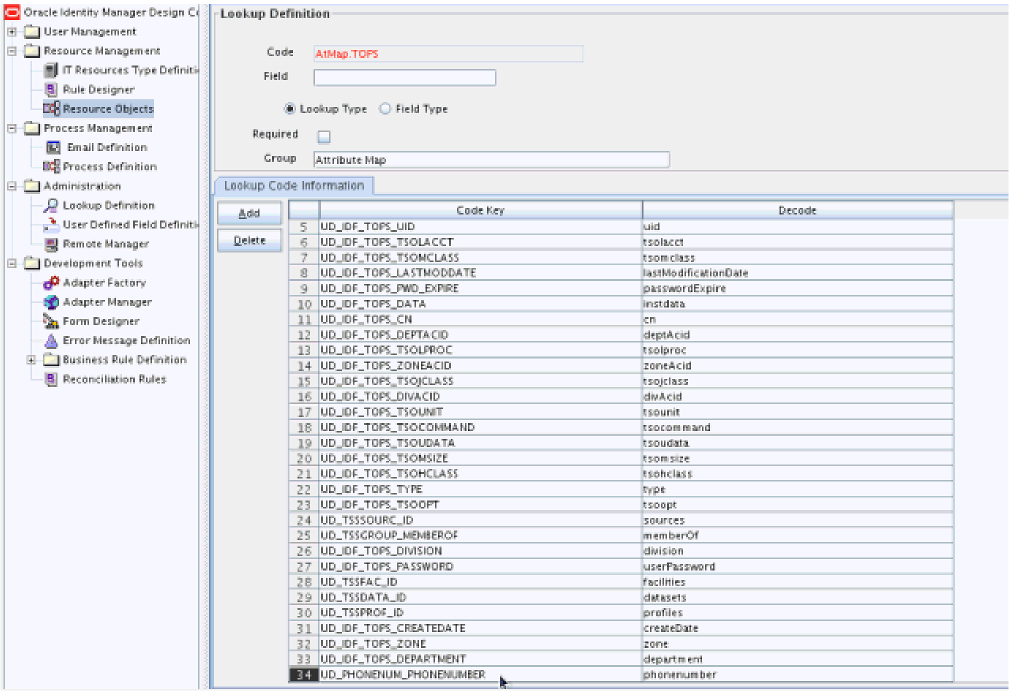
   a. Expand **Administration** and then double-click **Lookup Definition.**

   b. Search for the **AtMap.TOPS** lookup definition.

   c. Click **Add** and enter the Code Key and decode values for the field. The Code Key value is the name of the process form field that you created for the multivalued custom field in Step 3.3.d. The Decode value is the name of the target system field.

      For example, enter `UD_PHONENUM_PHONENUMBER` in the Code Key field and then enter `phonenumber` in the Decode field. Figure 6-4 shows the lookup code added to the lookup definition.

Figure 6-4   Entry Added in the Lookup Definition for CA Top Secret Connector



Code: AtMap.TOPS

Field:

● Lookup Type ○ Field Type

Required □

Group: Attribute Map

	Code Key	Decode
5	UD_IDF_TOPS_UID	uid
6	UD_IDF_TOPS_TSOLACCT	tsolacct
7	UD_IDF_TOPS_TSOMCLASS	tsomclass
8	UD_IDF_TOPS_LASTMODDATE	lastModificationDate
9	UD_IDF_TOPS_PWD_EXPIRE	passwordExpire
10	UD_IDF_TOPS_DATA	instdata
11	UD_IDF_TOPS_CN	cn
12	UD_IDF_TOPS_DEPTACID	deptAcid
13	UD_IDF_TOPS_TSOLPROC	tsolproc
14	UD_IDF_TOPS_ZONEACID	zoneAcid
15	UD_IDF_TOPS_TSOJCLASS	tsojclass
16	UD_IDF_TOPS_DIVACID	divAcid
17	UD_IDF_TOPS_TSOUNIT	tsounit
18	UD_IDF_TOPS_TSOCOMMAND	tsocommand
19	UD_IDF_TOPS_TSOUDATA	tsoudata
20	UD_IDF_TOPS_TSOMSIZE	tsomsize
21	UD_IDF_TOPS_TSOHCLASS	tsohclass
22	UD_IDF_TOPS_TYPE	type
23	UD_IDF_TOPS_TSOOPT	tsoopt
24	UD_TSSSOURC_ID	sources
25	UD_TSSGROUP_MEMBEROF	memberOf
26	UD_IDF_TOPS_DIVISION	division
27	UD_IDF_TOPS_PASSWORD	userPassword
28	UD_TSSFAC_ID	facilities
29	UD_TSSDATA_ID	datasets
30	UD_TSSPROF_ID	profiles
31	UD_IDF_TOPS_CREATEDATE	createDate
32	UD_IDF_TOPS_ZONE	zone
33	UD_IDF_TOPS_DEPARTMENT	department
34	UD_PHONENUM_PHONENUMBER	phonenumber

d. Click **Save.**

6. Create a reconciliation field mapping for the new multivalued field as follows:

   a. Expand **Process Management** and then double-click **Process Definition.**

   b. Search for and open the **OIMTopsProvisioningProcess** process definition.

   c. On the Reconciliation Field Mappings tab of the provisioning process, click **Add Table Map.**

   d. In the Add Reconciliation Table Mapping dialog box, select the **field name** and table name from the list, click **Save,** and then close the dialog box.

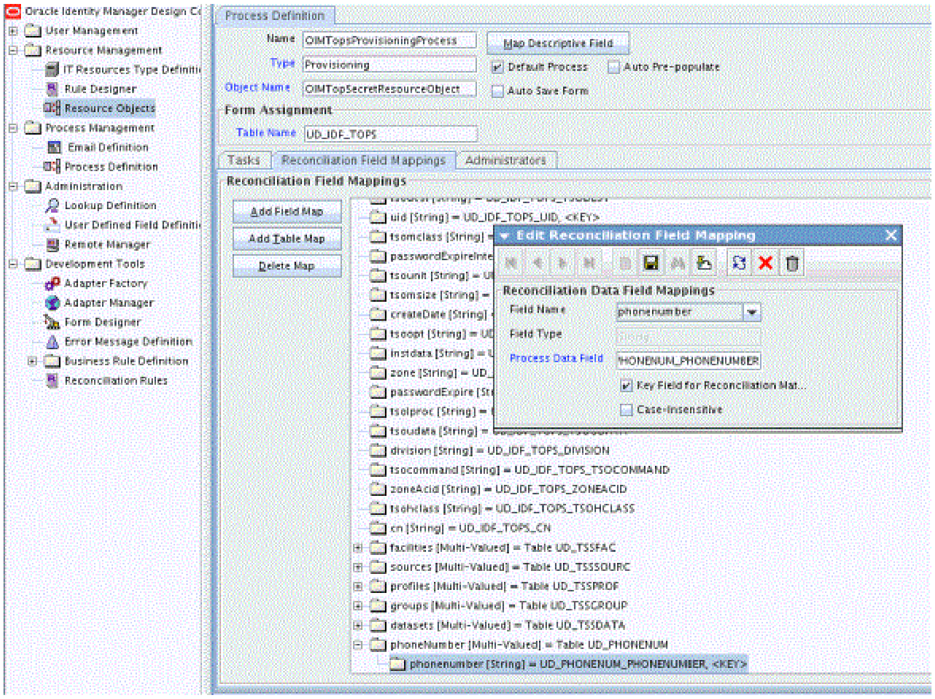
   e. Right-click the newly created field and select **Define Property Field Map.**

   f. In the Field Name field, select the value for the field that you want to add.

   g. Double-click the **Process Data** field, and then select **UD_PHONENUM_PHONENUMBER.**

   h. Select **Key Field** for Reconciliation Field Matching and click **Save.** Figure 6-5 shows the new reconciliation field mapped to a process data field in the process definition.

Figure 6-5 New Reconciliation Field Mapped to a Process Data Field for CA Top Secret Connector



Name: OIMTopsProvisioningProcess

Type: Provisioning

Object Name: OIMTopSecretResourceObject

☑ Default Process □ Auto Pre-populate

□ Auto Save Form

Form Assignment

Table Name: UD_IDF_TOPS

Reconciliation Field Mappings

- uid [String] = UD_IDF_TOPS_UID, <KEY>
- tsomclass [String] =
- passwordExpireInte
- tsounit [String] = UI
- tsomsize [String] =
- createDate [String]
- tsoopt [String] = UI
- instdata [String] = U
- zone [String] = UD_
- passwordExpire [St
- tsolproc [String] =
- tsoudata [String] =
- division [String] = UD_IDF_TOPS_DIVISION
- tsocommand [String] = UD_IDF_TOPS_TSOCOMMAND
- zoneAcid [String] = UD_IDF_TOPS_ZONEACID
- tsohclass [String] = UD_IDF_TOPS_TSOHCLASS
- cn [String] = UD_IDF_TOPS_CN
- facilities [Multi-Valued] = Table UD_TSSFAC
- sources [Multi-Valued] = Table UD_TSSSOURC
- profiles [Multi-Valued] = Table UD_TSSPROF
- groups [Multi-Valued] = Table UD_TSSGROUP
- datasets [Multi-Valued] = Table UD_TSSDATA
- phoneNumber [Multi-Valued] = Table UD_PHONENUM
  - phonenumber [String] = UD_PHONENUM_PHONENUMBER, <KEY>

Reconciliation Data Field Mappings

Field Name: phonenumber

Field Type: String

Process Data Field: HONENUM_PHONENUMBER

☑ Key Field for Reconciliation Mat...

□ Case-Insensitive

Adding Custom Fields for Provisioning

By default, the attributes listed in Table 1-4 are mapped for provisioning between Oracle Identity Manager and the target system. If required, you can map additional attributes for provisioning.

The connector does not support the use of custom attributes in CREATE USER operations that is, TSS CREATE. Instead, custom attribute modifications should be sent in an MODIFY USER operation that is, TSS ADDTO/REPLACE/REMOVE after the user has been provisioned a resource.

To add a new attribute for provisioning:

1. Log in to the Oracle Identity Manager Design Console.
2. Add the new attribute on the process form as follows:

   If you have added the field on the process form by performing Step 4 of Adding Custom Fields to Oracle Identity Manager, then you need not add the field again. If you have not added the field, then:

   a. Expand **Development Tools.**
   b. Double-click **Form Designer.**
   c. Search for and open the **UD_IDF_TOPS** process form.
   d. Click **Create New Version,** and then click **Add.**
   e. Enter the details of the attribute.

For example, if you are adding the Description field, enter **UD_IDF_TOPS_DESCRIPTION** in the Name field, and then enter the rest of the details of this field.

f. Click **Save** and then click **Make Version Active.**

> **Note:**
>
> OMVS and NETVIEW attributes must not be added to the AtMap.TOPS lookup definition as they are not supported for create provisioning operations.

3. To enable update of the attribute during provisioning operations, create a process task as follows:

a. Expand **Process Management,** and double-click **Process Definition.**

b. Search for and open the **OIMTopsProvisioningProcess** process definition.

c. Click **Add.**

d. On the General tab of the Creating New Task dialog box, enter a name and description for the task and then select the following:

Conditional

Required for Completion

Disable Manual Insert

Allow Cancellation while Pending

Allow Multiple Instances

e. Click **Save.**

f. Go to the Integration tab and click **Add.**

g. In the Handler Selection dialog box, select **Adapter**, click **adpMODIFYTOPSUSER**, and then click the Save icon.

The list of adapter variables is displayed on the Integration tab.

h. To create the mapping for the first adapter variable:

Double-click the number of the first row.

In the Edit Data Mapping for Variable dialog box, enter the following values:

Variable Name: `Adapter return value`

Data Type: `Object`

Map To: `Response code`

Click the Save icon.

i. To create mappings for the remaining adapter variables, use the data given in the following table:

Table 6-1 Values for the Variables, Map To, Qualifier, and Literal Value lists for each variable

Variable Number	Variable Name	Map To	Qualifier
Second	idfResource	Process Data	LDAP_SERVER
Third	uid	Process Data	LoginId
Fourth	attrName	String Literal	Enter the LDAP attribute name in the Literal Value field. Example: description Table 1-5 for a list of unmapped user attributes and their LDAP Gateway attribute names.
Fifth	attrValue	Process Data	Select the process form field from the drop-down list.Example: DESCRIPTION

j. On the Responses task, click **Add** to add at least the SUCCESS response code, with status C. This ensures that if the custom task is successfully run, then the status of task is displayed as Completed in Oracle Identity Manager.

k. Click the Save icon in the Editing Task dialog box, and then close the dialog box.

l. Click the Save icon to save changes to the process definition.

4. Create a new UI form and attach it to the application instance to make this new attribute visible. See Creating a New UI Form and Section Updating an Existing Application Instance with a New Form for the procedures.

Removing Attributes Mapped for Target Resource Reconciliation

The reconAttrs property contains the list of target system attributes that are mapped for real-time reconciliation. This property is found in the VOYAGER_ID.properties file. If you want to remove an attribute mapped for real-time reconciliation, then remove it from the reconAttrs property.

The SingleValueAttributes and MultiValuedAttributes properties contain the list of target system attributes that are mapped for scheduled task reconciliation. These properties are found in the Top Secret Reconcile All Users and Top Secret Reconcile All LDAP Users scheduled tasks. If you want to remove an attribute mapped for scheduled task reconciliation, then remove it from the SingleValueAttributes or MultiValuedAttributes property.

Adding Admin Multivalued Attributes for Provisioning and Reconciliation

To manage fine grained-administrative privileges to a user from Oracle Identity Manager, you can add Admin attributes for provisioning and reconciliation.

To grant or revoke admin privileges to a user:

1. Log in to the Oracle Identity Manager Design Console.
2. Create a child form for the admin authority field, (for example, RESOURCE) as follows:
    1. Expand **Development Tools** and double-click **Form Designer**.
    2. Create a form by specifying a table name and description, and then click **Save**.
    3. Click **Add** and enter the details of the field.
    4. Click **Save** and then click **Make Version Active**.
    5. Repeat Step 2.a through 2.d for each administrative authority whose grant or removal must be managed through Oracle Identity Manager.
3. Assign the form created for the admin authority field, (for example, RESOURCE) as a child form of the parent process form as follows:
    1. Search for and open the **UD_IDF_TOPS** process form, and then click **Create New Version**.
    2. Click the **Child Table(s)** tab and then click **Assign**.
    3. In the Assign Child Tables dialog box, select the newly created child form, click the right arrow, and then click **OK**.
    4. Click **Save** and then click **Make Version Active**.
4. Add the new admin authority field (for example, RESOURCE) to the list of reconciliation fields in the resource object as follows:
    1. Expand **Resource Management** and then double-click **Resource Objects**.
    2. Search for and open the **OIMTopSecretResourceObject** resource object. Then, on the **Object Reconciliation** tab, click **Add Field**.
    3. In the Add Reconciliation Fields dialog box, enter the details of the field.

       For example, enter `adminresource` in the **Field Name** field and select **Multi-Valued Attribute** from the Field Type list.
    4. Click **Save** and then close the dialog box.
    5. Right-click the newly created field and select **Define Property Fields**.
    6. In the Add Reconciliation Fields dialog box, enter the details of the field.

       For example, enter `adminresource` in the **Field Name** field and select **String** from the Field Type list.
    7. Click **Save**, and then close the dialog box.
    8. Click **Create Reconciliation Profile**. This copies changes made to the resource object into the MDS.
5. Create an entry for the field in the **AtMap.Tops** lookup definition, as follows:
    1. Expand **Administration** and then double-click **Lookup Definition**.
    2. Search for and open the **AtMap.Tops** lookup definition.
    3. Click **Add** and enter the Code Key and Decode values for the admin authority field. The Code Key value is the name of the process form field that you created Step 2. The Decode value is the name of the target system field.

       For example, enter `UD_ADM_RES_RESOURCE` in the Code Key field and then enter `adminResource` in the Decode field. Figure 5–4 shows the lookup code added to the lookup definition. (INSERT lookup_with_admin_field.png)

6. Create a reconciliation field mapping for the new field as follows:
   a. Expand **Process Management** and then double-click **Process Definition**.
   b. Search for and open the **OIMTopsProvisioningProcess** process definition. Then, on the Reconciliation Field Mappings tab, click **Add Table Map**.
   c. In the Add Reconciliation Table Mapping dialog box, select the field name (for example, adminResource) and table name (for example, UD_ADM_RES) from the list, click **Save**, and then close the dialog box.
   d. Right-click the newly created table map, and select **Define Property Field Map**.
   e. In the Field Name field, select the value for the field that you want to add. Then double-click the **Process Data Field** field and select the field name (for example, UD_ADM_RES_RESOURCE).
   f. Click **Save** and then close the dialog box.
7. Enable update provisioning operations on the admin authority field as follows:
   a. Expand **Process Management**, and then double-click **Process Definition**.
   b. Search for and open the **OIMTopsProvisioningProcess** process definition.
   c. Click **Add** and enter the task name and description. For example, enter `Administrative Authority Resource Granted` as the task name and `Grant Admin Authority Resource with Access to User as RESOURCE|ACCESS1,ACCESS2` as the task description.
   d. In the Task Properties section, select **Conditional**, **Disable Manual Insert**, **Required for Completion**, **Allow Cancellation while Pending**, and **Off-line**. In addition, from the Child Table list, select **UD_ADM_RES_RESOURCE** and from the Trigger Type list, select **insert**. <<INSERT create_new_task_adm_auth.png>>
   e. On the Integration tab, click **Add**. In the Handler Selection dialog box, select Adapter. Then from the list of adapter, select **adpMODIFYUSERATTRTOPS**, click Save and close the dialog box.
   f. In the Adapter Variables region, click the **paramValue** variable. In the dialog box that is displayed, create the following mapping, click **Save** and close the dialog box:

      Variable Name: paramValue

      Map To: Process Data

      Qualifier: adminResource
   g. Repeat Step 7.f for the remaining variables listed in the Adapter Variables region. The following table lists values that you must select from the Map To, Qualifier, and Literal Value lists for each variable:

Variable	Map To	Qualifier	Literal Value
idfserver	Process Data	LDAP_SERVER	NA
opType	Literal	Integer	1

Variable	Map To	Qualifier	Literal Value
containsITResource	Literal	Boolean	Select **True** if the attribute value is expected to contain *~IT_RESOURCE_KEY* as the prefix in your environment. Otherwise, select **False**.
auditInfo	Literal	String	Audit comment
uid	Process Data	USER_ID	NA
paramName	Literal	String	adminresource
Adapter return value	Response Code	NA	NA

   h. Click the **Save** icon, close the dialog box, and then save the process definition.

8. Add a new process task by using the adpMODIFYUSERATTRTOPS adapter for removing the admin authority. To do so, repeat Step 7 with the following difference:

   While performing Step 7.d, instead of selecting **Insert** from the Trigger Type list, select **Delete**.

9. Save the process task.

10. Replicate all changes made to the Form Designer of the Design Console in a new UI form as follows:

    a. Log in to Oracle Identity System Administration.

    b. Create and activate a sandbox.

    c. Create a new UI form to view the newly added field along with the rest of the fields.

    d. Associate the newly created UI form with the application instance of your target system. To do so, open the existing application instance for your resource, from the Form field, select the form (created in Step 10.c), and then save the application instance.

    e. Publish the sandbox.

Configuring the Connector for Provisioning to Multiple Installations of the Target System

You can configure the connector for multiple installations of the target system. You can also configure the connector for a scenario in which multiple logical partitions (LPARs), which are not associated with the first LPAR, are configured in the target system.

For each installation of the target system, you create an IT resource and configure an additional instance of the LDAP Gateway.

To configure the connector for the second installation of the target system:

> **Note:**
>
> Perform the same procedure for all installations of the target system.

1. Create an IT resource based on the OIMLDAPGatewayResourceType IT resource type.
2. Copy the current *LDAP_INSTALL_DIR* directory, including all the subdirectories, to a new location on the Oracle Identity Manager computer.

   > **Note:**
   >
   > In the remaining steps of this procedure, *LDAP_INSTALL_DIR* refers to the newly copied directory.

3. Extract the contents of the *LDAP_INSTALL_DIR*/dist/idfserver.jar file.
4. In the beans.xml file, change the value of the port in the <property name="port" value="xxxx"/> line to specify a port that is different from the port used for the first instance of the LDAP Gateway. The default port number is shown in the following example:

   ```
   <bean id="listener" class="com.identityforge.idfserver.nio.Listener">
   <constructor-arg><ref bean="bus"/></constructor-arg>
   <property name="admin"><value>false</value></property>
   <property name="config"><value>../conf/listener.xml</value></property>
   <property name="port" value="5389"/>
   </bean>
   ```

   When you change the port number, you must make the same change in the value of the idfServerPort parameter of the IT resource that you create by performing Step 1.
5. Save and close the beans.xml file.
6. Open the *LDAP_INSTALL_DIR*/conf/tops.properties file and set values for the following parameters:
   - _host_= Enter the IP address or host name of the mainframe.
   - _port_= Enter the port number for the second instance of the Provisioning agent.
   - _agentPort_= Enter the port number for the second instance of the Reconciliation agent.

     > **Note:**
     >
     > The value of the _agentPort_ parameter must not be the same as that of the first instance if a second LPAR, which is not associated with the first LPAR, is configured in the target system. This value can be the same as the value of the idfServerPort parameter if you have two mainframe servers with CA Top Secret running on each server.

7. Save and close the tops.properties file.
8. Delete the *LDAP_INSTALL_DIR*/etc/VOYAGER_ID.properties file.
9. In a Linux or Solaris environment, if there are not enough socket file descriptors to open up all the ports needed for the server, then:

a. In a text editor, open the run script from the *LDAP_INSTALL_DIR*/bin directory.

b. Add the following line in the file:

```
-Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider
```

c. Save and close the file.

> **Note:**
>
> When you perform provisioning in Identity Self Service, you can specify the IT resource corresponding to the CA Top Secret installation to which you want to provision the user.

Configuring the Generation of Single-Use Passwords for the Reset Password Operation

You can create and configure an adapter that generates single-use passwords when the Reset Password operation is performed.

To create the adapter:

1. Use the Adapter Factory to create a copy of the ResetPassword adapter.
2. Add the following variables to the adapter that you create:

   passwordExpire: boolean or String

   passwordExpireInterval: String
3. The idm.jar file is located in the JavaTasks directory. When you create and map the new adapter task, use the following functions defined in this file:
   - public String resetPassword(String idfUserId, String idfNewPwd, boolean expire, String expireInDays)
   - public String resetPassword(String idfUserId, String idfNewPwd, String expireNow, String expireInDays)

   In these functions, the expire and expireNow parameters expect the value `true` to expire users' passwords.
4. Compile the adapter.
5. Create a process task, and associate it with the object corresponding to the event for which you want single-use passwords to be generated. For example, you can associate the process task with the Password Updated task or with the event that the PWD_EXP check box on the process form is selected.

Customizing Log File Locations

The name and log locations of the main LDAP gateway log file (idfserver.log) and the CFILE XML error log file (idf.xml.error.log) can be modified by adding additional arguments to the LDAP gateway server STARTUP command. These arguments are optional, and you can include one, both, or neither in the STARTUP command.

1. In a text editor, open the run script from the *LDAP_INSTALL_DIR*/bin directory. This run script is used to start and stop the LDAP gateway.
   - If using a Windows system, open the run.bat file.
   - If using a UNIX system, open the run.sh file.
2. Add the arguments to the start command, located at the end of the run script:

   Add the arguments after the `-cp %CLASSPATH%` argument.

   To modify the idfserver.log path, use the argument `-Didf.logpath=`

   To modify the idf.xml.error.log path, use the argument `-Didf.xmllogpath=`

   In the following example, the start command will set the idfserver.log path to `C:/logs/ldap/idfserver.log` and the idf.xml.error.log path to `C:/logs/errors/idf.xml.error.log`:

   ```
   %JAVACMD% %DEBUG% %JVM_OPTS% %SECURE% -cp %CLASSPATH% -Didf.logpath="c:/logs/
   ldap/idfserver.log" -Didf.xmllogpath="c:/logs/errors/idf.xml.error.log" -
   Djava.library.path=%HOME%/lib com.identityforge.idfserver.Main %1 %2 %3 %4
   %5 %6 %7 %8 %9
   ```

Handling Pioneer Error Messaging Exceptions in the Gateway

The error handling routines let you configure what error messages to look for when deciding that a request sent to Pioneer has succeeded or failed. Use these instructions to configure error handling.

Enable or Disable the Ability to Examine the Pioneer SAF Code

Some commands will return SAF codes whenever a command fails.

To enable the ability to automatically throw an error whenever codes greater than 0 are returned, add the `check-return-codes` property to the tops.properties file (created in Setting Connection Properties) and set its value to `yes`.

> **Note:**
>
> Warning codes may also show up as codes greater than 0 depending on the type of mainframe environment that you are using. Ensure to check for false positives with testing before determining whether this is an appropriate capability to turn on before deploying to a production environment.

Configuring Custom Error Messages

Many commands will require parsing out the return value looking for error messages. The error handling has been expanded to include a configuration file that allows for extending the set of error messages you might encounter.

Each error message which is being searched, is defined as a regex signature.

The Topsecret connector comes with a default signatures file, errorMsgSignatures.xml, that you can extract from within the *LDAP_INSTALL_DIR*/dist/idfserver.jar compilation file. The errorMsgSignatures.xml file is located in the `com/identityforge/idfserver/backend/tops/repository/` directory of the `idfserver.jar` compilation file.

You can add, overwrite, or disable the defaults in favor of custom messages.

To do so, in the *LDAP_INSTALL_DIR*`/conf` directory, create a new XML file representing the messages to add, replace, or disable. For example, create a new XML file *LDAP_INSTALL_DIR*`/conf/custom-tops-error-sig-file.xml` and add your custom messages. Then, in the *LDAP_INSTALL_DIR*`/conf/tops.properties` file, add a reference to the newly created XML file by setting a value for the `errormsg-sig-file` property. For example:

```
errormsg-sig-file=../conf/custom-tops-error-sig-file.xml
```

Restart the LDAP gateway for the changes to take effect. At runtime, the contents of the custom signature file are merged into the default signatures file and the overrides or additions will be applied.

The following are examples of custom signatures:

Example 1: Suppose you create a new XML file *LDAP_INSTALL_DIR*`/conf/custom-tops-error-sig-file.xml` in the *LDAP_INSTALL_DIR*`/conf` directory with the following entries:

```
<?xml version="1.0" encoding="utf-8"?>
<Signatures>
    <Signature id="custom1" regex="^C4R541E .*" enabled="yes"/>
    <Signature id="custom2" regex="^ICH02005I .*" enabled="yes"/>
    <Signature id="custom3" regex="^IKJ56701I .*" enabled="yes"/>
</Signatures>
```

In this example, the first signature looks for `C4R541E` located at the beginning of the returned message from Pioneer. If found, it would get flagged as an error and the message returned.

The second signature looks for `ICH02005I` located at the beginning of the returned message from Pioneer. If found, it would get flagged as an error and the message returned. Modify as needed for example, signature 3 regex="^IKJ56701I .* to indicate. If found, it would get flagged as an error and the message returned.

In the preceding example, the `enabled="yes"` entry implies that the messages defined in the regex patterns must not be considered as errors.

Example 2: Suppose you create a new XML file *LDAP_INSTALL_DIR*`/conf/custom-tops-error-sig-file.xml` in the *LDAP_INSTALL_DIR*`/conf` directory with the following entries:

```
<?xml version="1.0" encoding="utf-8"?>
<Signatures>
    <Signature id="custom1" regex="^ICH\d{5}I .*" enabled="yes">
        <Exception regex="^ICH01432I .*"/>
        <Exception regex="^ICH05555I .*"/>
        <Exception regex="^ICH01024I .*"/>
    </Signature>
    <Signature id="custom2" regex=".*INVALID DEPARTMENT.*" enabled="yes"/>
    <Signature id="e2" enabled="no"/>
</Signatures>
```

In this example, the first signature looks for the `ICHxxxxxxI` pattern located at the beginning of the returned message from Pioneer. If found, it then examines the exceptions defined. If the message begins with `ICH01432I` or `ICH05555I`, then it is marked as a warning and ignored. Otherwise, it is flagged as an error and the message returned.

The second signature looks for `INVALID DEPARTMENT` to show up anywhere in the returned message. If found, then it is flagged as an error and the message returned.

The third signature is an example of disabling an existing default signature. All default signatures start with `e` in the `id` attribute followed by a number. By referencing the id, the default signature's regex, enablement flag, and or exceptions can be replaced with a custom override. The `enabled="yes"` entry implies that the messages defined in the regex patterns must not be considered as errors.

At any given point in time, locate and open the `errorMsgSignatures.xml` file to obtain the list of default signatures currently deployed.

> **Note:**
>
> Given that according to the CA Top Secret manual, "I" type messages are technically classified as informational and not error related, you need to make sure that it truly is a failure on the mainframe rather than something whereby the account gets created and Oracle Identity Manager considers it failed. We explicitly called out this SAF code as a warning as that is what the original implementation was doing.

7

Diagnostics and Troubleshooting the CA Top Secret Connector

Describes diagnostics and troubleshooting information for the connector that can assist in resolving issues.

- Understanding and Using the ENVINFO Diagnostic Tool
- Troubleshooting Information

Understanding and Using the ENVINFO Diagnostic Tool

Learn about the ENVINFO diagnostics tool and how to use it.

About the ENVINFO Diagnostics Tool

Whenever you need to report any issues related to the mainframe agents, Oracle recommends that you run the ENVINFO diagnostics tool. This tool fetches multiple setup and configuration values in your LPARs, which might be required to resolve the issue.

The ENVINFO tool is located in the default CLISTLIB (located inside `HLQ.CLISTLIB`) that is installed with the mainframe agents.

When you run the ENVINFO tool, it writes several system configuration information to the `HLQ.ENVINFO.OUTPUT` file that is created during installation of the mainframe agents. The following is some of the information that the ENVINFO tool is capable of fetching from the mainframe system and storing in the `HLQ.ENVINFO.OUTPUT` file:

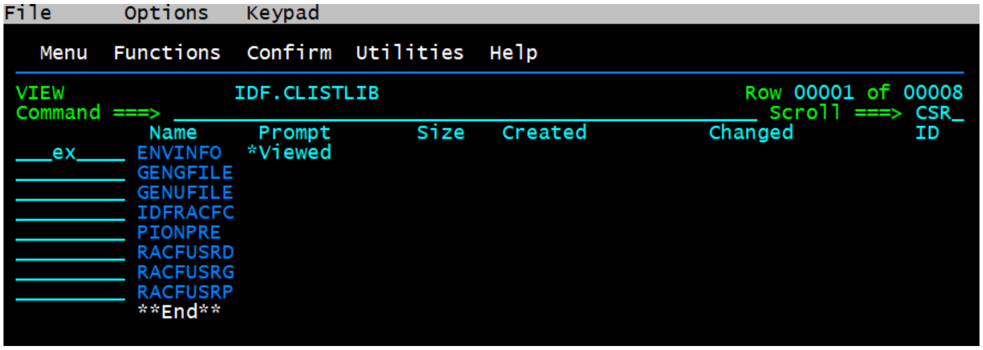
- System variables
- Storage variables
- CVT tables
- CPU info
- Codepage and character-set information
- Agent starter task definitions
- Agent version information fetched either from running starter task in spool or bind information in load module

Using the ENVINFO Tool

To use the ENVINFO tool:

1. Ensure that the DSN `<++ HLQ ++>.ENVINFO.OUTPUT` file is present and has been created by the CREATDSN job that is shipped along with the connector.
2. Go to the CLISTLIB (for example – IDF.CLISTLIB) that is created while installing the mainframe agents.
3. Look for the member ENVINFO.

4. Execute the ENVINFO rexx by issuing an `EX` against the member as shown in the following screenshot:



```
File       Options    Keypad

   Menu  Functions  Confirm  Utilities  Help

VIEW                 IDF.CLISTLIB                             Row 00001 of 00008
Command ===> _____________________________________________________ Scroll ===> CSR_
           Name     Prompt       Size    Created          Changed          ID
___ex____ ENVINFO  *Viewed
_________ GENGFILE
_________ GENUFILE
_________ IDFRACFC
_________ PIONPRE
_________ RACFUSRD
_________ RACFUSRG
_________ RACFUSRP
          **End**
```

5. Check the `<++` *`HLQ`* `++>. ENVINFO.OUTPUT` dataset for output.

Best Practices

If the rexx output reads that the Pioneer or Voyager job was not found, then the jobs need to be submitted and they should be up and running before attempting to execute this rexx again. In addition, this rexx relies on the fact that you are utilizing the agents from the libraries that were setup as described in this guide. If the starter tasks have been moved to a different location than the default ones, then the output of this rexx will be impacted and the starter task definition information may not be displayed.

Troubleshooting Information

You may encounter some problems with CA Top Secret configuration and these are some helpful tips to assist in resolving these problems.

The following table describes solutions to problems that you might encounter while using the connector.

Table 7-1 Troubleshooting Tips

Problem Description	Solution
Oracle Identity Manager cannot establish a connection with the target system.	• Ensure that the mainframe is running. • Verify that the required ports are working. • Due to the nature of the Provisioning Agent, the LDAP Gateway must be started first, and then the mainframe JCL started task must be started. This is a requirement based on how TCP/IP operates. Check that the IP address of the server that hosts the LDAP Gateway is configured in the Reconciliation Agent JCL. • Read the LDAP Gateway logs to determine if messages are being sent and received. • Examine the Oracle Identity Manager configuration to verify that the IP address, admin ID, and admin password are correct. • Check with the mainframe platform manager to verify that the mainframe user account and password have not been changed.

Table 7-1 (Cont.) Troubleshooting Tips

Problem Description	Solution
The mainframe does not appear to respond.	• Check the connection information that you have provided in the IT resource and the acf2Connection.properties file. • Check the logs. If any of the mainframe JCL jobs have reached an abnormal end, then make the required corrections and rerun the jobs.
A particular use case does not work as expected.	Check for the use case event in the LDAP Gateway logs. Then check for the event in the specific log assigned to the connector: • If the event has not been recorded in either of these logs, then investigate the connection between Oracle Identity Manager and the LDAP Gateway. • If the event is in the log but the command has not had the intended change on a mainframe user profile, then check for configuration and connections between the LDAP Gateway and the mainframe. Verify that the message transport layer is working.
The LDAP Gateway fails and stops working	If this problem occurs, then the Reconciliation Agent stops sending messages to the LDAP Gateway. Instead, it stores them in the subpool cache. When this happens, restart the LDAP Gateway instance so that the Reconciliation Agent reads the subpool cache and resends the messages.
The LDAP Gateway is running. However, the Reconciliation Agent fails and stops working	If this problem occurs, then all events are sent to the subpool cache. If the mainframe fails, then all messages are written to the disk. When this happens, restart the Reconciliation Agent instance so that it reads messages from the disk or subpool cache and resends the messages.
Top Secret reconciles users to Internal LDAP scheduled job (CFILE job) shows "Job Failure" as error message while it is still running. This is usually expected to happen while using Oracle Identity Governance 12*c* (12.2.1.4.0) only.	This job is failing due to an authentication error. In such a case, set the value of the `ServiceAccount.API.EncryptedParamsValue` and `ServiceAccount.ParamsValue.DBStore` system properties in Oracle Identity Governance to `True`.

8 Known Issues and Workarounds for CA Top Secret Connector

These are the known issues associated with this release of the connector.

Scheduled Tasks Contain Attributes for Trusted Recon

The scheduled tasks for reconciliation have attributes for trusted reconciliation settings even though the connector no longer supports trusted reconciliation.

There is no workaround for this issue.

A

Files and Directories in the CA Top Secret Connector Package

These are the files and directories on the connector installation package that comprise the CA Top Secret connector.

Table A-1 Files and Directories in the Installation Package

Files in the Installation Package Directory	Description
configuration/TopsAdv.xml	This XML file contains configuration information that is used during connector installation.
etc/LDAP Gateway/ IDF_LDAP_GATEWAY_v6.4.0.zip	This ZIP file contains the files required to deploy the LDAP Gateway.
etc/Provisioning and Reconciliation ConnectorTOPSECRET-AGENTS-201905311134-6.0.0.zip	This ZIP file contains the files required to deploy the Reconciliation and Provisioning Agents on the mainframe.
lib/topsecret-provisioning-adapter.jar	This JAR file contains the code for the adapters that are used during connector provisioning operations. During connector installation, this file is copied to the Oracle Identity Manager database.
lib/topsecret-scheduled-tasks.jar	This JAR file contains the code for the connector's scheduled tasks that perform lookup population and full reconciliation. During connector installation, this file is copied to the Oracle Identity Manager database.
Files in the resources directory	Each of these resource bundles contains locale-specific information that is used by the connector. During connector installation, this file is copied to the Oracle Identity Manager database. **Note:** A **resource bundle** is a file containing localized versions of the text strings that include GUI element labels and messages.
xml/oimTopsSecretAdvConnector.xml	This XML file contains definitions of the connector components, such as the IT resource and resource object. These objects are created in Oracle Identity Manager when you import the XML file.

B

Authorized Libraries

Authorizing libraries enables additional control over the functions that programs can access.

- About the APF Facility
- Finding APF-authorized Datasets

About the APF Facility

APF means "Authorized Program Facility". In a z/OS environment, APF is a facility that permits the identification of programs that are authorized to use restricted functions. APF-authorized programs must reside in one of the following authorized libraries:

- SYS1.LINKLIB
- SYS1.SVCLIB
- SYS1.LPALIB
- Authorized libraries specified by your installation

Authorized libraries are defined in an APF list. APF also prevents authorized programs (supervisor state, APF-authorized, PSW key 0-7, or PKM 0-7) from accessing a load module that is not in an APF-authorized library.

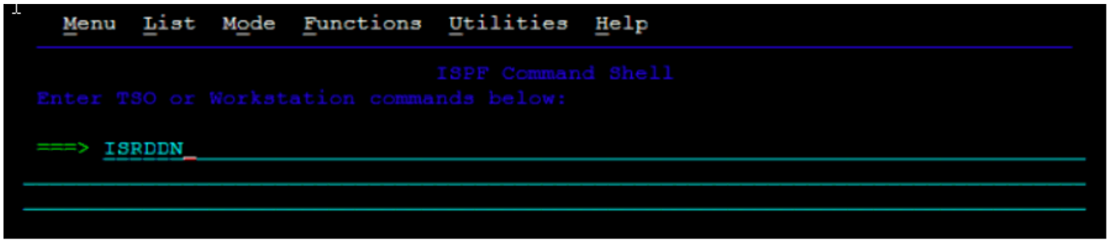
Finding APF-authorized Datasets

To find the datasets those are APF authorized:

1. Type TSO ISRDDN in your ISPF session (some shops need just ISRDDN with no TSO prefix) and press **enter**. See Figure B-1.
2. Type APF and press **enter**. It will bring up a list of all datasets that are APF authorized. See Figure B-2 and Figure B-3.

Remember that, if you like to use an APF authorized dataset in a job STEPLIB, make sure all the datasets in the STEPLIB are APF authorized.

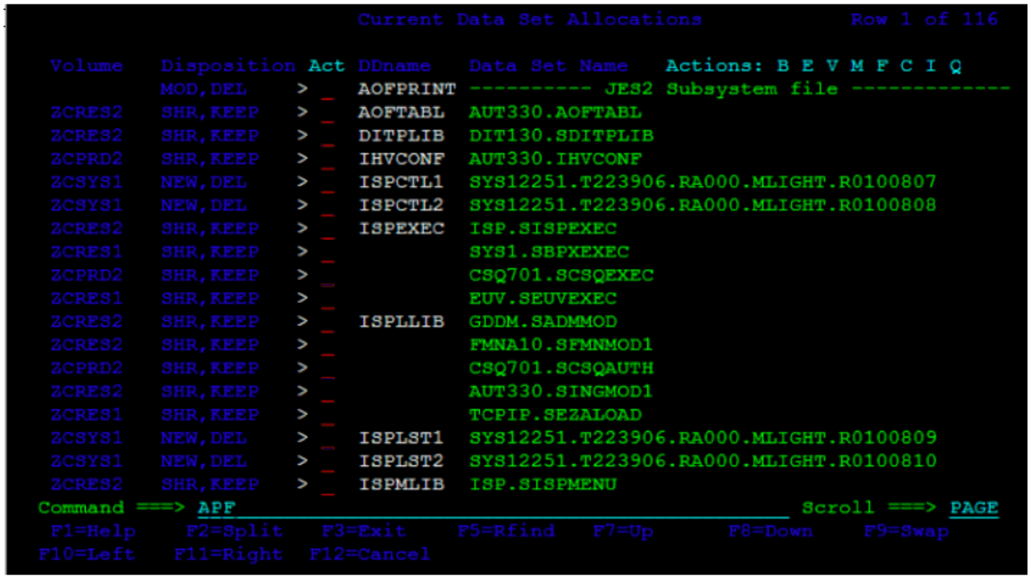
Figure B-1 ISPF Session Window



```
                                ISPF Command Shell
Enter TSO or Workstation commands below:

===> ISRDDN
```

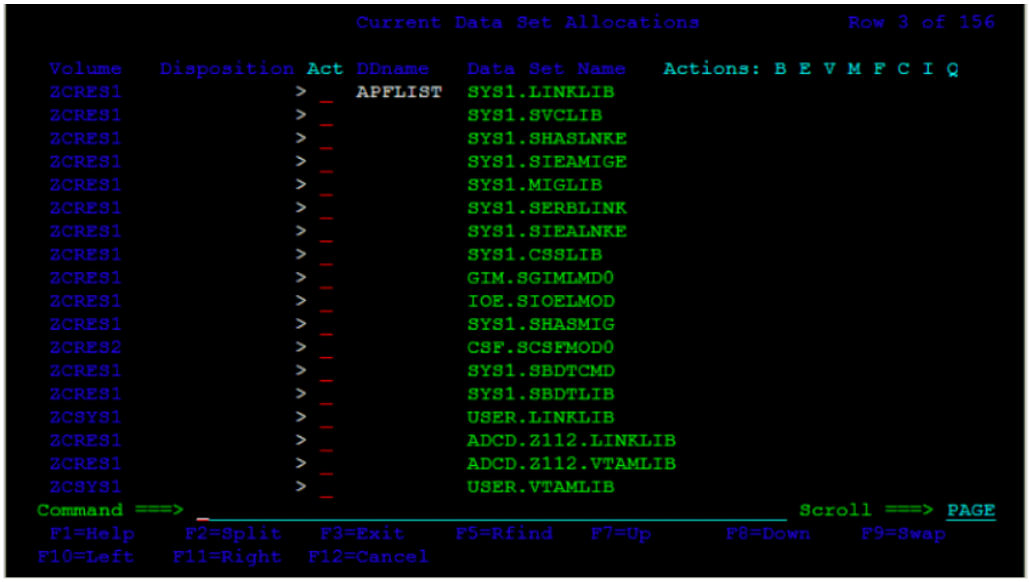
Figure B-2   List of All Datasets that are APF Authorized



```
                        Current Data Set Allocations            Row 1 of 116

   Volume    Disposition Act DDname    Data Set Name    Actions: B E V M F C I Q
             MOD,DEL     > _ AOFPRINT ---------- JES2 Subsystem file ------------
   ZCRES2    SHR,KEEP    > _ AOFTABL  AUT330.AOFTABL
   ZCRES2    SHR,KEEP    > _ DITPLIB  DIT130.SDITPLIB
   ZCPRD2    SHR,KEEP    > _ IHVCONF  AUT330.IHVCONF
   ZCSYS1    NEW,DEL     > _ ISPCTL1  SYS12251.T223906.RA000.MLIGHT.R0100807
   ZCSYS1    NEW,DEL     > _ ISPCTL2  SYS12251.T223906.RA000.MLIGHT.R0100808
   ZCRES2    SHR,KEEP    > _ ISPEXEC  ISP.SISPEXEC
   ZCRES1    SHR,KEEP    > _          SYS1.SBPXEXEC
   ZCPRD2    SHR,KEEP    > _          CSQ701.SCSQEXEC
   ZCRES1    SHR,KEEP    > _          EUV.SEUVEXEC
   ZCRES2    SHR,KEEP    > _ ISPLLIB  GDDM.SADMMOD
   ZCRES2    SHR,KEEP    > _          FMNA10.SFMNMOD1
   ZCPRD2    SHR,KEEP    > _          CSQ701.SCSQAUTH
   ZCRES2    SHR,KEEP    > _          AUT330.SINGMOD1
   ZCRES1    SHR,KEEP    > _          TCPIP.SEZALOAD
   ZCSYS1    NEW,DEL     > _ ISPLST1  SYS12251.T223906.RA000.MLIGHT.R0100809
   ZCSYS1    NEW,DEL     > _ ISPLST2  SYS12251.T223906.RA000.MLIGHT.R0100810
   ZCRES2    SHR,KEEP    > _ ISPMLIB  ISP.SISPMENU
  Command ===> APF____________________________________________ Scroll ===> PAGE
   F1=Help     F2=Split    F3=Exit     F5=Rfind    F7=Up       F8=Down     F9=Swap
  F10=Left    F11=Right   F12=Cancel
```

Figure B-3   APF Authorized Datasets



```
                        Current Data Set Allocations            Row 3 of 156

   Volume    Disposition Act DDname    Data Set Name    Actions: B E V M F C I Q
   ZCRES1                > _ APFLIST  SYS1.LINKLIB
   ZCRES1                > _          SYS1.SVCLIB
   ZCRES1                > _          SYS1.SHASLNKE
   ZCRES1                > _          SYS1.SIEAMIGE
   ZCRES1                > _          SYS1.MIGLIB
   ZCRES1                > _          SYS1.SERBLINK
   ZCRES1                > _          SYS1.SIEALNKE
   ZCRES1                > _          SYS1.CSSLIB
   ZCRES1                > _          GIM.SGIMLMD0
   ZCRES1                > _          IOE.SIOELMOD
   ZCRES1                > _          SYS1.SHASMIG
   ZCRES2                > _          CSF.SCSFMOD0
   ZCRES1                > _          SYS1.SBDTCMD
   ZCRES1                > _          SYS1.SBDTLIB
   ZCSYS1                > _          USER.LINKLIB
   ZCRES1                > _          ADCD.Z112.LINKLIB
   ZCRES1                > _          ADCD.Z112.VTAMLIB
   ZCSYS1                > _          USER.VTAMLIB
  Command ===> _______________________________________________ Scroll ===> PAGE
   F1=Help     F2=Split    F3=Exit     F5=Rfind    F7=Up       F8=Down     F9=Swap
  F10=Left    F11=Right   F12=Cancel
```

C

AES 128 User Key Definition and Usage

Pioneer and Voyager agents support the use of AES128 user key definitions. The customized key should consist of ASCII characters. Specifically, ASCII codes 33 – 127 are supported.

This appendix contains the following topics:

- Changing Pre-configured Key
- Configuring the LDAP Gateway

Changing Pre-configured Key

To change the pre-configured key, perform the following steps:

1. Copy the KEYMODR JCL to a new member in the PDS.
2. The Job stream is an AMASZAP, the module name is IDFRINFO.
3. Verify that the //SYSLIB is pointing to the LOADLIB or LINKLIB where Pioneer/Voyager modules are located.
4. Edit the copied member and modify the "REP 008A", the key is 32 bytes long.
5. Do not change REP displacement REP 009A and 00AA
6. Change REP 00BA to the date of the above change.
7. Submit the Job stream.

> **Note:**
>
> The AES128 key change effects both Pioneer/Voyager. Once an AES128 key is changed as stated above, shutdown Pioneer/Voyager and restart. Also the properties file on the LDAP must also be changed to contain the same key.

Configuring the LDAP Gateway

The LDAP gateway server settings must also be updated to use the new key. To configure the LDAP gateway, perform the following steps:

1. Stop the LDAP gateway server (if it is running).
2. Open the tops.properties file, located in the *LDAP_INSTALL_DIR*/conf directory.
3. Modify the value of the _secretKey_ property to match the new key.
4. Save and close the file.
5. Restart the LDAP gateway server.

D

CFILE LDAP Attribute Mapping for Top Secret Connector

CFILE LDAP attribute mapping is needed to reconcile child data through CFILE reconciliation.

This appendix contains the following topics:

- CFILE LDAP Attribute Mapping
- About LDAP ATTRIBUTE

> **Note:**
>
> Only a user's profiles and facilities child data is reconciled through CFILE reconciliation. No other child values are reconciled.

CFILE LDAP Attribute Mapping

Table D-1 lists CFILE record#, record type , LDAP attribute and the corresponding descriptions.

Table D-1 CFILE LDAP Attribute Mapping for Top Secret

CFILE RECORD #	RECORD TYPE	LDAP ATTRIBUTE	DESCRIPTION
NA	ACCESSORID	uid	ACID Unique ID
0100	NAME	cn	Full Name
0200	TYPE	type	ACID Type (USER, PROFILE, and so on)
0300	DEPT ACID	deptAcid	Department ID
0300	DEPT NAME	department	Department Descriptive Name
0400	DIV ACID	divAcid	Division ID
0400	DIV NAME	division	Division Descriptive Name
0450	ZONE ACID	zoneAcid	Zone ID
0450	ZONE NAME	zone	Zone Description Name
0500	CREATED	createDate	Create Date
0500	LAST MOD	lastModificationDate	Modify Date
0501	EXPIRES	expires	Expire Date
0502	SUSPENDED	suspendedUntilDate	Suspend Date
0600	PROFILES	profiles, memberOf	Profile Acids, Profile Acids Dn

Table D-1 (Cont.) CFILE LDAP Attribute Mapping for Top Secret

CFILE RECORD #	RECORD TYPE	LDAP ATTRIBUTE	DESCRIPTION
0650	GROUPS	groupIds, groupOf	Group Acids, Group Acids Dn
0700	ATTRIBUTES	attributes	Security Attributes
0800	BY PASSING	bypassing	Security ByPassing Attributes
0900	LAST USED	lastUsed	Last Used Date
1000	MASTER FAC	xresource	NA
1100	LOCK TIME	lockTime	NA
1200	LANGAUGE	language	Language
2002	XA DATASET	xresources	SEE BELOW (#A)
2005	XA xxxx (RESOURCE)	xresources	SEE BELOW (#B)
2014	PRIVPGM	xresources	SEE BELOW
2021	ACCESS	xresources	NA
2016	ACTION	xresources	NA
2100	FACILITY	facilities, facilityOf	Facility Acid, Facility Acid Dn
2200	SOURCES	sources	Source ACID
2300	OPIDENT	cicsOpident	CICS Operator Identification Value
2300	OPPRTY	cicsOpprty	CICS Operator Priority
2301	SITRAN	cicsSitran, cicsSitranFacility	CICS Transaction Following Facility Sign-In, CICS Facility Associated With Transaction
2400	OPCLASS	cicsOpclass	CICS Operator Classes
2500	SCTYKEY	cicsSctykey	CICS Security Keys
2600	INSTDATA	Instdata	255 Byte Text Field
2700	USER	NA	NA
2800	ACID	uniqueIds uniqueMember	Acids Profile Mem, Acids Dn
2901	FACILITIES	facilitiesp	Admin Facilities
2902	ACID	acid	Admin Acid
2903	LIST DATA	listData	Admin List Data
2904	MISC1	misc1	Admin Authority
2905	MISC9	misc9	Admin Authority
2906	RESOURCES	res	Admin Authority
2907	NA	NA	NA
2908	MISC2	misc2	Admin Authority
2909	SCOPE	scope	Admin Authority
2910	MISC8	misc8	Admin Authority
2911	ACCESS	access	Admin Authority

Table D-1 (Cont.) CFILE LDAP Attribute Mapping for Top Secret

CFILE RECORD #	RECORD TYPE	LDAP ATTRIBUTE	DESCRIPTION
2912	MISC3	misc3	Admin Authority
2913	MISC4	misc4	Admin Authority
2914	MISC5	misc5	Admin Authority
2921	ACCESS	xresources	See Below
3000	PASSWORD	passwordExpireDate passwordExpireInterval	Password Info
3500	TSOLPROC	tsolproc	TSO Logon Proc
3501	TSOLACCT	tsolacct	TSO Logon Account
3502	TSOJCLASS	tsojclass	TSO Job Class
3503	TSOMCLASS	tsomclass	TSO Message Class
3504	TSOLSIZE	tsolsize	TSO Region Size
3505	TSOUDATA	tsoudata	TSO User Data
3506	TSODEFPRFG	tsodefprfg	TSO Performance Group
3507	TSOOPT	tsoopt	TSO Options
3508	TSOCOMMAND	tsocommand	TSO Logon Command
3509	TSODEST	tsodest	TSO Output Destination
3510	TSOHCLASS	tsohclass	TSO Hold Class
3511	TSOMSIZE	tsomsize	TSO Max Region Size
3512	TSOSCLASS	tsosclass	TSO Sysout Class
3513	TSOUNIT	tsounit	TSO Unit
3700	FACILITY	facilities	Facility, All
4011 and 4012	USER-DEFINED	User Defined	User Defined Field Attribute will Match Field Name
4011	#APPL	lu62#appl	LU 6.2 #Appl
4011	#ENTITY	lu62#entity	LU 6.2 #Entity
4011	BC1CHAIN	lu62bc1chain	LU 6.2 Bc1chain
4011	BC2CHAIN	lu62bc2chain	LU 6.2 Bc2chain
4011	SET1DISP	lu62set1disp	LU 6.2 Set1disp
4011	SET2DISP	lu62set2disp	LU 6.2 Set2disp
4011	NETVCONS	netviewConsname	Netview Console Identifier
4011	NETVCTL	netviewControl	Netview Security Check Type
4011	NETVDMNS	netviewDomains	Netview Cross-Domain Sessions
4011	NETVIC	netviewInitCms	Netview Initial Command
4011	NETVMSGR	netviewMsgrecvr	Netview Receive Unsolicited Messages
4011	NETVNGMF	netviewNgmfadmn	Netview Authority To Graphic Monitor Facility
4011	NETVOPCL	netviewOpclass	Netview Scope Classes

Table D-1 (Cont.) CFILE LDAP Attribute Mapping for Top Secret

CFILE RECORD #	RECORD TYPE	LDAP ATTRIBUTE	DESCRIPTION
4401	UID	omvsUid	Omvs User ID
4402	GID	omvsGid	Omvs Group ID
4403	HOME	omvsHome	Omvs Home Subdirectory
4404	OMVSPRGM	omvsProgram	Omvs Program
4405	DFLTGRP	defaultGroup	Omvs Default Group
4406	ASSIZE	omvsAssize	Omvs Max Address Space Size
4407	MMAPAREA	omvsMmapArea	Omvs Max Data Space Pages
4408	OECPUTM	olecputm	Omvs Max Cpu Time
4409	OEFILEP	omvsOefilep	Omvs Max Files Per Process
4410	PROCUSER	omvsprocuser, procuser	Omvs Max Processes
4411	THREADS	omvsThreads	Omvs Max Pthreads Created

About LDAP ATTRIBUTE

LDAP ATTRIBUTE -> XRESOURCES

User is expected to read the xresources attribute and parse the data as needed for their application use.

:: Separates Field Name::Field Value

| Separates Different Fields

EXAMPLE DATA FOR DIFFERENT TYPES ->

acid-res::ACID|acid-auth::AMPIO#T|

rclass::$MOBIUS|rowner::DSAPP1|rres::DS.|alevel::READ|authfac::MOBIUST|authfac::MOBIUSP|

xauthclsn::DATASET|xauthdsno::DATASEX|xauthdsn::TXXXA.DUMMY4|alevel::READ|authfac::CICSPROD|authfacs::CICSTEST|

E

Provisioning Methods for OIM Adapters

The connector supports additional provisioning operations to CA Top Secret that are not shipped with a pre-configured child form, process task, or OIM adapter. Below is a list of method headers for functions included in the com.identityforge.idfTopsUserOperations java class (located in the topsecret-provisioning-adapter.jar). You can access these methods when creating or modifying an OIM adapter:

Primary Constructor:

```
public IdfTopsUserOperations(String idfServerHost, String idfServerPort, String
idfRootContext, String idfPrincipalDn, String idfPrincipalPwd, String ssl, String
trustStore, String trustStorePassword, String trustStoreType) throws Exception
```

Method Headers:

```
public String changePassword(String idfUserId, String idfCurrentPwd, String idfNewPwd)
public String deleteUser(String idfUserId)
public String resetPassword(String idfUserId, String idfNewPwd)
public String resetPassword(String idfUserId, String idfNewPwd, String expireNow,
String expireInDays)
public String revokeUser(String idfUserId)
public String revokeUser(String userId, String revokeUntil, String revokeFor)
public String revokeUserUntil(String userId, String revokeUntil)
public String resumeUser(String idfUserId)
public String resumeUserUntil(String userId, String type, String until)
public String resumeUser(String userId, String type, String until)
public String renameUser(String idfUserId, String newUid)
public String modifyUser(String idfUserDn, String idfAttrName, String idfAttrValue)
public String modifyUserRemove(String idfUserDn, String idfAttrName, String
idfAttrValue)
public String grantTsoAccess(String idfUserId, String idfTsoCommand, String
idfTsoAcctNum, String idfTsoSize, String idfTsoMaxSize, String idfTsoDest, String
idfHoldClass, String idfMsgClass, String idfJobClass, String idfProc, String
idfSysOutClass, String idfUnit, String idfUserData, String idfOpt)

public String addUserToDataset(String idfUserId, String idfDatasetId, String idfAccess)
public String addUserToDataset(String userId, String datasetId, String access, String
accessFor)
public String addUserToFacility(String idfUserId, String idfFacility, String idfAccess)
public String addUserToGroup(String idfUserId, String idfGroupId)
public String addUserToGroup(String uid, String groupId, String after, String before,
String first, String last, String forTime)
public String addUserToProfile(String uid, String profileId, String after, String
before, String first, String last, String forTime)
public String addUserToSource(String idfUserId, String idfSourceId)
public String removeUserFromDataset(String idfUserId, String idfDatasetId)
public String removeUserFromFacility(String idfUserId, String idfFaciltiy)
public String removeUserFromGroup(String idfUserId, String idfGroupId)
public String removeUserFromProfile(String uid, String profileId)
public String removeUserFromSource(String idfUserId, String idfSourceId)
public String generateCertificate(String idfUserId, String digicert, String dcdsn,
String keysize, String keyusage, String nbdate, String nbtime, String nadate, String
natime, String lablcert, String altname, String subjectn, String signwith, String
icsf, String dsa, String pcicc)
```

```
public String generateCertificateRequest(String idfUserId, String digicert,
String dcdsn, String lablcert)
```

F

Pioneer Searches Initiated from the LDAP

Top-Secret searches for all ACIDS, DEPTS, DATASETS, and so on can be requested by the LDAP at anytime.

Following is an overview of the LDAP-Pioneer search process and how it works:

1. Backend (OIM) or equivalent software application requests a "SEARCH ALL". This request is forwarded to the LDAP.
2. LDAP sends the Request to Pioneer through TCPIP
3. Pioneer validates the quest. Reads the //RECONJCL ddname dataset, this is a skeleton of MVS JCL, during this read Pioneer inserts the Rexx clist name for the desired function. For example: A SEARCHALL is a "%SRCHUSR DSN=xxxxx.xxxxx.xx", the DSN= is the QUEUE_DSN specified in the Pioneer control file. This is a 1 track enqueue/dequeue file. Please do not allocate this file. The modified skeleton JCL is then submitted to MVS through the Intrdr for execution. At this point Pioneer goes into a temporary wait, Pioneer is SINGLE thread only one TCB.
4. The submitted JCL is a PGM=IKJEFT01 with a //SYSPROC pointing to the Distribution Clist library. A ddname of FILEOUT pointing to the Pioneer ddname file "RECONOUT". The REXX clist begins execution by trying to find the QUEUE_DSN=, if the file is there it is DELETED. During this process Pioneer is waiting. If the QUEUE_DSN= is not found then TSS LIST commands are issued and a file is built (DDNAME = FILEOUT). When the REXX finishes, the QUEUE_DSN= is built and the REXX clist finishes. During the wait, Pioneer waits and trys every 10 seconds to allocate the QUEUE_DSN= dataset as DISP=OLD. When Pioneer can do this, it reads the file and sends it back to the LDAP. After all the data has been sent through Socket-writes to the LDAP, Pioneer deletes the records from the input file.

G Pioneer and Voyager LONG_FDTNAME=Y Processing

The Pioneer and/or Voyager agent, after receiving a TSS LIST request as a part of its internal logic, issues a TSS LIST for appropriate Top-Secret ACID.

When the list request is issued, Pioneer and Voyager will issue an @LSRCHUSR acid dsn through module IDFISCMD to IBM's System Rexx.

The new module uses MGCRE macro to issue this request. AXR (System Rexx) queues up the request and searches its REXXLIBs for the Rexx clist. The clists perform a TSO "LISTDS" for the dsname that is passed to System Rexx. If the dataset is found it is deleted.

A TSS LIST for the acid is then performed and the dsn is built. During this period, Pioneer and or Voyager will wait until it can allocate the dataset as OLD. When able to allocate Pioneer and/or Voyager will build messages for the LDAP, convert to ASCII, encrypt them and then issue a write socket.

The DSN specified must be unique for Pioneer and Voyager. The AXR STC (Started Task) must be able to update and delete these datasets.

The DSN specified in both Voyager and Pioneer to support LONG_FDTNAMES=Y must be unique. They are also preallocated by the LSRCHUSR Rexx clists in the REXXlib specified for System Rexx.

H

Pioneer and Voyager Messages

These are the messages generated by Pioneer and Voyager.

Pioneer Messages

All Provisioning Agent messages are prefixed with IDFTP. The next character after IDFTP defines the message type followed by 3 digit number that uniquely identifies the message in its specific sub-genre.

Table H-1 Pioneer Messages

Message ID	Message Text	Message Type
IDMP000I	Pioneer starting	Informational
IDMP001I	Pioneer Input Parameters are OK	Informational
IDMP001E	Pioneer Input Control File is Empty	Error
IDMP002I	Pioneer Detects IDF-Build <Build info>	Informational
IDMP003I	Pioneer Detects TCPIP Jobname <value>	Informational
IDMP004I	Pioneer Detects TCPIP IP Address of <value>	Informational
IDMP005I	Pioneer Detects TCPIP IP PORT of <value>	Informational
IDMP006I	Pioneer Detects Debugging is <value>	Informational
IDMP007I	Pioneer Detects Audit log is <value>	Informational
IDMP009I	Pioneer Detects Encryption Enabled	Informational
IDMP010I	Pioneer Detects Encryption Disabled	Warning
IDMP011I	Pioneer Detects CPUID <value>	Informational
IDMP012I	Pioneer Detects Sysplex Sysname <value>	Informational
IDMP013I	Pioneer Detects LPARNAME AS <value>	Informational
IDMP014I	Pioneer Detects Country Code of <value>	Informational
IDMP015I	Pioneer Detects Job Wait Time Of <value> Secs	Informational
IDFTPI001	Pioneer Detects RECON wait time of <value> Mins	Informational
IDMP020I	Pioneer Accepting Messages on <IP/Host>	Informational

Table H-1 (Cont.) Pioneer Messages

Message ID	Message Text	Message Type
IDMP020A	Pioneer Operator has Issued a Shutdown Command	Warning
IDMP021I	Pioneer has a Write Delay of <value> ms	Informational
IDMP030I	Pioneer INITAPI was successful	Informational
IDMP031I	Pioneer GETCLIENTID was successful	Informational
IDMP032I	CLIENT NAME/ID is <value>	Informational
IDMP033I	CLIENT TASK is <value>	Informational
IDMP035I	Pioneer BIND SOCKET was successful	Informational
IDMP036I	Pioneer Listening port is <value>	Informational
IDMP037I	Pioneer Listening Address is <IP/Host>	Informational
IDMP038I	Pioneer Listen Socket Call was successful	Informational
IDMP041I	Pioneer Socket Accept was successful	Informational
IDMP041E	Pioneer Socket Accept was not successful RC: <value>	Error
IDMP048I	Pioneer LDAP Connection Timed out	Informational
IDMP049I	Pioneer Has Been Idle for 60 Mins	Informational
IDMP050A	Pioneer Closing IP Connection	Warning
IDMP051I	Pioneer Close Socket Call was Successful	Informational
IDMP052I	Pioneer Shutdown Socket Call was Successful	Informational
IDMP054I	Pioneer Received TOPS Recon Request from LDAP	Informational
IDMP055I	Pioneer Recon Processing Started	Informational
IDMP056I	Pioneer Recon Processing Ended	Informational
IDMP057I	Pioneer Recon Processing Successful	Informational
IDMP070I	Pioneer <File> Is Now Open	Informational
IDMP071I	Pioneer <File> Is Now Closed	Informational
IDMP071A	Pioneer has cleared the <FileName> FILE	Informational
IDMP070E	Pioneer Could Not Open <File> RC: <value>	Error
IDMP080I	Pioneer Job Submitted to the Intrdr	Informational
IDMP100I	PIONEER (IN) MSGS PROCESSED IS <value>	Informational

Table H-1   (Cont.) Pioneer Messages

Message ID	Message Text	Message Type
IDFTPI002	Pioneer Message (READ) Bytes <value>	Informational
IDFTPI003	Pioneer Message (WRITE) Bytes <value>	Informational
IDFTPI004	Pioneer Debug Log Files Created <Logfile>	Informational
IDMP102I	Pioneer Terminating	Informational
IDMP200E	Pioneer Startup Parameter Error <value>	Error
IDMP201I	Pioneer <Parmout-msg> Status = Good	Informational
IDMP300I	*Debug* - <Debug-msg>	Debug
IDMP401E	*Parms* - <Message> <Status> <text>	Informational/ Parm
IDMP303I	Pioneer Job Wait was modified by the operator to <value> secs	Informational
IDMP304I	Pioneer Recon Wait time was modified by the operator to <value> secs	Informational
IDMP305I	Pioneer Debugging was Turned <status>	Informational
IDMP306I	Pioneer Received Status Query and is Alive	Informational
IDMP500I	*AUDIT* - <Func> <CMD> <ACID> <STAT> <INFO>	Audit
IDMP500I	*HEADER* - <Func> <CMD> <ACID> <STAT> <INFO>	Audit
IDMP500I	Generic	Audit
IDMP620I	*RADMIN* <value>	Informational
IDMP630I	*CLISTI* <value>	Informational
IDMP600I	*EXPMSG* <value>	Informational
IDMP605I	Pioneer waiting for the LDAP to respond Timeouts = <value> times	Informational
IDFTPE006	INITAPI ERRNO =156 - ERROR MSG - EMVSINITAL z/os Unix systems services process initial failure, review TSS USS Segment TSS OMVS Segment incorrectly defined Pioneer will terminate	Error
IDFTPE007	INITAPI ERRNO = 10102 - Missing SEZATCP. Loadlib Verify SEZATCP. Loadlib is either in the linklist or as a steplib Pioneer will terminate	Error
IDFTPE008	TCPN parameter in incorrect Must be the TCPIP STC Name Pioneer will terminate	Error

Table H-1 (Cont.) Pioneer Messages

Message ID	Message Text	Message Type
IDFTPI056	Setting next addrinfo address as <value>	Informational
IDFTPI057	Address of Output Next Addrinfo as <value>	Informational
IDFTPI058	ADDRESS OF INPUT-ADDRINFO-PTR IS <value>	Informational
IDFTPI059	NTOP-Family	Informational
IDFTPI060	PIONEERX : CLOSE RETCODE= <value>	Informational
IDFTPW004	MAX-TIMEOUTS TRIGGERED, GOING TO EXIT	Warning
IDFTPI061	SHUTDOWN RECEIVED FROM POLLOPER	Informational
IDFTPI062	STATUS CHECK RECEIVED FROM POLLOPER	Informational
IDFTPI063	DEBUG=<Y/N> RECEIVED FROM POLLOPER	Informational
IDFTPI064	PIONEER DEBUG ALL READY <ACTIVE/DEACTIVATED>	Informational
IDFTPI065	<RWAIT/JWAIT/VALUE>= RECEIVED FROM POLLOPER	Informational
IDFTPI066	SOCK# <value> is ON/OFF	Informational
IDFTPI067	<AUTH/CHG/ACTION> <value> ON USERID <value> RC= <value> OOPS(1) = <value> OOPS(2) = <value> OOPS(3) = <value>	Informational
IDFTPE009	CLIST Queue not found being created by system Rexx. Pioneer will retry allocate in 2 seconds	Error
IDFTPE010	Error Calling Program <value>	Error
IDFTPE011	Pioneerx is Ending due to Errors	Error
IDFTPE012	PARM-ERROR TYPE <value>	Error
IDFTPE013	DEBUG = Y and Debugout paramater is invalid must be DEBUGOUT=SYSOUT , CLASS(n) <OR> DEBUGOUT=FILE Please fix parameter and restart Pioneer	Error
IDFTPI068	DEBUGOUT Alloc OK	Informational
IDFTPE014	Debugout free failed	Error
IDFTPE015	DEBUGOUT spun to class <value> Debugout free failed	Error
IDFTPW005	** Invalid parameter in Control file **	Warning
IDFTPI069	PARMFLE:STATUS <value>	Informational

Table H-1 (Cont.) Pioneer Messages

Message ID	Message Text	Message Type
IDFTPE016	Pioneer control file must have control records in it see admin manual for details. Pioneer will abend	Error
IDFTPW006	Pioneer Control File is empty	Warning
IDFTPE017	Pioneer will Terminate	Error
IDFTPE018	Creation OF : <dsname> Failed RC <value>	Error
IDFTPE019	Open of Debugout failed RC: <value>	Error
IDFTPE020	Duplicate <ACTION/CREATE/ADDTP/REMOVE> encountered and will not be processed	Error
IDFTPE021	(PIONEER) ddname INCLR/RECONJCL/FILE is empty populate as described in Manual - Pioneer will abend with RC=100	Error
IDFTPE022	TEMPLIB1 WAS NOT FREED OK	Error
IDFTPE023	CLIST QUEUE NOT FOUND BEING CREATED IN BATCH. PIONEER WILL RETRY ALLOCATE IN 5 SECONDS.	Error
IDFTPI070	PROCESSED <rclist value>	Informational
IDFTPI071	RECONIN/LISTIN Closed ok	Informational
IDFTPI072	RECONIN records read <value>	Informational
IDFTPI073	RECONIN File now Empty	Informational
IDFTPW007	ALIAS FUNCTION TYPE NOT SUPPORTED	Warning
IDFTPW008	FILE-FREE TYPE NOT SUPPORTED	Warning
IDFTPI074	Pioneer Processing <value>	Informational
IDFTPI075	DYN-PARMS : <value>	Informational
IDFTPI076	Pioneer - Dynamic Alloc Of <libname> was OK	Informational
IDFTPE024	Pioneer - Dynamic Alloc of <libname> Failed BPXWDYN RC : Return code. Post Processing For <Libname> Membername <memname> will not be executed	Error
IDFTPI077	Pioneer free of <libname> Was OK	Informational
IDFTPI078	PIONEER - SYSPUNCH NOW CLOSED SUCCESSFULLY	Informational
IDFTPI079	PIONEER - CLOSE OF ' <LIBNAME> WAS SUCCESSFUL	Informational

Table H-1 (Cont.) Pioneer Messages

Message ID	Message Text	Message Type
IDFTPE025	BAD SYSREXX-CALL RC : <value>	Error
IDFTPI080	GOOD SYSREXX-CALL RC : 0	Informational
IDFTPI081	Dynamic Alloc : <dsn value> FREED	Informational
IDFTPE026	Dynamic Alloc : <dsn value> FAILED	Error
IDFTPE027	PIONEER COBOL FUNCTION CEE3DLY FAILED	Error
IDFTPI082	*** SUBMISSION JCL START ***	Informational
IDFTPI083	*** SUBMISSION JCL END ***	Informational
IDFTPE028	PIONEER - DYNAMIC FREE OF <LIBNAME > FAILED BPXWDYN RC: < RETURN-CODE> ALLOCATED PDS < LIBNAME > CAN NOT BE FREED	Error
IDMP400I	*Parms* - <parms-value>	Informational/ Parm

Voyager Messages

All Reconciliation Agent messages are prefixed with IDFTV. The next character after IDFTV defines the message type followed by 3 digit number that uniquely identifies the message in its specific sub-genre.

Table H-2 Voyager Messages

Message ID	Message Text	Message Type
IDMV000I	Voyager Reconciliation Agent Starting	Informational
IDFTVI001	Voyager is Executing from an APF Authorized Library	Informational
IDMV000E	Voyager is Not Executing from an APF Authorized Library	Error
IDFTVI002	Voyager Found TPSS Security Subsystem	Informational
IDFTVI003	Voyager Found Required Storage Subpool	Informational
IDMV002I	Voyager Build Level is at <Build Info>	Informational
IDMV004I	Voyager Detects (TCPIP) Jobname <value>	Informational
IDMV005I	Voyager Detects (TCPIP) IP Address of <value>	Informational
IDMV006I	Voyager Detects (TCPIP) IP PORT <value>	Informational
IDMV007I	Voyager Detects Encryption is ON	Informational
IDMV011I	Voyager Detects Encryption	Informational

Table H-2 (Cont.) Voyager Messages

Message ID	Message Text	Message Type
IDMV012I	Voyager Detects Debugging is ON	Warning
IDMV013I	Voyager Detects Debugging is OFF	Informational
IDMV015I	Voyager Detects Country Code of <value>	Informational
IDMV016E	Voyager Detects Bad Hostname of <value>	Error
IDMV016I	Voyager Detects Hostname of <value>	Informational
IDMV019E	Voyager Initialization of TCP API Failed RC: <value>	Error
IDMV019I	Voyager Initialization of TCP API was Successful	Informational
IDMV020E	Voyager TCPIP Socket Descriptors Exceeded - Fatal Error	Error
IDMV021E	Voyager Initialization of PTON failed RC: <value>	Error
IDMV021I	Voyager Accepting Messages on <IP-Addr> (OR) <hostname.com>	Informational
IDFTVI004	Voyager Initialization of PTON was Successful	Informational
IDMV025I	Voyager Connected to Gateway Server	Informational
IDMV025I	Connect messages will not be displayed	Warning
IDMV025I	Connect messages will be displayed	Informational
IDMV070I	Voyager <File DDname> is Now Open	Informational
IDMV071I	Voyager <File DDname> is Now Closed	Informational
IDMV070E	Voyager could not open <File DDname> RC: <value>	Error
IDMV001E	Voyager Input Control File is Empty	Error
IDMV100I	Voyager Shutdown Started	Informational
IDMV103I	Voyager has Ended with Non-Zero Return Code	Warning
IDMV104I	Voyager Sent Messages nnnnn Received Messages nnnnn	Informational
IDMV105I	Voyager Sent Bytes nnnnn Subpool Messages nnnnn	Informational
IDMV110I	Voyager Reconciliation Agent has Terminated	Informational
IDMV111I	Voyager has Ended with Zero Return Codes	Informational
IDMV130I	Voyager Probed Server <n> Tries	Informational
IDFTVI005	Voyager Run Startup To Allocate Sp231	Informational

Table H-2 (Cont.) Voyager Messages

Message ID	Message Text	Message Type
IDMV206E	Voyager Cachesav Open for Input Failed RC <value> SEE SYSLOG	Error
IDFTVE001	Voyager Cachesav Close Failed RC: <value> SEE SYSLOG	Error
IDMV206I	Voyager Cachesav Open For Output	Informational
IDMV207I	Voyager CACHESAV Open for Input	Informational
IDMV208I	Voyager Reads nnnnn CACHE Messages	Informational
IDMV209I	Voyager Wrote nnnnn CACHE Messages	Informational
IDFTVI006	Voyager Cacehsv Closed OK	Informational
IDMV210I	Voyager CACHESAV Cleared for Usage	Informational
IDMV151I	Voyager DNS Request <Hostname.com>	Informational
IDMV152I	Voyager IP Connect Request <IP-Addr>	Informational
IDMV200E	Voyager Config Parm Error - <value>	Error
IDMV202E	Voyager no Storage Token Found	Error
IDMV202I	Voyager Unable to Connect to new IP/Port	Informational
IDMV203E	Voyager Quiescing Because of the Subpool Not found	Error
IDMV204E	Voyager Subpool 231 Cannot be Found	Error
IDMV300I	*Debug* - <Debug-msg>	Debug
IDMV401I	*DEFAULT* <PARM - TEXT>	Informational
IDMV601I	Voyager IP-FUNC (GETCLIENTID/ SOCKET/PTON/SHUTDOWN/ READ/WRITE ERRNO (n)	Informational
IDMV602E	Voyager Connection Failed to LDAP - ERRNO(nnnnnn)	Error
IDMV603I	*CLISTI* - <value>	Informational
IDMV500I	*HEADER*[|GENERIC|AUDIT] - <Func> <CMD> <ACID> <STAT> <INFO>	Audit
IDMV604I	Write Successful - MSG=(A=x,P=xxxx,L=nnnnnn,U=xxxxxxxx)	Informational
IDMV999T	Write Good TRANS-CTR: nnnnn	Informational
IDMV999E	Voyager is Abending due to Bad Parms	Error
IDFTVI007	Please review Voyager SYSLOG	Error
IDFTVE002	Voyager is Abending	Error
IDMV001E	Voyager has Abended	Error
IDFTVI046	Voyager detects TCP Write is Off	Informational

Table H-2 (Cont.) Voyager Messages

Message ID	Message Text	Message Type
IDFTVE017	Voyager detects Encryption is OFF. Voyager now terminates	Error
IDFTVI047	Cachesave Had : <value>	Informational
IDFTVE018	*--- IDMVINFO --- * 1 VERIFY CACHESAVE HAS THE CORRECT Z/OS FILE ATTRIBUTES - LRECL=32 AND DSORG=PS 2 VERIFY HAS THE CORRECT HLQ FOR THE DATASET AND TOP-SECRET PERMISSIONS - PIONEER MUST BE ABLE TO READ AND WRITE TO THE DATASET	Error
IDFTVE019	VOYAGER CONTROL FILE MUST HAVE CONTROL RECORDS IN IT . SEE ADMIN MANUAL FOR CONTROL RECORD FORMAT AND SEQUENCE FILE IS EMPTY VOYAGER WILL NOW ABEND	Error
IDFTVW004	Cache overflow	Warning
IDFTVW005	Bailing out now	Warning
IDFTVE020	Error on <opening/closing/action> cachesave. Cachesave <open/close/action> error status <value>	Error
IDFTVI048	FILE FOUND	Informational
IDFTVI049	* IP SWITCH DETECTED *	Informational
IDFTVI050	HEARTBEAT - VOYAGER IS OK	Informational
IDFTVI051	DEBUGGING - WILL NOW BE ACTIVE	Informational
IDFTVW006	DEBUGGING - WILL NOW STOP	Warning
IDFTVW007	<-- LDAP GATEWAY ERROR ------------> ------ <TCP-ADDR-IN> HAS NOT RESPONDED IN <RETRY-SECS> SECS -- VOYAGER HAS EXCEEDED RETRY COUNT ----	Warning
IDFTVI052	--------- BUILD INFO -----------	Informational
IDFTVI053	--------- BUILD INFO END -----------	Informational
IDFTVE021	Voyager Timer Failed	Error

Table H-2 (Cont.) Voyager Messages

Message ID	Message Text	Message Type
IDFTVI054	*--- 050-GETHOST-BYNAME BEFORE ---* SOKET-GETHOSTBYNAME: <SOKET-GETHOSTBYNAME> GET-NAMELEN : <GET-NAMELEN> GET-NAME : <GET-NAME> GET-HOSTENT : <GET-HOSTENT> GET-RETCODE : <GET-RETCODE>	Informational
IDFTVI055	*--- 050-GETHOST-BYNAME AFTER ---* SOKET-GETHOSTBYNAME: <SOKET-GETHOSTBYNAME> GET-NAMELEN : <GET-NAMELEN> GET-NAME : <GET-NAME> GET-HOSTENT : <GET-HOSTENT> GET-RETCODE : <GET-RETCODE>	Informational
IDFTVI056	*--- EZACIC08 BEFORE CALL --* GET-HOSTENT : <GET-HOSTENT> GET-NAMELEN : <GET-NAMELEN> GET-NAME : <GET-NAME> HOST-ALIAS-COUNT: <HOST-ALIAS-COUNT> HOST-ALIAS-SEQ : <HOST-ALIAS-SEQ> HOST-ALIAS-LENGTH: <HOST-ALIAS-LENGTH> HOST-ALIAS-VALUE: <HOST-ALIAS-VALUE> HOST-ADDR-TYPE : <HOST-ADDR-TYPE> HOST-ADDR-LENGTH: <HOST-ADDR-LENGTH> HOST-ADDR-COUNT : <HOST-ADDR-COUNT> HOST-ADDR-SEQ: <HOST-ADDR-SEQ> HOST-ADDR-VALUE: <HOST-ADDR-VALUE>	Informational

Table H-2 (Cont.) Voyager Messages

Message ID	Message Text	Message Type
IDFTVI057	*--- EZACIC08 AFTER CALL --* GET-HOSTENT : <GET-HOSTENT> GET-NAMELEN : <GET-NAMELEN> GET-NAME : <GET-NAME> HOST-ALIAS-COUNT: <HOST-ALIAS-COUNT> HOST-ALIAS-SEQ : <HOST-ALIAS-SEQ> HOST-ALIAS-LENGTH: <HOST-ALIAS-LENGTH> HOST-ALIAS-VALUE: <HOST-ALIAS-VALUE> HOST-ADDR-TYPE : <HOST-ADDR-TYPE> HOST-ADDR-LENGTH: <HOST-ADDR-LENGTH> HOST-ADDR-COUNT : <HOST-ADDR-COUNT> HOST-ADDR-SEQ: <HOST-ADDR-SEQ> HOST-ADDR-VALUE: <HOST-ADDR-VALUE>	Informational
IDFTVW008	EZACIC08 FAILED	Warning
IDFTVI058	*--- SOKET-NTOP BEFORE ---* SOKET-NTOP: <SOKET-NTOP> NTOP-FAMILY: <NTOP-FAMILY> HOST-ADDR-VALUE: <HOST-ADDR-VALUE> PRESENTABLE-ADDR: <PRESENTABLE-ADDR> PRESENTABLE-ADDR-LEN: <PRESENTABLE-ADDR-LEN>	Informational
IDFTVI059	*--- SOKET-NTOP AFTER ---* SOKET-NTOP: <SOKET-NTOP> NTOP-FAMILY: <NTOP-FAMILY> HOST-ADDR-VALUE: <HOST-ADDR-VALUE> PRESENTABLE-ADDR: <PRESENTABLE-ADDR> PRESENTABLE-ADDR-LEN: <PRESENTABLE-ADDR-LEN>	Informational
IDFTVE022	DYNAMIC ALLOC : <dsn> FREE FAILED BPXWDYN RC : <value>	Error
IDMV400I	*PARM* - <parm-txt>	Informational/Parm