Oracle® Identity Manager

Connector Guide for UNIX

11.1.1

E17694-14

May 2020

Oracle Identity Manager Connector Guide for UNIX, 11.1.1

E17694-14

Primary Author: Gowri.G.R

Contributing Authors: Mike Howlett, Alankrita Prakash

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or “commercial computer software documentation” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government’s use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface

What's New in Oracle Identity Manager Connector for UNIX?

1 About the Connector

2 Deploying the Connector

4 Extending the Functionality of the Connector

5 Testing and Troubleshooting

6 Known Issues

A Privileges Required for Performing Provisioning and Reconciliation

B Sample Scripts for Updating Default Attributes for Reconciliation

List of Figures

List of Tables

Preface

This guide describes the connector that is used to integrate Oracle Identity Manager Connector with UNIX.

Audience

This guide is intended for resource administrators and target system integration teams.

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 Manager, visit the following Oracle Help Center page:

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

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

`http://docs.oracle.com/middleware/12213/oig/index.html`

Documentation Updates

Oracle is committed to delivering the best and most recent information available. For information about updates to the Oracle Identity Manager Connectors documentation, visit Oracle Technology Network at

`http://download.oracle.com/docs/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 Oracle Identity Manager Connector for UNIX?

This chapter provides an overview of the updates made to the software and documentation for release 11.1.1.7.0 of the UNIX connector.

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

- Software Updates

  This section describes updates made to the connector software. This section also points out the sections of this guide that have been changed in response to each software update.

- Documentation-Specific Updates

  These include major changes made to this guide. For example, the relocation of a section from the second chapter to the third chapter is a documentation-specific update. These changes are not related to software updates.

Software Updates

The following sections discuss the software updates:

- Software Updates in Release 11.1.1.7.0
- Software Updates in Release 11.1.1.6.0
- Software Updates in Release 11.1.1.5.0

Software Updates in Release 11.1.1.7.0

The following are issues resolved in release 11.1.1.7.0:

Bug Number	Issue	Resolution
16490411	When you created access policy with DNLA flag for the connector, it did not work as expected. After running the "Evaluate user policies" job, the account did not move to Enabled state but moved to Provisioned state.	This issue has been resolved.

Bug Number	Issue	Resolution
16489848	Entitlement, IT resource, Account Name, and Account ID tagging were missing in the process form fields in Oracle Identity Manager 11.1.2.	This issue has been resolved. The following form fields have been tagged in the connector: Secondary groups child form tagged as `Entitlement = true`. IT Resource Type field of the parent form tagged as `ITResource = true`. User Login field in the parent form tagged as `AccountName =true` and `AccountId = true`.
16240960	Unnecessary step for copying the third-party files was displayed in UI during the connector installation.	This step has been removed.

Software Updates in Release 11.1.1.6.0

There are no software updates in release 11.1.1.6.0.

Software Updates in Release 11.1.1.5.0

This is the first release of the Oracle Identity Manager Connector for UNIX based on Identity Connector Framework (ICF). The following software updates have been made in release 11.1.1.5.0:

- ICF Based Connector
- Support for Switching Between SSH and Telnet Protocols
- Support for Running Custom Scripts
- Support for Configuring the Connector for a New Target System
- Support for Multiple Instances and Multiple Versions of UNIX
- Support for Addition of New Fields
- Support for Resource Exclusion Lists
- Support for Transformation and Validation of Data
- Support for Connection Pooling

ICF Based Connector

The Identity Connector Framework (ICF) is a component that provides basic provisioning, reconciliation, and other functions that all Oracle Identity Manager connectors require.

The Oracle Identity Manager Connector for UNIX is an ICF-based connector. The ICF uses classpath isolation, which allows the connector to co-exist with legacy versions of the connector.

For more information about the ICF and its advantages, see Understanding the Identity Connector Framework in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager*.

Support for Switching Between SSH and Telnet Protocols

From this release onward, you can switch between SSH and Telnet protocols to connect to UNIX-based target systems. Earlier, there were separate UNIX connectors for SSH and Telnet connections.

You can specify the connection type by using the connectionType parameter of the IT Resource. The connector supports the following connection types:

- SSH - This is the default connection. Used for SSH with password-based authentication.
- SSHPUBKEY - Used for SSH with key-based authentication.
- TELNET - Used for Telnet connection.

See Configuring the IT Resource for the Target System for related information.

Support for Running Custom Scripts

From this release onward, you can run scripts on a computer where the UNIX connector is deployed. You can configure custom scripts to support additional flavors of UNIX.

You can configure the scripts to run before or after the create, update, or delete an account provisioning operations. For example, you could configure a script to run before a user is created by the connector. See Configuring Action Scripts for more information.

Support for Configuring the Connector for a New Target System

From this release onward, you can configure the connector to support an additional flavor of UNIX by using custom scripts.

By default, the connector uses pre-configured scripts to support AIX, HP-UX, Linux, and Solaris. You can customize these scripts to support an additional flavor of UNIX. See Configuring the Connector for a New Target System for more information.

Support for Multiple Instances and Multiple Versions of UNIX

This release of the connector supports multiple instances and multiple versions of UNIX.

You can deploy a single connector bundle on Oracle Identity Manager and create multiple IT resources for multiple instances and multiple versions of UNIX. Then, you can use Oracle Identity Manager to manage accounts on these target systems. See Configuring the Connector for Multiple Instances and Multiple Versions of the Target System for more information.

Support for Addition of New Fields

This release of the connector supports the addition of single-valued fields for provisioning and reconciliation.

See the following sections for more information:

- Adding Custom Attributes for Target Resource Reconciliation
- Adding Custom Attributes for Provisioning

Support for Resource Exclusion Lists

From this release onward, you can specify a list of accounts that must be excluded from reconciliation and provisioning operations. Accounts whose user IDs you specify in the exclusion list are not affected by reconciliation and provisioning operations.

Configuring Resource Exclusion Lists describes the procedure to add entries in these lookup definitions.

Support for Transformation and Validation of Data

You can configure transformation of data, such as process form field data or any other object, that is brought into Oracle Identity Manager during reconciliation. In addition, you can configure validation of data that is brought into or sent from Oracle Identity Manager during reconciliation and provisioning. See the following sections for more information:

- Configuring Validation of Data During Reconciliation and Provisioning
- Configuring Transformation of Data During User Reconciliation

Support for Connection Pooling

This release of the connector supports the connection pooling feature based on the ICF. In earlier releases, a connection with the target system was established at the start of a reconciliation run and closed at the end of the reconciliation run. With the introduction of connection pooling, multiple connections are established by the ICF and held in reserve for use by the connector.

See Setting up the Lookup Definition for Connection Pooling for more information.

Documentation-Specific Updates

The following sections discuss the documentation-specific updates:

- Documentation-Specific Updates in Release 11.1.1.7.0
- Documentation-Specific Updates in Release 11.1.1.6.0
- Documentation-Specific Updates in Release 11.1.1.5.0

Documentation-Specific Updates in Release 11.1.1.7.0

The following documentation-specific update has been made in the revision "14" of the release 11.1.1.7.0:

- The "Oracle Identity Governance or Oracle Identity Manager" row in Table 1-1 has been modified to include support for Oracle Identity Governance release 12c PS4 (12.2.1.4.0).

The following documentation-specific updates have been made in the revision "13" of the release 11.1.1.7.0:

- The "Target Systems" row in Table 1-1 has been modified to include the supported version Oracle Enterprise Linux 6.*x* and 7.*x*.
- A prerequisite step has been removed from Creating a Target System SUDO User Account for Connector Operations.

The following documentation-specific updates have been made in the revision "12" of the release 11.1.1.7.0:

- The "Target System" row of Table 1-1 has been modified to include Red Hat Linux 6.*x* and 7.*x* Enterprise Edition (64 bit).
- The "External code" row of Table 1-1 has been removed.

The following documentation-specific updates have been made in the revision "11" of the release 11.1.1.7.0:

- The procedure to configure SSH Public Key Authentication for Linux has been modified in Configuring SSH Public Key Authentication.
- The example path mentioned for Solaris, HP-UX, Linux, and AIX has been modified in Configuring SSH Public Key Authentication.
- The "Connector Server" row has been added to Table 1-1.
- The "JDK" row of Table 1-1 has been renamed to "Connector Server JDK".

The following documentation-specific update has been made in the revision "10" of the release 11.1.1.7.0:

A "Note" regarding trusted source IT resource has been added at the beginning of Configuring the IT Resource for the Target System.

The following documentation-specific update has been made in the revision "9" of the release 11.1.1.7.0:

Table 2-4 has been updated with information specific to the defaultConnectorShell attribute.

The following documentation-specific updates have been made in the revision "8" of the release 11.1.1.7.0:

- The "Oracle Identity Manager" row of Table 1-1 has been updated.
- Information specific to Oracle Identity Manager 11*g* Release 2 PS3 (11.1.2.3.0) has been added to Usage Recommendation for the UNIX Connector.

The following documentation-specific updates have been made in the revision "7" of the release 11.1.1.7.0:

- A "Note" has been added at the beginning of Extending the Functionality of the Connector .
- A note related to `NOPASSWD` has been added to Step 2 of the procedure in Creating a Target System SUDO User Account for Connector Operations.
- The "isSudoWithNoPasswd" row has been added to Table 2-4.

The following documentation-specific update has been made in the revision "6" of the release 11.1.1.7.0:

- Information about limited reconciliation has been modified in Limited Reconciliation.

The following documentation-specific updates have been made in the revision "5" of the release 11.1.1.7.0:

- TimeOut Errors has been updated with a new problem description and solution.
- The "Target System" row in Table 1-1 has been modified for the 7.1 AIX version.

The following are documentation-specific updates in this release:

- The "Oracle Identity Manager" row in Table 1-1 has been modified.
- A note has been added in the "xml/UNIX-RequestDatasets.xml" row of Table 2-1.
- The following sections have been added:
  - Configuring Oracle Identity Manager 11.1.2 or Later
  - Localizing Field Labels in UI Forms
  - Upgrade the UNIX Connector from Release 11.1.1.6.0 to 11.1.1.7.0
  - Configuring Provisioning in Oracle Identity Manager Release 11.1.2
- Instructions specific to Oracle Identity Manager release 11.1.2.*x* have been added in the following sections:
  - Configuring the IT Resource for the Target System
  - Configuring the IT Resource for the Connector Server
  - Configuring Scheduled Tasks

Documentation-Specific Updates in Release 11.1.1.6.0

There are no documentation-specific updates in this release.

Documentation-Specific Updates in Release 11.1.1.5.0

The following documentation-specific update has been made in the revision "2" of the release 11.1.1.5.0:

- Installation includes connector installation scenarios depending on where you want to run the connector code (bundle), either locally in Oracle Identity Manager or remotely in a Connector Server.
- In Table 1-1, the following target systems are included on this release:
  - ExaLogic X2-2
  - Oracle Enterprise Linux 5.2+(32-bit) and higher versions of 64-bit
  - Higher version of Red Hat Linux 5.5 Enterprise Edition (64-bit)

1 About 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. This guide discusses the connector that enables you to integrate Oracle Identity Manager with UNIX-based target systems using SSH or Telnet protocol. This connector enables you to use the target system as a managed (target) resource or as an authoritative (trusted) source of identity data for Oracle Identity Manager.

In the account management (target resource) mode of the connector, information about users 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.

In the identity reconciliation (trusted source) configuration of the connector, users are created or modified only on the target system and information about these users is reconciled into Oracle Identity Manage

This chapter contains the following sections:

> **Note:**
>
> In this guide, the term *Oracle Identity Manager server* refers to the computer on which Oracle Identity Manager is installed.

1.1 Certified Components

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

Table 1-1 lists the certified components for this connector.

Table 1-1 Certified Components

Component	Requirement
Oracle Identity Governance or Oracle Identity Manager	You can use one of the following releases of Oracle Identity Manager: • Oracle Identity Governance 12*c* (12.2.1.4.0) • Oracle Identity Governance 12*c* (12.2.1.3.0) • Oracle Identity Manager 11*g* Release 1 BP02 (11.1.1.5.2) and any later BP in this release track • Oracle Identity Manager 11*g* Release 2 BP04 (11.1.2.0.4) and any later BP in this release track • Oracle Identity Manager 11*g* Release 2 PS3 (11.1.2.3.0)
Target systems	The target system can be any one of the following operating systems: • HP-UX 11.11, 11.20, 11.31 • IBM AIX 5L Version 5.2, 5.3, 6.1, 7.1 • Oracle Enterprise Linux 5.2+(32-bit) and 64-bit versions of 5.2, 5.3, 5.4, 5.5, 5.6, 6.*x*, 7.*x* • Red Hat Enterprise Linux AS 2.1, 3, 4.x • Red Hat Enterprise Linux ES 3, 4.x • Red Hat Linux 5.5+ Enterprise Edition (64-bit) • Solaris 9, 10, 11 **Note:** You can also configure and use the connector on any other UNIX-based operating system that supports SSH and Telnet protocols. See Configuring the Connector for a New Target System for more information.
Connector Server	11.1.2.1.0
Connector Server JDK	JDK 1.6 Update 24 or later, or JRockit JDK 1.6 Update 24 or later
External code	Expect4j (expect4j-1.0.jar) is a third-party API used to connect and run commands or scripts on the target systems. It is included with the connector installation bundle, along with the following dependent libraries: • commons-net-3.0.1.jar • jakarta-oro-2.0.8.jar • jsch-0.1.41.jar
Other systems	OpenSSH, OpenSSL, operating system patches (HP-UX), and SUDO software (only if the SUDO Admin mode is required)
Target system user account	Depending on the target system that you are using, the target system user account can be one of the following: • For AIX, HP-UX, and Linux environments: root user or sudo user • For Solaris: root user, sudo user, RBAC user You provide the credentials of this user account while configuring the IT resource.
Character encoding supported by the target system	The target system must support the default C (POSIX) locale. Use the following command to check the locale that the target system supports: `locale -a`

> **Note:**
>
> The connector requires sh shell on the target system to run the scripts. Therefore, the connector switches to sh before running the commands.
>
> If the user account indicated in the loginUser basic configuration parameter has access to sh and the user account can switch to sh, then there is no restriction on the original login shell.

1.2 Usage Recommendation for the UNIX Connector

Depending on the Oracle Identity Manager version that you are using, you must deploy and use one of the following connectors:

- If you are using an Oracle Identity Manager release 9.1.0.1 or later and earlier than Oracle Identity Manager 11*g* Release 1 BP02 (11.1.1.5.2), then use the 9.0.4 version of this connector.
- If you are using Oracle Identity Manager 11*g* Release 1 BP02 (11.1.1.5.2) or later, Oracle Identity Manager 11*g* Release 2 BP04 (11.1.2.0.4) or later, or Oracle Identity Manager 11*g* Release 2 PS3 (11.1.2.3.0), then use the latest 11.1.1.*x* version of this connector.

1.3 Certified Languages for the UNIX Connector

These are the languages that the connector supports.

- Arabic
- Chinese (Simplified)
- Chinese (Traditional)
- Czech
- Danish
- Dutch
- English
- Finnish
- French
- German
- Greek
- Hebrew
- Hungarian
- Italian
- Japanese
- Korean
- Norwegian

- Polish
- Portuguese
- Portuguese (Brazilian)
- Romanian
- Russian
- Slovak
- Spanish
- Swedish
- Thai
- Turkish

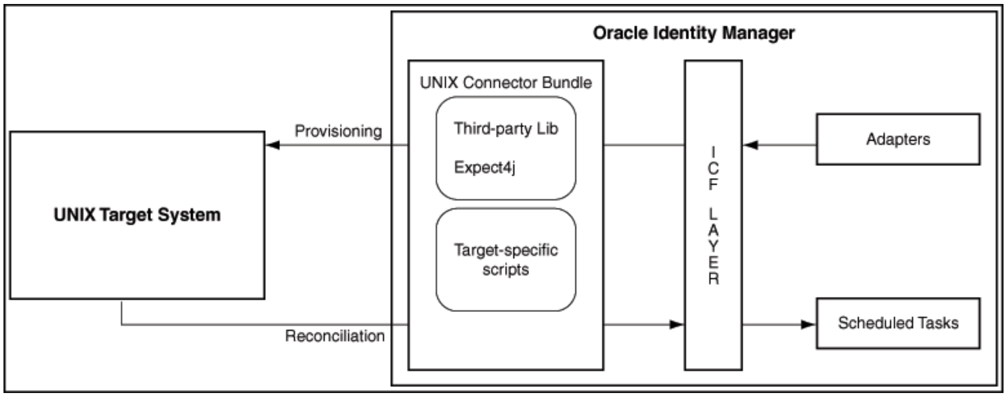
> **Note:**
>
> However, the connector does not support the entry of multibyte characters in some of the fields.

1.4 Architecture of the UNIX Connector

The UNIX connector is implemented by using the Identity Connector Framework (ICF).

This connector enables management of target system accounts through Oracle Identity Manager. Figure 1-1 shows the architecture of the connector.

Figure 1-1 Architecture of the Connector



The architecture of the connector can be explained in terms of the connector operations it supports:

- Reconciliation Process for the UNIX Connector
- Provisioning Process for the UNIX Connector

- Provisioning Functions of the UNIX Connector

1.4.1 Reconciliation Process for the UNIX Connector

This connector can be configured to perform either trusted source reconciliation or target resource reconciliation.

When you configure the target system as a target resource, the connector enables you to create and manage target accounts for OIM Users through provisioning. In addition, data related to newly created and modified target system accounts can be reconciled and linked with existing OIM Users and provisioned resources.

When you configure the target system as a trusted source, the connector fetches into Oracle Identity Manager, data about newly created or modified target system accounts. This data is used to create or update OIM Users.

> **See Also:**
>
> Reconciliation Based on the Object Being Reconciled in *Oracle Fusion Middleware Administering Oracle Identity Manager* for conceptual information about target resource reconciliation and trusted source reconciliation.

The following is an overview of the steps involved in reconciliation:

1. The scheduled task is run at the time or frequency that you specify. This scheduled task contains details of the mode of reconciliation (trusted source or target resource) that you want to perform.
2. The scheduled task establishes a connection with the target system by using the Expect4j third-party library.
3. The scheduled task performs the following tasks:
   - Reads the values that you set for the task attributes.
   - Reads the differences in the /etc/passwd, /etc/shadow and their corresponding mirror files to determine user records to be fetched into Oracle Identity Manager.
   - Fetches user records into Oracle Identity Manager.
4. If you have configured your target system as a trusted source, then:
   a. Each user record fetched from the target system is compared with existing OIM Users. The reconciliation rule is applied during the comparison process. See Reconciliation Rule for Trusted Source Reconciliation for information about the reconciliation rule.
   b. If a match is found between the target system record and the OIM User, then the OIM User attributes are updated with changes made to the target system record.
   c. If no match is found between the target system record and the OIM User, then the target system record is used to create an OIM User.
5. If you have configured your target system as a target resource, then:

a. Each user record fetched from the target system is compared with existing target system resources assigned to OIM Users. The reconciliation rule is applied during the comparison process. See Reconciliation Rule for Target Resource Reconciliation for information about the reconciliation rule.

b. If a match is found between the target system record and a resource provisioned to an OIM User, then the database user resource is updated with changes made to the target system record.

c. If no match is found between the target system record and a resource provisioned to an OIM User, then the target system user record is compared with existing OIM Users. The next step depends on the outcome of the matching operation:

   If a match is found, then the target system record is used to provision a resource for the OIM User.

   If no match is found, then the status of the reconciliation event is set to No Match Found.

1.4.2 Provisioning Process for the UNIX Connector

Provisioning involves creating and managing user accounts. When you allocate (or provision) a UNIX resource to an OIM User, the operation results in the creation of an account on the target system for that user. Similarly, when you update the resource on Oracle Identity Manager, the same update is made to the account on the target system.

The provisioning process can be started through one of the following events:

- Direct provisioning

  The Oracle Identity Manager administrator uses the Administrative and User Console to create a target system account for a user.

- Provisioning triggered by access policy changes

  An access policy related to accounts on the target system is modified. When an access policy is modified, it is reevaluated for all users to which it applies.

- Request-based provisioning

  In request-based provisioning, an individual creates a request for a target system account. The provisioning process is completed when an OIM User with the required privileges approves the request and provisions the target system account to the requester.

During provisioning operations, adapters carry provisioning data submitted through the process form to the Expect4j third-party library, which in turn submits the provisioning data to the target system. The user account maintenance commands accept provisioning data from the adapters, carry out the required operation on the target system, and return the response from the target system to the adapters. The adapters return the response to Oracle Identity Manager.

1.4.3 Provisioning Functions of the UNIX Connector

This section lists provisioning functions supported by the connector

Table 1-2 lists the provisioning functions and the Adapter column gives the name of the adapter that is used when the function is performed.

Table 1-2 Provisioning Functions

Function	Adapter
Create User	UNIXCreateUser
Delete User	UNIXDeleteUser
Disable User	UNIXDisableUser
Enable User	UNIXEnableUser
Expire Date Updated	UNIXUpdateUser
GECOS Updated	UNIXUpdateUser
Home Directory Updated	UNIXUpdateUser
Inactive Days Updated	UNIXUpdateUser
Password Updated	UNIXUpdateUser
Primary Group Updated	UNIXUpdateUser
Secondary Group Delete	UNIXUpdateChildTableValues
Secondary Group Insert	UNIXUpdateChildTableValues
Secondary Group Update	UNIXUpdateChildTableValues
UID Updated	UNIXUpdateUser
User Login Updated	UNIXUpdateUser
User Shell Updated	UNIXUpdateUser

1.5 Features of the UNIX Connector

The features of the connector include support for connector server, full reconciliation, and limited reconciliation.

The following are the features of this connector:

- Support for Switching Between SSH and Telnet Protocols
- Support for Running Custom Scripts
- Support for Configuring the Connector for a New Target System
- Support for Multiple Instances and Multiple Versions of UNIX
- Support for Both Target Resource and Trusted Source Reconciliation
- Support for Both Full and Incremental Reconciliation
- Support for Limited Reconciliation
- Support for Batched Reconciliation
- Support for Reconciliation of User Status from the Target System
- Support for Adding Custom Attributes for Reconciliation and Provisioning
- Transformation of Data
- Support for Resource Exclusion Lists

1.5.1 Support for Switching Between SSH and Telnet Protocols

You can switch between SSH and Telnet protocols to connect to UNIX-based target systems. You can specify the connection type by using the connectionType parameter of the IT Resource.

The connector supports the following connection types:

- SSH - This is the default connection. Used for SSH with password-based authentication.
- SSHPUBKEY - Used for SSH with key-based authentication.
- TELNET - Used for Telnet connection.

See Configuring the IT Resource for the Target System for related information.

1.5.2 Support for Running Custom Scripts

You can run scripts on a computer where the UNIX connector is deployed. You can configure custom scripts to support additional flavors of UNIX.

You can configure the scripts to run before or after the create, update, or delete an account provisioning operations. For example, you could configure a script to run before a user is created by the connector. See Configuring Action Scripts for more information.

1.5.3 Support for Configuring the Connector for a New Target System

You can configure the connector to support an additional flavor of UNIX by using custom scripts.

You can configure the connector to support an additional flavor of UNIX by using custom scripts.

By default, the connector uses pre-configured scripts to support AIX, HP-UX, Linux, and Solaris. You can customize these scripts to support an additional flavor of UNIX. See Configuring the Connector for a New Target System for more information.

1.5.4 Support for Multiple Instances and Multiple Versions of UNIX

The connector supports multiple instances and multiple versions of UNIX.

You can deploy a single connector bundle on Oracle Identity Manager and create multiple instances and multiple versions of UNIX. Then, you can use Oracle Identity Manager to manage accounts on these target systems. See Configuring the Connector for Multiple Instances and Multiple Versions of the Target System for more information.

1.5.5 Support for Both Target Resource and Trusted Source Reconciliation

You can use the connector to configure the target system as either a target resource or trusted source of Oracle Identity Manager.

See Configuring Reconciliation for more information.

1.5.6 Support for Both 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, incremental reconciliation is automatically enabled from the next run of the user reconciliation.

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

1.5.7 Support for Limited Reconciliation

You can set a reconciliation filter as the value of the Filter attribute of the scheduled tasks. This filter specifies the subset of newly added and modified target system records that must be reconciled.

See Limited Reconciliation for more information.

1.5.8 Support for Batched Reconciliation

You can break down a reconciliation run into batches by specifying the number of records that must be included in each batch.

See Batched Reconciliation for more information.

1.5.9 Support for Reconciliation of User Status from the Target System

From this release onward, the connector can reconcile user account status information from the target system.

1.5.10 Support for Adding Custom Attributes for Reconciliation and Provisioning

If you want to add custom attributes for reconciliation and provisioning, then perform the procedures described in Adding Custom Attributes for Target Resource Reconciliation and Adding Custom Attributes for Provisioning.

1.5.11 Transformation of Data

You can configure transformation of data that is brought into Oracle Identity Manager during reconciliation.

See Configuring Transformation of Data During User Reconciliation for more information.

1.5.12 Support for Resource Exclusion Lists

You can specify a list of accounts that must be excluded from reconciliation and provisioning operations.

Accounts whose user IDs you specify in the exclusion list are not affected by reconciliation and provisioning operations.

Configuring Resource Exclusion Lists describes the procedure to add entries in these lookup definitions.

1.6 User Attributes for Target Resource Reconciliation and Provisioning

The connector provides a default set of attribute mappings for reconciliation between Oracle Identity Manager and the target system. If required, you can add new user or group attributes for reconciliation.

Table 1-3 provides information about user attribute mappings for target resource reconciliation and provisioning.

Table 1-3 User Attributes for Target Resource Reconciliation and Provisioning

Process Form Field	Target System Field	Description
User Login	User Login	New login name, specified as a string of printable characters
Password	passwd	Password The length of the password field must be 32 characters. The supported special characters for password during provisioning are: `` ,./<>?:~!@#$%^*-_=+;'"()[]\{}|`& ``
Secondary Group Names	supplementary groups	List of supplementary groups, of which the user is also a member
User UID	uid	Numeric value of the user ID **Note:** It is recommended that you leave this entry blank. The target system will automatically assign a UID.
Primary Group Name	initial group	The group name or number of the user's initial login group **Note:** Primary Group cannot be updated to null.
Default Shell	shell	User's login shell
GECOS	comment	Generally, a short description of the login It is used as the field for the user's full name. This information is stored in the user's /etc/passwd file entry. **Note:** The entry of multibyte characters is supported for this attribute.
Home Directory	home directory	Login directory of the new user The final home directory name is obtained by appending the login name to the home directory value provided in this field. For example, if the login name is jdoe and the actual home directory desired on the target is `/home/jdoe`, then enter `/home` as the value for this field. Then, the default home directory is `/home/jdoe`.

Table 1-3 (Cont.) User Attributes for Target Resource Reconciliation and Provisioning

Process Form Field	Target System Field	Description
Expire Date	expire date	Date on which the user account is disabled **Note:** For AIX, this field is not reconciled.
Skeleton Directory	skeleton directory	Specifies the skeleton directory that contains information that can be copied to the new login's home directory An existing directory must be specified. The system provides a skeleton directory, /etc/skel, that can be used for this purpose.
Inactive Days	inactive days	Number of days after a password has expired before the account is disabled **Note:** For AIX, this field is neither provisioned nor reconciled.

1.7 User Attributes for Trusted Source Reconciliation

The connector provides a default set of attribute mappings for reconciliation between Oracle Identity Manager and the target system.

Table 1-4 lists user attributes for trusted source reconciliation.

Table 1-4 User Attributes for Trusted Source Reconciliation

OIM User Form Field	Target System Attribute	Description
User ID	UserLogin	UserLogin
Last Name	UserLogin	Last name
Employee Type	NA	Default value: `Full-Time`
User Type	NA	Default value: `End-User`
Organization	NA	Default value: `Xellerate Users`
TrustedStatus[TRUSTED]	Flag (Lock/Unlock)	User status

1.8 Roadmap for Deploying and Using the Connector

The following is the organization of information in the rest of this guide:

- Deploying the Connector describes procedures that you must perform on Oracle Identity Manager and the target system during each stage of connector deployment.
- Using the Connector describes guidelines on using the connector and the procedure to configure reconciliation runs and perform provisioning operations.
- Extending the Functionality of the Connector describes procedures that you can perform if you want to extend the functionality of the connector.
- Testing and Troubleshooting describes the procedure to use the connector testing utility for testing the connector.

- Known Issues lists known issues associated with this release of the connector.
- Privileges Required for Performing Provisioning and Reconciliation provides information about privileges required for successful provisioning operations and reconciliation runs.
- Sample Scripts for Updating Default Attributes for Reconciliation provides sample scripts for the procedure described in Adding Custom Attributes for Target Resource Reconciliation.
- Sample Scripts for Updating Default Attributes for Provisioning provides sample scripts for the procedure described in Adding Custom Attributes for Provisioning.

2

Deploying the Connector

This chapter is divided into the following sections:

- Preinstallation
- Installation
- Postinstallation
- Upgrading the Connector
- Postcloning Steps

2.1 Preinstallation

Preinstallation information is divided across the following sections:

- Files and Directories on the Installation Media
- Configuring the Target System

2.1.1 Files and Directories on the Installation Media

This section lists files and directories on the installation media.

Table 2-1 lists and describes them in detail.

Table 2-1 Files and Directories in the Installation Package

File in the Installation Package Directory	Description
bundle/org.identityconnectors.genericunix-1.0.0.jar	This JAR file contains the connector bundle.
configuration/GenericUNIX-CI.xml	This XML file contains configuration information that is used during the connector installation process.
Files in the resources directory	Each of these resource bundles contains language-specific information that is used by the connector. During connector installation, these resource bundles are copied to Oracle Identity Manager database. **Note:** A **resource bundle** is a file containing localized versions of the text strings that are displayed on the Administrative and User Console. These text strings include GUI element labels and messages.

Table 2-1 (Cont.) Files and Directories in the Installation Package

File in the Installation Package Directory	Description
Files in the test-utility directory: • example-config.groovy • README • test-utility.jar	These files are used by the testing utility to identify the cause of problems associated with connecting to the target system and performing basic operations on the target system. • The example-config.groovy file is a sample configuration that can be used to set the connection properties of the target system and the connector. • The README file contains instructions to configure and run the testing utility. • The test-utility.jar file contains the class files used by the testing utility.
upgrade/PostUpgradeScriptUnix.sql	This file is used after upgrading the connector. See Upgrading the Connector for more information.
util/privateKeyGen.sh	This file is used during SSH key-based authentication.
util/sudoers	This file contains the SUDO user specifications and configurations.
xml/UNIX-ConnectorConfig.xml	This XML file contains definitions for the connector components. These components include the following: • IT resource type • Process form • Process task and adapters (along with their mappings) • Resource object • Provisioning process • Prepopulate rules • Lookup definitions • Scheduled tasks
xml/UNIX-RequestDatasets.xml	This XML file contains request datasets that can be imported using Deployment Manager. It specifies the information to be submitted by the requester during a request-based provisioning operation. See Importing Request Datasets Using Deployment Manager for more information. **Note:** Use this file only if you are using Oracle Identity Manager release prior to 11.1.2.

2.1.2 Configuring the Target System

Depending on the target system and your requirements, perform some of the following procedures:

- Configuring Solaris and Linux
- Configuring AIX
- Configuring HP-UX
- Installing OpenSSH
- Creating a Target System SUDO User Account for Connector Operations
- Creating an RBAC User Account for Connector Operations on Solaris

- Configuring Public Key Authentication
- Configuring SSH Public Key Authentication

2.1.2.1 Configuring Solaris and Linux

Perform the following steps to configure Solaris and Linux environments:

1. Ensure that the /etc/passwd and /etc/shadow files are available on the UNIX server.
2. Create a directory on the target system where the connector can create mirror files for the /etc/passwd and /etc/shadow files.

   This directory is specified in the mirrorFilesLocation entry of the Lookup.UNIX.Configuration lookup definition. The default value is `/etc/connector_mirror_files`. If the directory path is different from the default value, then you must update the correct path in the lookup.The loginUser (sudo or root user) must have read and write privileges to this directory.

2.1.2.2 Configuring AIX

Perform the following steps for AIX environments:

1. Ensure that the /etc/passwd and /etc/security/user files are available on the server.
2. Create a directory on the target system where the connector can create mirror files for the /etc/passwd and /etc/shadow files.

   This directory is specified in the mirrorFilesLocation entry of the Lookup.UNIX.Configuration lookup definition. The default value is `/etc/connector_mirror_files`. If the directory path is different from the default value, then you must update the correct path in the lookup.The loginUser (sudo or root user) must have read and write privileges to this directory.

2.1.2.3 Configuring HP-UX

Perform the following steps for HP-UX environments:

1. If you want to switch to HP-UX Trusted mode, then:

   > **Note:**
   >
   > If you are converting the target system to the trusted system, then please make sure that no shadow file exists on the target after it is converted to trusted system.You can use `pwunconv` command to get rid of the shadow file, if it exists.

   a. Log in as root and then run the following command:

   ```
   /usr/bin/sam
   ```

   ```
   /usr/sbin/sam
   ```

   b. Select **Auditing and Security** and then select **System Security Policies.** A message is displayed asking if you want to switch to the trusted mode.

   c. Click **Yes.** The following message is displayed:

      ```
      System changed successfully to trusted system
      ```

2. Ensure that the /etc/passwd and /etc/shadow directories are available on the target server.
3. Create a directory on the target system where the connector can create mirror files for the /etc/passwd and /etc/shadow files.

   This directory is specified in the mirrorFilesLocation entry of the Lookup.UNIX.Configuration lookup definition. The default value is `/etc/connector_mirror_files.` If the directory path is different from the default value, then you must update the correct path in the lookup.The loginUser (sudo or root user) must have read and write privileges to this directory.

2.1.2.4 Installing OpenSSH

Follow these steps to install OpenSSH on the target system:

For Solaris 9

1. If SSH is not installed on the Solaris server, then install the appropriate OpenSSH.
2. Create a group with the name `sshd` and group ID `27`. Add a user with the name `sshadmin` to this group.
3. To enable root logins, change the value of `PermitRootLogin` in the /etc/ssh/sshd_config file as follows:

   ```
   PermitRootLogin yes
   ```

   > **Note:**
   >
   > Implement this change only if it does not violate local security policies. If Public Key Authentication is enabled, then you can change the value of `PermitRootLogin` to `without-password`.
   >
   > Instead of using the root account, if you can use a user account with sudo privileges, then you do not need to perform this step.

For Solaris 10 and Later Versions

By default, OpenSSH is installed on Solaris 10 and later versions. If it is not installed, then install the OpenSSH server from the operating system installation CD. To enable SSH, make the following changes in the /etc/ssh/ssh_config file:

1. Remove the comment character from the `Host *` line.
2. To enable root logins, change the value of `PermitRootLogin` in the /etc/ssh/sshd_config file as follows:

   ```
   PermitRootLogin yes
   ```

> **Note:**
>
> Implement this change only if it does not violate local security policies. If Public Key Authentication is enabled, then you can change the value of `PermitRootLogin` to `without-password`.
>
> Instead of using the root account, if you can use a user account with sudo privileges, then you do not need to perform this step.

For HP-UX

If SSH is not installed on the UNIX server, then install the appropriate OpenSSH from the installation media.

For Linux

By default, OpenSSH is installed on Red Hat Linux. If it is not installed, then install the OpenSSH server from the operating system installation CD.

For AIX

If SSH is not installed on the AIX server, then from the installation media:

1. Install OpenSSL.
2. Install PRNG.
3. Install OpenSSH.
4. To enable root logins, change the value of `PermitRootLogin` in the /etc/ssh/sshd_config file as follows:

   ```
   PermitRootLogin yes
   ```

> **Note:**
>
> Implement this change only if it does not violate local security policies. If Public Key Authentication is enabled, then you can change the value of `PermitRootLogin` to `without-password`.
>
> Instead of using the root account, if you can use a user account with sudo privileges, then you do not need to perform this step.

2.1.2.5 Creating a Target System SUDO User Account for Connector Operations

Oracle Identity Manager uses a target system account for performing reconciliation and provisioning operations. On all supported target systems, this account must be either the root user or sudo user.

> **See Also:**
>
> Privileges Required for Performing Provisioning and Reconciliation for information about the privileges required to perform connector operations

To create a target system user account with the minimum permissions required to perform connector operations, perform the following procedure:

1. If SUDO is not installed on the target system, then install it from the installation media.
2. Use the `visudo` command to edit and customize the /etc/sudoers file according to your requirements.

   > **Note:**
   >
   > If you cannot use the `visudo` command to edit the `sudoers` file, then:
   >
   > a. Enter the following command:
   >
   > ```
   > chmod 777 /etc/sudoers
   > ```
   >
   > b. Make the required changes in the `sudoers` file.
   >
   > c. Enter the following command:
   >
   > ```
   > chmod 440 /etc/sudoers
   > ```

   For example, if you have a group named `mqm` on the Linux server and require all members of the group to act as SUDO users with all possible privileges, then the `sudoers` file must contain the following line:

   ```
   mqm ALL= (ALL) ALL
   ```

   This example is only a sample configuration. If you need other group members or individual users to be SUDO users with specific privileges, then edit this file as was done for the sample value `mqm`.

   Therefore, the SUDO user must have the privileges required to run these commands.

   > **Note:**
   >
   > `NOPASSWD: ALL` option for any SUDO user or group is supported. To configure this, you may need to add a lookup field. For more information on adding a lookup field, refer Setting up the Lookup Definitions for Connector Configuration.

3. Edit the same `sudoers` file so that the SUDO user stays validated for 10 minutes after being validated once. You may need to increase the timeout if the reconciliation operation takes longer than 10 minutes and if you encounter errors such as "Permission denied". At the beginning of each operation, the connector

validates the user using `sudo -v` option so that the operation stays validated for a maximum of 10 minutes. After carrying out the operation, the connector runs the `sudo -k` to kill the validation.

Add the following line under the `# Defaults specification` header:

```
Defaults timestamp_timeout=10
```

This is a prerequisite for this connector to work successfully.

4. Create a SUDO user as follows:
   a. Enter the following command:

      ```
      useradd -g group_name -d /home/directory_name -m user_name
      ```

      In this command:

      `group_name` is the SUDO users group for which there is an entry in the `/etc/sudoers` file.

      `directory_name` is the name of the directory in which you want to create the default directory for the user.

   b. In the .bash_profile file, which is created in the /home/*directory_name* directory, add the following lines to set the `PATH` environment variable:

      ```
      PATH=/usr/sbin:$PATH
      export PATH
      ```

5. In the sudo user's .bashrc, .cshrc, or .kshrc file, which is created in the sudo user's home directory, add the following line to change the prompt end character from $ (dollar sign) to # (pound sign):

   ```
   PS1="[\\u@\\h:\\w]#"
   ```

   The encrypted passwords in the shadow file contain $ (dollar sign), which matches the default prompt end character. You must change the prompt end character to ensure that changes made to the shadow file are reconciled correctly.

6. Login with the sudo user.
7. Run the `sudo -k` command on the target system to clear the validation.
8. Run the `sudo -v` command on the target system and ensure that the password prompt is displayed.

   The connector would not work if the sudo user is not prompted for password at this step.

2.1.2.6 Creating an RBAC User Account for Connector Operations on Solaris

On Solaris, you can either create a sudo user or apply the role-based access control (RBAC) feature to create an account and assign to it the minimum privileges required for connector operations.

> **Note:**
>
> You use the IT resource to specify whether or not you want to use an RBAC user. Parameters of the IT resource are described later in this chapter.

To create an RBAC user account:

1. Run the following command to create a role for the user.

   ```
   roleadd -d /export/home/ROLE_NAME -m ROLE_NAME
   ```

   In this command, replace *ROLE_NAME* with the name that you want to assign to the role, for example, OIMRole.

2. Run the following command to assign a password to the role:

   ```
   passwd ROLE_NAME
   ```

   At the prompt, enter a password for the role.

   > **See Also:**
   >
   > Privileges Required for Performing Provisioning and Reconciliation for information about the privileges required to run the commands that are used for provisioning and reconciliation

3. Create a profile for the user as follows:

   a. Open the /etc/security/prof_attr file in a text editor and insert the following line in the file:

      ```
      PROFILE_NAME:::Oracle Identity Manager Profile:
      ```

      In this line, replace *PROFILE_NAME* with the name that you want to assign to the profile, for example, OIMProf.

   b. Save and close the file.

4. Add execution attribute entries in the /etc/security/exec_attr file. Each entry defines a task to be run and the uid that the role will assume when running the task.

   Open the /etc/security/exec_attr file in a text editor, and insert the following lines:

   > **Note:**
   >
   > There are seven fields in this file, and the colon (:) is used as the delimiting character.
   >
   > On Solaris 10, the value suser can be replaced with solaris.
   >
   > Some of the entries contain euid. These instances of euid can be replaced with uid.

   ```
   PROFILE_NAME:suser:cmd:::/usr/sbin/usermod:uid=0
   PROFILE_NAME:suser:cmd:::/usr/sbin/useradd:uid=0
   PROFILE_NAME:suser:cmd:::/usr/sbin/userdel:uid=0
   PROFILE_NAME:suser:cmd:::/usr/bin/passwd:uid=0
   PROFILE_NAME:suser:cmd:::/usr/bin/cat:euid=0
   PROFILE_NAME:suser:cmd:::/usr/bin/diff:euid=0
   PROFILE_NAME:suser:cmd:::/usr/bin/sort:euid=0
   PROFILE_NAME:suser:cmd:::/usr/bin/rm:uid=0
   ```

```
PROFILE_NAME:suser:cmd:::/usr/bin/grep:euid=0
PROFILE_NAME:suser:cmd:::/usr/bin/egrep:euid=0
PROFILE_NAME:suser:cmd:::/bin/echo:euid=0
PROFILE_NAME:suser:cmd:::/bin/sed:euid=0
```

You can add similar entries for other commands if you have customized the pre-configured Solaris scripts to use other commands.

5. Run the following command to associate the profile with the role:

   ```
   rolemod -P PROFILE_NAME ROLE_NAME
   ```

6. Run the following command to create the user:

   ```
   useradd -d /export/home/USER_NAME -m USER_NAME
   ```

7. Run the following command to assign a password to the user:

   ```
   passwd USER_NAME
   ```

8. Run the following command to grant the role to the user:

   ```
   usermod -R ROLE_NAME USER_NAME
   ```

9. To verify the changes that you have made, open the /etc/user_attr file in a text editor and verity that the following entries are present in the file:

   ```
   ROLE_NAME::::type=role;profiles=PROFILE_NAME
   USER_NAME::::type=normal;roles=ROLE_NAME
   ```

2.1.2.7 Configuring Public Key Authentication

To configure Public Key Authentication:

> **Note:**
>
> - If Public Key Authentication is used, then an RBAC user for a Solaris target system cannot be used.
> - This section contains the procedure to configure Public Key Authentication for a root user. It can also be configured for a SUDO user.

1. Copy the util/privateKeyGen.sh file from the installation media directory to any directory on the target system server.
2. Open this script file in a text editor and specify a working directory path other than the default value given in the file.
3. If required, enter the following command:

   For Solaris or Linux:

   ```
   dos2unix privateKeyGen.sh privateKeyGen.sh
   ```

   For HP-UX:

   ```
   dos2ux privateKeyGen.sh
   ```

4. Run the privateKeyGen.sh script on the UNIX server.

Provide a secure passphrase when prompted. Do not leave the passphrase blank. If you do so, the connector operations will be affected.

When these commands are run, the following files are created in the $HOME/.ssh directory:

- id_rsa: This is a private key file.
- authorized_keys: This file lists public keys that can be used to log in.

5. When the keys are generated successfully, edit the sshd_config file for Public Key Authentication and test login.
6. After successfully testing login, copy the id_rsa file to the following directory:

   ```
   OIM_HOME/server/ConnectorDefaultDirectory/SSH/config
   ```

   You can also copy the file to any directory that is readable and accessible by Oracle Identity Governance. The permissions for the keys should not be changed. If you change it for copying, then you must revert the permissions.

> **Note:**
>
> This release of the connector has been tested and certified only for RSA keys, and not DSA. In addition, this connector has been tested and certified for only single key configuration and not multiple keys.

2.1.2.8 Configuring SSH Public Key Authentication

To configure SSH Public Key Authentication:

For Solaris

1. Set the following parameters in the /etc/ssh/sshd_config file:

   ```
   PubKeyAuthorization yes
   PasswordAuthentication no
   PermitRootLogin yes
   ```

> **Note:**
>
> Change the value of `PermitRootLogin` to `yes` only if it does not violate local security policies. If Public Key Authentication is enabled, then you can change the value of `PermitRootLogin` to `without-password`.
>
> Instead of using the root account, if you can use a user account with sudo privileges, then you do not need to perform this step.

2. To restart the SSH server, enter the following commands:
   - `/etc/init.d/sshd stop`
   - `/etc/init.d/sshd start`
3. To test login:

   ```
   ssh -i /.ssh/id_rsa -l root server_IP_address
   ```

This command prompts you for the passkey before setting up the connection.

4. Set the `privateKey[LOADFROMURL]` advanced settings parameter to include the complete path of the `id_rsa` file with the prefix `file://`
   For example:

   ```
   file:///OIM_HOME/server/ConnectorDefaultDirectory/SSH/config/id_rsa
   ```

For HP-UX

1. Uncomment the following lines in the /etc/ssh/sshd_config file:

   ```
   PermitRootLogin yes
   PubkeyAuthentication yes
   AuthorizedKeysFile .ssh/authorized_keys
   ```

   > **Note:**
   >
   > Change the value of `PermitRootLogin` to `yes` only if it does not violate local security policies. If Public Key Authentication is enabled, then you can change the value of `PermitRootLogin` to `without-password`.
   >
   > Instead of using the root account, if you can use a user account with sudo privileges, then you do not need to perform this step.

2. To restart the SSH Server, enter the following command:

   ```
   /opt/ssh/sbin/sshd
   ```

3. To test login, enter the following command:

   ```
   ssh -i /.ssh/id_rsa -l root server_IP_address
   ```

   When prompted, enter the passkey to connect to the server.

4. In Oracle Identity Manager Design Console, create a new entry in the Lookup.UNIX.Configuration lookup definition with the following values:
   Code Key: `privateKey[LOADFROMURL]`

   Decode: Add the complete path of the `id_rsa` file, with the prefix file://.

   For example:

   ```
   file:///OIM_HOME/server/ConnectorDefaultDirectory/SSH/config/id_rsa
   ```

For Linux

1. Enter the following commands to restart the UNIX server:

   ```
   /etc/init.d/sshd stop
   /etc/init.d/sshd start
   ```

2. Copy the /.ssh/id_rsa file to the following directory:

   ```
   OIM_HOME/server/ConnectorDefaultDirectory/SSH/config
   ```

3. To check if you can connect to the target system using the SSH protocol, directly from the command prompt and without using a password, enter the following command:

> **Note:**
>
> The account used to run the OIM application server on UNIX should have the ownership of the id_rsa file.

```
ssh -i OIM_HOME/server/ConnectorDefaultDirectory/SSH/config/id_rsa -l root
host_ip_address
```

4. In Oracle Identity Manager Design Console, create a new entry in the Lookup.UNIX.Configuration lookup definition with the following values:
   Code Key: `privateKey[LOADFROMURL]`

   Decode: Add the complete path of the `id_rsa`file, with the prefix file://.

   For example:

   ```
   file:///OIM_HOME/server/ConnectorDefaultDirectory/SSH/config/id_rsa
   ```

For AIX

1. Use the /etc/ssh/sshd_config file to set the following parameters:

   ```
   export PATH=$PATH: /usr/sbin
   Installation path: /etc/ssh/
   sshd -- /usr/sbin/
   ```

2. Open the /etc/ssh/sshd_config file, and uncomment the following lines:

   ```
   AuthorizedKeysFile .ssh/authorized_keys
   PermitRootLogin yes
   PubkeyAuthentication yes
   ```

   > **Note:**
   >
   > Change the value of `PermitRootLogin` to `yes` only if it does not violate local security policies. If Public Key Authentication is enabled, then you can change the value of `PermitRootLogin` to `without-password`.
   >
   > Instead of using the root account, if you can use a user account with sudo privileges, then you do not need to perform this step.

3. To restart the SSH server, enter the following command:
   `/usr/sbin/sshd`

4. To test the login, enter the following command:

   ```
   ssh -i /.ssh/id_rsa -l root server_IP_address
   ```

   When prompted, enter the passkey to connect to the server.

5. In Oracle Identity Manager Design Console, create a new entry in the Lookup.UNIX.Configuration lookup definition with the following values:
   Code Key: `privateKey[LOADFROMURL]`

   Decode: Add the complete path of the `id_rsa` file, with the prefix file://.

   For example:

   ```
   file:///OIM_HOME/server/ConnectorDefaultDirectory/SSH/config/id_rsa
   ```

2.2 Installation

You must install the connector in Oracle Identity Manager. If necessary, you can also deploy the connector in a Connector Server.

> **Note:**
>
> In this guide, the term **Connector Installer** has been used to refer to the Connector Installer feature of the Oracle Identity Manager Administrative and User Console.

Depending on where you want to run the connector code (bundle), the connector provides the following installation options:

- To run the connector code locally in Oracle Identity Manager, perform the procedure described in Installing the Connector in Oracle Identity Manager
- To run the connector code remotely in a Connector Server, perform the procedures described in Installing the Connector in Oracle Identity Manager and Deploying the Connector Bundle in a Connector Server.

2.2.1 Installing the Connector in Oracle Identity Manager

In this scenario, you install the connector in Oracle Identity Manager using the Connector Installer.

> **Note:**
>
> In this guide, the term **Connector Installer** has been used to refer to the Connector Installer feature of the Oracle Identity Manager Administrative and User Console.

To run the connector code locally in Oracle Identity Manager, perform the following steps:

1. Copy the contents of the connector installation media directory into the following directory:

   OIM_HOME/server/ConnectorDefaultDirectory

   > **Note:**
   >
   > In an Oracle Identity Manager cluster, perform this step on each node of the cluster.

2. If you are using Oracle Identity Manager release 11.1.1, perform the following steps:

   a. Log in to the Administrative and User Console.
   b. On the Welcome to Identity Manager Advanced Administration page, in the System Management region, click **Manage Connector.**
3. If you are using Oracle Identity Manager release 11.1.2.*x*, perform the following steps:
   a. Log in to Oracle Identity System Administration.
   b. 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 **Generic UNIX Connector 11.1.1.7.0.** This list displays the names and release numbers of connectors whose installation files you copy into the default connector installation in Step 1.

   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 **Generic UNIX Connector 11.1.1.7.0.**
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 is displayed for the task. If a task fails, then an X mark and a message stating the reason for failure are displayed. Depending on the reason for the failure, make the required correction and then perform one of the following steps:

   - Retry the installation by clicking **Retry.**
   - Cancel the installation and begin again from Step 1.
8. If all three tasks of the connector installation process are successful, then a message indicating successful installation is displayed. In addition, a list of the steps that you must perform after the installation is displayed. These steps are as follows:
   a. Ensuring that the prerequisites for using the connector are addressed

> **Note:**
>
> At this stage, run the Oracle Identity Manager PurgeCache utility to load the server cache with content from the connector resource bundle in order to view the list of prerequisites. See Clearing Content Related to Connector Resource Bundles from the Server Cache for information about running the PurgeCache utility.
>
> There are no prerequisites for some predefined connectors.

b. Configuring the IT resource for the connector

   Record the name of the IT resource displayed on this page. The procedure to configure the IT resource is described later in this guide.

c. Configuring the scheduled tasks

   Record the names of the scheduled tasks displayed on this page. 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 on the Installation Media.

2.2.2 Deploying the Connector Bundle in a Connector Server

To deploy the connector bundle remotely in a Connector Server, you must first deploy the connector in Oracle Identity Manager.

To do so, perform the procedure described in Installing the Connector in Oracle Identity Manager.

> **Note:**
>
> - You can download the Connector Server from the Oracle Technology Network web page.
> - See Configuring the IT Resource for the Connector Server for related information.
> - See Using an Identity Connector Server in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager* for more information about installing, configuring, and running the Connector Server.

To install the connector in the Connector Server:

1. Stop the Connector Server.
2. Copy the connector bundle JAR file from the bundle directory of the connector installation media into the following directory:

   CONNECTOR_SERVER_HOME/bundles
3. Start the Connector Server.

2.3 Postinstallation

Postinstallation for the connector involves configuring Oracle Identity Manager, enabling logging to track information about all connector events, and configuring the IT resources. It also involves performing some optional configurations such as localizing the user interface, setting up lookup definitions for connection pooling, and so on.

Postinstallation steps are divided across the following sections:

- Configuring Oracle Identity Manager 11.1.2 or Later

- Configuring the IT Resource for the Target System
- Configuring the IT Resource for the Connector Server
- Setting up the Lookup Definitions for Connector Configuration
- Setting up the Lookup Definition for Connection Pooling
- Setting up the Lookup Definitions for User Operations
- Setting up the Lookup Definitions for Attribute Mappings
- Enabling Logging
- Changing to the Required Input Locale
- Clearing Content Related to Connector Resource Bundles from the Server Cache
- Localizing Field Labels in UI Forms

2.3.1 Configuring Oracle Identity Manager 11.1.2 or Later

If you are using Oracle Identity Manager release 11.1.2 or later, 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

2.3.1.1 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. Select the sandbox that you created.
8. From the table showing the available sandboxes in the Manage Sandboxes page, select the newly created sandbox that you want to activate.

9. On the toolbar, click Activate Sandbox.
   The sandbox is activated.

2.3.1.2 Creating a New UI Form

Create a new UI form as follows. For detailed instructions, see Creating Forms By Using the Form Designer in *Oracle Fusion Middleware Administering Oracle Identity Manager*.

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.
4. Enter a form name and click Create.

2.3.1.3 Creating an Application Instance

Create an application instance and associate it with the form created in Creating a New UI Form. Then, publish the application instance to an organization to make the application instance available for requesting and subsequent provisioning to users. See the following sections in *Oracle Fusion Middleware Administering Oracle Identity Manager* for detailed instructions on creating and publishing application instances:

- Creating Application Instances
- Publishing an Application Instance to Organizations

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 for a particular organization.

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

2.3.1.5 Harvesting Entitlements and Sync Catalog

To harvest entitlements and sync catalog:

1. Run the scheduled jobs for lookup field synchronization listed in Scheduled Tasks for Lookup Field Synchronization.
2. Run the Entitlement List scheduled job to populate Entitlement Assignment schema from child process form table. See Predefined Scheduled Tasks in *Oracle Fusion Middleware Administering Oracle Identity Manager* for more information about this scheduled job.
3. Run the Catalog Synchronization Job scheduled job. See Predefined Scheduled Tasks in *Oracle Fusion Middleware Administering Oracle Identity Manager* for more information about this scheduled job.

2.3.1.6 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.

2.3.2 Configuring the IT Resource for the Target System

> **Note:**
>
> If you have configured your target system as a trusted source, then create an IT resource of type **UNIX.** For example, UNIX Trusted. The parameters of this IT resource are the same as the parameters of the IT resources described in Table 2-2 of this section. See Creating IT Resources in *Fusion Middleware Administering Oracle Identity Manager*.

The IT resource for the target system contains connection information about the target system. Oracle Identity Manager uses this information for reconciliation and provisioning.

For both provisioning and reconciliation, the connector uses the UNIX Server IT Resource. This IT resource is created with default parameter values as part of the connector installation. You must update the IT resource parameters with information about the target system.

To configure the UNIX Server IT resource:

1. Depending on the Oracle Identity Manager release you are using, perform one of the following steps:
   - For Oracle Identity Manager release 11.1.1: Log in to the Administrative and User Console

- For Oracle Identity Manager release 11.1.2.*x*: Log in to Oracle Identity System Administration

2. If you are using Oracle Identity Manager release 11.1.1, then:

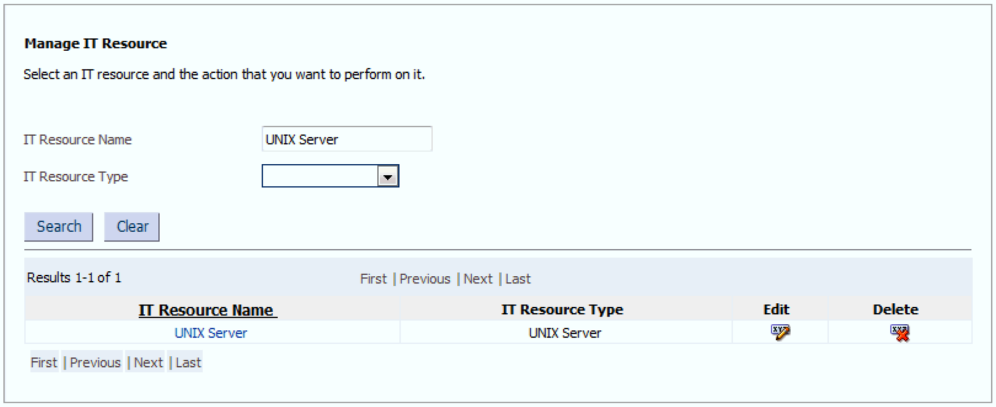
   a. On the Welcome page, click **Advanced** in the upper right corner.

   b. On the Welcome to Oracle Identity Manager Advanced Administration page, in the Configuration region, click **Manage IT Resource**.

3. If you are using Oracle Identity Manager release 11.1.2.*x*, then in the left pane under Configuration, click **IT Resource**.

4. In the IT Resource Name field on the Manage IT Resource page, enter `UNIX Server` and then click **Search.** Figure 2-1 shows the Manage IT Resource page.

Figure 2-1 Manage IT Resource Page



Manage IT Resource

Select an IT resource and the action that you want to perform on it.

IT Resource Name: UNIX Server

IT Resource Type:

Results 1-1 of 1

IT Resource Name	IT Resource Type	Edit	Delete
UNIX Server	UNIX Server		

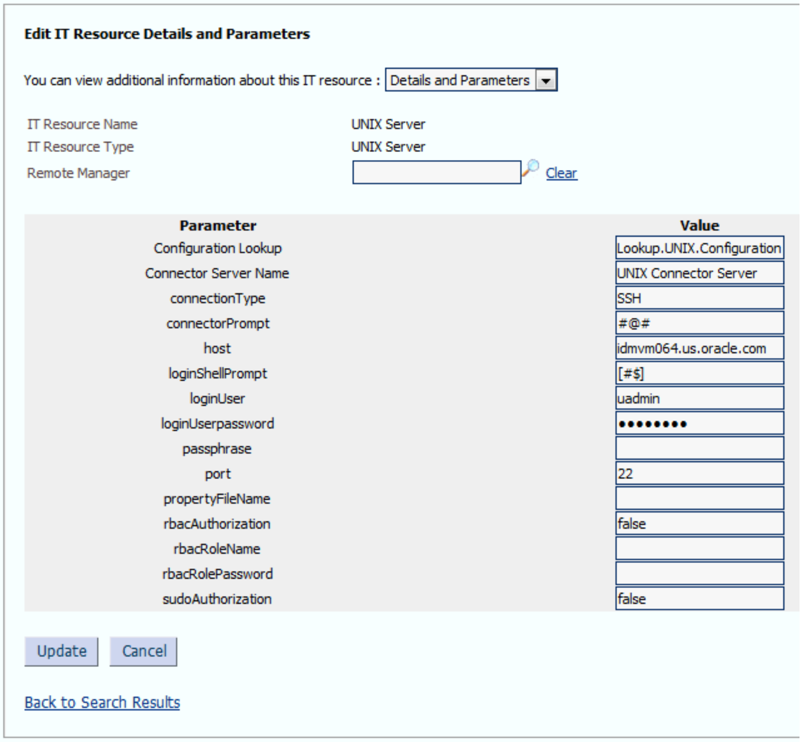
Search
Clear
First | Previous | Next | Last
First | Previous | Next | Last

5. Click the edit icon corresponding to the UNIX Server IT resource.

6. From the list at the top of the page, select **Details and Parameters**.

7. Specify values for the parameters of the UNIX Server IT resource. Figure 2-2 shows the Edit IT Resource Details and Parameters page.

Figure 2-2 Edit IT Resource Details and Parameters Page



Edit IT Resource Details and Parameters

You can view additional information about this IT resource : Details and Parameters

IT Resource Name	UNIX Server
IT Resource Type	UNIX Server
Remote Manager	Clear

Parameter	Value
Configuration Lookup	Lookup.UNIX.Configuration
Connector Server Name	UNIX Connector Server
connectionType	SSH
connectorPrompt	#@#
host	idmvm064.us.oracle.com
loginShellPrompt	[#$]
loginUser	uadmin
loginUserpassword	••••••••
passphrase	
port	22
propertyFileName	
rbacAuthorization	false
rbacRoleName	
rbacRolePassword	
sudoAuthorization	false

Table 2-2 describes each parameter of the UNIX Server IT resource.

Table 2-2 Parameters of the UNIX Server IT Resource for the Target System

Parameter	Description
Configuration Lookup	Name of the lookup definition that stores configuration information used during reconciliation and provisioning To use the target system as a target resource, set the following value (default): `Lookup.UNIX.Configuration` To use the target system as a trusted source, set the following value: `Lookup.UNIX.Configuration.Trusted`
Connector Server Name	Name of the IT resource of type "Connector Server" A default IT resource for the Connector Server is created during the connector installation. See Configuring the IT Resource for the Connector Server for information about modifying the default IT resource. By default, this field is blank. If you use a Connector Server, then the default value is: `UNIX Connector Server`
connectionType	Protocol used by the connector to connect to the target system The connector supports the following connection types: • `SSH` - Used for SSH with password-based authentication. • `SSHPUBKEY` - Used for SSH with key-based authentication. • `TELNET` - Used for Telnet connection. Default value: `SSH`

Table 2-2 (Cont.) Parameters of the UNIX Server IT Resource for the Target System

Parameter	Description
connectorPrompt	Shell prompt set by the connector for its operations on the target system Default value: `#@#` **Note:** If this value occurs in user login names, comment fields, directory names, and so on, some connector operations may be affected. In such a case, the value for the connector prompt can be changed to a value that does not occur in the names.
host	Host name or the IP address of the target system computer
loginShellPrompt	Shell prompt that you encounter when you login to the target system using the loginUser account Default value: `[#$]` **Note:** This value is a regular expression. By default, the connector works if the shell prompt on the target system is either # or $. However, if the shell prompt is different, for example >, then you must change the value of this parameter to the actual prompt. To know the loginShellPrompt, perform the following steps on the target system: **a.** Log in to the target system using the user and the password specified in the loginUser and loginUserPassword parameters. Note the login prompt. For example, `#.` **b.** Run the `sh` command. Note the shell prompt, if it is different from the previous prompt. For example, `$.` **c.** Run the `sudo -k` command. Note the shell prompt, if it is different from the previous prompt. For example, `$.` **d.** Run the `sudo -v` command. This will prompt you for the password if loginUser is a SUDO user. Enter the password and continue. Note the shell prompt, if it is different from the previous prompt. For example, `$.` **e.** Run the `sudo -s` command. Note the shell prompt, if it is different from the previous prompt. For example, `$.` For the values shown in the examples, the loginShellPrompt parameter value should be `[#$].` In addition, if the shell prompt displayed in any of the previous steps is similar to `home/jdoe>,` then the prompt is > (not the entire string, `home/jdoe>`).
loginUser	User ID of the administrator to perform connector operations `root` or `jdoe` Here, `jdoe` can be the SUDO user ID, for the SUDO Admin mode. Alternatively, on Solaris, it can be the user ID of the account to which you assign the minimum privileges required to perform connector operations. See Creating a Target System SUDO User Account for Connector Operations for more information.
loginUserpassword	Password of the administrator
passphrase	Passphrase for the key file to use with key based authentication **Note:** You must provide a passphrase if you use key-based authentication.

Table 2-2 (Cont.) Parameters of the UNIX Server IT Resource for the Target System

Parameter	Description
port	Port at which the SSH or Telnet service is running on the server Default value for SSH: 22 Default value for Telnet: 23
propertyFileName	Relative path of the ScriptProperties.properties file of the target system You can leave this field blank if you want to use the default scripts. However, if you want to use custom scripts other than the OOTB scripts, then you must provide a value for this field. The connector will try to determine the path of the properties file by running the `uname -a` command on the target system. If the connector is unable to determine an appropriate value (when an exception is encountered), then it will display the following error message: `Unable to determine UNIX Type. Please provide property file name in IT Resource.` In the case of an error message, enter one of the following values (or a different path if you want to use customized scripts) depending on the target system and the user account: • `scripts/solaris/sudo/ScriptProperties.properties` • `scripts/solaris/nonsudo/ScriptProperties.properties` • `scripts/linux/sudo/ScriptProperties.properties` • `scripts/linux/nonsudo/ScriptProperties.properties` • `scripts/aix/sudo/ScriptProperties.properties` • `scripts/aix/nonsudo/ScriptProperties.properties` • `scripts/hpux/sudo/ScriptProperties.properties` • `scripts/hpux/nonsudo/ScriptProperties.properties`
rbacAuthorization	Indicates whether the user provided in the loginUser parameter is a RBAC user Default value: `false` See Creating an RBAC User Account for Connector Operations on Solaris for more information.
rbacRoleName	If you specify the rbacAuthorization parameter as `true`, then enter the name of the role assigned to the RBAC user. Otherwise, do not specify a value for this parameter.
rbacRolePassword	If you specify the rbacAuthorization parameter as `true`, then enter the password of the role assigned to the RBAC user. Otherwise, do not specify a value for this parameter.
sudoAuthorization	Indicates whether the user provided in the loginUser parameter is a SUDO user Default value: `false`

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

2.3.3 Configuring the IT Resource for the Connector Server

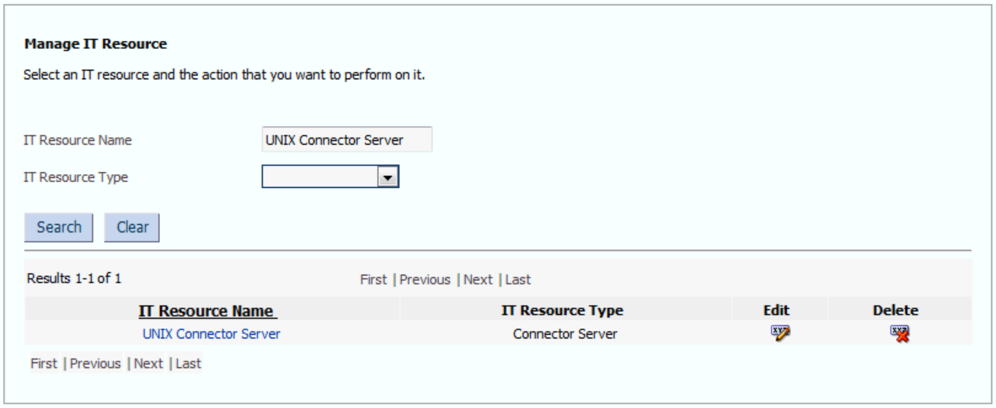
Perform the procedure described in this section only if you have installed the connector bundle in a Connector Server.

The procedure to install the connector bundle in a Connector Server is described in Deploying the Connector Bundle in a Connector Server. During the installation of the connector, a default IT resource for the Connector Server for UNIX is created with the name, UNIX Connector Server.

To configure or modify the IT resource for the Connector Server:

1. Depending on the Oracle Identity Manager release you are using, perform one of the following steps:
   - For Oracle Identity Manager release 11.1.1: Log in to the Administrative and User Console
   - For Oracle Identity Manager release 11.1.2.*x*: Log in to Oracle Identity System Administration
2. If you are using Oracle Identity Manager release 11.1.1, then:
   a. On the Welcome page, click **Advanced** in the upper right corner.
   b. On the Welcome to Oracle Identity Manager Advanced Administration page, in the Configuration region, click **Manage IT Resource**.
3. If you are using Oracle Identity Manager release 11.1.2.*x*, then in the left pane under Configuration, click **IT Resource**.
4. In the IT Resource Name field on the Manage IT Resource page, enter `UNIX Connector Server` and then click **Search.** Figure 2-3 shows the Manage IT Resource page.

Figure 2-3 Manage IT Resource Page for Connector Server IT Resource



Manage IT Resource

Select an IT resource and the action that you want to perform on it.

IT Resource Name: UNIX Connector Server

IT Resource Type:

Results 1-1 of 1

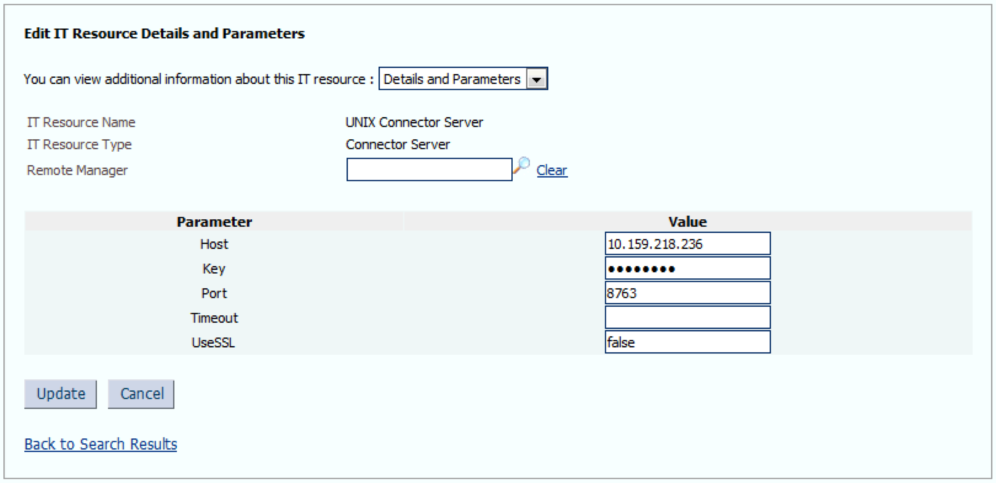
First | Previous | Next | Last

IT Resource Name	IT Resource Type	Edit	Delete
UNIX Connector Server	Connector Server	✎	✗

First | Previous | Next | Last

5. Click the edit icon corresponding to the Connector Server IT resource.
6. From the list at the top of the page, select **Details and Parameters**.
7. Specify values for the parameters of the Connector Server IT resource. Figure 2-4 shows the Edit IT Resource Details and Parameters page.

Figure 2-4 Edit IT Resource Details and Parameters Page for the Connector Server IT Resource



Edit IT Resource Details and Parameters

You can view additional information about this IT resource : Details and Parameters

IT Resource Name	UNIX Connector Server
IT Resource Type	Connector Server
Remote Manager	

Parameter	Value
Host	10.159.218.236
Key	••••••••
Port	8763
Timeout	
UseSSL	false

Table 2-3 provides information about the parameters of the IT resource.

Table 2-3 Parameters of the IT Resource for the UNIX Connector Server

Parameter	Description
Host	Enter the host name or IP address of the computer hosting the Connector Server. Sample value: `HostName`
Key	Enter the key for the Connector Server.
Port	Enter the number of the port at which the Connector Server is listening. By default, this value is blank. You must enter the port number that is displayed on the terminal when you start the Connector Server. For example: `8763`
Timeout	Enter an integer value which specifies the number of milliseconds after which the connection between the Connector Server and Oracle Identity Manager times out. If the value is zero or if no value is specified, the timeout is unlimited. Recommended value: `0`
UseSSL	Enter `true` to specify that you will configure SSL between Oracle Identity Manager and the Connector Server. Otherwise, enter `false`. Default value: `false`

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

2.3.4 Setting up the Lookup Definitions for Connector Configuration

The configuration lookup definitions are created in Oracle Identity Manager when you deploy the connector.

These lookup definitions are either prepopulated with values or you must manually enter values in them after the connector is deployed. The lookup definitions are as follows:

- Lookup.UNIX.Configuration

This lookup definition holds connector configuration entries that are used during reconciliation and provisioning operations.

- Lookup.UNIX.Configuration.Trusted

  This lookup definition holds connector configuration entries when the target system is configured as a trusted source.

Table 2-4 lists the default entries in these lookup definitions.

Table 2-4 Entries in the UNIX Configuration Lookup Definitions

Code Key	Decode	Description
Bundle Name	`org.identityconnectors.genericunix`	Name of the connector bundle package Do not modify this entry.
Bundle Version	1.0.0	Version of the connector bundle class Do not modify this entry.
commandTimeout	100000	Time in milliseconds for which the connector would wait for a response from the target systemAfter this time, the connector will throw timeout exception. You can increase this value if you encounter a 'command timed out' exception for connector operations.
configPropertiesOnScripts	`moveHomeDirContents,shadow,defaultHomeBaseDir, defaultPriGroup,defaultShell,nisPwdDir, nisBuildDirectory,removeHomeDirContents,forceDeleteUserHome,syncToken, mirrorFilesLocation,connectorPrompt`	Lists the properties that are sent to the scripts For example, if during provisioning, you want to set a default shell for the users. To do so:1. Verify that the 'defaultShell' property is a part of this list.2. Add an entry for this property in this lookup.Set the Code Key value to defaultShell.Set the Decode value to `/bin/sh`. If the target-specific script supports the defaultShell property, it would be set. Not all scripts support all the attributes listed in the Decode column. You must manually check the script contents for supported attributes.
Connector Name	`org.identityconnectors.genericunix.GenericUnixConnector`	Name of the connector class Do not modify this entry.

Table 2-4 (Cont.) Entries in the UNIX Configuration Lookup Definitions

Code Key	Decode	Description

mirrorFilesLocation	• For Lookup.UNIX.Configuration:
	`/etc/connector_mirror_files`
	• For Lookup.UNIX.Configuration.Trusted:
	`/etc/connector_mirror_files_trusted`
		Directory used by the connector to store copies of the /etc/passwd and shadow files
		Note: This directory has to be manually created on the target before performing reconciliation.If you want to specify a different directory, ensure that the directory exists on the target system and the loginUser has read-write access to the directory.

moveHomeDirContents	Default value: `true`	Specifies whether the old home directory contents should be moved to the new directory location when changing the Home Directory. You can enter `true` or `false` as the Decode value.

passwordExpectExpressions	`new[\s](unix[\s])?password:,new[\s](unix[\s])?password([\s]again)?:`
	Note: The third-party library, Expect4j, matches these expected expressions to the actual contents of the console output on the UNIX target system.
	Therefore, you must ensure that these fields have correct values. Incorrect values may impact the connector operations.
		Regular expression for the two comma-separated password prompts that are displayed on the target system when a password is set for a user
		If the regular expression does not work on your target system, then you can specify the exact prompts in this lookup entry.
		For example, if you set the password for a user and you get the following prompt:
		`Enter Password for USER1:`
		`Re-enter Password for USER1:`
		Then, you can set the Decode value as follows:
		`enter password,re-enter password`

Table 2-4 (Cont.) Entries in the UNIX Configuration Lookup Definitions

Code Key	Decode	Description

prePasswdExpectExpression
Note: This entry does not exist by default. You must add it to the configuration lookup if your target displays extra prompts such as the prompt shown in the description column.
	For the example shown in the description column, the sample value for the choice p will be:
	`Enter choice here:,p`
		Some target systems such as HP-UX may display additional options before prompting for passwords while running the `passwd` command.
		For example:
		`Do you want (choose one letter only):`
		`pronounceable passwords generated for you (g)`
		`a string of letters generated (l) ?`
		`to pick your passwords (p)?`
		`Enter choice here:`
		In such a case, you can enter these Code Key and Decode entries to the lookup definition.

privateKey[LOADFROMURL] **Note:** This entry does not exist by default. You must add it to the configuration lookup if you want to enable key-based authentication.	Sample value: `file:///scratch/files/jars/unix/id_rsa_linux`	Path to the id_rsa file
rbacRoleExpectExpressions **Note:** This entry is applicable only to Lookup.UNIX.Configuration.	`password:,[$#]` **Note:** The third-party library, Expect4j, matches these expected expressions to the actual contents of the console output on the UNIX target system. Therefore, you must ensure that these fields have correct values. Incorrect values may impact the connector operations.	Regular expressions for the two comma-separated prompts The first prompt (password:) is the password prompt displayed on the Solaris target system when you enter the SUDO mode for the RBAC role. If the target system displays a different prompt, then you must change this password prompt. The second prompt ([$#]) is the shell prompt displayed after running the previous command in SUDO mode. If the target system displays a different prompt, then you must change this shell prompt.
sudoPasswdExpectExpression	`password` **Note:** The third-party library, Expect4j, matches these expected expressions to the actual contents of the console output on the UNIX target system. Therefore, you must ensure that these fields have correct values. Incorrect values may impact the connector operations.	Regular expression for the password prompt displayed on the target system when you enter the SUDO mode If the target system displays a different prompt, then you must change this password prompt.
supportedLanguage	Bourne	Shell script language supported on the target system

Table 2-4 (Cont.) Entries in the UNIX Configuration Lookup Definitions

Code Key	Decode	Description
targetDateFormat	yyyy-MM-dd **Note:** You must ensure that this is the correct Java date format for the target system. An incorrect format may affect provisioning of the Expire Date attribute. For information about the date format, see `http://docs.oracle.com/javase/6/docs/api/java/text/SimpleDateFormat.html` and `http://docs.oracle.com/javase/6/docs/api/java/text/DateFormat.html`.	Format of the date on the target system
telnetAuthenticationPrompts **Note:** This entry is applicable for Telnet connection, when the connectionType parameter of the IT Resource is set to TELNET.	`login:,Password:` **Note:** The third-party library, Expect4j, matches these expected expressions to the actual contents of the console output on the UNIX target system. Therefore, you must ensure that these fields have correct values. Incorrect values may impact the connector operations.	The login and password prompts on a target system using Telnet connection.

User Configuration Lookup	• For Lookup.UNIX.Configuration:
	`Lookup.UNIX.UM.Configuration`
	• For Lookup.UNIX.Configuration.Trusted:
	`Lookup.UNIX.UM.Configuration.Trusted`
		Name of the lookup definition that contains user-specific configuration properties
		Do not modify this entry.

whitelistRegex	`[A-Za-z0-9_//]*`	Specifies characters that are allowed as a part of the field values For example: The regular expression, `[A-Za-z0-9_//]*`, allows all alphanumeric, underscore, and forward slash characters. You can add more characters if needed. **Note:** For information about the supported regular expressions, you can refer to a guide such as `http://www.zytrax.com/tech/web/regex.htm` This regular expression does not apply to the GECOS field, which can have any characters.

Table 2-4 (Cont.) Entries in the UNIX Configuration Lookup Definitions

Code Key	Decode	Description
isSudoWithNoPasswd **Note:** This entry does not exist by default. If you want the support for SUDO user with NoPasswd, then you must add it to the configuration lookup definition.	true/false	true: If NoPasswd is configured for SUDO user. If not, false.
defaultConnectorShell	`sh` **Note:** If you are using RBAC, then the decode value must be changed from `sh` to `pfsh.`	This is the defaultShell used for connector operations. Do not modify this entry unless you are using RBAC

2.3.5 Setting up the Lookup Definition for Connection Pooling

By default, this connector uses the ICF connection pooling.

Table 2-5 lists the connection pooling properties, their description, and default values set in ICF:

Table 2-5 Connection Pooling Properties

Property	Description
Pool Max Idle	Maximum number of idle objects in a pool. Default value: `10`
Pool Max Size	Maximum number of connections that the pool can create. Default value: `10`
Pool Max Wait	Maximum time, in milliseconds, the pool must wait for a free object to make itself available to be consumed for an operation. Default value: `150000`
Pool Min Evict Idle Time	Minimum time, in milliseconds, the connector must wait before evicting an idle object. Default value: `120000`
Pool Min Idle	Minimum number of idle objects in a pool. Default value: `1`

If you want to modify the connection pooling properties to use values that suit requirements in your environment, then:

1. Log in to the Design Console.
2. Expand **Administration,** and then double-click **Lookup Definition.**
3. Search for and open the **Lookup.UNIX.Configuration** lookup definition.
4. On the Lookup Code Information tab, click **Add.**

   A new row is added.
5. In the **Code Key** column of the new row, enter `Pool Max Idle.`

6. In the **Decode** column of the new row, enter a value corresponding to the Pool Max Idle property.
7. Repeat Steps 4 through 6 for adding each of the connection pooling properties listed in Table 2-5.
8. Click the save icon.

2.3.6 Setting up the Lookup Definitions for User Operations

The user management lookup definitions are created in Oracle Identity Manager when you deploy the connector. These lookup definitions are either prepopulated with values or values must be manually entered in them after the connector is deployed. The lookup definitions are as follows:

- Lookup.UNIX.UM.Configuration
- Lookup.UNIX.UM.Configuration.Trusted

2.3.6.1 Lookup.UNIX.UM.Configuration

The Lookup.UNIX.UM.Configuration lookup definition holds configuration entries that are specific to the user object type. This lookup definition is used during user management operations.

Table 2-6 lists the default entries in this lookup definition.

Table 2-6 Entries in the Lookup.UNIX.UM.Configuration

Code Key	Decode	Description
Provisioning Attribute Map	Lookup.UNIX.UM.ProvAttrMap	This entry holds the name of the lookup definition that maps process form fields and target system attributes. See Lookup.UNIX.UM.ProvAttrMap for more information about this lookup definition.
Recon Attribute Map	Lookup.UNIX.UM.ReconAttrMap	This entry holds the name of the lookup definition that maps resource object fields and target system attributes. See Lookup.UNIX.UM.ReconAttrMap for more information about this lookup definition.
Recon Transformation Lookup **Note:** This entry does not exist by default. You must add it if you want to enable transformation during reconciliation.	Lookup.UNIX.UM.ReconTransformation	This entry holds the name of the lookup definition that is used to configure transformation of attribute values that are fetched from the target system during user reconciliation. See Configuring Transformation of Data During User Reconciliation for more information about adding entries in this lookup definition.
Recon Validation Lookup **Note:** This entry does not exist by default. You must add it if you want to enable validation during reconciliation.	Lookup.UNIX.UM.ReconValidation	This entry holds the name of the lookup definition that is used to configure validation of attribute values that are fetched from the target system during reconciliation. See Configuring Validation of Data During Reconciliation and Provisioning for more information about adding entries in this lookup definition.

Table 2-6 (Cont.) Entries in the Lookup.UNIX.UM.Configuration

Code Key	Decode	Description
Provisioning Validation Lookup **Note:** This entry does not exist by default. You must add it if you want to enable validation during provisioning.	Lookup.UNIX.UM.ProvValidation	This entry holds the name of the lookup definition that is used to configure validation of attribute values entered on the process form during provisioning operations. See Configuring Validation of Data During Reconciliation and Provisioning for more information about adding entries in this lookup definition.
Recon Exclusion Lookup **Note:** This entry does not exist by default. You must add it if you want to enable resource exclusions during reconciliation.	Lookup.UNIX.UM.ProvExclusionList	This entry holds the name of the lookup definition that is used to configure resource exclusion lists during reconciliation. See Configuring Resource Exclusion Lists for more information.
Provisioning Exclusion Lookup **Note:** This entry does not exist by default. You must add it if you want to enable resource exclusions during provisioning.	Lookup.UNIX.UM.ReconExclusionList	This entry holds the name of the lookup definition that is used to configure resource exclusion lists during provisioning operations. See Configuring Resource Exclusion Lists for more information about adding entries in this lookup definition.

2.3.6.2 Lookup.UNIX.UM.Configuration.Trusted

The Lookup.UNIX.UM.Configuration.Trusted lookup definition holds configuration entries that are specific to the user object type when the target system is configured as a trusted source. This lookup definition is used during user management operations.

Table 2-7 lists the default entries in this lookup definition.

Table 2-7 Entries in the Lookup.UNIX.UM.Configuration.Trusted

Code Key	Decode	Description
Recon Attribute Defaults	Lookup.UNIX.UM.ReconAttrMap.TrustedDefaults	This entry holds the name of the lookup definition that maps process form fields and target system attributes. See Lookup.UNIX.UM.ReconAttrMap.TrustedDefaults for more information about this lookup definition.
Recon Attribute Map	Lookup.UNIX.UM.ReconAttrMap.Trusted	This entry holds the name of the lookup definition that maps resource object fields and target system attributes. See Lookup.UNIX.UM.ReconAttrMap.Trusted for more information about this lookup definition.

2.3.7 Setting up the Lookup Definitions for Attribute Mappings

The attribute mapping lookup definitions are created in Oracle Identity Manager when you deploy the connector. These lookup definitions are either prepopulated with values or values must be manually entered in them after the connector is deployed. The lookup definitions are as follows:

- Lookup.UNIX.UM.ProvAttrMap
- Lookup.UNIX.UM.ReconAttrMap
- Lookup.UNIX.UM.ReconAttrMap.Trusted
- Lookup.UNIX.UM.ReconAttrMap.TrustedDefaults
- Lookup.UNIX.YesNo.Options

2.3.7.1 Lookup.UNIX.UM.ProvAttrMap

The Lookup.UNIX.UM.ProvAttrMap lookup definition holds mappings between process form fields (Code Key values) and target system attributes (Decode values) used during provisioning operations.

You can add entries to this lookup if you want to map new target system attributes for provisioning. See Adding Custom Attributes for Provisioning for more information.

Table 2-8 lists the default entries in this lookup definition.

Table 2-8 Entries in the Lookup.UNIX.UM.ProvAttrMap

Code Key	Decode
Create home directory	CREATE_HOME_DIR
Expire Date[DATE]	EXP_DATE##DATE##
GECOS	COMMENTS##COMMENTS##
Home Directory	HOME_DIR
Inactive Days	INACTIVE
Password	__PASSWORD__
Primary Group[LOOKUP]	PGROUP
ReturnValue	__UID__
Skeleton Directory	SKEL_DIR
UD_UNIX_CH~Secondary Group[LOOKUP]	SECONDARYGROUP
UID	USID
User Login	__NAME__ **Note:** This value is a target system attribute, used by the connector for internal purposes.
User Shell[LOOKUP]	USER_SHELL

2.3.7.2 Lookup.UNIX.UM.ReconAttrMap

The Lookup.UNIX.UM.ReconAttrMap lookup definition holds mappings between resource object fields (Code Key values) and target system attributes (Decode values) used during reconciliation operations.

You can add entries to this lookup definition if you want to map new target system attributes for reconciliation. See Adding Custom Attributes for Target Resource Reconciliation for more information.

Table 2-9 lists the default entries in this lookup definition.

Table 2-9   Entries in the Lookup.UNIX.UM.ReconAttrMap

Code Key	Decode
Create home directory	CREATE_HOME_DIR
Expire Date[DATE]	EXP_DATE##DATE##
GECOS	COMMENTS
Home Directory	HOME_DIR
Inactive Days	INACTIVE
Primary Group[LOOKUP]	PGROUP
ReturnValue	__UID__ **Note:** This value is a target system attribute, used by the connector for internal purposes.
Secondary Groups~Secondary Group[LOOKUP]	SECONDARYGROUP
Status	__ENABLE__
UID	USID
User Login	__NAME__
User Shell[LOOKUP]	USER_SHELL

2.3.7.3 Lookup.UNIX.UM.ReconAttrMap.Trusted

The Lookup.UNIX.UM.ReconAttrMap.Trusted lookup definition holds mappings between resource object fields (Code Key values) and target system attributes (Decode values) used during reconciliation operations. This lookup definition is used during user management operations when the target system is configured as a trusted source.

Table 2-10 lists the default entries in this lookup definition.

Table 2-10   Entries in the Lookup.UNIX.UM.ReconAttrMap.Trusted

Code Key	Decode
LastName	__NAME__
TrustedStatus[TRUSTED]	__ENABLE__

Table 2-10 (Cont.) Entries in the Lookup.UNIX.UM.ReconAttrMap.Trusted

Code Key	Decode
User ID	__UID__

2.3.7.4 Lookup.UNIX.UM.ReconAttrMap.TrustedDefaults

The Lookup.UNIX.UM.ReconAttrMap.TrustedDefaults lookup definition is used when the target system is configured as a trusted source.

These entries are OIM User attributes. The fields are not mapped to any UNIX target system fields. The default values are set for these fields in this lookup.

Table 2-11 lists the default entries in this lookup definition.

Table 2-11 Entries in the Lookup.UNIX.UM.ReconAttrMap.TrustedDefaults

Code Key	Decode
Organization	Xellerate Users
Role	Full-Time
User Type	End-User

2.3.7.5 Lookup.UNIX.YesNo.Options

The Lookup.UNIX.YesNo.Options lookup definition is used to set value for a process form field that is boolean in nature, such as true or false. Do *not* modify the entries in this lookup definition.

This lookup contains the following entries by default:

Code Key	Decode
false	false
true	true

2.3.8 Enabling Logging

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

> **Note:**
>
> In an Oracle Identity Manager cluster, perform this procedure on each node of the cluster. Then, restart each node.

- 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.

These log levels are mapped to ODL message type and level combinations as shown in

Table 2-12 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
FINE	TRACE:1
FINER	TRACE:16
FINEST	TRACE:32

The configuration file for OJDL is logging.xml, which is located at the following path:

DOMAIN_HOME/config/fmwconfig/servers/*OIM_SERVER*/logging.xml

Here, *DOMAIN_HOME* and *OIM_SERVER* are the domain name and server name specified during the installation of Oracle Identity Manager.

To enable logging in Oracle WebLogic Server:

1. Edit the logging.xml file as follows:

   a. Add the following blocks in the file:

```
<log_handler name='unix-handler' level='[LOG_LEVEL]'
class='oracle.core.ojdl.logging.ODLHandlerFactory'>
<property name='logreader:' value='off'/>
     <property name='path' value='[FILE_NAME]'/>
     <property name='format' value='ODL-Text'/>
     <property name='useThreadName' value='true'/>
     <property name='locale' value='en'/>
     <property name='maxFileSize' value='5242880'/>
     <property name='maxLogSize' value='52428800'/>
     <property name='encoding' value='UTF-8'/>
   </log_handler>

<logger name="ORG.IDENTITYCONNECTORS.GENERICUNIX" level="[LOG_LEVEL]"
useParentHandlers="false">
     <handler name="unix-handler"/>
     <handler name="console-handler"/>
   </logger>
```

b. Replace both occurrences of **`[LOG_LEVEL]`** with the ODL message type and level combination that you require. Table 2-12 lists the supported message type and level combinations.

   Similarly, replace **`[FILE_NAME]`** with the full path and name of the log file in which you want log messages to be recorded.

   The following blocks show sample values for **`[LOG_LEVEL]`** and **`[FILE_NAME]`** :

```
<log_handler name='unix-handler' level='NOTIFICATION:1'
class='oracle.core.ojdl.logging.ODLHandlerFactory'>
<property name='logreader:' value='off'/>
     <property name='path' value='F:\MyMachine\middleware\user_projects
\domains\base_domain1\servers\oim_server1\logs\oim_server1-
diagnostic-1.log'/>
     <property name='format' value='ODL-Text'/>
     <property name='useThreadName' value='true'/>
     <property name='locale' value='en'/>
     <property name='maxFileSize' value='5242880'/>
     <property name='maxLogSize' value='52428800'/>
     <property name='encoding' value='UTF-8'/>
   </log_handler>

<logger name="ORG.IDENTITYCONNECTORS.GENERICUNIX" level="NOTIFICATION:1"
useParentHandlers="false">
     <handler name="telnetssh-handler"/>
     <handler name="console-handler"/>
   </logger>
```

With these sample values, when you use Oracle Identity Governance, all messages generated for this connector that are of a log level equal to or higher than the `NOTIFICATION:1` level are recorded in the specified file.

2. Save and close the file.

3. Set the following environment variable to redirect the server logs to a file:

   For Microsoft Windows:

   ```
   set WLS_REDIRECT_LOG=FILENAME
   ```

   For UNIX:

   ```
   export WLS_REDIRECT_LOG=FILENAME
   ```

   Replace ***FILENAME*** with the location and name of the file to which you want to redirect the output.

4. Restart the application server.

2.3.9 Changing to the Required Input Locale

Changing to the required input locale (language and country setting) involves installing the required fonts and setting the required input locale.

> **Note:**
>
> In an Oracle Identity Manager cluster, you must perform this step on each node of the cluster. Then, restart each node.

You may require the assistance of the system administrator to change to the required input locale.

2.3.10 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 the 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.

> **Note:**
>
> In an Oracle Identity Manager cluster, you must perform this step on each node of the cluster. Then, restart each node.

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.
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.

2.3.11 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.

> **Note:**
>
> Perform the procedure described in this section only if you are using Oracle Identity Manager release 11.1.2.*x* or later and you want to localize UI form field labels.

To localize field label 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:
   - For Oracle Identity Manager 11g Release 2 PS2 (11.1.2.2.0) or later releases:

     SAVED_LOCATION\xliffBundles\oracle\iam\ui\runtime\BizEditorBundle_en.xlf
   - For releases prior to Oracle Identity Manager 11*g* Release 2 PS2 (11.1.2.2.0):

     SAVED_LOCATION\xliffBundles\oracle\iam\ui\runtime\BizEditorBundle.xlf
6. Edit the BizEditorBundle.xlf file in the following manner:
   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 French:

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

c. Search for the application instance code. This procedure shows a sample edit for UNIX application instance. The original code is:

```
<trans-unit id="$
{adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResourceBundle']
['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity.userEO.UD_U
NIX_GRPNAME__c_description']}">
<source>Primary Group</source>
</target>
</trans-unit>
<trans-unit
id="sessiondef.oracle.iam.ui.runtime.form.model.UNIX.entity.UNIXEO.UD_UNIX_GR
PNAME__c_LABEL">
<source>Primary Group</source>
</target>
</trans-unit>
```

d. Open the resource file from the connector package, for example UNIX_fr.properties, and get the value of the attribute from the file, for example, global.udf.UD_UNIX_GRPNAME= Groupe principal.

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_U
NIX_GRPNAME__c_description']}">
<source> Primary Group</source>
<target> Groupe principal</target>
</trans-unit>
<trans-unit
id="sessiondef.oracle.iam.ui.runtime.form.model.UNIX.entity.UNIXEO.UD_UNIX_GR
PNAME__c_LABEL">
<source> Primary Group</source>
<target> Groupe principal</target>
</trans-unit>
```

f. Repeat Steps 6.a through 6.d 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_fr.xlf.

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

> **See Also:**
>
> Deploying and Undeploying Customizations in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance* for more information about exporting and importing metadata files

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

2.4 Upgrading the Connector

If you have already deployed an earlier release of this connector, then upgrade the connector to the current release 11.1.1.7.0.

> **Note:**
>
> Before you perform the upgrade procedure:
>
> - It is strongly recommended that you create a backup of the Oracle Identity Manager database. Refer to the database documentation for information about creating a backup.
> - As a best practice, first perform the upgrade procedure in a test environment.
> - If you have added custom attributes to an earlier release of the connector, you must retain and configure all the custom attributes after the upgrade procedure.

The following sections discuss the procedure to upgrade the connector:

- Preupgrade Steps
- Upgrade the UNIX Connector from Release 11.1.1.6.0 to 11.1.1.7.0
- Upgrade Steps
- Postupgrade Steps

2.4.1 Preupgrade Steps

Perform the following preupgrade steps:

1. Perform a reconciliation run to fetch all latest updates to Oracle Identity Manager.
2. Perform the preupgrade procedure documented.
3. Define the source connector (an earlier release of the connector that must be upgraded) in Oracle Identity Manager. You define the source connector to update the Deployment Manager XML file with customization changes made to the connector.

2.4.2 Upgrade the UNIX Connector from Release 11.1.1.6.0 to 11.1.1.7.0

You can upgrade the UNIX connector from release 11.1.1.6.0 to this release of the connector.

To do so, perform the following procedures:

- Setting Entitlement Tagging.
- Setting IT Resource, Account ID, and Account Name Tagging.
- Setting the Status of Task to Object Status Mapping of the Secondary Group Update Process Task to None .
- Updating the Connector Bundle.

2.4.2.1 Setting Entitlement Tagging

To set entitlement tagging for secondary groups child form (UD_UNIX_CH), do the following:

1. Log in to the Oracle Identity Manager Design Console.
2. Expand **Development Tools** and then double-click **Form Designer.**
3. Enter the name of the UNIX child form, `UD_UNIX_CH,` in the Table Name field and click the **Query for records** button.
4. Click **Create New Version.**
5. In the Create a New Version dialog box, specify the version name in the **Label** field, save the changes, and then close the dialog box.
6. From the **Current Version** list, select the newly created version.
7. Click the **Properties** tab.
8. Select the Secondary Group field, and click **Add Property.**
9. From the Property Name list, select **Entitlement.**
10. In the Property Value field, enter `true.`
11. Click **Make Version Active.**

2.4.2.2 Setting IT Resource, Account ID, and Account Name Tagging

To set IT resource, Account ID, and Account Name tagging in the process form (UD_UNIX), do the following:

1. In the Oracle Identity Manager Design Console, expand **Development Tools** and then double-click **Form Designer.**
2. Enter the name of the parent form, `UD_UNIX,` in the Table Name field and click the **Query for records** button.
3. Click **Create New Version.**
4. In the Create a New Version dialog box, specify the version name in the **Label** field, save the changes, and then close the dialog box.

5. From the **Current Version** list, select the newly created version.
6. Click the **Properties** tab.
7. Select the IT Resource field, and click **Add Property.**
8. From the Property Name list, select **ITResource.**
9. In the Property Value field, enter `true`.
10. Select the User Login field, and click **Add Property.**
11. From the Property Name list, select **AccountName.**
12. In the Property Value field, enter `true`.
13. Select the User Login field, and click **Add Property.**
14. From the Property Name list, select **AccountID.**
15. In the Property Value field, enter `true`.
16. Update the parent form to add the child form created in Step 1.
17. Click **Make Version Active.**
18. Recreate the form in the user interface (UI) and update the application instance with the new form as described in Updating an Existing Application Instance with a New Form .

2.4.2.3 Setting the Status of Task to Object Status Mapping of the Secondary Group Update Process Task to None

To set the status of task to object status mapping of the secondary group update process task to none, do the following:

1. In the Oracle Identity Manager Design Console, expand **Process Management** and then double-click **Process definition.**
2. In the Name field, enter `UNIX` and then click the **Query for records** button.
3. Under Tasks, open the **Secondary Group Update** task.
4. In the Task to Object Status Mapping tab, change the Object Status of status C from Provisioned to `None`.

2.4.2.4 Updating the Connector Bundle

Update the connector bundle in the Oracle Identity Manager database with the latest bundle JAR from this release.

2.4.3 Upgrade Steps

Depending on the environment in which you are upgrading the connector, perform one of the following steps:

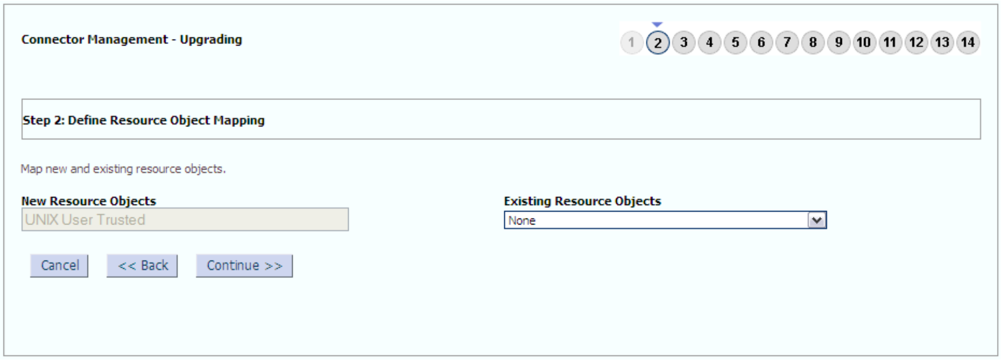
- Staging Environment

  Perform the upgrade procedure by using the wizard mode.
- Production Environment

  Perform the upgrade procedure by using the silent mode.

See Procedure to Upgrade a Connector in *Oracle Fusion Middleware Administering Oracle Identity Manager* for detailed information about the wizard and silent modes.

The following sample screenshots show the connector artifacts to be mapped between the new and the old connectors:



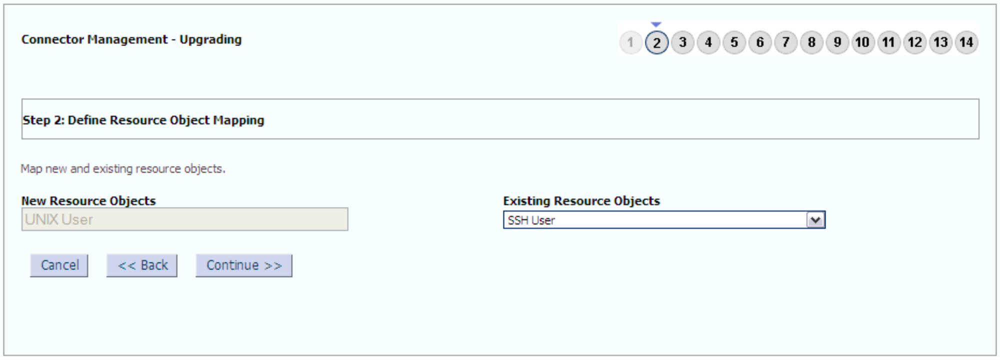
Connector Management - Upgrading

Step 2: Define Resource Object Mapping

Map new and existing resource objects.

New Resource Objects: UNIX User Trusted

Existing Resource Objects: None



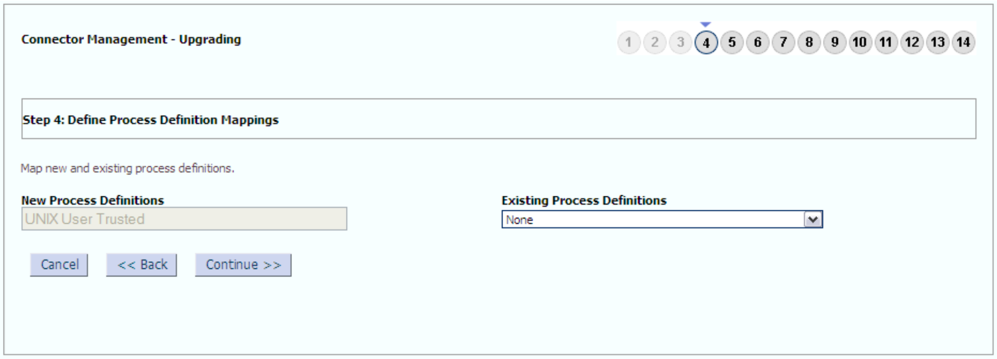
Connector Management - Upgrading

Step 2: Define Resource Object Mapping

Map new and existing resource objects.

New Resource Objects: UNIX User

Existing Resource Objects: SSH User



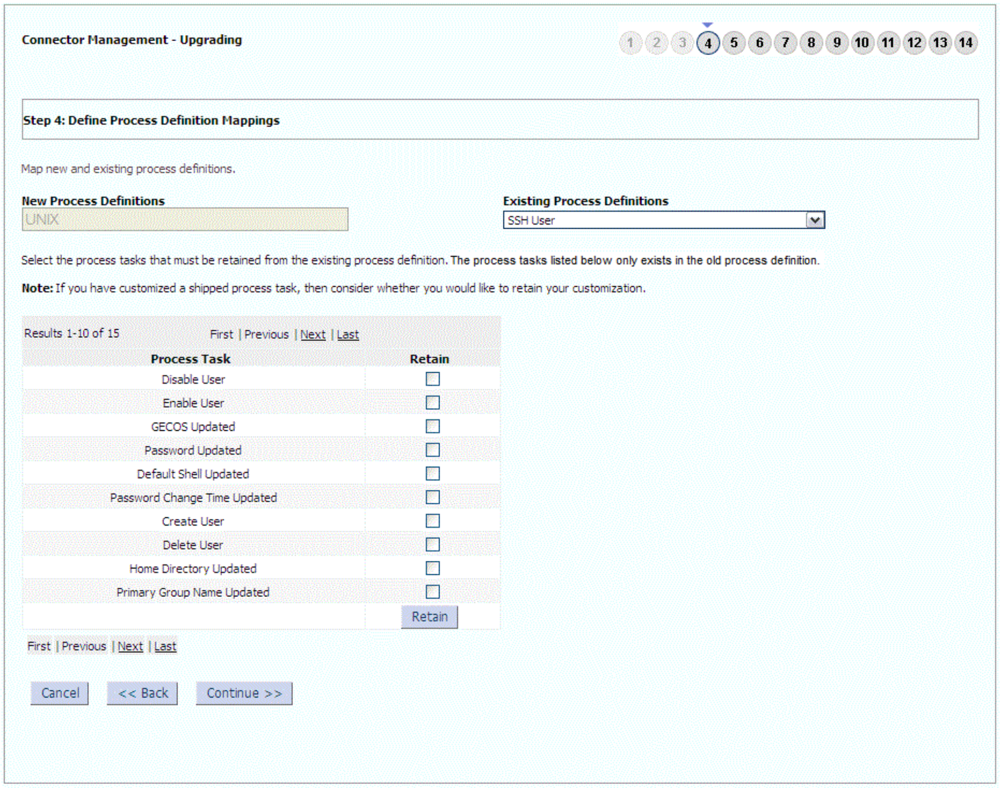
Connector Management - Upgrading

Step 4: Define Process Definition Mappings

Map new and existing process definitions.

New Process Definitions: UNIX User Trusted

Existing Process Definitions: None



Connector Management - Upgrading

Step 4: Define Process Definition Mappings

Map new and existing process definitions.

New Process Definitions: UNIX

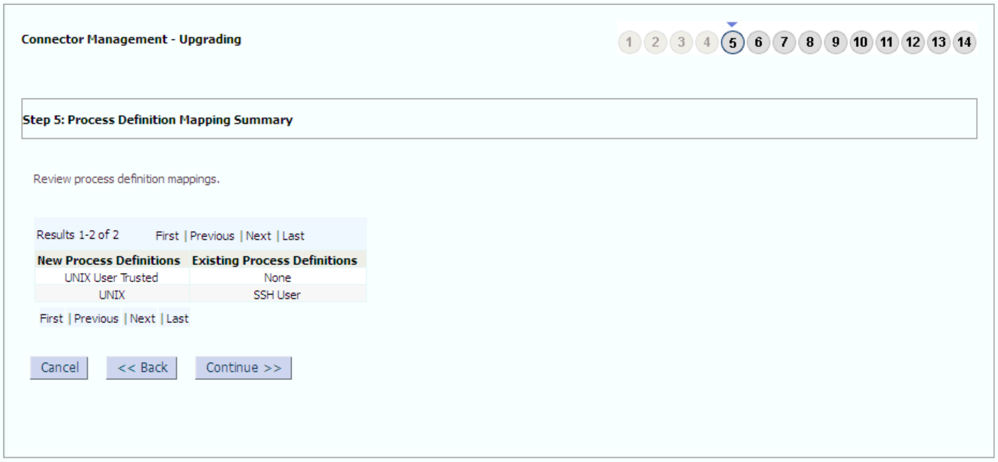
Existing Process Definitions: SSH User

Select the process tasks that must be retained from the existing process definition. The process tasks listed below only exists in the old process definition.

Note: If you have customized a shipped process task, then consider whether you would like to retain your customization.

Results 1-10 of 15 First | Previous | Next | Last

Process Task	Retain
Disable User	□
Enable User	□
GECOS Updated	□
Password Updated	□
Default Shell Updated	□
Password Change Time Updated	□
Create User	□
Delete User	□
Home Directory Updated	□
Primary Group Name Updated	□



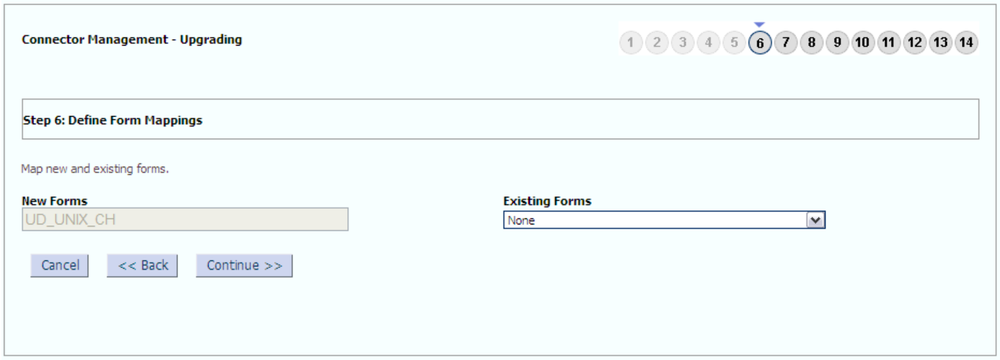
Connector Management - Upgrading

Step 5: Process Definition Mapping Summary

Review process definition mappings.

Results 1-2 of 2 First | Previous | Next | Last

New Process Definitions	Existing Process Definitions
UNIX User Trusted	None
UNIX	SSH User



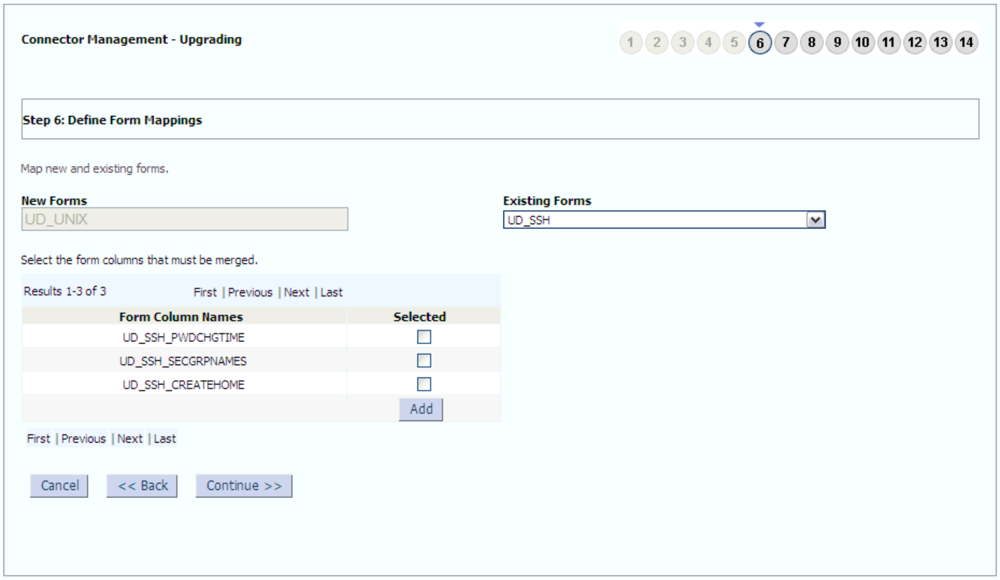
Connector Management - Upgrading

Step 6: Define Form Mappings

Map new and existing forms.

New Forms: UD_UNIX_CH

Existing Forms: None


1
2
3
4
5
6
7
8
9
10
11
12
13
14
First | Previous | Next | Last
Add
First | Previous | Next | Last
Cancel
<< Back
Continue >>

Connector Management - Upgrading

Step 6: Define Form Mappings

Map new and existing forms.

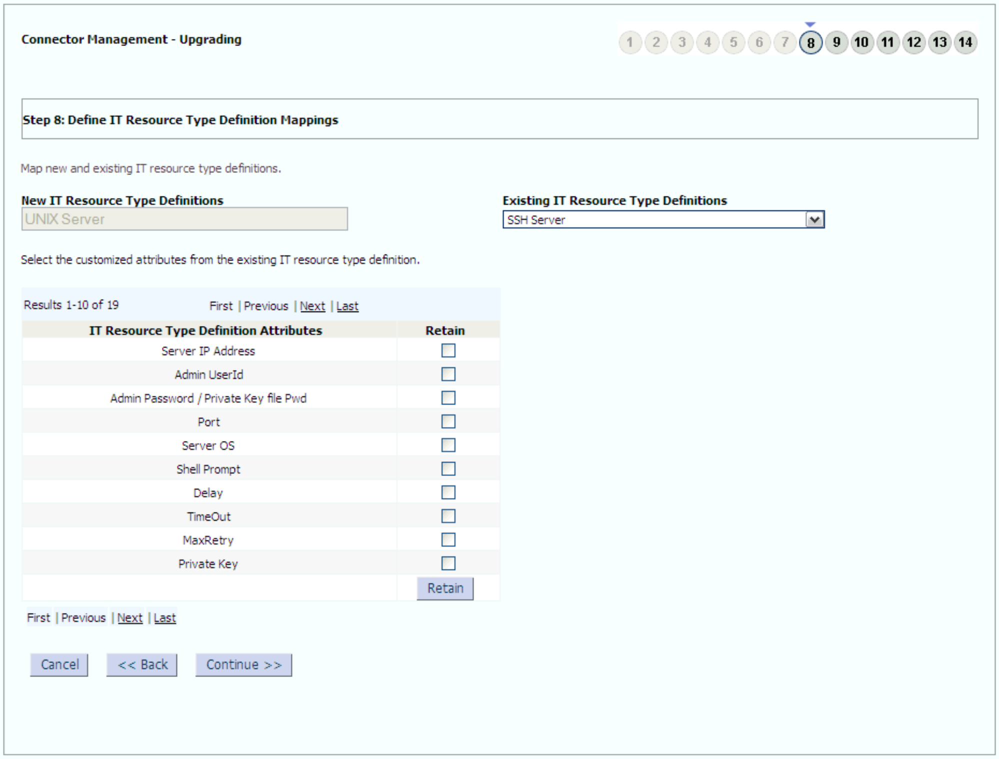
New Forms: UD_UNIX

Existing Forms: UD_SSH

Select the form columns that must be merged.

Results 1-3 of 3

Form Column Names	Selected
UD_SSH_PWDCHGTIME	☐
UD_SSH_SECGRPNAMES	☐
UD_SSH_CREATEHOME	☐



Connector Management - Upgrading

Step 8: Define IT Resource Type Definition Mappings

Map new and existing IT resource type definitions.

New IT Resource Type Definitions: UNIX Server

Existing IT Resource Type Definitions: SSH Server

Select the customized attributes from the existing IT resource type definition.

Results 1-10 of 19 First | Previous | Next | Last

IT Resource Type Definition Attributes	Retain
Server IP Address	□
Admin UserId	□
Admin Password / Private Key file Pwd	□
Port	□
Server OS	□
Shell Prompt	□
Delay	□
TimeOut	□
MaxRetry	□
Private Key	□

2.4.4 Postupgrade Steps

Perform the following postupgrade steps:

1. If you are using Oracle Identity Manager release 11.1.2.*x* or later, you must create a new UI form and attach it to an existing application instance to view the user-defined fields (UDFs or custom attributes). For more information about UDFs, see Configuring Custom Attributes in *Oracle Fusion Middleware Administering Oracle Identity Manager*.
2. Modify the parent form and the child form as follows:
   a. Create a new version of the parent form, UD_UNIX, and make it active.

      For example: v_11.1.1.7.2

   b. Ensure that the child form, UD_UNIX_CH, is linked to the parent form UD_UNIX in the Design Console.

   > **Note:**
   >
   > You must perform these steps as a workaround for the known issue where the parent form is not linked to the child form after upgrading the connector. This issue is also described under Bug 13690646 in Known Issues.

3. Re-configure the IT resource of the source connector (an earlier release of the connector that must be upgraded). See Configuring the IT Resource for the Target System for information.

4. Run the Form Version Control (FVC) utility to manage data changes on a form after an upgrade operation. To do so:

    a. In a text editor, open the fvc.properties file located in the *OIM_DC_HOME* directory and include the following entries:

    ```
    FormName;UD_UNIX
    FromVersion;3
    ToVersion;v_11.1.1.7.2
    ParentParent;UD_UNIX_USERLOGIN;UD_UNIX_RETURNVALUE
    Parent;UD_UNIX_CREATE_HOME_DIR;false
    ```

    > **Note:**
    >
    > The value of the ToVersion field must match the version of the child form created in Step 3.a.

    b. Run the FVC utility. This utility is copied into the following directory when you install the design console:

    For Microsoft Windows:

    OIM_DC_HOME/fvcutil.bat

    For UNIX:

    OIM_DC_HOME/fvcutil.sh

    When you run this utility, you are prompted to enter the login credentials of the Oracle Identity Manager administrator, and the logger level and log file location.

5. Run the PostUpgradeScriptUnix.sql script as follows:

    a. Connect to the Oracle Identity Manager database by using the OIM User credentials.

    b. Run the PostUpgradeScriptUnix.sql located in the *OIM_HOME*/server/ConnectorDefaultDirectory/UNIX_Package/Upgrade directory.

6. Setup incremental reconciliation as follows:

    a. On the target system, copy the password mirror file (`/etc/passwd1`), the shadow mirror file (`/etc/shadow1`), and the group file (`/etc/group`) to the location specified by the **mirrorFilesLocation** attribute (`/etc/connector_mirror_files`) in the configuration lookup definition (`Lookup.UNIX.Configuration`).

    b. Get the current date and time on the target system by running one of the following commands:

    For Linux, use `$(date '+%d%m%Y%s%N')`

    For Solaris, use `(date '+%m%d%y%H%M%S'$random)`

    For HPUX and AIX, use `($(date '+%m%d%Y%S')$RANDOM)`

    c. Save this value as **syncToken.**

    For example, `syncToken = '090420121333955808939929000'`

    d. Rename the **/etc/connector_mirror_files/passwd1** file to *SYNCTOKEN*.passwd.

For example: `/etc/connector_mirror_files/ 0904201213333955808939929000.passwd`.

e. Rename the **/etc/connector_mirror_files/shadow1** file to *SYNCTOKEN*.shadow.

   For example: `/etc/connector_mirror_files/ 0904201213333955808939929000.shadow.`

f. Rename the **/etc/connector_mirror_files/group** file to *SYNCTOKEN*.group.

   For example: `/etc/connector_mirror_files/ 0904201213333955808939929000.group.`

g. Log in to the Oracle Identity Manager Administrative and User Console.

h. On the Welcome to Oracle Identity Manager Self Service page, click **Advanced** in the upper-right corner of the page.

i. Search for and open the UNIX Target Incremental Resource User Reconciliation scheduled task.

j. On the Job Details tab, in the Parameters region, specify the following value for the **Sync Token** attribute of the scheduled task:

   ```
   <String>0904201213333955808939929000</String>
   ```

   > **Note:**
   >
   > For other flavors of Unix, you can use the same syncToken format for the Sync Token attribute of the incremental reconciliation scheduled task.

k. After specifying the attribute, click **Apply** to save the changes.

2.5 Postcloning Steps

You can clone the connector by setting new names for some of the objects that comprise the connector. The outcome of the process is a new connector XML file. Most of the connector objects, such as Resource Object, Process Definition, Process Form, IT Resource Type Definition, IT Resource Instances, Lookup Definitions, Adapters, Reconciliation Rules and so on in the new connector XML file have new names.

> **See Also:**
>
> Cloning Connectors in *Oracle Fusion Middleware Administering Oracle Identity Manager* for information about the privileges required to perform connector operations

After a copy of the connector is created by setting new names for connector objects, some objects might contain the details of the old connector objects. Therefore, you must modify the following Oracle Identity Manager objects to replace the base connector artifacts or attribute references with the corresponding cloned artifacts or attributes:

- Lookup Definition

  If the lookup definition contains the old lookup definition details, then you must modify it to provide the new cloned lookup definition names. If the Code Key and Decode values are referring the base connector attribute references, then replace these with new cloned attributes.

- Scheduled Task

  You must replace the base connector resource object name in the scheduled task with the cloned resource object name. If the scheduled task parameter has any data referring to the base connector artifacts or attributes, then these must be replaced with the new cloned connector artifacts or attributes.

- Child Table

  You must reassign the adapter and add a new literal value to the childTableName variable of a child table after cloning the connector.

  To update a child table:

  1. Log in to Design Console.
  2. Open the process task and click Integrations tab.
  3. Click **Remove** to unassign the adapter to the process task.
  4. Click **Add** to assign the same adapter to the process task.
  5. Assign a new literal value to the childTableName variable.
  6. Map the other adapter variables as per the previous mappings.

- Localization Properties

  You must update the resource bundle of a user locale with new names of the process form attributes for proper translations after cloning the connector. You can modify the properties file of your locale in the resources directory of the connector bundle.

  For example, the process form attributes are referenced in the Japanese properties file, UNIX_ja.properties, as `global.udf.UD_UNIX_ALIASNAME`. During cloning, if you change the process form name from `UD_UNIX` to `UD_UNIX1`, then you must update the process form attributes to `global.udf.UD_UNIX1_ALIASNAME`.

3 Using the Connector

After you deploy the connector, you must configure it to meet your requirements.

This chapter contains the following sections:

> **Note:**
>
> These sections provide both conceptual and procedural information about configuring the connector. It is recommended that you read the conceptual information before you perform the procedures.

- Configuring Reconciliation
- Scheduled Tasks
- Configuring Provisioning in Oracle Identity Manager Release 11.1.1
- Configuring Provisioning in Oracle Identity Manager Release 11.1.2
- Configuring Action Scripts

3.1 Configuring Reconciliation

Reconciliation involves duplicating in Oracle Identity Manager the creation of and modifications to user accounts on the target system. While configuring the connector, the target system can be designated as a trusted source or target resource.

If you designate the target system as a trusted source, then during a reconciliation run:

- For each newly created user on the target system, an Oracle Identity Manager User is created.
- Updates made to each user on the target system are propagated to the corresponding Oracle Identity Manager User.

If you designate the target system as a target resource, then during a reconciliation run:

- For each account created on the target system, a resource is assigned to the corresponding Oracle Identity Manager User.
- Updates made to each account on the target system are propagated to the corresponding resource.

This section discusses the following topics related to configuring reconciliation:

- Full Reconciliation
- Limited Reconciliation
- Batched Reconciliation

- Reconciliation Rule for Target Resource Reconciliation
- Reconciliation Action Rules for Target Resource Reconciliation
- Configuring the Target System As a Trusted Source
- Reconciliation Rule for Trusted Source Reconciliation
- Reconciliation Action Rule for Trusted Source Reconciliation

3.1.1 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.

To perform a full reconciliation run, remove (delete) any value currently assigned to the Filter attribute of the UNIX Target Resource Full User Reconciliation scheduled task. See Scheduled Tasks for Reconciliation for information about this scheduled task.

During a full reconciliation run, if you provide both batching parameters and filters, the connector processes the data in batches. Then, filters are applied to the processed data.

3.1.2 Limited Reconciliation

By default, all target system records that are added or modified after the last reconciliation run are reconciled during the current reconciliation run. You can customize this process by specifying the subset of added or modified target system records that must be reconciled.

The connector provides a Filter attribute that allows you to use any of the UNIX resource attributes to filter the target system records.

You can perform limited reconciliation by creating filters for the reconciliation module. This connector provides a Filter attribute (a scheduled task attribute) that allows you to use any of the UNIX resource attributes to filter the target system records.

For detailed information about ICF Filters, see ICF Filter Syntax in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager.*

While deploying the connector, follow the instructions in Configuring Scheduled Tasks to specify attribute values.

3.1.3 Batched Reconciliation

During a reconciliation run, all changes in the target system records are reconciled into Oracle Identity Manager. Depending on the number of records to be reconciled, this process may require a large amount of time. In addition, if the connection breaks during reconciliation, then the process would take longer to complete.

You can configure batched reconciliation to avoid these problems.

To configure batched reconciliation, you must specify values for the following user reconciliation scheduled task attributes:

- Batch Size: Use this attribute to specify the number of records that must be included in each batch.

- Batch Start Index: Use this attribute to specify the position from which the records will be included in each batch.
- No. of Batches: Use this attribute to specify the total number of batches that must be reconciled.

By default, the values of all attributes is `0,` indicating that all records will be included (no batched reconciliation). The following example illustrates this:

Suppose that of a total 314 records, only 200 records were processed before encountering an exception or an error. During the next reconciliation run, you can set Batch Start Index to 200 to process the records from 200 to 314.

You specify values for these attributes by following the instructions described in Configuring Scheduled Tasks.

3.1.4 Reconciliation Rule for Target Resource Reconciliation

The following is the process-matching rule:

Rule name: UNIX User Recon Rule

Rule element: User Login equals User Login

In this rule:

- User Login for Oracle Identity Manager:

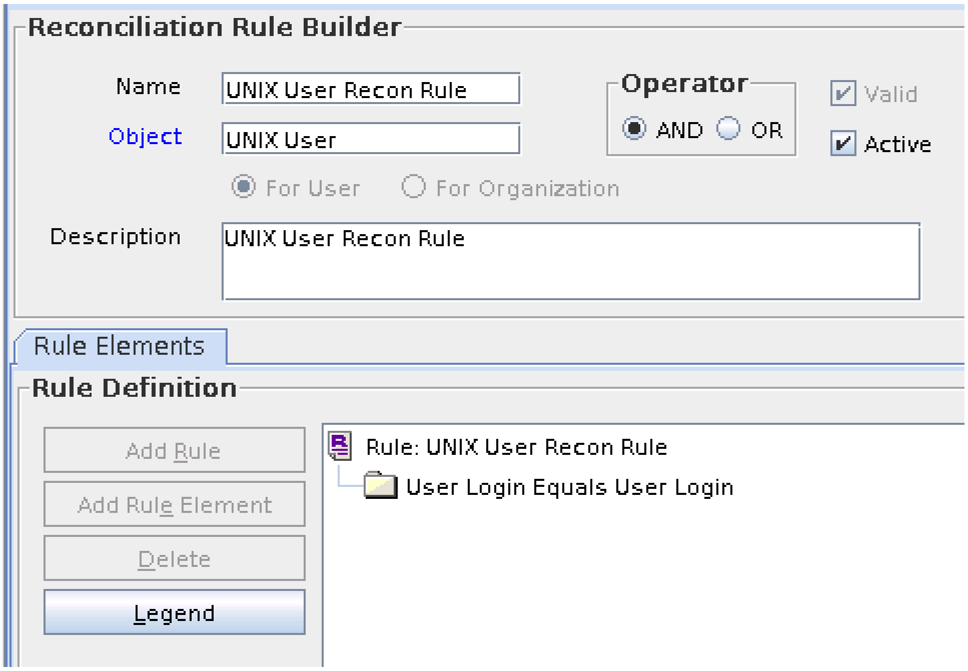
  User ID attribute on the OIM User form
- User Login is the User Login Name attribute of the target system.

After you deploy the connector, you can view the reconciliation rule for target resource reconciliation by performing the following steps:

> **Note:**
>
> Perform the following procedure only after the connector is deployed.

1. Log in to the Oracle Identity Manager Design Console.
2. Expand **Development Tools.**
3. Double-click **Reconciliation Rules.**
4. Search for **UNIX User Recon Rule**. Figure 3-1 shows the reconciliation rule for target resource reconciliation.

Figure 3-1   Reconciliation Rule for Target Resource Reconciliation



Name: UNIX User Recon Rule

Object: UNIX User

● For User ○ For Organization

Description: UNIX User Recon Rule

Operator: ● AND ○ OR

☑ Valid

☑ Active

Rule: UNIX User Recon Rule

User Login Equals User Login

3.1.5 Reconciliation Action Rules for Target Resource Reconciliation

This section lists the action rules for target resource reconciliation and suggests the procedure to view them.

Table 3-1 lists the action rules for target resource reconciliation.

Table 3-1   Action Rules for Target Resource Reconciliation

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

> **Note:**
>
> No action is performed for rule conditions that are not predefined for this connector. You can define your own action rule for such rule conditions. See Setting a Reconciliation Action Rule in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager* for information about modifying or creating reconciliation action rules.

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

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

2. Expand **Resource Management**.
3. Double-click **Resource Objects**.
4. Search for and open the **UNIX User** resource object.
5. 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.

3.1.6 Configuring the Target System As a Trusted Source

To configure trusted source reconciliation:

> **Note:**
>
> Skip this section if you do not want to designate the target system as a trusted source for reconciliation.

1. In Oracle Identity Manager Administrative and User Console, update the Configuration Lookup parameter of the IT Resource to `Lookup.UNIX.Configuration.Trusted`.

   You can change the entries in this configuration lookup if needed.

2. Create a directory on the target system where the connector can create mirror files for the /etc/passwd and /etc/shadow files.

   This directory is specified in the mirrorFilesLocation entry of the Lookup.UNIX.Configuration.Trusted lookup definition. The default value is `/etc/connector_mirror_files_trusted`. If the directory path is different from the default value, then you must update the correct path in the lookup.

   The loginUser (sudo or root user) must have read and write privileges to this directory.

3.1.7 Reconciliation Rule for Trusted Source Reconciliation

This section describes the reconciliation rule for trusted source reconciliation.

> **Note:**
>
> Skip this section if the target system is not designated as a trusted source for reconciliation.

The following is the process matching rule:

Rule name: UNIX User Trusted Recon Rule

Rule element: User Login Equals User ID

In this rule element:

- User Login for Oracle Identity Manager:

User ID attribute on the OIM User form.

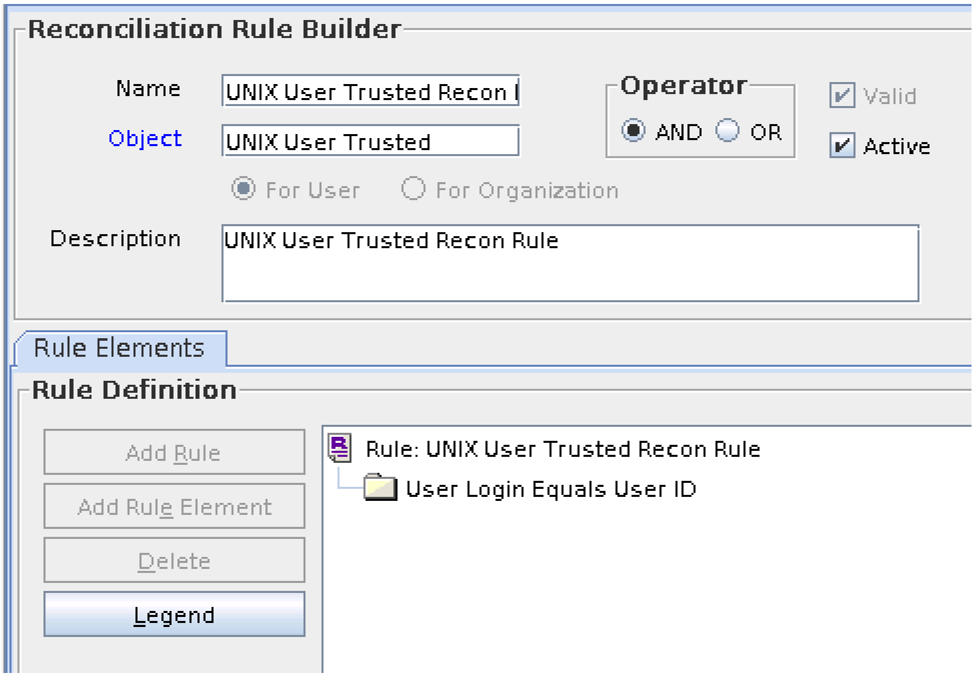
- User ID is the User Login Name attribute of the target system.

After you deploy the connector, you can view the reconciliation rule for trusted source reconciliation by performing the following steps:

> **Note:**
>
> Perform the following procedure only after the connector is deployed.

1. Log in to the Oracle Identity Manager Design Console.
2. Expand **Development Tools**.
3. Double-click **Reconciliation Rules**.
4. Search for **UNIX User Trusted Recon Rule.** Figure 3-2 shows the reconciliation rule for trusted source reconciliation.

Figure 3-2 Reconciliation Rule for Trusted Source Reconciliation



Reconciliation Rule Builder

Name: UNIX User Trusted Recon I

Object: UNIX User Trusted

Operator: ● AND ○ OR

☑ Valid

☑ Active

● For User ○ For Organization

Description: UNIX User Trusted Recon Rule

Rule Elements

Rule Definition

Rule: UNIX User Trusted Recon Rule
User Login Equals User ID

3.1.8 Reconciliation Action Rule for Trusted Source Reconciliation

This section lists the action rules for trusted resource reconciliation and suggests the procedure to view them.

> **Note:**
>
> Skip this section if the target system is not designated as a trusted source for reconciliation.

Table 3-2 lists the action rules for trusted resource reconciliation.

Table 3-2 Action Rules for Trusted Source Reconciliation

Rule Condition	Action
No Matches Found	Create User
One Entity Match Found	Establish Link
One Process Match Found	Establish Link

> **Note:**
>
> No action is performed for rule conditions that are not predefined for this connector. You can define your own action rule for such rule conditions. See Setting a Reconciliation Action Rule in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager* for information about modifying or creating reconciliation action rules.

After you deploy the connector, you can view the reconciliation action rules for trusted source reconciliation by performing the following steps:

1. Log in to the Oracle Identity Manager Design Console.
2. Expand **Resource Management**.
3. Double-click **Resource Objects**.
4. Search for and open the **UNIX User Trusted** resource object.
5. 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.

3.2 Scheduled Tasks

When you run the Connector Installer or import the connector XML file, reconciliation scheduled tasks are automatically created in Oracle Identity Manager.

This section discusses the following topics related to scheduled tasks:

- Scheduled Tasks for Lookup Field Synchronization
- Scheduled Tasks for Reconciliation
- Configuring Scheduled Tasks

3.2.1 Scheduled Tasks for Lookup Field Synchronization

The following scheduled tasks are used for lookup field synchronization:

- UNIX User Primary Group Lookup Reconciliation

  This scheduled task is used to synchronize the primary group names available on the target system into the Lookup.UNIX.PrimaryGroup lookup definition.
- UNIX User Shell Lookup Reconciliation

  This scheduled task is used to synchronize the default shell on the target system into the Lookup.UNIX.UserShell lookup definition.

You must specify values for the attributes of these scheduled jobs. Table 3-3 describes the attributes of these scheduled tasks. The procedure to configure scheduled tasks is described later in the guide.

Table 3-3 Attributes of the Scheduled Tasks for Lookup Field Synchronization

Attribute	Description
Code Key Attribute	Enter the name of the connector or target system attribute that is used to populate the Code Key column of the lookup definition (specified as the value of the Lookup Name attribute). Depending on the scheduled task you are using, the default values are as follows: • For UNIX User Primary Group Lookup Reconciliation: `GID` • For UNIX User Shell Lookup Reconciliation: `__NAME__` **Note:** You must not change the value of this attribute.
Decode Attribute	Enter the name of the connector or target system attribute that is used to populate the Decode column of the lookup definition (specified as the value of the Lookup Name attribute). Sample value: `__NAME__`
IT Resource Name	Enter the name of the IT resource for the target system installation from which you want to reconcile user records. Default value: `UNIX Server`
Lookup Name	This attribute holds the name of the lookup definition that maps each lookup definition with the data source from which values must be fetched. Depending on the scheduled task you are using, the default values are as follows: • For UNIX User Primary Group Lookup Reconciliation: `Lookup.UNIX.PrimaryGroup` • For UNIX User Shell Lookup Reconciliation: `Lookup.UNIX.UserShell`
Object Type	Enter the type of object whose values must be synchronized. Depending on the scheduled task you are using, the default values are as follows: • For UNIX User Primary Group Lookup Reconciliation: `User` • For UNIX User Shell Lookup Reconciliation: `__SHELLS__` **Note:** You must not change the value of this attribute.

Table 3-3 (Cont.) Attributes of the Scheduled Tasks for Lookup Field Synchronization

Attribute	Description
Resource Object Name	Enter the name of the resource object that is used for reconciliation. Default value: `UNIX User`

3.2.2 Scheduled Tasks for Reconciliation

You can use reconciliation scheduled job to reconcile user account data from the target system.

To reconcile user data in the target resource (account management) mode of the connector:

- UNIX Target Resource Full User Reconciliation

  This scheduled task is used to reconcile all user data from the target system.
- UNIX Target Incremental Resource User Reconciliation

  This scheduled task is used to reconcile only records created, modified, or deleted after the last reconciliation run.

> **Note:**
>
> Full target and trusted reconciliation scheduled tasks do not support reconciliation of deleted records.

To reconcile user data in the trusted source (identity management) mode of the connector:

- UNIX User Trusted Recon

  This scheduled task is used to reconcile all user data from the target system.
- UNIX User Trusted Incremental Recon

  This scheduled task is used to reconcile only records created, modified, or deleted after the last reconciliation run.

The following is the behavior of incremental reconciliation scheduled tasks:

- Incremental reconciliation scheduled tasks do not support filtering of records.
- Incremental reconciliation scheduled tasks fetch data from the target system in alphabetical order.
- If you run an incremental reconciliation scheduled task for the first time, or if you run the task after removing the value of **Sync Token** parameter, then the following directories (or the directory specified in the configuration lookup definition) must be empty:
  connector_mirror_files

  connector_mirror_files_trusted

- After an incremental reconciliation scheduled task completes, the following files will be generated in the **connector_mirror_files** or **connector_mirror_files_trusted** directory (or in the directory specified in the configuration lookup definition). Here, SYNC_TOKEN refers to the value of the Sync Token parameter.
  - *SYNC_TOKEN.***passwd** file contains previous copy of the password file in the /etc directory, for example, `/etc/passwd`.
  - *SYNC_TOKEN.***shadow** file contains previous copy of the shadow file in the /etc directory, for example, `/etc/shadow`.
  - *SYNC_TOKEN.***group** file contains previous copy of the group file in the /etc directory, for example, `/etc/group`.
  - **passwd_difference_incr** file contains differences between the `/etc/passwd` and the SYNC_TOKEN.`passwd` files.
  - **shadow_difference_incr** file contains differences between the `/etc/shadow` and the SYNC_TOKEN.`shadow` files.
  - **group_difference_incr** file contains differences between the `/etc/group` and the SYNC_TOKEN.`group` files.
  - **record** file contains the actual records that will be sent back to Oracle Identity Manager in alphabetically sorted order.

Table 3-4 describes the attributes of the scheduled tasks.

Table 3-4 Attributes of the Scheduled Tasks for Reconciliation

Attribute	Description
Batch Size **Note:** The data fetched by an incremental reconciliation scheduled task is sorted in alphabetical order.	Specify the number of records that must be included in each batch Default value: `0` See Batched Reconciliation for more information.
Batch Start Index **Note:** The data fetched by an incremental reconciliation scheduled task is sorted in alphabetical order.	Specify the position from which the records will be included in each batch Default value: `0`
No. of Batches **Note:** The data fetched by an incremental reconciliation scheduled task is sorted in alphabetical order.	Specify the total number of batches that must be reconciled Default value: `0`
Filter **Note:** Incremental reconciliation scheduled task does not support filtering of records.	Expression for filtering records that must be reconciled by the scheduled task By default, the value of this attribute is empty. Sample value: `equalTo('logonname','SEPT12USER1')` See Limited Reconciliation for the syntax of this expression.
IT Resource Name	Name of the IT resource for the target system installation from which you want to reconcile user records Default value: `UNIX Server`

Table 3-4 (Cont.) Attributes of the Scheduled Tasks for Reconciliation

Attribute	Description
Object Type	Type of object you want to reconcile Default value: `User`
Resource Object Name	Name of the resource object that is used for reconciliation Default value for UNIX Target Resource Full User Reconciliation: `UNIX User` Default value for other scheduled tasks: `UNIX User Trusted`
Scheduled Task Name	Name of the scheduled task **Note:** For the scheduled task shipped with this connector, you must not change the value of this attribute. However, if you create a copy of the task, then you can enter the unique name for that scheduled task as the value of this attribute.
Sync Token	Time stamp at which the last reconciliation run started **Note:** Do not enter a value for this attribute. The reconciliation engine automatically enters a value for this attribute. If you set this attribute to an empty value, then incremental reconciliation operations fetch all the records (perform full reconciliation).

3.2.3 Configuring Scheduled Tasks

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

To configure a scheduled task:

1. Depending on the Oracle Identity Manager release you are using, perform one of the following steps:
   - For Oracle Identity Manager release 11.1.1:
     a. Log in to the Administrative and User Console.
     b. On the Welcome to Oracle Identity Manager Self Service page, click **Advanced** in the upper-right corner of the page.
   - For Oracle Identity Manager release 11.1.2.*x:*
     a. Log in to Oracle Identity System Administration. For detailed instructions on creating and activating a sandbox, see Managing Sandboxes in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager*.
     b. Create and activate a sandbox.
     c. In the left pane, under System Management, click **Scheduler.**
2. Search for and open the scheduled job as follows:
   a. If you are using Oracle Identity Manager release 11.1.1, then on the Welcome to Oracle Identity Manager Advanced Administration page, in the System Management region, click **Search Scheduled Jobs**.
   b. 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.

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

3. On the Job Details tab, you can modify the following parameters:

   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.

   > **Note:**
   >
   > See Creating Jobs in *Oracle Fusion Middleware Administering Oracle Identity Manager* for detailed information about schedule types.

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

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

   > **Note:**
   >
   > - Attribute values are predefined in the connector XML file that you import. Specify values only for those attributes that you want to change.
   > - Attributes of the scheduled task are discussed in Scheduled Tasks for Reconciliation.

5. After specifying the attributes, click **Apply** to save the changes.

   > **Note:**
   >
   > The Stop Execution option is available in the Administrative and User Console. You can use the Scheduler Status page to either start, stop, or reinitialize the scheduler.

3.3 Configuring Provisioning in Oracle Identity Manager Release 11.1.1

Provisioning a resource for an OIM User involves using Oracle Identity Manager to create a target system account for the user.

If you have configured the connector for request-based provisioning, then the process form is suppressed and the object form is displayed. In other words, direct provisioning is disabled when you configure the connector for request-based provisioning. If you want to revert to direct provisioning, then perform the steps described in Switching Between Request-Based Provisioning and Direct Provisioning.

The following are types of provisioning operations:

- Direct provisioning
- Request-based provisioning
- Provisioning triggered by policy changes

> **See Also:**
>
> Manually Completing a Task in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Manager* for information about the types of provisioning

This section discusses the following topics:

- Guidelines on Performing Provisioning Operations
- Configuring Direct Provisioning
- Configuring Request-Based Provisioning
- Switching Between Request-Based Provisioning and Direct Provisioning

3.3.1 Guidelines on Performing Provisioning Operations

These are the guidelines that you must apply while performing provisioning operations.

Apply the following guidelines while performing provisioning operations:

- While performing a Create User provisioning operation, before populating the Skeleton directory field, data must be populated in the Home Directory field and the Create Home Directory check box must also be selected.
- The value in the Home Directory field must not contain spaces.
- The maximum permitted date value for account expiry is 31-Dec-2099. It should also be a future date.
- On HP-UX, while performing a Create User provisioning operation, the Inactive Days field must be populated only when the UNIX server is configured in trusted mode.
- The Primary Group Name and the Secondary Group Names must have different values. The target system ignores the duplicate values.

3.3.2 Configuring Direct Provisioning

When you install the connector on Oracle Identity Manager, the direct provisioning feature is automatically enabled. This means that the process form is enabled when you install the connector.

In direct provisioning, the Oracle Identity Manager administrator uses the Administrative and User Console to create a target system account for a user.

To provision a resource by using the direct provisioning approach:

1. Log in to the Administrative and User Console.
2. On the Welcome to Identity Administration page, in the Users region, click **Create User**.
3. On the Create User page, enter values for the OIM User fields, and then click **Save**.
4. If you want to provision a target system account to an existing OIM User, then:
    - On the Welcome to Identity Administration page, search for the OIM User by selecting **Users** from the list on the left pane.
    - From the list of users displayed in the search results, select the OIM User. The user details page is displayed on the right pane.
5. On the user details page, click the **Resources** tab.
6. From the Action menu, select **Add Resource**. Alternatively, you can click the add resource icon with the plus (+) sign. The Provision Resource to User page is displayed in a new window.
7. On the Step 1: Select a Resource page, select **UNIX User** from the list and then click **Continue**.
8. On the Step 2: Verify Resource Selection page, click **Continue**.
9. On the Step 5: Provide Process Data for User Details page, enter the details of the account that you want to create on the target system and then click **Continue**.
10. On the Step 6: Verify Process Data page, verify the data that you have provided and then click **Continue**.
11. Close the window displaying the "Provisioning has been initiated" message.
12. On the Resources tab, click **Refresh** to view the newly provisioned resource.

3.3.3 Configuring Request-Based Provisioning

The following sections give you a brief about request-based provisioning and how to enable them:

- About Request-Based Provisioning
- Enabling Request-Based Provisioning

3.3.3.1 About Request-Based Provisioning

In request-based provisioning, an end user creates a request for a resource by using the Administrative and User Console. Administrators or other users can also create requests for a particular user. Requests for a particular resource on the resource can be viewed and approved by approvers designated in Oracle Identity Manager.

The following are features of request-based provisioning:

- A user can be provisioned only one resource (account) on the target system.

> **Note:**
>
> Direct provisioning allows the provisioning of multiple target system accounts on the target system.

- Direct provisioning cannot be used if you enable request-based provisioning.

> **Note:**
>
> The request dataset provided with the connector does not contain the User Login field, which is usually fed directly from Oracle Identity Manager user profile to the process form using a prepopulate adapter.
>
> To include the User Login field in request dataset, perform the following procedure:
>
> 1. Export the current dataset using the MDS export utility.
> 2. Update the dataset to include the User Login field.
> 3. Import the updated dataset using the MDS import utility.
> 4. Purge the cache, as described in Clearing Content Related to Connector Resource Bundles from the Server Cache.

3.3.3.2 Enabling Request-Based Provisioning

The following sections discuss the steps to be performed to enable request-based provisioning:

> **Note:**
>
> The procedures described in these sections are built on an example in which the end user raises or creates a request for provisioning a target system account. This request is then approved by the approver.

- End User's Role in Request-Based Provisioning
- Approver's Role in Request-Based Provisioning
- Importing Request Datasets Using Deployment Manager
- Enabling the Auto Save Form Feature
- Running the PurgeCache Utility

3.3.3.2.1 End User's Role in Request-Based Provisioning

The following steps are performed by the end user in a request-based provisioning operation:

1. Log in to the Administrative and User Console.

2. On the Welcome page, click **Advanced** in the upper-right corner of the page.
3. On the Welcome to Identity Administration page, click the **Administration** tab, and then click the **Requests** tab.
4. From the Actions menu on the left pane, select **Create Request**.

   The Select Request Template page is displayed.
5. From the Request Template list, select **Provision Resource** and click **Next**.
6. On the Select Users page, specify a search criterion in the fields to search for the user that you want to provision the resource, and then click **Search**. A list of users that match the search criterion you specify is displayed in the Available Users list.
7. From the **Available Users** list, select the user to whom you want to provision the account.

   If you want to create a provisioning request for more than one user, then from the **Available Users** list, select users to whom you want to provision the account.
8. Click **Move** or **Move All** to include your selection in the Selected Users list, and then click **Next**.
9. On the Select Resources page, click the arrow button next to the Resource Name field to display the list of all available resources.
10. From the Available Resources list, select **UNIX User**, move it to the Selected Resources list, and then click **Next**.
11. On the Resource Details page, enter details of the account that must be created on the target system, and then click **Next**.
12. On the Justification page, you can specify values for the following fields, and then click **Finish**.
    - Effective Date
    - Justification

    On the resulting page, a message confirming that your request has been sent successfully is displayed along with the Request ID.
13. If you click the request ID, then the Request Details page is displayed.
14. To view details of the approval, on the Request Details page, click the **Request History** tab.

3.3.3.2.2 Approver's Role in Request-Based Provisioning

The following are steps performed by the approver in a request-based provisioning operation:

1. Log in to the Administrative and User Console.
2. On the Welcome page, click **Self-Service** in the upper-right corner of the page.
3. On the Welcome to Identity Manager Self Service page, click the **Tasks** tab.
4. On the **Approvals** tab, in the first section, you can specify a search criterion for request task that is assigned to you.
5. From the search results table, select the row containing the request you want to approve, and then click **Approve Task**.

   A message confirming that the task was approved is displayed.

3.3.3.2.3 Importing Request Datasets Using Deployment Manager

A request dataset is an XML file that specifies the information to be submitted by the requester during a provisioning operation. These request datasets specify information about the default set of attributes for which the requester must submit information during a request-based provisioning operation.

To import a request dataset XML file by using the Deployment Manager:

1. Log in to the Oracle Identity Manager Administrative and User Console.
2. Click the **Deployment Management** link on the left navigation bar.
3. Click the **Import** link under Deployment Management.

   A dialog box for opening files is displayed.
4. Locate and open the request dataset XML file, UNIX-RequestDatasets.xml, which is in the xml directory of the installation media.

   Details of this XML file are shown on the **File Preview** page.
5. Click **Add File**.

   The Substitutions page is displayed.
6. Click **Next**.

   The Confirmation page is displayed.
7. Click **Import**.
8. Close the Deployment Manager dialog box.

   The request dataset is imported into Oracle Identity Manager.

3.3.3.2.4 Enabling the Auto Save Form Feature

To enable the Auto Save Form feature:

1. Log in to the Design Console.
2. Expand **Process Management,** and then double-click **Process Definition.**
3. Search for and open the **SSH User** process definition.
4. Select the **Auto Save Form** check box.
5. Click **Save**.

3.3.3.2.5 Running the PurgeCache Utility

Run the PurgeCache utility to clear content belonging to the Metadata category from the server cache. See Clearing Content Related to Connector Resource Bundles from the Server Cache for instructions.

The procedure to configure request-based provisioning ends with this step.

3.3.4 Switching Between Request-Based Provisioning and Direct Provisioning

During the provisioning operation, you can switch between request request-based provisioning to direct provisioning.

> **Note:**
>
> It is assumed that you have performed the procedure described in Configuring Request-Based Provisioning.

To switch from request-based provisioning to direct provisioning:

1. Log in to the Design Console.
2. Disable the Auto Save Form feature as follows:
   a. Expand **Process Management**, and then double-click **Process Definition**.
   b. Search for and open the **UNIX User** process definition.
   c. Deselect the **Auto Save Form** check box.
   d. Click the save icon.
3. If the Self Request Allowed feature is enabled, then:
   a. Expand **Resource Management**, and then double-click **Resource Objects**.
   b. Search for and open the **UNIX User** resource object.
   c. Deselect the **Self Request Allowed** check box.
   d. Click the save icon.

To switch from direct provisioning back to request-based provisioning:

1. Log in to the Design Console.
2. Enable the Auto Save Form feature as follows:
   a. Expand **Process Management**, and then double-click **Process Definition**.
   b. Search for and open the **UNIX User** process definition.
   c. Select the **Auto Save Form** check box.
   d. Click the save icon.
3. If you want to enable end users to raise requests for themselves, then:
   a. Expand **Resource Management**, and then double-click **Resource Objects**.
   b. Search for and open the **UNIX User** resource object.
   c. Select the **Self Request Allowed** check box.
   d. Click the save icon.

3.4 Configuring Provisioning in Oracle Identity Manager Release 11.1.2

Provisioning involves creating and updating users on UNIX through Oracle Identity Manager.

To configure provisioning operations in Oracle Identity Manager release 11.1.2.*x:*

> **Note:**
>
> The time required to complete a provisioning operation that you perform the first time by using this connector takes longer than usual.

1. Log in to Oracle Identity Administrative and User console.
2. Create a user. See Managing Users in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Manager* for more information about creating a user.
3. On the Account tab, click **Request Accounts.**
4. In the Catalog page, search for and add to cart the application instance, and then click **Checkout.**
5. Specify values for fields in the application form and then click **Ready to Submit.**
6. Click **Submit.**
7. If you want to provision a UNIX User, then:
   a. On the Users page, search for the required user.
   b. On the user details page, click **Accounts.**
   c. Click the **Request Accounts** button.
   d. Search for the UNIX application instance in the catalog search box and select it.
   e. Click **Add to Cart.**
   f. Click **Checkout.**
   g. Specify values for fields in the application form and then click **Ready to Submit.**
   h. Click **Submit.**

3.5 Configuring Action Scripts

Actions are scripts that you can configure to run before or after the create, update, or delete an account provisioning operations. For example, you could configure a script to run before every user creation.

Every connector should specify which scripting language and which target it supports. The UNIX connector supports the following script:

- **shell**: shell script
- **target**: Resource

The target refers to the location where the script is executed. In this case, the script is executed on the computer where the target resource is running (and is typically interpreted by the target computer).

To configure the action:

1. Log in to the Design Console.
2. Search for and open the **Lookup.UNIX.UM.Configuration** lookup definition.
3. Add the following new values:
   - **Code Key**: `Before Create Action Language`
   - **Decode**: Enter the scripting language of the script you want to execute
   - **Example**: `shell`
4. Add these new values:
   - **Code Key**: `Before Create Action File`
   - **Decode**: Enter the full path to the file containing the script to be executed (Oracle Identity Manager must be able to access this file.)
   - **Example**: `/home/scripts/testscript.sh`
5. Add these new values:
   - **Code Key**: `Before Create Action Target`
   - **Decode**: `Resource`
6. Save the lookup definition.

Now, this action will be executed every time you create a user. You must configure these three values for each action you want to execute.

4

Extending the Functionality of the Connector

After you deploy the connector, you must configure it to meet your requirements.

This chapter discusses the following connector configuration procedures:

> **Note:**
>
> From Oracle Identity Manager Release 11.1.2 onward, lookup queries are not supported. See Managing Lookups in *Oracle Fusion Middleware Administering Oracle Identity Manager* guide for information about managing lookups by using the Form Designer in the Oracle Identity Manager System Administration console.

- Configuring the Connector for a New Target System
- Configuring the Connector for Multiple Instances and Multiple Versions of the Target System
- Adding Custom Attributes for Target Resource Reconciliation
- Adding Custom Attributes for Provisioning
- Configuring Validation of Data During Reconciliation and Provisioning
- Configuring Transformation of Data During User Reconciliation
- Configuring Resource Exclusion Lists

4.1 Configuring the Connector for a New Target System

You can configure the connector to support an additional flavor of UNIX.

> **Note:**
>
> Perform this procedure only if you want to configure the connector for an additional flavor of UNIX other than the target systems listed in Certified Components.

By default, the connector uses pre-configured scripts to support Linux, Solaris, AIX, and HP-UX.

The scripts are available in the connector bundle JAR file. You can download the bundle from Oracle Identity Governance database using the DownloadJars utility in *OIM_HOME/* bin directory or from the installation media. If you are using Connector

Server, then you can copy the bundle from *CONNECTOR_SERVER_HOME/* bundles directory.

You can add similar scripts with similar directory structure to support an additional flavor of UNIX. For example, you can add connector support for a target system with BSD/OS flavor of UNIX. To do so:

1. Create the following directories, which will be packaged into the connector bundle JAR:

   scripts/bsdos/nonsudo/

   scripts/bsdos/sudo/

2. Create the following scripts for sudo and non-sudo authentication types. Then, drop them in the corresponding directories created in the previous step.

   > **Note:**
   >
   > It is recommended that the script files have read-only permissions.

   Table 4-1 Custom Scripts to Support New Flavor of UNIX Target System

Script Name	Description
CreateNativeUser.txt	Create a user on target
DeleteNativeUser.txt	Delete a user from target
FetchAllGroupRecords.txt	For group lookup reconciliation
FetchAllUserRecords.txt	For full user reconciliation
FetchAllShellRecords.txt	For Shell lookup reconciliation
FetchSingleUserRecord.txt	Get one user
NativeUserIncrementalRecon.txt	Used by SyncOp for incremental reconciliation
UpdateNativeUser.txt	For user updates

3. Create and update the ScriptProperties.properties file with details of all the scripts.

   The values should be paths to the new scripts. See the scripts/linux/ScriptProperties.properties file for sample values. For example:

   ```
   CREATE_USER_SCRIPT=scripts/bsdos/sudo/CreateNativeUser.txt
   DELETE_USER_SCRIPT=scripts/bsdos/sudo/DeleteNativeUser.txt
   FETCH_SINGLE_USER=scripts/bsdos/sudo/FetchSingleUserRecord.txt
   FETCH_FULL_RECON_SCRIPT=scripts/bsdos/sudo/FetchAllUserRecords.txt
   INCREMENTAL_RECON_SCRIPT=scripts/bsdos/sudo/NativeUserIncrementalRecon.txt
   ```

4. Ensure that the values returned by the scripts are appropriate format, as expected by the bundle. See scripts/linux/ for sample scripts.

5. Create and update the ResponseMapping.properties file in the scripts/bsdos directory.

   The ResponseMapping.properties file contains mapping between the message to be expected and the exception class with which the message has to be wrapped and thrown. See the scripts/linux/ResponseMapping.properties file for sample values. For example:

```
User already
exists=org.identityconnectors.framework.common.exceptions.AlreadyExistsException
Group already
exists=org.identityconnectors.framework.common.exceptions.AlreadyExistsException
```

6. Run the following command to update the bundle JAR file with the new scripts:

   ```
   jar uvf org.identityconnectors.genericunix-1.0.0.jar scripts/bsdos/
   ```

7. In the PropertyFileName basic configuration parameter, specify the value of the path to the properties file.

   For example: `scripts/bsdos/nonsudo/ScriptProperties.properties` (for non-sudo authentication)

   See Table 2-2 for more information about the PropertyFileName parameter.

4.2 Configuring the Connector for Multiple Instances and Multiple Versions of the Target System

> **Note:**
>
> Perform this procedure only if you want to configure the connector for multiple installations of the target system.

You may want to configure the connector for multiple installations of the target system.

The following example illustrates this requirement:

The Tokyo, London, and New York offices of Example Multinational Inc. have their own installations of the target system. The company has recently installed Oracle Identity Manager, and they want to configure Oracle Identity Manager to link all the installations of the target system.

To meet the requirement posed by such a scenario, you must configure the connector for multiple installations of the target system.

To configure the connector for multiple installations of the target system:

1. Create and configure one IT resource for each target system installation.

   The IT Resources form is in the Resource Management folder. An IT resource is created when you import the connector XML file. You can use this IT resource as the template for creating the remaining IT resources, of the same resource type.

2. Configure reconciliation for each target system installation. See Scheduled Tasks for instructions. Note that you only need to modify the attributes that are used to specify the IT resource and to specify whether or not the target system installation is to be set up as a trusted source.

3. If required, modify the fields to be reconciled for the **UNIX User** resource object.

When you use the Administrative and User Console to perform provisioning, you can specify the IT resource corresponding to the target system installation to which you want to provision the user.

4.3 Adding Custom Attributes for Target Resource Reconciliation

The connector provides a default set of attribute mappings for reconciliation between Oracle Identity Manager and the target system. If required, you can add new user or group attributes for reconciliation.

> **Note:**
>
> In this section, the term "attribute" refers to the identity data fields that store user data.

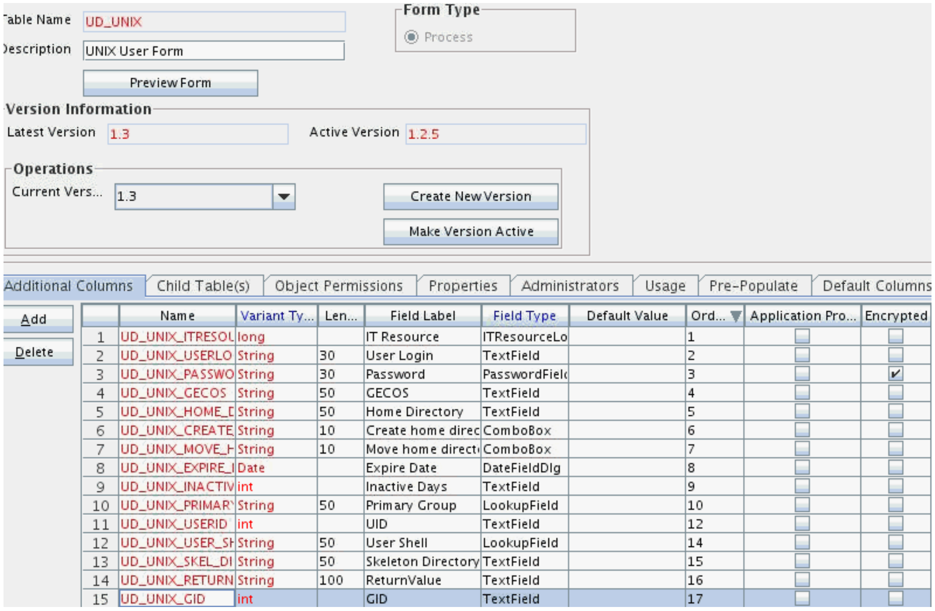
By default, the attributes listed in User Attributes for Target Resource Reconciliation and Provisioning are mapped for reconciliation between Oracle Identity Manager and the target system. If required, you can map additional attributes for reconciliation as follows:

1. Log in to the Oracle Identity Manager Design Console.
2. Create a new version of the process form as follows:
   a. Expand **Development Tools**.
   b. Double-click **Form Designer**.
   c. Search for and open the **UD_UNIX** process form.
   d. Click **Create New Version**.

   On the Create a new version dialog box, enter a new version in the Label field, and then click the save icon.
3. Add the new field on the process form as follows:.
   a. Click **Add**.
   A field is added to the list. Enter the details of the field.

   For example, if you are adding the GID field, enter `UD_UNIX_GID` in the **Name** field and then enter the rest of the details of this field.
   b. Click **Save**.
   c. To activate the newly created form, click **Make Version Active**.
   Figure 4-1 is a sample screenshot of the new version of process form.

Figure 4-1 Adding a New Version of Process Form



able Name: UD_UNIX
escription: UNIX User Form
Latest Version: 1.3
Active Version: 1.2.5
Current Vers...: 1.3

	Name	Variant Ty...	Len...	Field Label	Field Type	Default Value	Ord...	Application Pro...	Encrypted
1	UD_UNIX_ITRESOL	long		IT Resource	ITResourceLo		1	□	□
2	UD_UNIX_USERLO	String	30	User Login	TextField		2	□	□
3	UD_UNIX_PASSWO	String	30	Password	PasswordFielc		3	□	☑
4	UD_UNIX_GECOS	String	50	GECOS	TextField		4	□	□
5	UD_UNIX_HOME_I	String	50	Home Directory	TextField		5	□	□
6	UD_UNIX_CREATE	String	10	Create home direc	ComboBox		6	□	□
7	UD_UNIX_MOVE_H	String	10	Move home direct	ComboBox		7	□	□
8	UD_UNIX_EXPIRE_I	Date		Expire Date	DateFieldDlg		8	□	□
9	UD_UNIX_INACTIV	int		Inactive Days	TextField		9	□	□
10	UD_UNIX_PRIMAR'	String	50	Primary Group	LookupField		10	□	□
11	UD_UNIX_USERID	int		UID	TextField		12	□	□
12	UD_UNIX_USER_SH	String	50	User Shell	LookupField		14	□	□
13	UD_UNIX_SKEL_DI	String	50	Skeleton Directory	TextField		15	□	□
14	UD_UNIX_RETURN	String	100	ReturnValue	TextField		16	□	□
15	UD_UNIX_GID	int		GID	TextField		17	□	□

4. Create an entry for the field in the lookup definition for reconciliation as follows:

   a. Expand **Administration**.

   b. Double-click **Lookup Definition**.

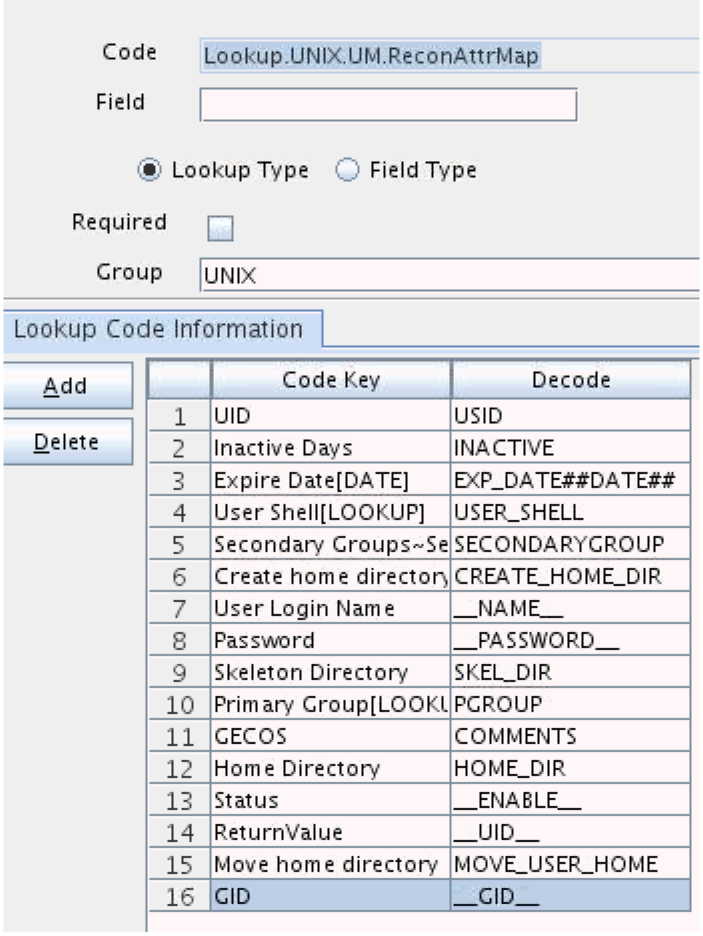
   c. Search for and open the **Lookup.UNIX.UM.ReconAttrMap** lookup definition.

   d. Click **Add** and enter the Code Key and Decode values for the field.
   The Code Key value must be the form field name. The Decode value must be the name that would be used for this attribute in the shell scripts.

      For example, enter `GID` in the **Code Key** field and then enter `__GID__` in the **Decode** field.

   e. Click the save icon.
   Figure 4-2 is a sample screenshot of the new entry added to the reconciliation lookup definition.

Figure 4-2 Adding an Entry to Reconciliation Lookup



Code: Lookup.UNIX.UM.ReconAttrMap

Field:

● Lookup Type ○ Field Type

Required □

Group: UNIX

	Code Key	Decode
1	UID	USID
2	Inactive Days	INACTIVE
3	Expire Date[DATE]	EXP_DATE##DATE##
4	User Shell[LOOKUP]	USER_SHELL
5	Secondary Groups~Se	SECONDARYGROUP
6	Create home director	CREATE_HOME_DIR
7	User Login Name	__NAME__
8	Password	__PASSWORD__
9	Skeleton Directory	SKEL_DIR
10	Primary Group[LOOKU	PGROUP
11	GECOS	COMMENTS
12	Home Directory	HOME_DIR
13	Status	__ENABLE__
14	ReturnValue	__UID__
15	Move home directory	MOVE_USER_HOME
16	GID	__GID__

5. Copy the connector bundle JAR file.

   You can download the bundle from Oracle Identity Manager database using the DownloadJars utility in *OIM_HOME/*bin directory or from the installation media. If you are using Connector Server, then you can copy the bundle from *CONNECTOR_SERVER_HOME/*bundles directory.

6. Extract the JAR file to edit the scripts.

   For example, to extract the script for Linux, non-sudo user for reconciliation, run the following command:

   ```
   jar xvf org.identityconnectors.genericunix-1.0.0.jar scripts/linux/nonsudo/
   FetchAllUserRecords.txt
   ```

   The FetchAllUserRecords.txt script is called when you run full reconciliation without the equalsTo filter. If you want, you can edit additional reconciliation scripts:

   - FetchSingleUserRecord.txt - this script is called when you run full reconciliation with the equalsTo filter.
   - NativeUserIncrementalRecon.txt - this script is called during incremental reconciliation.

7. Open the script for editing.

> **Note:**
>
> You must have a good knowledge of bash scripts to edit the scripts. Before editing the scripts, you can create a backup of the original scripts.

For example, you can construct a block for the __GID__ attribute similar to other blocks.

Add the following line after line 9 starting with PGROUP to fetch the __GID__ field:

```
__GID__=$( id -G $__NAME__ | cut -d' ' -f1);
```

Add an entry to line 32 starting with RESULT as follows:

```
RESULT=__NAME__:$__NAME__:__GID__:$__GID__:__ENABLE__:$ENABLE
```

Add an entry to line 41 starting with unset as follows:

```
unset inputline __NAME__ USID COMMENTS HOME_DIR USER_SHELL PGROUP secgrplist
__GID__;
```

> **See Also:**
>
> Sample Scripts for Updating Default Attributes for Reconciliation for the original and updated FetchAllUserRecords.txt script

8. Save the script and update the bundle as follows:

   ```
   jar uvf org.identityconnectors.genericunix-1.0.0.jar scripts/linux/nonsudo/
   FetchAllUserRecords.txt
   ```

9. Replace the old bundle by using UpdateJars utility in *OIM_HOME/*bin directory.

   If you are using the Connector Server, stop it. Then, replace the JAR in the *CONNECTOR_SERVER_HOME/*bundles directory and restart the Connector Server.

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

    a. Expand **Resource Management**.

    b. Double-click **Resource Objects**.

    c. Search for and open the **UNIX User** resource object.

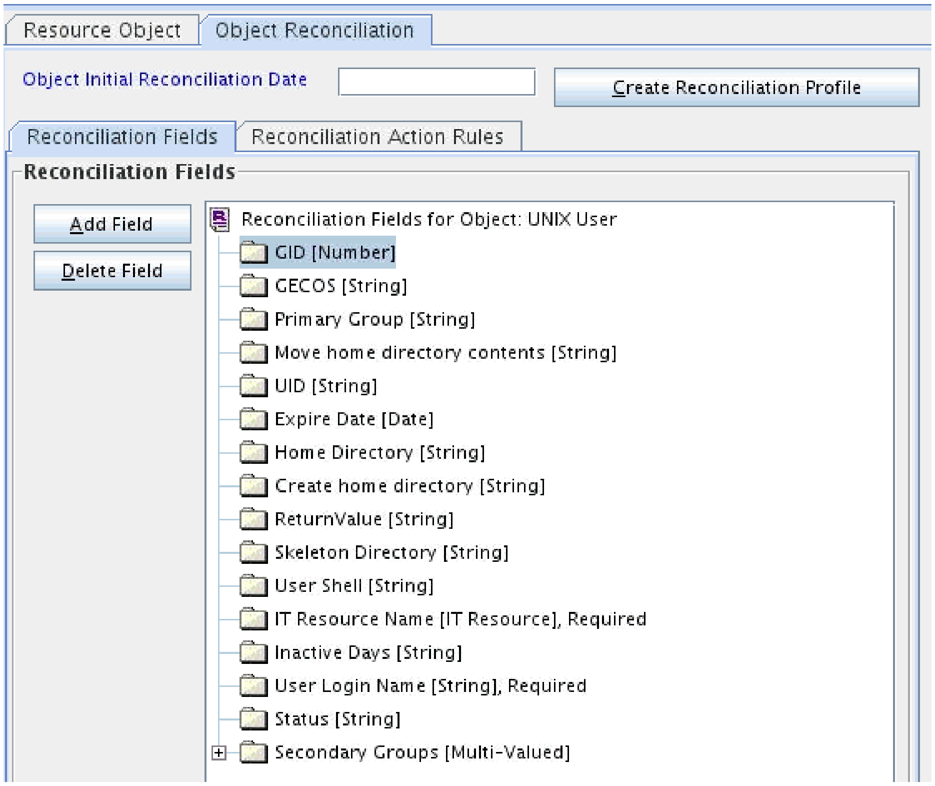
    d. On the Object Reconciliation tab, click **Add Field**.

    e. In the Add Reconciliation Field dialog box, enter the details of this field.
    For example, enter `GID` in the **Field Name** field and select **Number** from the Field Type list.

    f. Click the save icon.

    g. Click **Create Reconciliation Profile.** This copies changes made to the resource object into the MDS.
    Figure 4-3 is a sample screenshot of the newly added reconciliation field.

Figure 4-3   Adding a New Reconciliation Field



Reconciliation Fields for Object: UNIX User
- GID [Number]
- GECOS [String]
- Primary Group [String]
- Move home directory contents [String]
- UID [String]
- Expire Date [Date]
- Home Directory [String]
- Create home directory [String]
- ReturnValue [String]
- Skeleton Directory [String]
- User Shell [String]
- IT Resource Name [IT Resource], Required
- Inactive Days [String]
- User Login Name [String], Required
- Status [String]
- Secondary Groups [Multi-Valued]

11. Create a reconciliation field mapping for the new field on the process form as follows:

    a. Expand **Process Management**.

    b. Double-click **Process Definition**.

    c. From the Process Definition table, select and open the **UNIX User** resource object.

    d. Click **Reconciliation Field Mappings** and then click **Add Field Map**.

    e. In the Field Name field, select the value for the field that you want to add. For example, select `GID`.

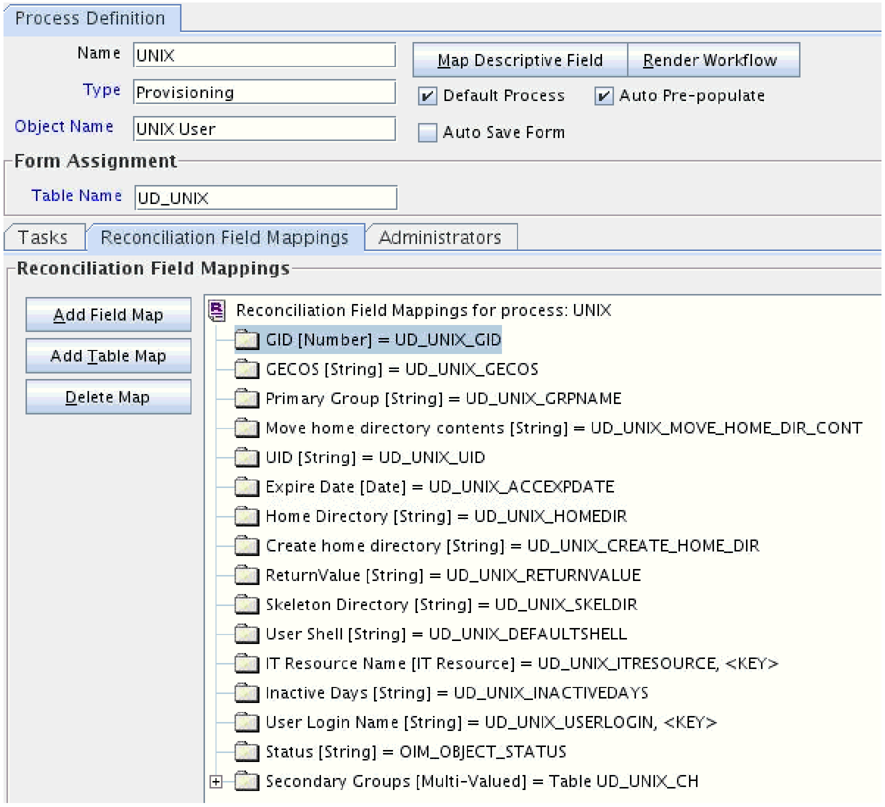
    f. In the **Field Type** field, select the type of the field that is prepopulated.

    g. Double-click the **Process Data Field** field.
    A list of process data columns is displayed. From the list, select the process data column corresponding to the process data field.

       For example, select `GID [Number] = UD_UNIX_GID`.

    h. Click the save icon.
    Figure 4-4 is a sample screenshot of the reconciliation field mappings.

Figure 4-4 Adding a Reconciliation Field Mapping



Name: UNIX

Type: Provisioning

Object Name: UNIX User

☑ Default Process

☑ Auto Pre-populate

□ Auto Save Form

Form Assignment

Table Name: UD_UNIX

Reconciliation Field Mappings

Reconciliation Field Mappings for process: UNIX

- GID [Number] = UD_UNIX_GID
- GECOS [String] = UD_UNIX_GECOS
- Primary Group [String] = UD_UNIX_GRPNAME
- Move home directory contents [String] = UD_UNIX_MOVE_HOME_DIR_CONT
- UID [String] = UD_UNIX_UID
- Expire Date [Date] = UD_UNIX_ACCEXPDATE
- Home Directory [String] = UD_UNIX_HOMEDIR
- Create home directory [String] = UD_UNIX_CREATE_HOME_DIR
- ReturnValue [String] = UD_UNIX_RETURNVALUE
- Skeleton Directory [String] = UD_UNIX_SKELDIR
- User Shell [String] = UD_UNIX_DEFAULTSHELL
- IT Resource Name [IT Resource] = UD_UNIX_ITRESOURCE, <KEY>
- Inactive Days [String] = UD_UNIX_INACTIVEDAYS
- User Login Name [String] = UD_UNIX_USERLOGIN, <KEY>
- Status [String] = OIM_OBJECT_STATUS
- Secondary Groups [Multi-Valued] = Table UD_UNIX_CH

12. On the Resource Objects form, click **Create Reconciliation Profile**.

13. Perform reconciliation to verify the GID attribute.

    You should be able to see GID in the connector logs. Ensure that they appear in the reconciliation events.

14. If you are using Oracle Identity Manager release 11.1.2.*x* or later, 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.

4.4 Adding Custom Attributes for Provisioning

The connector provides a default set of attribute mappings for provisioning between Oracle Identity Manager and the target system. If required, you can add new user or group attributes for provisioning.

> **Note:**
>
> In this section, the term "attribute" refers to the identity data fields that store user data.

By default, the attributes listed in User Attributes for Target Resource Reconciliation and Provisioning are mapped for provisioning between Oracle Identity Manager and the target system. If required, you can map additional attributes for provisioning as follows:

1. Log in to the Oracle Identity Manager Design Console.
2. Create a new version of the process form as follows:
   a. Expand **Development Tools**.
   b. Double-click **Form Designer**.
   c. Search for and open the **UD_UNIX** process form.
   d. Click **Create New Version**.
      On the Create a new version dialog box, enter a new version in the Label field, and then click the save icon.
3. Add the new field on the process form as follows:
   a. Click **Add**.
      A field is added to the list. Enter the details of the field.

      For example, if you are adding the GID field, enter `UD_UNIX_GID` in the **Name** field and then enter the rest of the details of this field.
   b. Click **Save**.
   c. To activate the newly created form, click **Make Version Active**.
      Figure 4-5 is a sample screenshot of the new version of process form.

Figure 4-5 Adding a New Version of Process Form

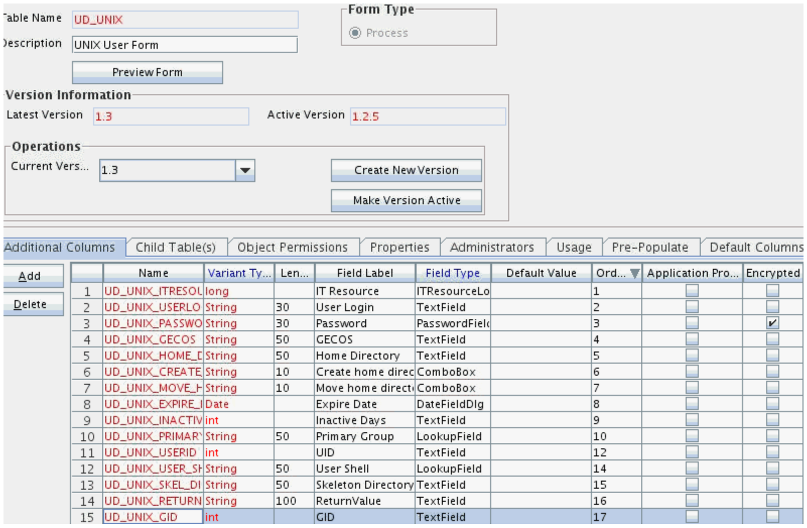


Table Name: UD_UNIX
Description: UNIX User Form

Form Type: ● Process

Version Information
Latest Version: 1.3
Active Version: 1.2.5

Operations
Current Vers...: 1.3

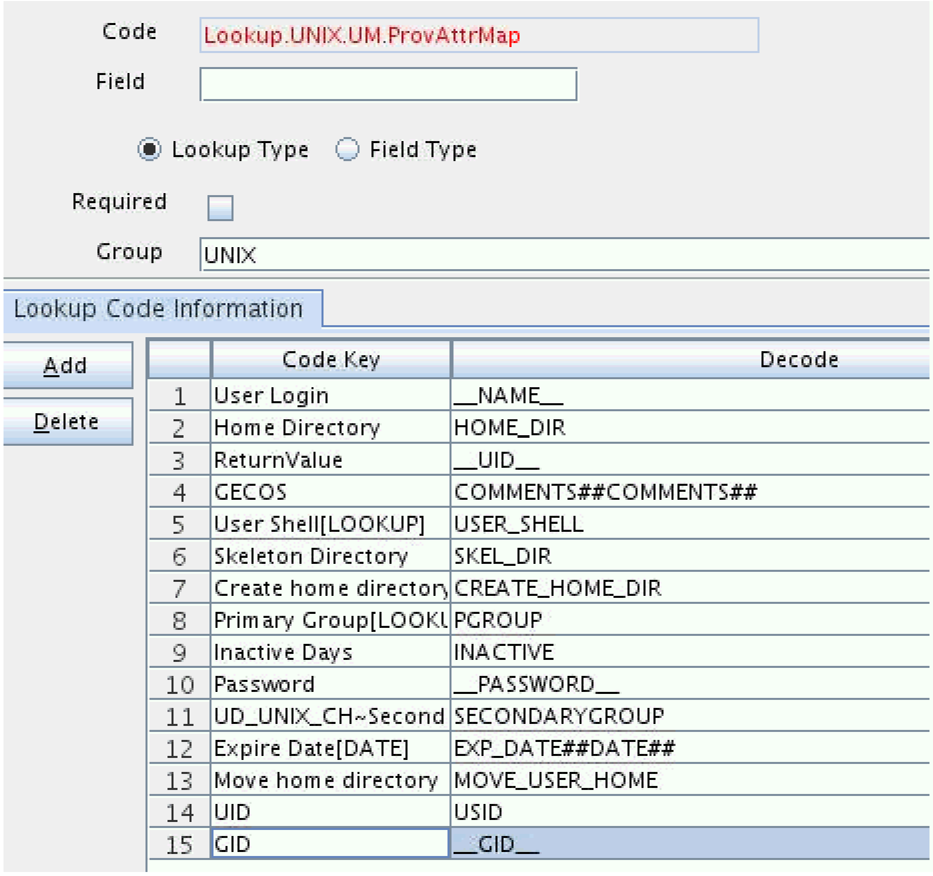
	Name	Variant Ty...	Len...	Field Label	Field Type	Default Value	Ord...	Application Pro...	Encrypted
1	UD_UNIX_ITRESOL	long		IT Resource	ITResourceLo		1	□	□
2	UD_UNIX_USERLO	String	30	User Login	TextField		2	□	□
3	UD_UNIX_PASSWO	String	30	Password	PasswordFiel		3	□	☑
4	UD_UNIX_GECOS	String	50	GECOS	TextField		4	□	□
5	UD_UNIX_HOME_D	String	50	Home Directory	TextField		5	□	□
6	UD_UNIX_CREATE	String	10	Create home direc	ComboBox		6	□	□
7	UD_UNIX_MOVE_H	String	10	Move home direct	ComboBox		7	□	□
8	UD_UNIX_EXPIRE_I	Date		Expire Date	DateFieldDlg		8	□	□
9	UD_UNIX_INACTIV	int		Inactive Days	TextField		9	□	□
10	UD_UNIX_PRIMAR	String	50	Primary Group	LookupField		10	□	□
11	UD_UNIX_USERID	int		UID	TextField		12	□	□
12	UD_UNIX_USER_SH	String	50	User Shell	LookupField		14	□	□
13	UD_UNIX_SKEL_DI	String	50	Skeleton Directory	TextField		15	□	□
14	UD_UNIX_RETURN	String	100	ReturnValue	TextField		16	□	□
15	UD_UNIX_GID	int		GID	TextField		17	□	□

4. Create an entry for the field in the lookup definition for reconciliation as follows:
   a. Expand **Administration**.
   b. Double-click **Lookup Definition**.
   c. Search for and open the **Lookup.UNIX.UM.ProvAttrMap** lookup definition.
   d. Click **Add** and enter the Code Key and Decode values for the field.
      The Code Key value must be the form field name. The Decode value must be the name that would be used for this attribute in the shell scripts.

      For example, enter `GID` in the **Code Key** field and then enter `_GID_` in the **Decode** field.

e. Click the save icon.
   Figure 4-6 is a sample screenshot of the new entry added to the reconciliation lookup definition.

Figure 4-6 Adding an Entry to Provisioning Lookup



Code	Lookup.UNIX.UM.ProvAttrMap
Field	
● Lookup Type	○ Field Type
Required	□
Group	UNIX

	Code Key	Decode
1	User Login	__NAME__
2	Home Directory	HOME_DIR
3	ReturnValue	__UID__
4	GECOS	COMMENTS##COMMENTS##
5	User Shell[LOOKUP]	USER_SHELL
6	Skeleton Directory	SKEL_DIR
7	Create home director	CREATE_HOME_DIR
8	Primary Group[LOOKU	PGROUP
9	Inactive Days	INACTIVE
10	Password	__PASSWORD__
11	UD_UNIX_CH~Second	SECONDARYGROUP
12	Expire Date[DATE]	EXP_DATE##DATE##
13	Move home directory	MOVE_USER_HOME
14	UID	USID
15	GID	__GID__

5. Copy the connector bundle JAR file.

   You can download the bundle from Oracle Identity Manager database using the DownloadJars utility in *OIM_HOME/*bin directory or from the installation media. If you are using Connector Server, then you can copy the bundle from *CONNECTOR_SERVER_HOME/*bundles directory.

6. Extract the JAR file to edit the scripts.

   For example, to extract the script for Linux, non-sudo user for provisioning, run the following command:

```
jar xvf org.identityconnectors.genericunix-1.0.0.jar scripts/linux/nonsudo/
CreateNativeUser.txt
```

   This script is used to enable create operations on the newly added attribute. Similarly, you can edit the UpdateNativeUser.txt script to enable update operations.

7. Open the script for editing.

> **Note:**
>
> You must have a good knowledge of bash scripts to edit the scripts. Before editing the scripts, you can create a backup of the original scripts.

For example, you can construct a block for the __GID__ attribute similar to other blocks, as follows (lines 76 to 78):

```
if [ ! -z $__GID__ ] ;then
    command="$command -g $__GID__";
fi;
```

Add an entry to line 91 starting with unset as follows:

```
unset defaultHomeBaseDir homedir checkHomeBaseDir grp defaultPriGroup __GID__;
```

> **See Also:**
>
> Sample Scripts for Updating Default Attributes for Provisioning for the original and updated CreateNativeUser.txt script

8. Save the script and update the bundle as follows:

   ```
   jar uvf org.identityconnectors.genericunix-1.0.0.jar scripts/linux/nonsudo/
   CreateNativeUser.txt
   ```

9. Replace the old bundle by using UpdateJars utility in *OIM_HOME/*bin directory.

   If you are using the Connector Server, stop it. Then, replace the JAR in the *CONNECTOR_SERVER_HOME/*bundles directory and restart the Connector Server.

10. If you are using Oracle Identity Manager release 11.1.2.*x* or later, 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.

4.5 Configuring Validation of Data During Reconciliation and Provisioning

You can configure validation of reconciled and provisioned single-valued data according to your requirements.

The Lookup.UNIX.UM.ProvValidation and Lookup.UNIX.UM.ReconValidation lookup definitions hold single-valued data to be validated during provisioning and reconciliation operations, respectively.

For example, you can validate data fetched from the First Name attribute to ensure that it does not contain the number sign (#). In addition, you can validate data entered in the First Name field on the process form so that the number sign (#) is not sent to the target system during provisioning operations.

Note:

The Lookup.UNIX.UM.ProvValidation and Lookup.UNIX.UM.ReconValidation lookup definitions are optional and do not exist by default.

You must add these lookups as decode values to the Lookup.UNIX.UM.Configuration lookup definition to enable exclusions during provisioning and reconciliation operations. See Lookup.UNIX.UM.Configuration for more information.

To configure validation of data:

1. Write code that implements the required validation logic in a Java class with a fully qualified domain name (FQDN), such as `org.identityconnectors.genericunix.extension.UNIXValidator`.

   This validation class must implement the validate method. The following sample validation class checks if the value in the First Name attribute contains the number sign (#):

   ```
   package com.validationexample;

   import java.util.HashMap;

   public class MyValidator {
       public boolean validate(HashMap hmUserDetails, HashMap hmEntitlementDetails,
   String sField) throws ConnectorException {

           /* You must write code to validate attributes. Parent
                    * data values can be fetched by using hmUserDetails.get(field)
                    * For child data values, loop through the
                    * ArrayList/Vector fetched by hmEntitlementDetails.get("Child
   Table")
                    * Depending on the outcome of the validation operation,
                    * the code must return true or false.
                    */
           /*
           * In this sample code, the value "false" is returned if the field
           * contains the number sign (#). Otherwise, the value "true" is
           * returned.
           */
           boolean valid = true;
           String sFirstName = (String) hmUserDetails.get(sField);
           for (int i = 0; i < sFirstName.length(); i++) {
               if (sFirstName.charAt(i) == '#') {
                   valid = false;
                   break;
               }
           }
           return valid;

       }
   }
   ```

2. Log in to the Design Console.
3. Create one of the following new lookup definitions:
   - To configure validation of data for reconciliation:

    `Lookup.UNIX.UM.ReconValidation`

  - To configure validation of data for provisioning:

    `Lookup.UNIX.UM.ProvValidation`

4. In the **Code Key** column, enter the resource object field name that you want to validate. For example, `SimpleDisplayName.`
5. In the **Decode** column, enter the class name. For example, `org.identityconnectors.genericunix.extension.UNIXValidator.`
6. Save the changes to the lookup definition.
7. Search for and open the **Lookup.UNIX.UM.Configuration** lookup definition.
8. In the **Code Key** column, enter one of the following entries:
   - To configure validation of data for reconciliation:

     `Recon Validation Lookup`

   - To configure validation of data for provisioning:

     `Provisioning Validation Lookup`

9. In the **Decode** column, enter one of the following entries:
   - To configure validation of data for reconciliation:

     `Lookup.UNIX.UM.ReconValidation`

   - To configure validation of data for provisioning:

     `Lookup.UNIX.UM.ProvValidation`

10. Save the changes to the lookup definition.
11. Create a JAR with the class and upload it to the Oracle Identity Manager database as follows:

    Run the Oracle Identity Manager Upload JARs utility to post the JAR file created in Step 7 to the Oracle Identity Manager database. This utility is copied into the following location when you install Oracle Identity Manager:

    > **Note:**
    >
    > Before you use this utility, verify that the `WL_HOME` environment variable is set to the directory in which Oracle WebLogic Server is installed.

    For Microsoft Windows:

    OIM_HOME/server/bin/UploadJars.bat

    For UNIX:

    OIM_HOME/server/bin/UploadJars.sh

    When you run the utility, you are prompted to enter the login credentials of the Oracle Identity Manager administrator, URL of the Oracle Identity Manager host computer, context factory value, type of JAR file being uploaded, and the location from which the JAR file is to be uploaded. Select 1 as the value of the JAR type.

12. Run the PurgeCache utility to clear content related to request datasets from the server cache.

13. Perform reconciliation or provisioning to verify validation for the field, for example, SimpleDisplayName.

4.6 Configuring Transformation of Data During User Reconciliation

You can configure transformation of reconciled data according to your requirements.

The Lookup.UNIX.UM.ReconTransformation lookup definition holds single-valued user data to be transformed during reconciliation operations. For example, you can use First Name and Last Name values to create a value for the Full Name field in Oracle Identity Manager.

> **Note:**
>
> The Lookup.UNIX.UM.ReconTransformation lookup definition is optional and does not exist by default.
>
> You must add this lookup as decode value to the Lookup.UNIX.UM.Configuration lookup definition to enable exclusions during provisioning and reconciliation operations. See Lookup.UNIX.UM.Configuration for more information.

To configure transformation of single-valued user data fetched during reconciliation:

1. Write code that implements the required transformation logic in a Java class with a fully qualified domain name (FQDN), such as `org.identityconnectors.genericunix.extension.UNIXTransformation`.

   This transformation class must implement the transform method. The following sample transformation class creates a value for the Full Name attribute by using values fetched from the First Name and Last Name attributes of the target system:

```
package com.transformationexample;

import java.util.HashMap;


public class MyTransformer {
    public Object transform(HashMap hmUserDetails, HashMap hmEntitlementDetails,
String sField) throws ConnectorException {
        /*
        * You must write code to transform the attributes.
        * Parent data attribute values can be fetched by
        * using hmUserDetails.get("Field Name").
        * To fetch child data values, loop through the
        * ArrayList/Vector fetched by hmEntitlementDetails.get("Child
Table")
        * Return the transformed attribute.
        */
        String sFirstName = (String) hmUserDetails.get("First Name");
        String sLastName = (String) hmUserDetails.get("Last Name");
```

```
            return sFirstName + "." + sLastName;

        }
    }
```

2. Log in to the Design Console.
3. Create a new lookup definition, **Lookup.UNIX.UM.ReconTransformation.**
4. In the **Code Key** column, enter the resource object field name you want to transform. For example, `SimpleDisplayName`.
5. In the **Decode** column, enter the class name. For example, `org.identityconnectors.genericunix.extension.UNIXTransformation`.
6. Save the changes to the lookup definition.
7. Search for and open the **Lookup.UNIX.UM.Configuration** lookup definition.
8. In the **Code Key** column, enter `Recon Transformation Lookup`.
9. In the **Decode** column, enter `Lookup.UNIX.UM.ReconTransformation`.
10. Save the changes to the lookup definition.
11. Create a JAR with the class and upload it to the Oracle Identity Manager database as follows:

    Run the Oracle Identity Manager Upload JARs utility to post the JAR file created in Step 7 to the Oracle Identity Manager database. This utility is copied into the following location when you install Oracle Identity Manager:

    > **Note:**
    >
    > Before you use this utility, verify that the `WL_HOME` environment variable is set to the directory in which Oracle WebLogic Server is installed.

    For Microsoft Windows:

    OIM_HOME/server/bin/UploadJars.bat

    For UNIX:

    OIM_HOME/server/bin/UploadJars.sh

    When you run the utility, you are prompted to enter the login credentials of the Oracle Identity Manager administrator, URL of the Oracle Identity Manager host computer, context factory value, type of JAR file being uploaded, and the location from which the JAR file is to be uploaded. Select 1 as the value of the JAR type.
12. Run the PurgeCache utility to clear content related to request datasets from the server cache.
13. Perform reconciliation to verify transformation of the field, for example, SimpleDisplayName.

4.7 Configuring Resource Exclusion Lists

The Lookup.UNIX.UM.ProvExclusionList and Lookup.UNIX.UM.ReconExclusionList lookup definitions hold user IDs of target system accounts for which you do not want to perform provisioning and reconciliation operations, respectively.

> **Note:**
>
> The Lookup.UNIX.UM.ProvExclusionList and Lookup.UNIX.UM.ReconExclusionList lookup definitions are optional and do not exist by default.
>
> You must add these lookups as decode values to the Lookup.UNIX.UM.Configuration lookup definition to enable exclusions during provisioning and reconciliation operations. See Lookup.UNIX.UM.Configuration for more information.

The following is the format of the values stored in these lookups:

Code Key	Decode	Sample Values
User Login resource object field name	User ID of a user	Code Key: User Login Decode: User001
User Login resource object field name with the [PATTERN] suffix	A regular expression supported by the representation in the `java.util.regex.Pattern` class	Code Key: User Login[PATTERN] To exclude users matching any of the user ID 's User001, User002, User088, then: Decode: User001|User002|User088 To exclude users whose user ID 's start with 00012, then: Decode: 00012* **See Also:** For information about the supported patterns, visit `http://download.oracle.com/javase/6/docs/api/java/util/regex/Pattern.html`

To add entries in the lookup for exclusions during provisioning operations:

1. On the Design Console, expand **Administration** and then double-click **Lookup Definition.**
2. Create a new lookup definition, **Lookup.UNIX.UM.ProvExclusionList.**

   > **Note:**
   >
   > To specify user IDs to be excluded during reconciliation operations, create a new lookup definition called Lookup.UNIX.UM.ReconExclusionList and add entries to that lookup.

3. Click **Add.**
4. In the Code Key and Decode columns, enter the first user ID to exclude.

> **Note:**
>
> The Code Key represents the resource object field name on which the exclusion list is applied during provisioning operations.

5. Repeat Steps 3 and 4 for the remaining user IDs to exclude.

   For example, if you do not want to provision users with user IDs User001, User002, and User088 then you must populate the lookup definition with the following values:

Code Key	Decode
User Login	User001
User Login	User002
User Login	User088

   You can also perform pattern matching to exclude user accounts. You can specify regular expressions supported by the representation in the `java.util.regex.Pattern` class.

> **See Also:**
>
> For information about the supported patterns, visit `http://download.oracle.com/javase/6/docs/api/java/util/regex/Pattern.html`

   For example, if you do not want to provision users matching any of the user IDs User001, User002, and User088, then you must populate the lookup definition with the following values:

Code Key	Decode
User Login[PATTERN]	User001|User002|User088

   If you do not want to provision users whose user IDs start with 00012, then you must populate the lookup definition with the following values:

Code Key	Decode
User Login[PATTERN]	00012*

6. Click the save icon.

5 Testing and Troubleshooting

After you create the application, you must test the connector to ensure that it functions as expected.

This chapter discusses the following topics related to connector testing:

- Testing the UNIX Connector
- Troubleshooting

5.1 Testing the UNIX Connector

You can use the testing utility, supplied with the connector package, to identify the cause of problems associated with connecting to the target system and performing basic operations on the target system.

The test-utility directory of connector installation media contains the following files:

- The example-config.groovy file is a sample configuration that can be used to set the connection properties of the target system and the connector.
- The README file contains instructions to configure and run the testing utility.
- The test-utility.jar file contains the class files used by the testing utility.

> **Note:**
>
> The testing utility does not support separate update and delete actions. You can update and delete a user while creating the user in the same operation.

To use the testing utility, perform the following steps:

1. Ensure JDK 1.6 is installed.
2. Unzip the connector zip file.
3. Locate and switch to the `test-utility` directory in the contents of the extracted zip file.

   The test-utility.jar and example-config.groovy files already exist in this directory.

   > **Note:**
   >
   > On a Microsoft Windows computer, the bundle package (`org.identityconnectors.genericunix-1.0.0.jar`) must be in a directory other than the test-utility directory.

4. Copy the following JAR files to the test-utility directory:
   - `connector-framework.jar`
   - `connector-framework-internal.jar`
   - `groovy-all.jar`

   > **Note:**
   >
   > These are files are delivered as part of the OIM EAR application, and they are located in the `oim.ear/APP-INF/lib` directory.
   >
   > You must add these three JAR files to the `JAVA_HOME/jre/lib/ext` directory to run the test utility.
   >
   > If you copy these JAR files into the test-utility directory, you can exclude these files in the Java command provided in Step 6.

5. Update the `example-config.groovy` file to reflect your local settings.
6. Run one of the following commands:

   On a UNIX computer:

```
java -classpath ./test-utility.jar:./connector-framework.jar:./connector-
framework-internal.jar:./groovy-all.jar:./test-utility.jar
oracle.iam.connectors.testutility.Main example-config.groovy | tee test.log
```

   On a Microsoft Windows computer (assuming the current directory is c:\test-utility):

```
java -classpath C:\test-utility\test-utility.jar;C:\test-utility\connector-
framework.jar;C:\test-utility\connector-framework-internal.jar;C:\test-utility
\groovy-all.jar;C:\test-utility\test-utility.jar;
oracle.iam.connectors.testutility.Main example-config.groovy
```

5.2 Troubleshooting

You can troubleshoot the UNIX connector depending on the type of error you encounter.

The following sections list solutions to some commonly encountered errors of the following types:

- Connection Errors
- Create User Errors
- Delete User Errors
- Edit User Errors
- TimeOut Errors

5.2.1 Connection Errors

This table lists the solution to a commonly encountered Connection error.

Problem Description	Solution
Oracle Identity Manager cannot establish a connection to the target system. **Returned Error Message:** UNIX connection exception	• Ensure that the target system is running. • Ensure that Oracle Identity Manager is working (that is, the database is running). • Ensure that all the adapters have been compiled. • Examine the Oracle Identity Manager record (from the IT Resources form). Ensure that values for all the IT resource parameters have been correctly specified.

5.2.2 Create User Errors

This table lists the solution to a commonly encountered Create User error.

Problem Description	Solution
Oracle Identity Governance cannot create a user. **Returned Error Message:** User already exists	A user with the assigned ID already exists in the target system.

5.2.3 Delete User Errors

This table lists the solution to a commonly encountered Delete User error.

Problem Description	Solution
Oracle Identity Governance cannot delete a user. **Returned Error Message:** User does not exist in target system	The specified user does not exist in the target system.

5.2.4 Edit User Errors

This table lists the solution to a commonly encountered Edit User error.

Problem Description	Solution
Oracle Identity Governance cannot update a user. **Returned Error Message:** User does not exist in target system	Review the log for more details.

5.2.5 TimeOut Errors

This table lists the solution to a commonly encountered TimeOut error.

Problem Description	Solution
Connection timeout during connector operations, typically during reconciliation operations. **Returned Error Message:** Command timed out	Increase the value of the commandTimeout entry in the config lookup definition. See Setting up the Lookup Definitions for Connector Configuration for more information.
Reconciliation fails with an error. **Returned Error Message:** org.identityconnectors.genericunix.common. GenericUnixUtil :getMatchedResponseInfo : Buffer:-> > > > >Password:	Change the value of the sudoPasswdExpectExpression entry in the Lookup.Unix.Configuration.Trusted lookup definition from Password: to `password.` **Note**: The letter "P" must be in lowercase.

6 Known Issues

The following are known issues associated with this release of the connector:

- **Bug 13979826**

  After performing a trusted delete reconciliation operation, a user remains in provisioned status if the user is deleted from Oracle Identity Manager (OIM).

  As a work around, run the target delete reconciliation scheduled task after running the trusted delete reconciliation task.

  For any use case where identity is reconciled authoritatively from a target system to OIM, and also account daily changes are provisioned from OIM back to the target system, this behavior will be observed.

  The solution is to leverage on the existing infrastructure of account reconciliation and schedule it to run at the optimal frequency so that it follows trusted delete reconciliation scheduled task run. After the account reconciliation is completed, the scheduled task will convert the Account status to Revoked state. Eventually, the identity data maintained inside OIM would look correct from operational as well as audit perspective.

- **Bug 13975499**

  This issue applies to Oracle Identity Manager releases prior to 11.1.2. The request dataset provided with the connector does not contain the User Login field, which is usually fed directly from Oracle Identity Manager user profile to the process form using a prepopulate adapter.

  To include the User Login field in request dataset, perform the following procedure:

  1. Export the current dataset using the MDS export utility.
  2. Update the dataset to include the User Login field.
  3. Import the updated dataset using the MDS import utility.
  4. Purge the cache, as described in Clearing Content Related to Connector Resource Bundles from the Server Cache.

- **Bug 13650341**

  During a provisioning operation, if the connector encounters a timeout error, an incorrect response message is displayed in the resource history of the scheduled task.

  For example:

  ```
  Response: CONNECTOR_EXCEPTION
  ```

  ```
  Response Description: Unknown response received
  ```

- **Bug 13690646**

  After upgrading the connector, the parent form is not linked to the child form. As a work around, you must create a new version of the parent form and make it active.

- **Bug 13695758**

  If you update an encrypted basic configuration parameter such as loginUserPassword, other encrypted parameters such as passphrase and rbacRolePassword also appear to have been updated.

A Privileges Required for Performing Provisioning and Reconciliation

These are the privileges required for successful provisioning operations and reconciliation runs.

This appendix includes the following topics:

- Privileges Required for Running Commands on Solaris and Linux
- Privileges Required for Running Commands on HP-UX
- Privileges Required for Running Commands on AIX

A.1 Privileges Required for Running Commands on Solaris and Linux

These are the privileges and permissions required to run commands on Solaris and Linux for performing provisioning operations and reconciliation runs.

Users must have privileges to run the following commands:

```
usermod, useradd, userdel, passwd, chage, id, cut, touch, awk, uniq
```

In addition, the users must have execute permissions for the following commands:

```
sed, cat, diff, sort, rm, grep, egrep, echo, /usr/bin/sh, /bin/sh
```

Users must have read and write permissions on the `/home`, `/tmp` and `/etc/connector_mirror_files` (or the mirror files directory specified in the configuration lookup definition) directories.

A.2 Privileges Required for Running Commands on HP-UX

These are the privileges and permissions required to run commands on HP-UX for performing provisioning operations and reconciliation runs.

Users must have privileges to run the following commands:

```
modprpw, usermod, useradd, userdel, passwd, chage, id, cut, touch, awk,
uniq
```

In addition, the users must have execute permissions for the following commands:

```
sed, cat, diff, sort, rm, grep, egrep, echo, /usr/bin/sh, /bin/sh
```

Users must have read and write permissions on the `/home`, `/tmp` and `/etc/connector_mirror_files` (or the mirror files directory specified in the configuration lookup definition) directories.

A.3 Privileges Required for Running Commands on AIX

These are the privileges and permissions required to run commands on AIX for performing provisioning operations and reconciliation runs.

User must have privileges to execute the following commands:

`mkuser, useradd, chuser, rmuser, lsuser, /usr/bin/usermod, /usr/chuser`, `id, cut, touch, awk, uniq`

In addition, the users must have execute permissions for the following commands:

`/usr/bin/bdiff, sh, cat, /usr/bin/sort, /usr/bin/rm, /usr/bin/grep, /bin/echo, /bin/sed, command.`

Users must have read and write permissions on the `/home`, `/tmp` and `/etc/connector_mirror_files` (or the mirror files directory specified in the configuration lookup definition) directories.

B

Sample Scripts for Updating Default Attributes for Reconciliation

This appendix provides sample scripts for editing the default attribute mappings for reconciliation.

Sample scripts for the procedure described in Adding Custom Attributes for Target Resource Reconciliation are as follows:

- Original Sample Script
- Updated Sample Script

B.1 Original Sample Script

This is the original FetchAllUserRecords.txt script.

```
while read inputline ;do
    __NAME__=$(echo $inputline | cut -d: -f1);
    USID=$(echo $inputline | cut -d: -f3);
    COMMENTS=$(echo $inputline | cut -d: -f5);
    HOME_DIR=$(echo $inputline | cut -d: -f6);
    CREATE_HOME_DIR="false";
    if [ -d "$HOME_DIR" ] ;then CREATE_HOME_DIR="true";fi;
    USER_SHELL=$(echo $inputline | cut -d: -f7);
    PGROUP=$( id -G -n $__NAME__ | cut -d' ' -f1);
    shadowRecord=$(cat /etc/shadow |grep $__NAME__);
    INACTIVE=$(echo $shadowRecord | cut -d: -f7);
    EXP_DATE=$(echo $shadowRecord | cut -d: -f8);
    secgrplist="";
    id -G -n $__NAME__ | grep -q " ";
    if [ $? -eq 0 ];then
        secgrplist=$( id -G -n $__NAME__ | cut -d ' ' -f2- | sed 's/ /~~~/g');
    fi;
    ENABLE="true";
    if [ ! -z "$__NAME__" ] ;then
        passwd -S $__NAME__ | grep -w LK >/dev/null;
         if [ $? -eq 0 ] ;then
            ENABLE="false";
         fi;
         passwd -S $__NAME__ | grep -w locked >/dev/null;
         if [ $? -eq 0 ] ;then
            ENABLE="false";
         fi;
    fi;

RESULT1=__NAME__:$__NAME__:__ENABLE__:$ENABLE:COMMENTS:$COMMENTS:USID:$USID:USER_SHEL
L:$USER_SHELL:HOME_DIR:$HOME_DIR:;
        RESULT2=CREATE_HOME_DIR:$CREATE_HOME_DIR:SECONDARYGROUP:$secgrplist:;

RESULT3=PGROUP:$PGROUP:INACTIVE:$INACTIVE:EXP_DATE:$EXP_DATE:__UID__:$__NAME__;
        RESULT="$RESULT1$RESULT2$RESULT3";
    echo "$RESULT" | grep -q -w "$connectorPrompt";
```

```
    if [ $? -gt 0 ];then
        echo "RESULT_START $RESULT RESULT_END";
    else
        echo "Record contains connector prompt. Hence ignored";
    fi;
done < /etc/passwd;[ $? -eq 0 ] && echo "SUCCESS";
unset inputline __NAME__ USID COMMENTS HOME_DIR USER_SHELL PGROUP secgrplist;
unset ENABLE passwordFull passwordF passwordS RESULT RESULT1 RESULT2 RESULT3 __UID__
INACTIVE EXP_DATE shadowRecord;
```

B.2 Updated Sample Script

This is the FetchAllUserRecords.txt script that has been updated to include the newly added __GID__ attribute. The updated lines are represented in bold font.

```
while read inputline ;do
    __NAME__=$(echo $inputline | cut -d: -f1);
    USID=$(echo $inputline | cut -d: -f3);
    COMMENTS=$(echo $inputline | cut -d: -f5);
    HOME_DIR=$(echo $inputline | cut -d: -f6);
    CREATE_HOME_DIR="false";
    if [ -d "$HOME_DIR" ] ;then CREATE_HOME_DIR="true";fi;
    USER_SHELL=$(echo $inputline | cut -d: -f7);
    PGROUP=$( id -G -n $__NAME__ | cut -d' ' -f1);
    __GID__=$( id -G $__NAME__ | cut -d' ' -f1);
    shadowRecord=$(cat /etc/shadow |grep $__NAME__);
    INACTIVE=$(echo $shadowRecord | cut -d: -f7);
    EXP_DATE=$(echo $shadowRecord | cut -d: -f8);
    secgrplist="";
    id -G -n $__NAME__ | grep -q " ";
    if [ $? -eq 0 ];then
        secgrplist=$( id -G -n $__NAME__ | cut -d ' ' -f2- | sed 's/ /~~~/g');
    fi;
    ENABLE="true";
    if [ ! -z "$__NAME__" ] ;then
        passwd -S $__NAME__ | grep -w LK >/dev/null;
         if [ $? -eq 0 ] ;then
            ENABLE="false";
         fi;
         passwd -S $__NAME__ | grep -w locked >/dev/null;
         if [ $? -eq 0 ] ;then
            ENABLE="false";
         fi;
    fi;

RESULT1=__NAME__:$__NAME__:__ENABLE__:$ENABLE:COMMENTS:$COMMENTS:USID:$USID:USER_SHEL
L:$USER_SHELL:HOME_DIR:$HOME_DIR:;
        RESULT2=CREATE_HOME_DIR:$CREATE_HOME_DIR:SECONDARYGROUP:$secgrplist:;

RESULT3=PGROUP:$PGROUP:INACTIVE:$INACTIVE:EXP_DATE:$EXP_DATE:__UID__:$__NAME__:__GID_
_:$__GID__;
        RESULT="$RESULT1$RESULT2$RESULT3";
    echo "$RESULT" | grep -q -w "$connectorPrompt";
    if [ $? -gt 0 ];then
        echo "RESULT_START $RESULT RESULT_END";
    else
        echo "Record contains connector prompt. Hence ignored";
    fi;
done < /etc/passwd;[ $? -eq 0 ] && echo "SUCCESS";
unset inputline __NAME__ USID COMMENTS HOME_DIR USER_SHELL PGROUP secgrplist __GID__;
```

```
unset ENABLE passwordFull passwordF passwordS RESULT RESULT1 RESULT2 RESULT3 __UID__
INACTIVE EXP_DATE shadowRecord;
```

C

Sample Scripts for Updating Default Attributes for Provisioning

This appendix provides sample scripts for editing the default attribute mappings for provisioning.

Sample scripts for the procedure described in Adding Custom Attributes for Provisioning are as follows:

- Original Sample Script
- Updated Sample Script

C.1 Original Sample Script

This is the original CreateNativeUser.txt script.

```
if [ ! -z "$__UID__" ] ;then
    __NAME__=$__UID__;
else
    __NAME__=$__NAME__;
fi;
if id $__NAME__ > /dev/null 2>&1 ;then
    echo "User already exists";
else
    globalVar="true";
    if [ ! -z $SECONDARYGROUP ] ;then
        command="$command -G $SECONDARYGROUP";
    fi;

    homedir="";
    if [ ! -z $HOME_DIR ] ;then
        homedir=$HOME_DIR;
    else
        if [ ! -z $defaultHomeBaseDir ] ;then
            homedir=$defaultHomeBaseDir;
        fi;
    fi;
    if [ ! -z $homedir ] ;then
        checkHomeBaseDir=$(test -d $homedir && echo "true" || echo "false");
        if [ $checkHomeBaseDir == "true" ]; then
            command="$command -d $homedir/$__NAME__";
        else
            globalVar="false";
            echo "useradd: cannot create directory $homedir/$__NAME__";
        fi;
    fi;
    if [ ! -z $EXP_DATE ] ;then
        command="$command -e $EXP_DATE";
    fi;
    if [ ! -z $INACTIVE ] ;then
        command="$command -f $INACTIVE";
    fi;
    if [ ! -z $PGROUP ] ;then
        grp=$PGROUP;
    else
        if [ ! -z $defaultPriGroup ] ;then
            grp=$defaultPriGroup;
        fi;
    fi;
    if [ ! -z $grp ] ;then
        getent group $grp;
        if [ $? -ne 0 ] ;then
            echo "PGROUP=$grp";
            echo "Invalid primary group :- $grp";
            globalVar="false";
        else
            command="$command -g $grp";
        fi;
    fi;
    if [ ! -z $CREATE_HOME_DIR ] && [  $CREATE_HOME_DIR == "true" ] ;then
        command="$command -m";
            if [ ! -z $SKEL_DIR ] ;then
                command="$command -k $SKEL_DIR";
            fi;
    fi;
    if [ ! -z $CREATE_USER_GROUP ] && [  $CREATE_USER_GROUP == "false" ] ;then
        command="$command -n";
    fi;
    if [ ! -z $USER_SHELL ] ;then
        command="$command -s $USER_SHELL";
    else
        if [ ! -z $defaultShell ] ;then
            command="$command -s $defaultShell";
        fi;
    fi;
    if [ ! -z $USID ] && [  $USID -gt 0 ] ;then
        command="$command -u $USID";
        if [ ! -z $UNIQUE_USID ] && [  $UNIQUE_USID == "false" ] ;then
            command="$command -o";
        fi;
    fi;
    if [ $globalVar == "true" ] ;then
        echo "useradd $command $__NAME__";
        useradd $command $__NAME__;
        if [ $? -eq 0 ]; then echo "SUCCESS";
            if [ ! -z "$COMMENTS" ] ; then echo "usermod -c
\""$COMMENTS"\" $__NAME__";
                usermod -c "$COMMENTS" $__NAME__;
            fi;
        fi;
    fi;
fi;
unset bar COMMENTS HOME_DIR PGROUP grp EXP_DATE SKEL_DIR UNIQUE_USID __NAME__
__UID__ CREATE_HOME_DIR;
unset USER_SHELL USID CREATE_USER_GROUP INACTIVE SECONDARYGROUP command globalVar
name;
unset defaultHomeBaseDir homedir checkHomeBaseDir grp defaultPriGroup;
```

C.2 Updated Sample Script

This is the CreateNativeUser.txt script that has been updated to include the newly added __GID__ attribute. The updated lines are represented in bold font.

```
if [ ! -z "$__UID__" ] ;then
    __NAME__=$__UID__;
else
    __NAME__=$__NAME__;
fi;
if id $__NAME__ > /dev/null 2>&1 ;then
    echo "User already exists";
else
    globalVar="true";
    if [ ! -z $SECONDARYGROUP ] ;then
        command="$command -G $SECONDARYGROUP";
    fi;

    homedir="";
    if [ ! -z $HOME_DIR ] ;then
        homedir=$HOME_DIR;
    else
        if [ ! -z $defaultHomeBaseDir ] ;then
            homedir=$defaultHomeBaseDir;
        fi;
    fi;
    if [ ! -z $homedir ] ;then
        checkHomeBaseDir=$(test -d $homedir && echo "true" || echo "false");
        if [ $checkHomeBaseDir == "true" ]; then
            command="$command -d $homedir/$__NAME__";
        else
            globalVar="false";
            echo "useradd: cannot create directory $homedir/$__NAME__";
        fi;
    fi;
    if [ ! -z $EXP_DATE ] ;then
        command="$command -e $EXP_DATE";
    fi;
    if [ ! -z $INACTIVE ] ;then
        command="$command -f $INACTIVE";
    fi;
    if [ ! -z $PGROUP ] ;then
        grp=$PGROUP;
    else
        if [ ! -z $defaultPriGroup ] ;then
            grp=$defaultPriGroup;
        fi;
    fi;
    if [ ! -z $grp ] ;then
        getent group $grp;
        if [ $? -ne 0 ] ;then
            echo "PGROUP=$grp";
            echo "Invalid primary group :- $grp";
            globalVar="false";
        else
            command="$command -g $grp";
        fi;
    fi;
    if [ ! -z $CREATE_HOME_DIR ] && [  $CREATE_HOME_DIR == "true" ] ;then
```

```
        command="$command -m";
            if [ ! -z $SKEL_DIR ] ;then
                command="$command -k $SKEL_DIR";
            fi;
    fi;
    if [ ! -z $CREATE_USER_GROUP ] && [  $CREATE_USER_GROUP == "false" ] ;then
        command="$command -n";
    fi;
    if [ ! -z $USER_SHELL ] ;then
        command="$command -s $USER_SHELL";
    else
        if [ ! -z $defaultShell ] ;then
            command="$command -s $defaultShell";
        fi;
    fi;
    if [ ! -z $USID ] && [  $USID -gt 0 ] ;then
        command="$command -u $USID";
        if [ ! -z $UNIQUE_USID ] && [  $UNIQUE_USID == "false" ] ;then
            command="$command -o";
        fi;
    fi;
    if [ ! -z $__GID__ ] ;then
        command="$command -g $__GID__";
    fi;
    if [ $globalVar == "true" ] ;then
        echo "useradd $command $__NAME__";
        useradd $command $__NAME__;
        if [ $? -eq 0 ]; then echo "SUCCESS";
            if [ ! -z "$COMMENTS" ] ; then echo "usermod -c
\""$COMMENTS"\" $__NAME__";
                usermod -c "$COMMENTS" $__NAME__;
            fi;
        fi;
    fi;
fi;
unset bar COMMENTS HOME_DIR PGROUP grp EXP_DATE SKEL_DIR UNIQUE_USID __NAME__
__UID__ CREATE_HOME_DIR;
unset USER_SHELL USID CREATE_USER_GROUP INACTIVE SECONDARYGROUP command globalVar
name;
unset defaultHomeBaseDir homedir checkHomeBaseDir grp defaultPriGroup __GID__;
```