Oracle® Identity Governance

Configuring the Oracle NetSuite Application

12.2.1.3
F84737-01
August 2023

Oracle Identity Governance Configuring the Oracle NetSuite Application, 12.2.1.3

F84737-01

Primary Author: Maya Chakrapani

Contributing Authors: Syam Kumar Battu

Contents

1 Introduction to the Connector

1.1 Certified Components 1-1
1.2 Usage Recommendation 1-2
1.3 Certified Languages 1-2
1.4 Supported Connector Operations 1-3
1.5 Connector Architecture 1-3
1.6 Use Cases Supported by the Connector 1-5
1.7 Connector Features 1-5
1.7.1 User Provisioning 1-6
1.7.2 Full Reconciliation 1-6
1.7.3 Limited (Filtered) Reconciliation 1-7
1.7.4 Support for the Connector Server 1-7
1.7.5 Transformation and Validation of Account Data 1-7
1.7.6 Support for Cloning Applications and Creating Instance Applications 1-7
1.7.7 Secure Communication to the Target System 1-8

2 Creating an Application by Using the Connector

2.1 Prerequisites for Creating an Application by Using the Connector 2-1
2.1.1 Configuring target system to perform connector Operations 2-1
2.1.2 OAuth2.0 Flow to Generate the User-Level Tokens 2-3
2.1.3 Downloading the Connector Installation Package 2-5
2.2 Process Flow for Creating an Application by Using the Connector 2-6
2.3 Creating an Application by Using the NetSuite Cloud Connector 2-6

3 Configuring the Connector

3.1 Basic Configuration Parameters 3-1
3.2 Advanced Settings Parameters 3-3
3.3 Attribute Mappings 3-3
3.3.1 Attribute Mappings for the Target Application 3-3
3.4 Correlation Rules 3-8
3.4.1 Correlation Rules for the Target Application 3-9

3.5 Reconciliation Jobs 3-11

4 Performing Post configuration Tasks for the Connector

4.1 Configuring Oracle Identity Governance 4-1
4.1.1 Creating and Activating a Sandbox 4-1
4.1.2 Creating a New UI Form 4-2
4.1.3 Publishing a Sandbox 4-2
4.1.4 Updating an Existing Application Instance with a New Form 4-2
4.2 Harvesting Entitlements and Sync Catalog 4-3
4.3 Managing Logging for the Connector 4-3
4.3.1 Understanding Logging on the Connector Server 4-3
4.3.2 Enabling Logging for the Connector Server 4-4
4.3.3 Understanding Log Levels 4-4
4.3.4 Enabling Logging 4-5
4.4 Configuring the IT Resource for the Connector Server 4-6
4.5 Localizing Field Labels in UI Forms 4-7
4.6 Configuring SSL 4-9

5 Using the Connector

5.1 Configuring Reconciliation 5-1
5.1.1 Performing Full Reconciliation 5-1
5.1.2 Performing Limited (Filtered) Reconciliation 5-1
5.2 Configuring Reconciliation Jobs 5-2
5.3 Configuring Provisioning 5-3
5.3.1 Guidelines on Performing Provisioning Operations 5-3
5.3.2 Performing Provisioning Operations 5-4

6 Extending the Functionality of the Connector

6.1 Configuring Transformation and Validation of Data 6-1
6.2 Configuring Action Scripts 6-1
6.3 Configuring the Connector for Multiple Installations of the Target System 6-2

7 Known Issues and Workarounds

8 Files and Directories in the Connector Installation Package

List of Figures

1-1	Oracle NetSuite Connector Architecture	1-4
2-1	Overall Flow of the Process for Creating an Application By Using the Connector	2-6
3-1	Default Attribute Mappings for Oracle NetSuite User Account	3-6
3-2	Default Attribute Mappings for NetSuite Roles	3-7
3-3	Default Attribute Mappings for NetSuite Groups	3-8
3-4	Default Attribute Mappings for NetSuite Global Permission	3-8
3-5	Simple Correlation Rule for NetSuite Target Application	3-10
3-6	Predefined Situations and Responses for a Oracle NetSuite Target Application	3-11

1

Introduction to the Connector

This chapter introduces the Oracle NetSuite Application connector.

Oracle Identity Governance is a centralized identity management solution that provides self-service, compliance, provisioning, and password management services for applications residing on-premises or on the Cloud. Oracle Identity Governance connectors are used to integrate Oracle identity Governance with the external identity-aware applications.

The Oracle NetSuite Connector lets you create and onboard Oracle NetSuite applications in Oracle Identity Governance.

Note:

In this guide, the connector that is deployed using the **Applications** option on the **Manage** tab of Identity Self Service is referred to as an **AOB application.**

From Oracle Identity Governance release 12.2.1.3.0 onward, connector deployment is handled using the application onboarding capability of Oracle Identity Self Service. This capability lets business users to onboard applications with minimum details and effort. The connector installation package includes a collection of predefined templates (XML files) that contain all the information required for provisioning and reconciling data from a given application or target system. These templates also include basic connectivity and configuration details specific to your target system. The connector uses information from these predefined templates allowing you to onboard your applications quickly and easily using only a single and simplified UI.

Application onboarding is the process of registering or associating an application with Oracle Identity Governance and making that application available for provisioning and reconciliation of user information.

The following topics provide a high-level overview of the connector:

1. Certified Components
2. Usage Recommendation
3. Certified Languages
4. Supported Connector Operations
5. Connector Architecture
6. Use Cases Supported by the Connector
7. Connector Features

1.1 Certified Components

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

Table 1-1 Certified Components

Component	Requirement for AOB Application
Oracle Identity Governance or Oracle Identity Manager	You can use any one of the following releases: • Oracle Identity Governance 12c PS4 (12.2.1.4.0) or later. • Oracle Identity Governance 12*c* PS3 (12.2.1.3.0) or later.
Oracle Identity Governance or Oracle Identity Manager JDK	JDK 1.8 and later
Target systems	Oracle NetSuite Release 2023.1
Connector Server	11.1.2.1.0 or 12.2.1.3.0
Connector Server JDK	JDK 1.8 and later
Target API version	NetSuite v1 and NetSuitePort_2022_1

1.2 Usage Recommendation

If you are using Oracle Identity Governance 12*c* (12.2.1.3.0) or later, then use the latest 12.2.1.*x* version of this connector. Deploy the connector using the **Applications** option on the **Manage** tab of Identity Self Service.

1.3 Certified Languages

These are the languages that the connector supports.

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

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

1.4 Supported Connector Operations

These are the list of operations that the connector supports for your target system.

Table 1-2 Supported Connector Operations

Operation	Supported?
User Management	
Create user	Yes
Update user	Yes
Enable user	Yes
Disable user	Yes
Delete user	Yes
Reset Password	Yes
Role Grant Management	
Assign and Revoke Roles	Yes
Group Grant Management	
Assign and Revoke Group	Yes
Global Permission Management	
Global permissions (Reconciliation only)	Yes

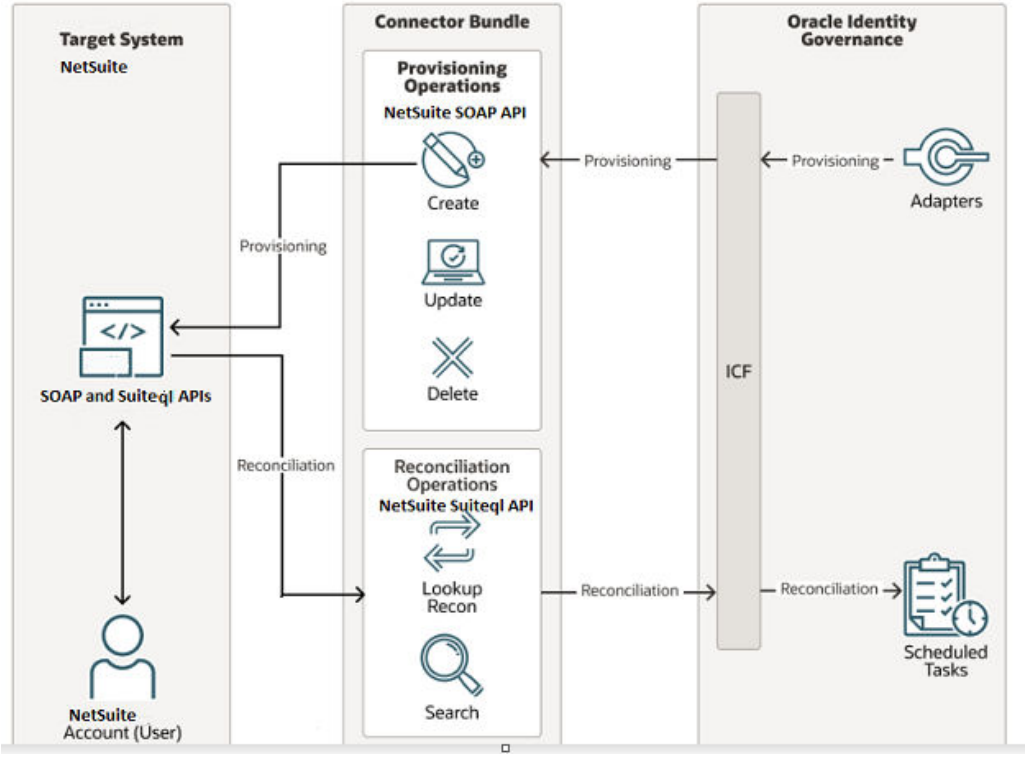
1.5 Connector Architecture

The Oracle NetSuite is implemented by using the Identity Connector Framework (ICF).

The ICF is a component that is required in order to use Identity Connector. ICF provides basic reconciliation and provisioning operations that are common to all Oracle Identity Governance connectors. In addition, ICF provides common features that developers would otherwise need to implement on their own, such as, buffering, time outs, and filtering. ICF is distributed together with Oracle Identity Governance. Therefore, you do not need to configure or modify ICF.

The following figure shows the architecture of the Oracle NetSuite.

Figure 1-1 Oracle NetSuite Connector Architecture



The connector is configured to run in one of the following modes:

Account management

Account management is also known as target resource management. In this mode, the target system is used as a target resource and the connector enables the following operations:

Provisioning

Provisioning involves creating, updating, or deleting users on the target system through Oracle Identity Governance. During provisioning, the Adapters invoke ICF operation, ICF in turn invokes create operation on the NetSuite Identity Connector Bundle and then the bundle calls the target system API (NetSuite API) for provisioning operations. The API on the target system accepts provisioning data from the bundle, carries out the required operation on the target system, and returns the response from the target system back to the bundle, which passes it to the adapters.

Target resource reconciliation

During reconciliation, a scheduled task invokes an ICF operation. ICF in turn invokes a search operation on the Oracle NetSuite Identity Connector Bundle and then the bundle calls NetSuite API for Reconciliation operation. The API extracts user records that match the reconciliation criteria and hands them over through the bundle and ICF back to the scheduled task, which brings the records to Oracle Identity Governance.

Each record fetched from the target system is compared with NetSuite resources that are already provisioned to OIM Users. If a match is found, then the update made to the NetSuite record from the target system is copied to the NetSuite resource in Oracle

Identity Governance. If no match is found, then the Name of the record is compared with the User Login of each OIM User. If a match is found, then data in the target system record is used to provision a NetSuite resource to the OIM User.

The Oracle NetSuite Identity Connector Bundle communicates with the NetSuite API using the HTTPS protocol. The NetSuite API provides programmatic access to NetSuite through Suite-QL API and SOAP API endpoints. Apps can use the Suite-QL API and SOAP API to perform create, read, update, and delete (CRUD) operations on directory data and directory objects, such as users, roles, global permissions and groups.

> **See Also:**
>
> Understanding the Identity Connector Framework in *Oracle Fusion Middleware Developing and Customizing Applications* for Oracle Identity Governance for more information about ICF.

1.6 Use Cases Supported by the Connector

Oracle NetSuite is used to integrate Oracle Identity Governance with Oracle NetSuite to ensure that all Oracle NetSuite accounts are created, updated, and deactivated on an integrated cycle with the rest of the identity-aware applications in your enterprise. NetSuite supports management of identities for Cloud Identity, Synchronized Identity, and Federated Identity models of Oracle NetSuite. In a typical IT scenario, an organization using Oracle Identity Governance wants to manage accounts, groups, roles across Oracle NetSuite Cloud Service. The following are some of the most common scenarios in which this connector can be used:

- **Oracle NetSuite User Management**:

An organization using Oracle NetSuite wants to integrate with Oracle Identity Governance to manage identities. The organization wants to manage its user identities by creating them in the target system using Oracle Identity Governance. The organization also wants to synchronize user identity changes performed directly in the target system with Oracle Identity Governance. In such a scenario, a quick and an easy way is to install Oracle NetSuite and configure it with your target system by providing connection information.

To create a new user in the target system, fill in and submit the OIM process form to trigger the provisioning operation. The connector executes the CreateOp operation against your target system and the user is created on successful execution of the operation. Similarly, operations like delete and update can be performed.

To search or retrieve the user identities, you must run a scheduled task from Oracle Identity Governance. The connector will run the corresponding SearchOp against the user identities in the target system and fetch all the changes to Oracle Identity Governance

1.7 Connector Features

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

The following table provides the list of features supported by the AOB application.

Table 1-3 Supported Connector Features Matrix

Feature	AOB Application
User Provisioning	Yes
Full reconciliation	Yes
Limited (Filtered)reconciliation	Yes
Delete reconciliation	No
Use connector server	Yes
Transformation and validation of account data	Yes
Perform connector operations in multiple domains	Yes
Support for pagination	Yes
Test connection	Yes
Clone applications or create new application instances	Yes
Provide secure communication to the target system through SSL	Yes
Reset password	Yes

The following topics provide more information on the features of the AOB application:

1. User Provisioning
2. Full Reconciliation
3. Limited (Filtered) Reconciliation
4. Support for the Connector Server
5. Transformation and Validation of Account Data
6. Support for Cloning Applications and Creating Instance Applications
7. Secure Communication to the Target System

1.7.1 User Provisioning

User provisioning involves creating or modifying the account data on the target system through Oracle Identity Governance.

> **Note:**
>
> For more information, see Performing Provisioning Operations.

1.7.2 Full Reconciliation

You can perform full reconciliation to bring all existing user data from the target system to Oracle Identity Governance.

For more information, see Performing Full Reconciliation.

1.7.3 Limited (Filtered) Reconciliation

You can reconcile records from the target system based on a specified filter criterion. To limit or filter the records that are fetched into Oracle Identity Governance during a reconciliation run, you can specify the subset of added or modified target system records that must be reconciled.

You can set a reconciliation filter as the value of the Filter Suffix attribute of the user reconciliation scheduled job. The Filter Suffix attribute helps you to assign filters to the API based on which you get a filtered response from the target system.

For more information, see Performing Limited (Filtered) Reconciliation.

1.7.4 Support for the Connector Server

Connector Server is one of the features provided by ICF. By using one or more connector servers, the connector architecture permits your application to communicate with externally deployed bundles.

A Java connector server is useful when you do not want to execute a Java connector bundle in the same VM as your application. It can be beneficial to run a Java connector on a different host for performance improvements if the bundle works faster when deployed on the same host as the native managed resource.

> **See Also:**
>
> Using an Identity Connector Server in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager* for more information about installing and configuring connector server and running the connector server.

1.7.5 Transformation and Validation of Account Data

You can configure transformation and validation of account data that is brought into or sent from Oracle Identity Governance during reconciliation and provisioning operations by writing Groovy scripts while creating your application.

For more information, see Validation and Transformation of Provisioning and Reconciliation Attributesin *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

1.7.6 Support for Cloning Applications and Creating Instance Applications

You can configure this connector for multiple installations of the target system by cloning applications or by creating instance applications.

When you clone an application, all the configurations of the base application are copied into the cloned application. When you create an instance application, it shares all configurations as the base application.

For more information about these configurations, see Cloning Applications and Creating Instance Applications in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

1.7.7 Secure Communication to the Target System

To provide secure communication to the target system, SSL is required. You can configure SSL between Oracle Identity Governance and the Connector Server and between the Connector Server and the target system.

If you do not configure SSL, passwords can be transmitted over the network in clear text. For example, this problem can occur when you are creating a user or modifying a user's password.

For more information, see Configuring SSL.

2

Creating an Application by Using the Connector

Learn about onboarding applications using the connector and the prerequisites for doing so.

- Prerequisites for Creating an Application by Using the Connector
- Process Flow for Creating an Application by Using the Connector
- Creating an Application by Using the NetSuite Cloud Connector

2.1 Prerequisites for Creating an Application by Using the Connector

Learn about the tasks that you must complete before you create the application.

- Configuring target system to perform connector Operations
- OAuth2.0 Flow to Generate the User-Level Tokens
- Downloading the Connector Installation Package

2.1.1 Configuring target system to perform connector Operations

This is a high-level summary of the tasks to be performed on the target system before you create the application.

Pre-installation for the NetSuite connector involves performing a series of tasks on the target system.

Pre-installation involves the following tasks:

1. Login to Oracle NetSuite.
2. Go to ***Setup > Company > Enable Features*** .
3. Click **SuiteCloud** sub-tab and enable the following features from the respective menu items.
    a. **SuiteBuilder**
    Enable the following boxes:
        i. ITEM OPTIONS
        ii. CUSTOM RECORDS
        iii. ADVANCED PDF/HTML TEMPLATES
        iv. REMOVE PERSONAL INFORMATION
    b. **SuiteScript**:
        i. CLIENT SUITESCRIPT
        ii. SERVER SUITESCRIPT

c. **SuiteFlow**
   i. SUITEFLOW
d. **SuiteGL**
   i. CUSTOM GL LINES
   ii. CUSTOM TRANSACTIONS
   iii. CUSTOM SEGMENTS
e. **SuiteBundler**
   i. CREATE BUNDLES WITH SUITEBUNDLER
f. **SuiteTalk**
   i. SOAP WEB SERVICES
   ii. REST WEB SERVICES
g. **Manage Authentication**
   i. SUITESIGNON
   ii. TOKEN-BASED AUTHENTICATION
   iii. OAUTH 2.0
h. **SuiteCloud Development Framework**
   i. SUITECLOUD DEVELOPMENT FRAMEWORK

4. Click **SAVE**.

To create an integration record for an application, follow the below steps:

- 1. Go to **Setup > Integration> Manage Integration > New**.
  2. Enter a name for your application in the Name field.
  3. Enter a description in the Description field, if preferred.
  4. Select Enabled in the State field.
  5. Enter a note in the Note field, if preferred.
  6. On the **Authentication** tab, check the appropriate boxes for your application:
     a. Token-based Authentication
        – i. TOKEN-BASED AUTHENTICATION
          ii. TBA: AUTHORIZATION FLOW
          iii. Define the CALLBACK URL.
     b. O-Auth 2.0
        i. AUTHORIZATION CODE GRANT
        ii. Scope
          i. RESTLETS
          ii. REST WEB SERVICES
        iii. Provide a valid REDIRECT URI
  7. Click **SAVE**.

8. Ensure to copy the Client Credentials details that will appear on the screen as it is one-time display.
   For Example:

   consumerKey = "fcb9ec7e7d386fab36566e9c4159bXXXXXXX2875841d828aee7e"

   consumerSecret = "bd7780d4396715f5f4586d874379XXXXXX38c42a525c95f70"

To create and assign a Token Based Authentication token:

1. Log in as a user with the **Access Token Management** permission.
2. Go to **Setup > Users/Roles > Access Tokens.**
3. On the Access Tokens page, click **New Access Token**.
4. On the Access Token page:
   a. Select the **Application Name**.
   b. Select the **User**.
   c. Select the **Role**.
   d. The **Token Name** is already populated by default with a concatenation of Application Name, User, and Role. Enter your own name for this token, if preferred.
5. Click **Save**.
6. Ensure to copy the Token details that will appear on the screen as it is one-time display.
   For example:
   tokenId = "0948d37f7XXXXXXXXXXXXXX8075";

   tokenSecret = "86b7bb19cXXXXXXXXabfa0eb401e2c2c24b”

2.1.2 OAuth2.0 Flow to Generate the User-Level Tokens

To generate the user-level access and refresh tokens, there are two steps you must complete manually, and these values should be provided in authToken in Oracle NetSuite Connector basic configuration for authentication.

The following steps must be completed by users who are opting in for Authorization Code Grant:

You must pass the Authorization code grant URL in the internet browser or use Postman to generate the tokens.

1. **Requesting the Authorization Code**

> **Note:**
>
> The token URI for the developer environment is as follows:
> `https://<host name>/services/rest/auth/oauth2/v1/token.`

   a. Enter the following URL in a browser as provided in the example.
   Example:

```
https://<host name> /app/login/oauth2/authorize.nl?
redirect_uri={callback}&response_type=code&scope=restlets+rest_webservice
s&state=ykv2XLx1BpT5Q0F3MRPHb94j&client_id={ConsumerKey}.
```

Replace `{ConsumerKey}` with your Consumer key / Client id and `{callback}` with your redirect URI. The URL above includes the signature scope required for the eSignature REST API.

This URL opens the Oracle NetSuite authentication screen.

b. After you enter your Oracle NetSuite account email address and password and give consent for the requested scopes and then once you redirect to the login Browser Enter the user Credentials to Login and authenticate then Click on the Continue to allow Oracle NetSuite to access your information to Provide the code. The browser will redirect to your redirect URI with a long string returned for the code parameter embedded in the URL.
Request:

```
https://<host name>/app/login/oauth2/authorize.nl?
redirect_uri=http://
example.com&response_type=code&scope=restlets+rest_webservices&stat
e=ykv2XLx1BpT5Q0F3MRPHb94j&client_id=7e1c238e-xxxx-xxxx-xxxx-
abcea08a3171
```

Response: `https://example.com/?state=ykv2XLx1BpT5Q0F3MRPHb94j&role=3&entity=4622&company=TSTDRVXXXXXX&code=096835b6aced….....457b00e3c`

2. **Generating Refresh Tokens Using the Code Generated in Step 1**

a. To request a refresh token, send a POST request containing your authorization code to the NetSuite authentication service.

b. Paste the values of Consumer Key and Consumer secret key as User name and Password respectively under Authorization in the Refresh token request with the type as Basic Auth in Postman.

c. In addition, the refresh token request contains a set of body parameters namely grant_type and code.

i. Update the key as code with value <code>.

> **Note:**
>
> <code> is nothing but the authorization code that you received from the callback in step 1.
> For example, code=096835b6aced.....457b00e3c.

ii. Similarly, update one or more body parameter with the key as `grant_type` and value as `authorization_code` and another body parameter with key as `redirect_uri` and value as the same provided in the step 1.

d. Execute the Authorize Code Grant Refresh Token request to generate an access token and a refresh token.

i. In the response, you will get elements, namely, access_token, token_type, refresh_token, and expires_in.

ii. Copy/save the values of refresh_token.

For more information about how to get a refresh token with Auth Code Grant, see NetSuite Applications Suite.

Examples:

Request:

```
curl --location --request POST " https://<host name>/services/rest/auth/
oauth2/v1/token"--header "Authorization: Basic
N2UxYzIzOGU1Zj........GI3Njg3MzMzMTZm" --header "Content-Type:
application/x-www-form-urlencoded" --data-urlencode
"code=34e8dec4289........a52fe26" --data-urlencode "redirect_uri=https://
example.com" --data-urlencode "grant_type=authorization_code"
```

Response:

```
{ "access_token":"eyJ0eXAi......mX9f7k1g", "token_type":"Bearer",
"refresh_token":"eyJ0eXAi......mruC5c3A", "expires_in":3600 }
```

Table 2-1 Required element for OAuth2.0 authentication

Element	Description
refresh_token	A token that is used to obtain a new access token without requiring user consent and Use this token in the Authorization header of all NetSuite API calls. Providing Values for NetSuite Connector Basic Configuration. After you have obtained the refresh_token value, you must provide these values in authToken under NetSuite Connector basic configuration. For information about configuration, see Configuring the NetSuite Connector. For example, eyJ0eXAi.....mX9f7k1g
refresh_token value	The full refresh token value that is received from authentication.

2.1.3 Downloading the Connector Installation Package

You can obtain the installation package for your connector on the Oracle Technology Network (OTN) website.

To download the connector installation package:

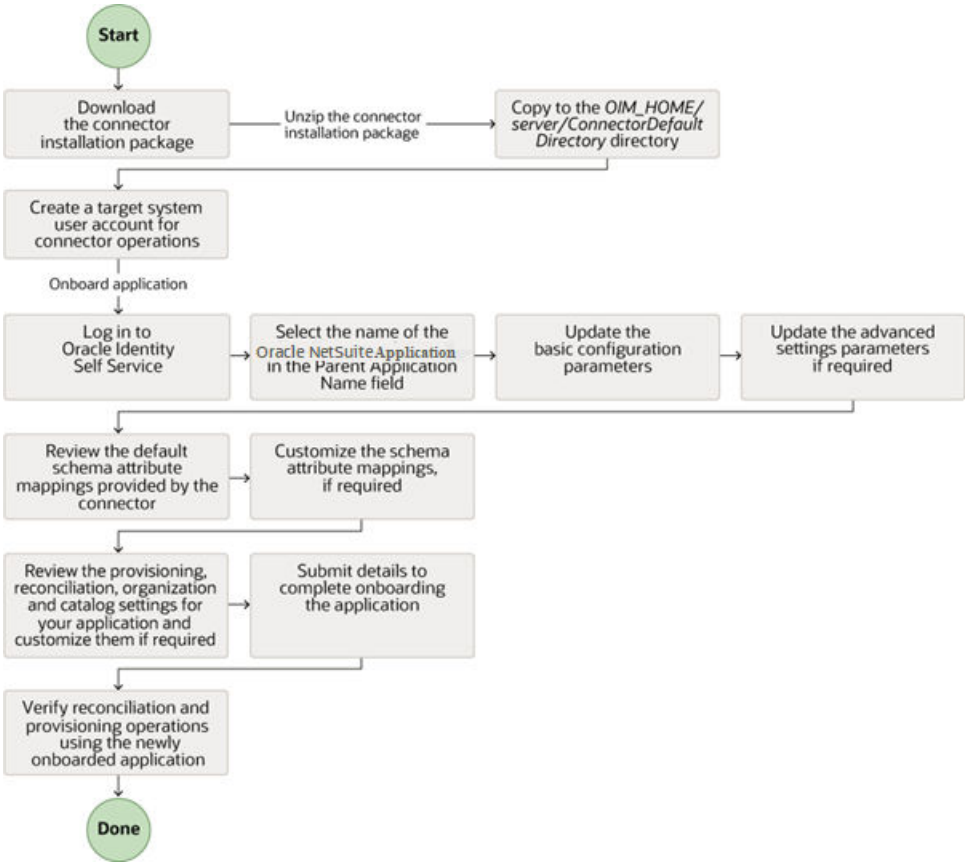
1. Navigate to the OTN website at http://www.oracle.com/technetwork/middleware/id-mgmt/downloads/connectors-101674.html.
2. Click **OTN License Agreement** and read the license agreement.
3. Select the **Accept License Agreement** option.
   You must accept the license agreement before you can download the installation package.
4. Download and save the installation package to any directory on the computer hosting Oracle Identity Governance.
5. Extract the contents of the installation package to any directory on the computer hosting Oracle Identity Governance. This creates a directory named *CONNECTOR_NAME-RELEASE_NUMBER*.
6. Copy the *CONNECTOR_NAME-RELEASE_NUMBER* directory to the *OIG_HOME*/server/ConnectorDefaultDirectory directory.

2.2 Process Flow for Creating an Application by Using the Connector

From Oracle Identity Governance release 12.2.1.3.0 onward, connector deployment is handled using the application onboarding capability of Identity Self Service.

Figure 2-1 is a flowchart depicting high-level steps for creating an application in Oracle Identity Governance by using the connector installation package.

Figure 2-1 Overall Flow of the Process for Creating an Application By Using the Connector



2.3 Creating an Application by Using the NetSuite Cloud Connector

You can onboard an application into Oracle Identity Governance from the connector package by creating a Target application. To do so, you must log in to Identity Self Service and then choose the **Applications** box on the **Manage** tab.

The following is the high-level procedure to create an application by using the connector:

Note:

For detailed information regarding each step in this procedure, see Creating Applications of *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

1. Create an application in Identity Self Service. The high-level steps are as follows:
   a. Log in to Identity Self Service either by using the **System Administration** account or an account with the **ApplicationInstanceAdministrator** admin role.
   b. Ensure that the **Connector Package** option is selected when creating an application.
   c. Update the basic configuration parameters to include connectivity-related information.
   d. If required, update the advanced setting parameters to update configuration entries related to connector operations.
   e. Review the default user account attribute mappings. If required, add new attributes or you can edit or delete existing attributes.
   f. Review the provisioning, reconciliation, organization, and catalog settings for your application and customize them if required. For example, you can customize the default correlation rules for your application if required.
   g. Review the details of the application and click **Finish** to submit the application details.
   The application is created in Oracle Identity Governance.
   h. When you are prompted whether you want to create a default request form, click **Yes** or **No**.
   If you click **Yes**, then the default form is automatically created and is attached with the newly created application. The default form is created with the same name as the application. The default form cannot be modified later. Therefore, if you want to customize it, click **No** to manually create a new form and attach it with your application.
2. Verify reconciliation and provisioning operations on the newly created application.

> **Note:**
>
> - Configuring the Connector of for details on basic configuration and advanced settings parameters, default user account attribute mappings, default correlation rules, and reconciliation jobs that are predefined for this connector.
> - Configuring Oracle Identity Governance for details on creating a new form and associating it with your application if you chose not to create the default form.

3
Configuring the Connector

Configure connection-related parameters while creating a target application. These parameter values will be used to connect Oracle Identity Governance with your target system and perform connector operations. In addition, you can view and edit attribute mappings between the process form fields in Oracle Identity Governance and target system columns, predefined correlation rules, situations and responses, and reconciliation jobs.

- Basic Configuration Parameters
- Advanced Settings Parameters
- Attribute Mappings
- Correlation Rules
- Reconciliation Jobs

3.1 Basic Configuration Parameters

These are the connection-related parameters that Oracle Identity Governance requires to connect to Oracle NetSuite Application.

> **Note:**
>
> Unless specified, do not modify entries in the below table.

Table 3-1 Parameters in the Basic Configuration

Parameter	Mandatory?	Description
Host	Yes	Enter the host name of the system on which your NetSuite target application is running. This is a mandatory attribute while creating an application. **Sample Value**: TSTDRVXXXXXXXX.suitetalk.api.netsuite.com
account	Yes	Enter the name of account. **Sample Value**: TSTDRVXXXXXXXX
consumerKey	Yes	Enter the consumerKey. **Sample Value**: 7e1c238e538bafXXXXXXXXbcea08a3171
consumerSecret	Yes	Enter the consumerSecret. **Sample Value**: fff0b23810704056XXXXXXXXXX0b768733316f

Table 3-1 (Cont.) Parameters in the Basic Configuration

Parameter	Mandatory?	Description
tokenId	Yes	Enter the tokenId. **Sample Value**: 3e23ecc14bc7dXXXXXXXd400e56177ed
tokenSecret	Yes	Enter the tokenSecret **Sample Value**: cd750404ee67653aXXXXXXXXXX646422da64c
nonce	Yes	Enter any 10 letters characters. **Sample value:** abcdefghij
auth URL	Yes	Enter the URL of the authentication server that validates the client ID and client secret for your target system
auth Token	Yes	Enter the Refresh Token Values. This value can be fetched by performing OAuth code authorization flow. **Sample value**: "access_token=eyJ0eXAiOiJNVCIsImFsZyI6IlJTMjU2Iiwia2lkIjoiNjgxODVmZjEtNGU1MS00Y2U5LWFmMWMtNjg5ODEyMjAzMzE3In0.AQoAAAABAAUABwCA8Kx7sbjaSAgAgDDQifS42kgCAGcjU3expKxCtXXXXXXXXXXXFAAAADQAkAAAANDdhZWE4OWQtNWViYy00NmMyLWI0YmYtNjE5MDRhMjE0MTE1IgAkAAAANDdhZWE4OWQtNWViYy00NmMyLWI0YmYtNjE5MDRhMjE0MTE1MACABwhGsbjaSDcAC1hTwTsYB0GKF0Qif6kfLg.Lk45d4mcBPIrBghYun1S2pVa0EE0XHYTU66cqWpEuPMgSieVTRgwF3wyTOSgyPuiJNf18QTJcG6js4LvVL7sPw8IJwQ6bd
ConnectorServerName	No	This field is blank. If you are using this connector with the Java Connector Server, then provide the name of Connector Server IT Resource here.
port	No	Enter the port number at which the target system is listening. **Sample value**: 443
proxyHost	No	Enter the name of the proxy host used to connect to an external target. **Sample value**: www.example.com
proxyPassword	No	Enter the password for the proxy user.
proxyPort	No	Enter the proxy port number. **Sample value**: 1105
proxyUser	No	Enter the proxy username if you are using a proxy server to access the internet.
SSL	No	If the target system requires SSL connectivity, then set the value of this parameter to true. Otherwise set the value to false. **Default value:** true

3.2 Advanced Settings Parameters

These are the configuration-related entries that the connector uses during reconciliation and provisioning operations.

Note:

- Unless specified, do not modify entries in the below table.
- All parameters in the table below are mandatory.

Table 3-2 Advanced Settings Parameters

Parameter	Mandatory?	Description
Connector Name	Yes	This entry holds the name of the connector class. **Default value**: org.identityconnectors.netsuite.NetSuiteConnector
Bundle Name	Yes	This entry holds the name of the connector bundle. **Default value**: org.identityconnectors.netsuite
Bundle Version	Yes	This entry holds the version of the connector bundle. **Default value**: 12.3.0
soapURL	Yes	Enter the SOAP endpoint URL **Default value**: /services/NetSuitePort_2022_1
suiteqlURL	Yes	Enter the Suiteql endpoint URL **Default value**: /services/rest/query/v1/suiteql
pageSize	Yes	The number of users that appears on a page for a search operation. **Default value**:100

3.3 Attribute Mappings

The following topic provides the attribute mappings details.

- Attribute Mappings for the Target Application

3.3.1 Attribute Mappings for the Target Application

The Schema page for a target application displays the default schema (provided by the connector) that maps Oracle Identity Governance attributes to target system attributes. The connector uses these mappings during reconciliation and provisioning operations.

The following table lists the user-specific attribute mappings between the process form fields in Oracle Identity Governance and NetSuite target application attributes The table also lists whether a specific attribute is used during provisioning or reconciliation and whether it is a matching key field for fetching records during reconciliation.

If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application of *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-3 Default Attribute for NetSuite Target Application

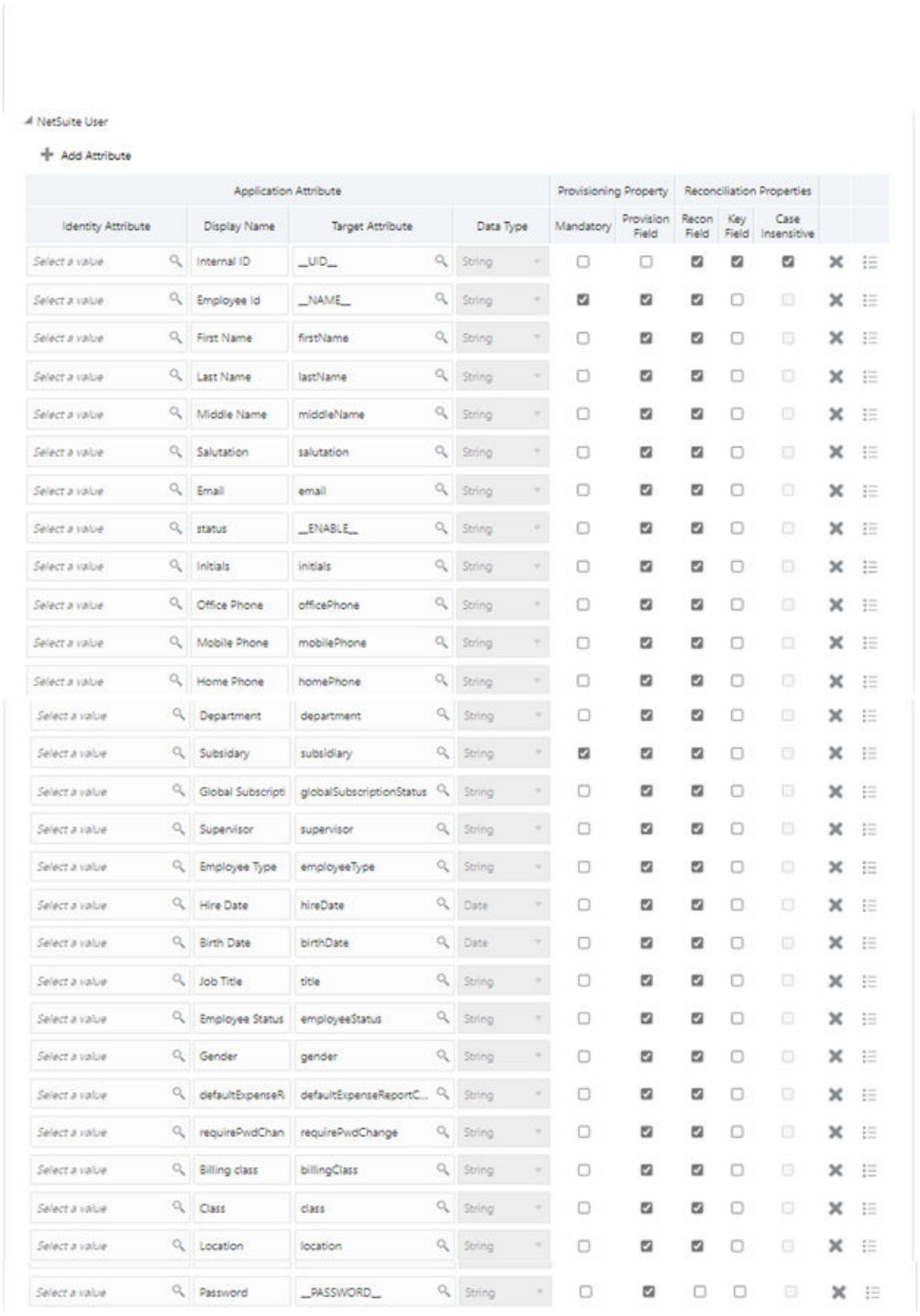
Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Provision Field?	Recon Field?	Key Field?	Case Insensitive?
Internal ID	__UID__	String	No	No	Yes	Yes	Yes
Employee ID	__NAME__	String	Yes	Yes	Yes	No	Not applicable
First Name	firstName	String	No	Yes	Yes	No	Not applicable
Last Name	lastName	String	No	Yes	Yes	No	Not applicable
Middle Name	middleName	String	No	Yes	Yes	No	Not applicable
Salutation	salutation	String	No	Yes	Yes	No	Not applicable
Email	email	String	No	Yes	Yes	No	Not applicable
status	__ENABLE__	String	No	Yes	Yes	No	Not applicable
Intials	intials	String	No	Yes	Yes	No	Not applicable
Office Phone	officePhone	String	No	Yes	Yes	No	Not applicable
Mobile Phone	mobilePhone	String	No	Yes	Yes	No	Not applicable
Home Phone	homePhone	String	No	Yes	Yes	No	Not applicable
Department	department	String	No	Yes	Yes	No	Not applicable
Subsidiary	subsidiary	String	Yes	Yes	Yes	No	Not applicable
Global Subscription Status	globalSubscriptionStatus	String	No	Yes	Yes	No	Not applicable

Table 3-3 (Cont.) Default Attribute for NetSuite Target Application

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Provision Field?	Recon Field?	Key Field?	Case Insensitive?
Supervisor	supervisor	String	No	Yes	Yes	No	Not applicable
Employee Type	employeeType	String	No	Yes	Yes	No	Not applicable
Hire Date	hireDate	Date	No	Yes	Yes	No	Not applicable
Birth Date	birthDate	Date	No	Yes	Yes	No	Not applicable
Job Title	title	String	No	Yes	Yes	No	Not applicable
Employee Status	employeeStatus	String	No	Yes	Yes	No	Not applicable
Gender	gender	String	No	Yes	Yes	No	Not applicable
defaultExpenseReportCurrency	defaultExpenseReportCurrency	String	No	Yes	Yes	No	Not applicable
requirePwdChange	requirePwdChange	String	No	Yes	Yes	No	Not applicable
Billing class	billingClass	String	No	Yes	Yes	No	Not applicable
Class	class	String	No	Yes	Yes	No	Not applicable
Location	location	String	No	Yes	Yes	No	Not applicable
Password	__PASSWORD__	String	No	Yes	No	No	Not applicable
IT Resource Name		Long	No	No	Yes	No	Not applicable

The following figure shows the default User account attribute mappings.

Figure 3-1 Default Attribute Mappings for Oracle NetSuite User Account



NetSuite Role Entitlement

The following table lists the roles forms attribute mappings between the process form fields in Oracle Identity Governance and NetSuite target application attributes. The table lists whether a given attribute is mandatory during provisioning. It also lists

whether a given attribute is used during reconciliation and whether it is a matching key field for fetching records during reconciliation.

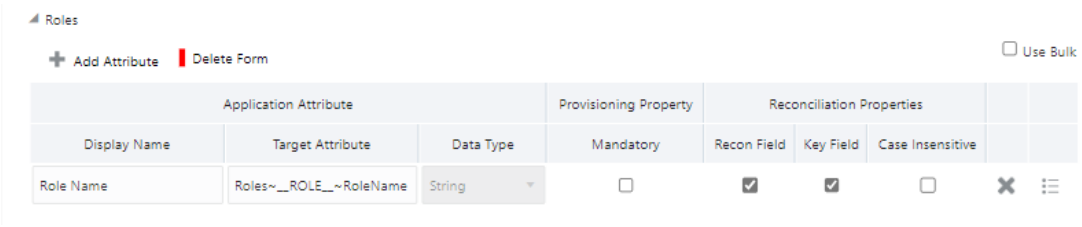
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application of *Oracle Fusion Middleware Performing Self Service Tasks* with Oracle Identity Governance.

Table 3-4 Default Attribute Mappings for Roles

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Recon Field	Key Field?	Case Insensitive?
Role Name	Roles~__ROLE__~RoleName	String	No	Yes	Yes	No

The following figure shows the default Roles Entitlement mapping.

Figure 3-2 Default Attribute Mappings for NetSuite Roles



NetSuite Groups Entitlement

The following table lists the group forms attribute mappings between the process form fields in Oracle Identity Governance and NetSuite target application attributes. The table lists whether a given attribute is mandatory during provisioning. It also lists whether a given attribute is used during reconciliation and whether it is a matching key field for fetching records during reconciliation.

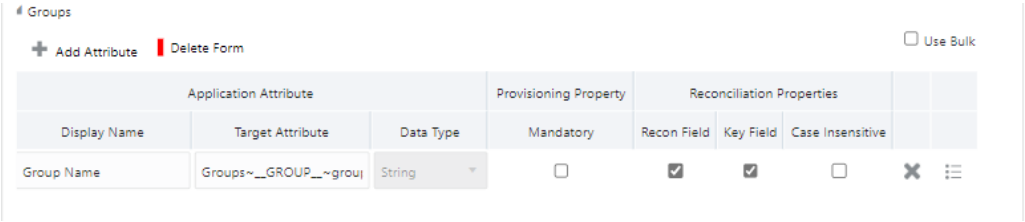
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application *Oracle Fusion Middleware Performing Self Service Tasks* with Oracle Identity Governance.

Table 3-5 Default Attribute Mappings for Roles

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Recon Field	Key Field?	Case Insensitive?
Group Name	Groups~__GROUP__~groupName	String	No	Yes	Yes	No

The following table shows the default Groups Entitlement mapping.

Figure 3-3 Default Attribute Mappings for NetSuite Groups



NetSuite Global Permission

The following table lists the Global Permission forms attribute mappings between the process form fields in Oracle Identity Governance and Oracle NetSuite target application attributes. The table lists whether a given attribute is mandatory during provisioning. It also lists whether a given attribute is used during reconciliation and whether it is a matching key field for fetching records during reconciliation.

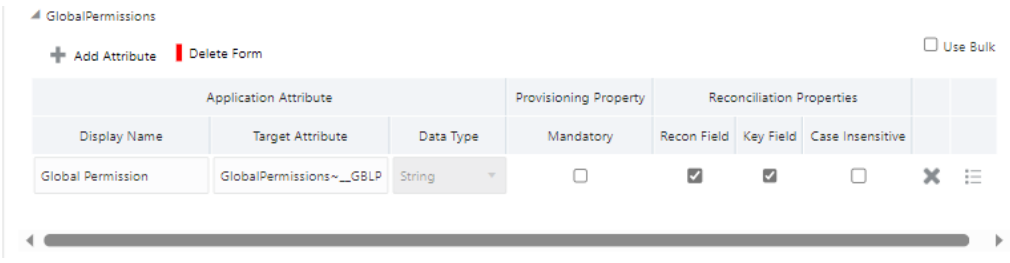
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described Creating a Target Application of *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-6 Default Attribute Mappings for Global Permission

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property ?	Recon Field	Key Field ?	Case Insensitive?
Global Permission	GlobalPermissions~__GBLPERMISSIONS__~GlobalPermissionsName	String	No	Yes	Yes	No

The following figure shows the default Global Permission Entitlement mapping.

Figure 3-4 Default Attribute Mappings for NetSuite Global Permission



3.4 Correlation Rules

Learn about the predefined rules, responses, and situations for Target applications. The connector uses these rules and responses for performing reconciliation.

- Correlation Rules for the Target Application

3.4.1 Correlation Rules for the Target Application

When you create a target application, the connector uses correlation rules to determine the identity to which Oracle Identity Governance must assign a resource.

Predefined Identity Correlation Rules

By default, the Oracle NetSuite connector provides a simple correlation rule when you create a target application. The connector uses this correlation rule to compare the entries in Oracle Identity Governance repository and the target system repository, determine the difference between the two repositories, and apply the latest changes to Oracle Identity Governance.

The following table lists the default simple correlation rule for a Oracle NetSuite connector. If required, you can edit the default correlation rule or add new rules. You can create complex correlation rules also. For more information about adding or editing simple or complex correlation rules, see Updating Identity Correlation Rules in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-7 Predefined Identity Correlation Rule for an Oracle NetSuite Connector

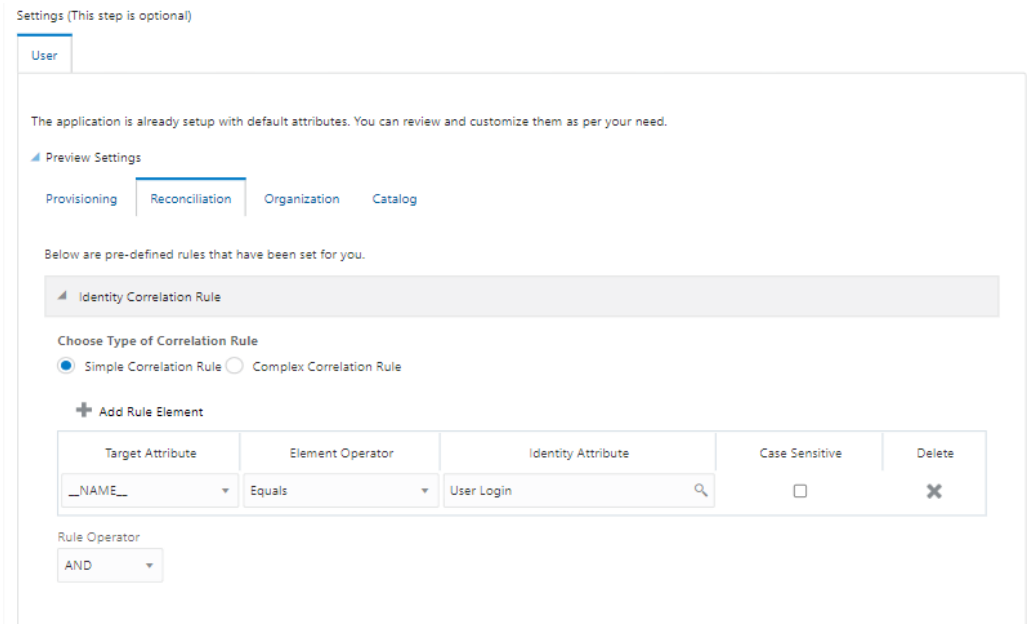
Target Attribute	Element Operator	Identity Attribute	Case Sensitive?
__NAME__	Equals	User Login	No

In this identity rule:

- __NAME__ is a single-valued attribute on the target system that identifies the user account.
- User Login is the field on the OIG User form.

The following figure shows the Simple Correlation Rule for NetSuite Target Application

Figure 3-5 Simple Correlation Rule for NetSuite Target Application



Predefined Situations and Responses

The Oracle NetSuite connector provides a default set of situations and responses when you create a target application. These situations and responses specify the action that Oracle Identity Governance must take based on the result of a reconciliation event.

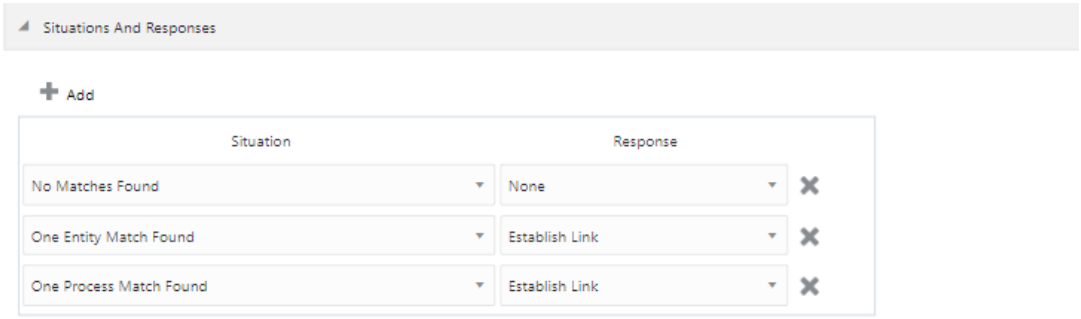
The following table lists the default situations and responses for a Oracle NetSuite Target application. If required, you can edit these default situations and responses or add new ones. For more information about adding or editing situations and responses, see Updating Situations and Responses in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-8 Predefined Situations and Responses for a Oracle NetSuite Target Application

Situation	Response
No Matches Found	None
One Entity Match Found	Establish Link
One Process Match Found	Establish Link

The following figure shows the situations and responses for a NetSuite that the connector provides by default.

Figure 3-6 Predefined Situations and Responses for a Oracle NetSuite Target Application



3.5 Reconciliation Jobs

These are the reconciliation jobs that are automatically created in Oracle Identity Governance after you create the application.

User Reconciliation Jobs

You can either use these predefined jobs or edit them to meet your requirements. Alternatively, you can create custom reconciliation jobs. For information about editing these predefined jobs or creating new ones, see Updating Reconciliation Jobs in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

The following reconciliation jobs are available for reconciling user data:

- NetSuite Full User Reconciliation: Use this reconciliation job to reconcile user data from a target application.
- NetSuite Limited User Reconciliation: Use this reconciliation job to reconcile records from the target system based on a specified filter criterion.

The following table describes the parameters of the NetSuite Full User Reconciliation job.

Table 3-9 Parameters of the NetSuite Full User Reconciliation Job

Parameter	Description
Application name	Name of the AOB application with which the reconciliation job is associated. This value is the same as the value that you provided for the Application Name field while creating your target application. Do *not* change the default value.
Filter Suffix	Enter the search filter for fetching user records from the target system during a reconciliation run. Filter suffix for single user: **Sample value:** 10245 In this example, the record whose Internal ID value is 10245 is reconciled. For more information about creating filters, see Performing Limited (Filtered) Reconciliation.

Table 3-9 (Cont.) Parameters of the NetSuite Full User Reconciliation Job

Parameter	Description
Object Type	This parameter holds the name of the object type for the reconciliation run. **Default value**: User Do *not* change the default value.
Scheduled Task Name	Name of the scheduled task used for reconciliation. Do *not* modify the value of this parameter.

Reconciliation Jobs for Entitlements

The following jobs are available for reconciling entitlements:

- NetSuite Group Lookup Reconciliation
- NetSuite Role Lookup Reconciliation
- NetSuite Department Lookup Reconciliation
- NetSuite Subsidiary Lookup Reconciliation
- NetSuite Employee Status Lookup Reconciliation
- NetSuite Supervisor Lookup Reconciliation
- NetSuite Default Expense Report Currency Lookup Reconciliation
- NetSuite Employee Type Lookup Reconciliation
- NetSuite Billing Class Lookup Reconciliation
- NetSuite Class Lookup Reconciliation
- NetSuite location Lookup Reconciliation

The parameters for all the reconciliation jobs are the same.

Table 3-10 Parameters of the Reconciliation Jobs for Entitlements

Parameter	Description
Application Name	Current AOB application name with which the reconciliation job is associated. Do *not* modify this value.
Code Key Attribute	Name of the connector attribute that is used to populate the Code Key column of the lookup definition. (Specified as the value of the Lookup Name attribute). Default value: __UID__ Do *not* modify this value.
Decode Attribute	Name of the connector attribute that is used to populate the Decode column of the lookup definition. (Specified as the value of the Lookup Name attribute). Default value: __NAME__

Table 3-10 (Cont.) Parameters of the Reconciliation Jobs for Entitlements

Parameter	Description
Lookup Name	Enter the name of the lookup definition in Oracle Identity Governance that must be populated with values fetched from the target system. Depending on the Reconciliation job that you are using, the default values are as follows: • For NetSuite Group Lookup Reconciliation: Lookup.NetSuite.Groups • For NetSuite Role Lookup Reconciliation: Lookup.NetSuite.Roles • For NetSuite Department Lookup Reconciliation: Lookup.NetSuite.department • For NetSuite Subsidiary Lookup Reconciliation: Lookup.NetSuite.subsidiary • For NetSuite Supervisor Lookup Reconciliation: Lookup.NetSuite.Supervisor • For NetSuite Employee Status Lookup Reconciliation: Lookup.NetSuite.employeeStatus • For NetSuite Default Expense Report Currency Lookup Reconciliation: Lookup.NetSuite.defaultExpenseReportCurrency • For NetSuite Employee Type Lookup Reconciliation: Lookup.NetSuite.employeeType • For NetSuite Billing Class Lookup Reconciliation: Lookup.NetSuite.billingClass • For NetSuite Class Lookup Reconciliation: Lookup.NetSuite.class • For NetSuite location Lookup Reconciliation: Lookup.NetSuite.location If you create a copy of any of these lookup definitions, then enter the name of that new lookup definition as the value of the Lookup Name attribute.
Object Type	Enter the type of object you want to reconcile. Depending on the reconciliation job that you are using, the default values are as follows: • For NetSuite Group Lookup Reconciliation: __GROUP__ • For NetSuite Role Lookup Reconciliation: __ROLE__ • For NetSuite Department Lookup Reconciliation: __DEPARTMENT__ • For NetSuite Subsidiary Lookup Reconciliation: __SUBSIDIARY__ • For NetSuite Supervisor Lookup Reconciliation: __SUPERVISOR__ • For NetSuite Employee Status Lookup Reconciliation: __EMPSTATUS__ • For NetSuite Default Expense Report Currency Lookup Reconciliation: __EXPREPORT__ • For NetSuite Employee Type Lookup Reconciliation: __EMPTYPE__ • For NetSuite Billing Class Lookup Reconciliation: __BILLING__ • For NetSuite Class Lookup Reconciliation: __CLASS__ • For NetSuite location Lookup Reconciliation: __LOCATION__ **Note:** Do not change the value of this parameter.

4

Performing Post configuration Tasks for the Connector

These are the tasks that you can perform after creating an application in Oracle Identity Governance.

- Configuring Oracle Identity Governance
- Harvesting Entitlements and Sync Catalog
- Managing Logging for the Connector
- Configuring the IT Resource for the Connector Server
- Localizing Field Labels in UI Forms
- Configuring SSL

4.1 Configuring Oracle Identity Governance

During application creation, if you did not choose to create a default form, then you must create a UI form for the application that you created by using the connector.

> **Note:**
>
> Perform the procedures described in this section only if you did not choose to create the default form during creating the application.

The following topics describe the procedures to configure Oracle Identity Governance:

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

4.1.1 Creating and Activating a Sandbox

You must create and activate a sandbox to begin using the customization and form management features. You can then publish the sandbox to make the customizations available to other users.

See Creating a Sandbox and Activating a Sandbox in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*.

4.1.2 Creating a New UI Form

You can use Form Designer in Oracle Identity System Administration to create and manage application instance forms.

See Creating Forms By Using the Form Designer in *Oracle Fusion Middleware Administering Oracle Identity Governance*.

While creating the UI form, ensure that you select the resource object corresponding to the newly created application that you want to associate the form with. In addition, select the **Generate Entitlement Forms** check box.

4.1.3 Publishing a Sandbox

Before publishing a sandbox, perform this procedure as a best practice to validate all sandbox changes made till this stage as it is difficult to revert the changes after a sandbox is published.

1. In Identity System Administration, deactivate the sandbox.
2. Log out of Identity System Administration.
3. Log in to Identity Self Service using the xelsysadm user credentials and then activate the sandbox that you deactivated in Step 1.
4. In the Catalog, ensure that the application instance form for your resource appears with correct fields.
5. Publish the sandbox. See Publishing a Sandbox in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*.

4.1.4 Updating an Existing Application Instance with a New Form

For any changes that you do in the schema of your application in Identity Self Service, 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 and activate a sandbox.
2. Create a new UI form for the resource.
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.

See Also:

- Creating a Sandbox and Activating a Sandbox in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*
- Creating Forms By Using the Form Designer in *Oracle Fusion Middleware Administering Oracle Identity Governance*
- Publishing a Sandbox in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*

4.2 Harvesting Entitlements and Sync Catalog

You can populate Entitlement schema from child process form table, and harvest roles, application instances, and entitlements into catalog. You can also load catalog metadata.

To harvest entitlements and sync catalog:

1. Run the scheduled jobs for lookup field synchronization listed in Reconciliation Jobs
2. Run the Entitlement List scheduled job to populate Entitlement Assignment schema from child process form table.
3. Run the Catalog Synchronization Job scheduled job.

See Also:

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

4.3 Managing Logging for the Connector

Oracle Identity Governance uses the Oracle Diagnostic Logging (ODL) logging service for recording all types of events pertaining to the connector.

The following topics provide detailed information about logging:

- Understanding Logging on the Connector Server
- Enabling Logging for the Connector Server
- Understanding Log Levels
- Enabling Logging

4.3.1 Understanding Logging on the Connector Server

When you enable logging, the connector server stores in a log file information about events that occur during the course of provisioning and reconciliation operations for different statuses. By default, the connector server logs are set at INFO level and you can change this level to any one of these.

- Error
  This level enables logging of information about errors that might allow connector server 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 operation.
- FINE, FINER, FINEST
  These levels enable logging of information about fine-grained events, where FINEST logs information about all events.

4.3.2 Enabling Logging for the Connector Server

Edit the logging properties file located in the CONNECTOR_SERVER_HOME/Conf directory to enable logging.

1. Open the `logging.properties` file in a text editor.
2. Navigate to the *CONNECTOR_SERVER_HOME*/Conf directory.
3. Edit the following entry by replacing INFO with the required level of logging:`level=INFO`
4. Save and close the file.
5. Restart the connector server.

4.3.3 Understanding Log Levels

When you enable logging, Oracle Identity Governance automatically stores in a log file information about events that occur during the course of provisioning and reconciliation operations.

ODL is the principle logging service used by Oracle Identity Governance and 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:

- 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 Governance 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 message types are mapped to ODL message type and level combinations as shown in the below table.

Table 4-1 Log Levels and ODL Message Type:Level Combinations

Java Level	ODL Message Type:Level
SEVERE.intValue()+100	INCIDENT_ERROR:1
SEVERE	ERROR:1
WARNING	WARNING:1

Table 4-1 (Cont.) Log Levels and ODL Message Type:Level Combinations

Java Level	ODL Message Type:Level
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 Governance.

4.3.4 Enabling Logging

Perform this procedure to enable logging in Oracle WebLogic Server.

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='NetSuite-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> Copy<logger name="
        ORG.IDENTITYCONNECTORS.NETSUITE" level="[LOG_LEVEL]"
          useParentHandlers="false">   <handler
          name="NetSuite-handler"/>   <handler  name="console-
handler"/> </logger>
```

   b. Replace both occurrences of **[LOG_LEVEL]** with the ODL message type and level combination that you require. The Table 4-1 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= 'NetSuite -handler'

level='NOTIFICATION:1'class='oracle.core.ojdl.logging.ODLHandlerFactor
y'>
```

```
            <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.NETSUITE" level="NOTIFICATION:1"
          useParentHandlers="false">   <handler name="NetSuite-
handler"/>
            <handler
          name="console-handler"/>
 </logger>
```

2. 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.
3. Save and close the file.
4. Set the following environment variable to redirect the server logs to a file:
   a. For Microsoft Windows: set WLS_REDIRECT_LOG= ***FILENAME***
   b. 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.
5. Restart the application server.

4.4 Configuring the IT Resource for the Connector Server

If you have used the Connector Server, then you must configure values for the parameters of the Connector Server IT resource.

After you create the application for your target system, you must create an IT resource for the Connector Server as described in Creating IT Resources of *Oracle Fusion Middleware Administering Oracle Identity Governance.* While creating the IT resource, ensure to select Connector Server from the IT Resource Type list. In addition, specify values for the parameters of IT resource for the Connector Server listed in the following table.

For more information about searching for IT resources and updating its parameters, see Managing IT Resources in Oracle Fusion Middleware Administering Oracle Identity Governance.

Table 4-2 Parameters of the IT Resource for the Oracle NetSuite 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. Sample value: 8763
Timeout	Enter an integer value which specifies the number of milliseconds after which the connection between the Connector Server and Oracle Identity Governance times out. If the value is zero or if no value is specified, the timeout is unlimited. Sample value: 0 (recommended value)
UseSSL	Enter true to specify that you will configure SSL between Oracle Identity Governance and the Connector Server. Otherwise, enter false. Default value: false

Note:

It is recommended that you configure SSL to secure communication with the connector server. To configure SSL, see Configuring the Java Connector Server with SSL for OIG in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*.

4.5 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. Resource bundles are available in the connector installation media.

To localize field labels that is added to the 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 (`oracle.iam.console.identity.sysadmin.ear_V2.0_metadata.zip`) to the local computer.
5. Extract the contents of the archive, and open the following file in a text editor:

   `SAVED_LOCATION\xliffBundles\oracle\iam\ui\runtime\BizEditorBundle.xlf`

> **Note:**
>
> You will not be able to view the BizEditorBundle.xlf file unless you complete creating the application for your target system or perform any customization such as creating a UDF.

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 Japanese:

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

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

```
 <trans-unit
      id="$
{adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResou
rceBundle']
['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity
.userEO.UD_
      USER_FIRST_NAME__c_description']}"><source>User Principal
      Name</source><target/></trans-unit><trans-unit

id="sessiondef.oracle.iam.ui.runtime.form.model.RSAForm.entity.Ne
tSuiteFormEO.UD_USER_FIRST_NAME



__c_LABEL"><source>First Name</source><target/> </trans-unit>
```

d. Open the resource file from the connector package, for example `NetSuite_ja.properties`, and get the value of the attribute from the file, for example,

```
global.udf.UD_GA_USR_USER_FIRST_NAME =\u540D
```

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

```
<trans-unit id="$
{adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResourceBu
 ndle']
['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity.use
rEO.UD_GA_USR_USER_FIRST_NAME __c_description']}"><source>First Name</
source> <target>\u540D</target></trans-unit> <trans-
unitid="sessiondef.oracle.iam.ui.runtime.form.model.NetSuite.entity
sEO.UD_GA_USR_FIRST_NAME__c_LABEL"><source> First Name </source>
<target>\u540D </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_ja.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.

4.6 Configuring SSL

Configure SSL to secure data communication between Oracle Identity Governance and the Oracle NetSuite target system.

> **Note:**
>
> If you are using this connector along with a Connector Server, then there is no need to configure SSL. You can skip this section.

To configure SSL:

1. Obtain the SSL public key certificate of Oracle NetSuite.
2. Copy the public key certificate of Oracle NetSuite to the computer hosting Oracle Identity Governance.

3. Run the following `keytool` command to import the public key certificate into the identity key store in Oracle Identity Governance:`keytool -import -alias ALIAS -trustcacerts -file CERT_FILE_NAME -keystore KEYSTORE_NAME -storepass PASSWORD`

In this command:

- *ALIAS* is the public key certificate alias.
- *CERT_FILE_NAME* is the full path and name of the certificate store (the default is cacerts).
- *KEYSTORE_NAME* is the name of the keystore.
- *PASSWORD* is the password of the keystore.
- `keytool -import -alias serverwl -trustcacerts -file supportcert.pem -keystore client_store.jks -storepass weblogic1`
  - `keytool -import -keystore <JAVA_HOME>/jre/lib/security/cacerts -file <Cert_Location>/NetSuite.crt -storepass changeit -alias NetSuite_1`
  - `keytool -import -keystore <WL_HOME>/server/lib/DemoTrust.jks -file <Cert_Location>/NetSuite.crt -storepass DemoTrustKeyStorePassPhrase -alias NetSuite_2`

> **Note:**
>
> - Change the parameter values passed to the keytool command according to your requirements. Ensure that there is no line break in the keytool arguments
> - In the Oracle Identity Governance cluster, perform this procedure on each node of the cluster and then restart each node.
> - Ensure that the system date for Oracle Identity Governance is in sync with the validity date of the SSL certificate to avoid any errors during SSL communication.

5
Using the Connector

You can use the connector for performing reconciliation and provisioning operations after configuring your application to meet your requirements.

- Configuring Reconciliation
- Configuring Reconciliation Jobs
- Configuring Provisioning

5.1 Configuring Reconciliation

You can configure the connector to specify the type of reconciliation and its schedule.

This section discusses the following topics related to configuring reconciliation:

- Performing Full Reconciliation
- Performing Limited (Filtered) Reconciliation

5.1.1 Performing Full Reconciliation

Full reconciliation involves reconciling all existing user records from the target system into Oracle Identity Governance. After you create the application, you must first perform full reconciliation.

To perform a full reconciliation run, remove (delete) any value currently assigned to the Filter suffix parameters and run one of the reconciliation jobs listed in the Reconciliation Jobs section.

5.1.2 Performing Limited (Filtered) Reconciliation

Limited or filtered reconciliation is the process of limiting the number of records being reconciled based on a set filter criterion.

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. You do this by creating filters for the reconciliation module.

This connector provides a Filter Suffix attribute (a scheduled task attribute) that allows you to use any of the attributes of the target system to filter target system records. You specify a value for the Filter Suffix attribute while configuring the user reconciliation scheduled job.

You can perform limited reconciliation by creating filter for the reconciliation module. The connector only supports Internal ID filter.

Filter Suffix: 10245

In this example, the record whose Internal ID value is 10245 is reconciled.

5.2 Configuring Reconciliation Jobs

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

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

To configure a reconciliation job:

1. Log in to Identity System Administration.
2. In the left pane, under System Management, click **Scheduler**.

> **Note:**
>
> If you are using OIG 12cPS4 with 2022O CTBP or later version, log in to Identity Console, click **Manage,** under S**ystem Configuration**, click **Scheduler**.

3. Search for and open the scheduled job as follows:
   a. In the Search field, enter the name of the scheduled job as the search criterion. Alternatively, you can click **Advanced Search** and specify the search criterion.
   b. In the search results table on the left pane, click the scheduled job in the Job Name column.
4. On the Job Details tab, you can modify the parameters of the scheduled task:
   a. **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.
   b. **Schedule Type**: Depending on the frequency at which you want the job to run, select the appropriate schedule type. See Creating Jobs in *Oracle Fusion Middleware Administering Oracle Identity Governance*.
   In addition to modifying the job details, you can enable or disable a job.
5. On the **Job Details** tab, in the Parameters region, specify values for the attributes of the scheduled task.

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

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

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

5.3 Configuring Provisioning

You can configure the provisioning operation for the Oracle NetSuite connector.

This section provides information on the following topics:

- Guidelines on Performing Provisioning Operations
- Performing Provisioning Operations

5.3.1 Guidelines on Performing Provisioning Operations

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

Provisioning attributes required to create user account

To create User provisioning operation, follow the following values as required:

- Employee Id: The user's entity Id
- Subsidiary: The user's subsidiary.
- Password: The password of the user.

> **Note:**

Target does not allow to create a user with existing Employee id which has already been used in Oracle NetSuite cloud.

Attributes required to be updated in the parent form.

- First Name: The user's first name.
- Last Name: The user's last name.
- Middle Name: The user’s middle name.
- Email: The user's email ID.
- Initials: The user's initials
- Office Phone: The user's office phone number.
- Mobile Phone: The user's mobile number.
- Home Phone: The user's home phone number
- Department: The user's department
- Subsidiary: The user's subsidiary.
- Global Subscription Status:

- Supervisor: The user's supervisor
- Employee Type: The user's Employee type.
- Hire Date: The user's hire date.
- Birth Date: The user's birth date
- Job Title: The user's job title.
- Employee Status: Employee status of the user.
- Gender: Gender of the user.
- defaultExpenseReportCurrency: Expense currency used by user.
- requirePwdChange: Indicates if user password needs to be reset after login.
- Billing class: The user's billing class.
- Class: The user's class.
- Location: The user's location.
- Password: The password of the user.

5.3.2 Performing Provisioning Operations

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

To perform provisioning operations in Oracle Identity Governance, perform the following steps:

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

6
Extending the Functionality of the Connector

You can extend the functionality of the connector to address your specific business requirements.

This section discusses the following topics:

- Configuring Transformation and Validation of Data
- Configuring Action Scripts
- Configuring the Connector for Multiple Installations of the Target System

6.1 Configuring Transformation and Validation of Data

Configure transformation and validation of user account data by writing Groovy script logic while creating your application.

You can configure transformation of reconciled single-valued user data according to your requirements. For example, you can use First Name and Last Name values to create a value for the Full Name field in Oracle Identity Governance.

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

To configure transformation or validation of user account data, you must write Groovy scripts while creating your application. For more information about writing Groovy script-based validation and transformation logic, see Validation and Transformation of Provisioning and Reconciliation Attributes of *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

6.2 Configuring Action Scripts

You can configure **Action Scripts** by writing your own Groovy scripts while creating your application.

These scripts can be configured to run before or after create, update, or delete an account provisioning operation. For example, you can configure a script to run before every user creation operation.

For information on adding or editing action scripts, see Updating the Provisioning Configuration in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

6.3 Configuring the Connector for Multiple Installations of the Target System

You must create copies of configurations of your base application to configure it for multiple installations of the target system.

The following example illustrates this requirement:

The London and New York offices of Example Multinational Inc have their own installations of the target system, including independent schema for each. The company has recently installed Oracle Identity Governance, and they want to configure it to link all the installations of the target system.

To meet the requirement posed by such a scenario, you must clone your application which copies all configurations of the base application into the cloned application. For more information about cloning applications, see Cloning Applications in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

7

Known Issues and Workarounds

The following are the known issues and limitations with the Oracle NetSuite connector.

Known Issues

1. To assign a role to a specific user, you must have a valid User account, password, and email address in Oracle NetSuite target, else you will encounter an error.
   Error Message:

   <ORG.IDENTITYCONNECTORS.NETSUITE.UTILS.NETSUITESOAPHELPER> <org.identityconnectors.netsuite.utils.NetSuiteSOAPHelper : checkResponseStatus: Please enter a valid email address.

   <ORG.IDENTITYCONNECTORS.NETSUITE.UTILS.NETSUITESOAPHELPER> <org.identityconnectors.netsuite.utils.NetSuiteSOAPHelper : checkResponseStatus: The password must be at least 10 characters long.
2. If an end-user must log in to Oracle NetSuite, the user must be assigned with a role which has access to Oracle NetSuite. This is a mandatory pre-requirement to access the target.

Workaround

1. If you encounter the message 'Test Connection failed' with an error message '**CONNECTOR_EXCEPTION_TEST_OP**', then use the following workaround.
   Connect to a proper internet connection. You can update the credentials or regenerating the AuthToken of the application.

8

Files and Directories in the Connector Installation Package

These are the components of the connector installation package that comprise the Oracle NetSuite connector.

Table 8-1 Files and Directories in the Oracle NetSuite Connector Installation Package

File in the Installation Package	Description
/bundle/org.identityconnectors.netsuite-12.3.0.jar	This JAR is the ICF connector bundle.
configuration/NetSuite-CI.xml	This XML file contains configuration information.
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 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.
xml/NetSuite-target-template.xml	This file contains definitions for the connector objects required for creating a Target application. It includes certain details required to connect Oracle Identity Governance with the target system. It also includes configuration details specific to your target system, attribute mappings, correlation rules, and reconciliation jobs.
xml/NetSuite-pre-config.xml	This XML file contains definitions for the connector objects associated. it contains definitions of Lookups